The game of software engineering

José Juan Domínguez Veiga, Bsc (Hons) (CIT)

A dissertation submitted to the University of Dublin,
in partial fulfilment of the requirements for the degree of
Master of Science in Technology & Learning

2009

Declaration

I declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed: ____________________
José Juan Domínguez Veiga, BSc(Hons) (CIT)
13/05/2009

Permission to lend and/or copy

I agree that Trinity College Library may lend or copy this dissertation upon request.

Signed: ____________________
José Juan Domínguez Veiga
13/05/2009

Acknowledgement

to **all** the participants for making this possible.

The Game of Software Engineering

Abstract

Computer science and programming languages have very much evolved, along with digital technology, during the last few decades. Advanced hardware and communication technologies are the basis of complex software systems, providing solutions fitted to the most diverse tasks and business domains. However, teaching approaches in computer science, as in many other educational fields, have failed to keep pace with the technological tools available, lacking efficacy in today's evolving world. In programming courses, teaching practices derived from procedural programming are still being widely applied to other programming paradigms. The focus on syntax and implementation that characterises the more traditional curriculum is not suited for the abstract concepts of the new paradigms (Machanick, 1998). Furthermore the isolation in which the learners are forced to approach the different courses is not beneficial to develop and understanding of s*oftware engineering* as a profession (Garlan, Gluch, & Tomayko, 1997). A different approach is needed to introduce the student in the complex design and architecture style of modern software, and in the social context of the software profession.

Massive multiplayer online games and other virtual environments have benefited from the advances in technology. Virtual worlds are currently considered as more than a game. They have since the spring of 2006 "*gone mainstream*" becoming platforms for corporations, artist and also universities (Ludlow & Wallace, 2007). Research about how these virtual worlds can enhance the educational environment is thriving at the moment. The main characteristics of a virtual world, such as its communication tools, the possibility to create visual 3D content, and role playing, could benefit any teaching experience.

This exploratory study investigates the possibility of applying new approaches to teaching programming, in the completely different environment provided by an immersive virtual world, and through the design of a game. Based on the theories of Epistemic Games (Shaffer, 2006), and Cognitive Apprenticeship (Collins, J.S. Brown, & Newman, 1989), a role-playing experience has been designed. In the game, a fictitious company, with headquarters in a virtual world, develop educational tools to be used within the world itself and also to be integrated with external services. A group of experienced senior software engineers joined a group of undergraduate students in each of the teams. A total of sixteen participants, from different countries and different cultural and educational backgrounds, participated in this study, and their interactions were recorded and analysed.
The experience was considered to provide the appropriate support for the application of the epistemic games theory and cognitive apprenticeship model. The environment was also considered appropriate, characterised by such attributes as *engaging*, *interactive*, and *fun*, qualities that new generations of students crave from educational settings nowadays (Prensky, 2001a).

This exploratory study followed the research design of a *case study.* The main datasets used were the text log files and the reflection work of the researcher, recorded during the virtual sessions and also the design and implementation of the artefact. The outcomes of the sessions were available to other participants through the use of web tools, giving the groups a sense of community (Holmes, Tangney, FitzGibbon, Savage, & Meehan, 2001), and helping to develop the *external design grammar* of the game, and the process of *critical learning* (Gee, 2007).

Table of Contents

Illustration Index

Abbreviations

DGBL	Digital Game-Based Learning
ICT	Information and Communications Technologies
IDE	Integrated Development Environment
IM	Instant Messaging
OpenSim	OpenSimulator
SL	Second Life ®
svn	Subversion (source control system)
TCD	Trinity College Dublin
VLE	Virtual Learning Environment

Chapter 1 Introduction

The area of interest in this project is *software engineering* as a **profession**. The main problems observed are the isolation in which teaching of programming subjects occurs in most institutions (Garlan et al., 1997), and the obstacles that junior software engineers face when joining the workplace (Machanick, 1998).

The proposed solution is a *different* approach to teaching programming, involving the idea of epistemic games (Shaffer, 2005), and implemented following the cognitive apprenticeship model (Collins et al., 1989). Epistemic games are characterised by being shaped around a professional practicum for the area in hand. They emphasise five different areas, skills, knowledge, identity, values and epistemology. Apprenticeship is characterised by the activities of modelling, scaffolding, fading and coaching. Both theories share many points in common and they stress the need to ***situate the learner in real tasks*** that are important in the real world, as well as ***reflective practice***.

The project aims to **situate learners** in a real software development environment and involve them in real world projects. This will provide skills and knowledge by using real professional tools. A sense of identity is fostered by treating the participants as software engineers, giving them responsibilities and mentoring them to mature always within their zone of proximal development (Vygotsky, 1978). The project also allows the chance to play different roles in different projects. The values of different business models are also embedded in the simulation.

This project has been developed in the form of a **case study:** Two fictitious companies were established within two different, but similar, virtual world grids, Second Life ® (SL) and OSGrid. Each team was formed by a mix of professional senior software engineers (developers and project managers), and undergraduate students from the school of computer science and statistics. There were 5 professionals and 9 students divided in 2 groups. The two teams followed different business models, one being an open source team and the other a proprietary code company. There were also 2 more people acting as clients of the proprietary company. Towards the end of the study, both teams were merged into a single open source community, with a total of 16 members. Both teams developed educational tools for virtual worlds. The virtual environment provides facilities to create objects with different shapes and forms that can also be linked to create bigger structures as buildings, vehicles, blackboards, etc. The objects can also be scripted and made interactive, with the possibility to communicate with external applications. The students worked along with the professionals in developing tools such as an avatar radar, a slide viewer and a pointer for the viewer.

The main goal of the project was to answer the following two research questions:

RQ1: *To what extent are the elements of the epistemic games theory supported by the game-based learning experience within the virtual world settings?*

RQ2: *To what extent does the artefact created within the virtual world setting allow for the application of the cognitive apprenticeship model through the interaction among students and professionals?*

The data collected from the sessions, added to reflection work from both the participants and the researcher, resulted in a positive support of the literature.

The rest of this document is divided in the following sections: Chapter 2 provides a literature review in the main topics of the study. Chapter 3 provides the design of the artefact that accompanies this study, based in the key areas coming from the literature. Chapter 4 discusses the methodology followed during this study, an exploratory research work in the form of a case study, and the implementation of the artefact. Chapter 5 provides the findings from the analysis of all the data sources in the project, opening a discussion to try and answer the research questions from the point of view of the data collected. Chapter 6 is a final discussion of the project, reflecting on the findings section, limitation of the work, and potential future work. Additional sections containing bibliography and appendices close this document.

Chapter 2 Literature review

This chapter provides a literature review in the areas under investigation, trying to focus on a possible solution to the problems at hand. This solution will inform the artefact to be designed and implemented as part of this exploratory research study.

2.1 Learning programming languages

Programming languages can be a very effective tool in the hands of experienced software engineers, but they are not an easy topic to master. There are many implications and problems to teaching programming languages, as it will be discussed in this section.

2.1.1 Programming education

In the programming teaching domain, emphasis is placed on syntax and functionality, instead of on abstract concepts and problem solving (Sherrell & Mills, 2008). Teaching a programming language involves more than syntax, and programming languages are not generally well suited for teaching purposes (Machanick, 1998). Knowledge about the domain at hand is difficult to achieve and also very specific to each concrete problem, but the acknowledgement of its existence is something that students are rarely confronted with. The problems of teaching programming concepts, specially the difficult abstractions used in some high level languages as in the *Object Oriented* paradigm, have been recorded in the literature for some time (Berg, Cline, & Girou, 1995). Efforts had been made to apply Bloom's Taxonomy in different ways, both within the view of general education, and also as a concrete application to the computer science field. The taxonomy is used mainly for assessment (Starr, Manaris, & Stalvey, 2008), instead of for computing courses description and design.

2.1.2 Abstract-first approach

An interesting application of the taxonomy was proposed by Machanick (Machanick, 1998). His view on teaching a programming language 'backwards' (Machanick, 2007), promotes an *abstraction-first* approach, justified by the need of building a base knowledge of readily available resources, before developing design skills. It is based on the assumption that presenting abstract concepts first should ease the transfer of advanced concepts, and avoid future problems in the workplace. He has also recorded the fact that junior developers must unlearn their habits and mental models of software development techniques in order to learn new ways, focused on *reuse*. As an outcome of traditional approaches, students focus their efforts in the implementation of the particular object they are working on, in terms of implementation. Best practice in software design warns against this very idea (Venners, 2005), promoting a design process focused on interfaces instead of implementation, which is consistent with Machanick's view.

2.1.3 Agents of change

The rapidly changing environment of technology stresses an education system where not only current technologies should be taught, but also training is needed to adapt to new technologies and trends. Students need to be provided with models, skills, and analytical techniques that will benefit their future career (Garlan et al., 1997). The key issues of a professional software engineering program are analysed in Garlan's paper, and contrasted with the different approach used in the MSE program at Carnegie Mellon, based in a mentored, long-term software project, developed for real external clients, as opposed to the observed failures in the traditional approach regarding problem solving skills and isolation in courses and material. The authors claim that this program produces engineers that can act as *agents of change* in the technological field.

2.1.4 Other challenges

There are other challenges for educators, including the type of audience, limited time for a prearranged set of course objectives, or a rigid curriculum (Ryoo, F. Fonseca, & Janzen, 2008). Ryoo and his colleagues have tried to cope with the obstacles described by using a problem-based learning approach as a pedagogical model, setting the assignments of creating games to promote engagement and creativity.

Many other problems regarding teaching and learning programming are recorded in the literature. Some of them are:

- Many students have learning problems due to the abstract nature of the subject.
- Students are affected by a lack of personal instruction.
- Novices are typically limited to surface knowledge of programs, approaching programs “line by line”. They cannot combine syntax and semantics into programs.
- The perception of teachers and students about learning is very different (Lahtinen, Ala-Mutka, & Jarvinen, 2005).

The problems of teaching programming are varied and complex. Traditional tuition focuses on syntax and semantics, isolating the students' view of the system. Some of the research to overcome these problems consists of abstracting the view of the students, and preparing them for the changing field of technology by making use of a valid context and personal tuition. This study will take into account all these possible solutions to create a learning experience.

2.2 Digital natives

The fact that most new students are *digital natives* (Prensky, 2001a), aggravates the problem of teaching programming. This term applies to the new generations born along with the main advances in digital technology during the latest decades of the last century and onwards. A whole new set of variables and lifestyle choices come into play in every aspect of life, being education one of the most affected systems. Motivation and engagement of digital natives might be only achieved within their own terms and using their own language. Visual appeal, entertainment, and technology, seem to be key values for the new generations. But as things go slow pace in academia, the students of today are not receiving what they crave, stuck in a system designed by instructors that had experienced a completely different lifestyle. According to Marc Prensky (Prensky, 2001b), the time dedicated to reading books halves the amount of time dedicated to playing video games. Students today watch television about double the time that they play video games. This indicates that their information sources are quite different to previous generations, with information travelling very fast and being received and scanned in the same fast way. This leads to multitasking and parallel processing, different thinking patterns, and also interactivity and the need for rapid feedback. The random access that the web offers takes priority over the *step by step* methods of the past. Networking, and the availability of technologies enabling remote collaboration, also shape this new communications world. Added to this, the rewarding mechanisms found in the games that natives are used to play, are difficult to match. Also the *reciprocal understanding* between interactors in a classroom (Järvelä, 1995) might be negatively affected. Another area that seems to be suffering the effects of this new lifestyle is *reflection* (Prensky, 2005). An educational environment where instructors and students receive, process, and reflect about information in such radical different ways, is bound to fail sooner or later. Added to all this, the diverse backgrounds, the ubiquitous presence of computing systems in society, and prior computing experience in each student case, also accentuates the problem (Pedroni & Meyer, 2006).

The needs of the new generations could be catered for inside a virtual world, making use of the capabilities of integration with other tools such as blogs, wikis, or community portals. This will be reflected in the design of the artefact for this study.

2.3 Evolution of VLEs

Virtual learning environments can be as simple as web-like portals, focusing on community work processes and easy access to resources, or as complex as 3D environments, being more suited for real-time cooperation, trying to mimic real life behaviours using online *personas* known as avatars. Their use can be beneficial in different aspects to the learning experience, such as in terms of content, communication and cooperation (Prasolova-Forland, Divitini, & Lindas, 2007). A study in the Portuguese university of *Trás-os-Montes e Alto Douro* (UTAD) provides a good example of the success on motivation and generating interest in the students during a Second Life ® based experience to teach programming (Esteves, Morgado, & B. Fonseca, 2007). Their exploratory study uses the virtual world to situate the learner in the proper environment.

In this project, the author will pursue the possibility of blending the more static systems with immersive environments.

2.4 Digital game-based learning

Game researchers have been caught unaware after years of promoting the benefits of DGBL(Van Eck, 2006). Games can provide a solution to some of the challenges of teaching digital natives, and most gaming studies have focused on demonstrating that fact. A shift is needed in gaming research covering topics as:

- Research explaining why DGBL is engaging and effective
- Practical guidance (when, with whom, and under what conditions) games can be integrated into the learning process to maximise their learning potential (Van Eck, 2006).

A balance is needed in game design between learning theories and gameplay. Many have recorded failed attempts on leading design by educational principles. Seymour Papert has coined the term “Shavian reversals” (Papert, 1998), to reference most of the game titles classified under the “edutainment” tag, a mating of education and entertainment. In his words, offspring that keep the bad features of each parent and lose the good ones, or boring games and 'drill-and-kill' learning. Games provide a mechanism for situated cognition that goes beyond the real world. They also create intrinsic motivation but they do not appeal to everyone. Making digital games compulsory in the curriculum might mean losing some of the engaging properties (Squire, 2005). Attempts can be found in the literature to provide models to design and evaluate educational computer games. The *experiential gaming model* provides a link between educational theory and game design (Kiili, 2005).

Some video games incorporate good learning principles. There is a whole body of literature about this topic, lead by the seminal work of James Paul Gee, W*hat video games have to teach us about learning and literacy* (Gee, 2007). Gamers are willing to put many hours into video games, and they would not accept easy and short ones. To play a game, the gamer has to be able to learn it, so the principles have to be within the game itself. But gaming is quite different to schooling. Learning facts are far less important than living an embodied experience, acquiring knowledge and being able to apply it to solve the problems at hand. In his book, Gee identifies up to 36 learning principles found in *good* video games, of which the author of this study has chosen the subset that better fit the purposes of this project, for discussion:
As described in the ***Active, Critical Learning principle*** passive learning is to be avoided, encouraging participation, boosting reflection, and providing the basis for critical learning. Virtual worlds and other game-like media provide an environment where risks can be taken with lowered real world consequences, and a ground where multiple real-world identities can be reflected upon (***Psychosocial moratorium* and *Identity Principles***). There are three main identities involved in playing games, the *real identity* as a person, the *multiple virtual identities* that the game allows the real person to experiment with, and the projection of the real into the virtual, denominated *projective identity*. Taking on different identities makes possible to see

situations from different points of view.

Very little input in the virtual domain provides a magnified amount of output. In the case of virtual worlds as used in this study, the fast feedback that the programming of objects provides, gives a real boost to the participants. This is reflected in Gee's ***Amplification of output principl***e.

Cognitive researchers ask how something so hard as a game can be taken so seriously. There are a number of clues leading to rewards inside the game. But a game should always reward at an appropriate level (***Achievement Principle***) or it would not be perceived as fair. If rewards are too easy to get, the gamer will lose interest. Too much effort leading to a poor reward would have the same effect.

Learning is generally considered as a cycle of probing, reflecting, forming hypothesis and re-probing (Kolb, 1984). Several choices and paths should be given to solve a problem, and the ability to probe and reflect in and on action. This is intrinsic to every good game, and it has been recorded as the ***Multiple routes Principle*** and ***Probing Principle***, in Gee's work.

Situated meaning through embodied experiences are more likely to be worked on, and for longer. The multimodal capabilities of a virtual world are a good ground for this, and the possibilities to integrate with other communication tools makes it even more appealing (S***ituated Meaning***, ***Text***, and ***multimodal Principles***).

Following the ***Subset*** and ***Incremental principles***, as well as the ***Discovery Principle*** would be paramount for any game design to succeed. All learning should start in a simplified sub-domain of the real domain, where earlier generalisations would be consistent with the ongoing work.

The ***affinity group*** and ***insider principles*** connect gaming theories with broader sociology and pedagogical theories as *communities of practise* and social *constructivism*. Learners and gamers constitute groups with shared goals and practices, where they can both be consumers and producers, being able to customise the learning experience.

All these principles will have an implication in the design of the learning experience for this study.

2.5 Epistemic Games

Sometimes defined as the *study of knowledge*, or the *study of what it means to know something*, epistemology marks the rules by which it is decided what is true or not in a certain domain. An epistemic game can be defined as a game that has been shaped around a "*practicum*" of professional practise, the epistemic frame of that community (Shaffer, 2006). As any game, it is an activity followed by a series of individuals taking on a role, and behaving according to the rules associated with such role. With such a definition, almost any profession or social activity could be considered as a game. So the domain, and more concretely its epistemology, is an important factor of every activity.

The two main ideas in this theory are *authenticity* and *professionalism* (Shaffer, 2005). The author aims to link games, simulations, and professional practise. Shaffer proposes an authentic setting, with real tools and tasks that are important in the participants' real world, as a recipe to engage the learner. There are multiple contrasted theories involved in such a complex context definition (Dewey, Schon, Gee etc.), and creating an environment like that one is not an easy task. This is where simulations and computers can help with the construction of such settings.

There are five key aspects to epistemic games, knowledge, skills, identity, values, and epistemology. They are all bound together, drawing from the situated and embodied experience created by the game, and the fact that it has been established following authentic and professional principles.

Construction of knowledge can be aided by the environment in different ways, be it by fostering collaboration, easing the access to content, etc. In the same manner, the possibilities of using simulations of tools will allow participants to gain new skills. Identity experimentation is a good way to relate and get involved with the "*reality*" of the experience. And framing all this, the epistemology of the domain at hand will shape the interactions and behaviours of the participants, within the values of the profession or subgroup of it.

The community targeted in this project is software engineering, and the design of the learning experience will try to create the proper epistemic frame within the virtual environment setting.

2.6 Cognitive Apprenticeship

Considered by many as *the natural way to learn*, apprenticeship has mostly been replaced by formal schooling except in certain areas as children's learning of language, in some aspects of graduate education, and in on-job training (Collins, John Seely Brown, & Holum, 1991). Cognitive Apprenticeship is a different model of instruction that mixes apprenticeship with some elements of modern schooling (Collins et al., 1989). The main processes in traditional apprenticeship have been characterised as modelling, scaffolding, fading, and coaching. *Modelling* refers to the fact that the master demonstrates how to do the work affecting each part of the task while the apprentice observes. When the apprentice starts working in a task, the master will support this work through *scaffolding*. This support should fade away to let the student evolve and not be always dependant of a master. That is what it's known as *fading*, drifting the responsibility towards the student side. *Coaching* occurs throughout the whole experience, in which after all, the students are still in the hands of the master, following his advice, getting support and the much needed feedback and encouragement.

There is an interesting application of the modelling activity in the book *Mathematical Problem Solving* (Schoenfeld, 1985) where the author models the whole process of solving a mathematics problem that he has not encountered before. In the process, the use of different mathematics rules and control strategies can be observed by the student, as opposed to most school books that show only the right solution for the problem at hand. The main goal of the activity is to show the students that the process might fail a few times before reaching a good solution.

There are two important factors in apprenticeship, *observation* and the *social context* of the task. Observation is paramount in aiding learners to develop a conceptual model of the task at hand, aiding in organising the initial attempts to execute a skill, the interpretation of feedback, and allowing the student to be more independent (Lave & Wenger, 1991). It also allows for the development of self-monitoring and correction skills, and integrating skills and conceptual knowledge. Regarding the social context in which traditional apprenticeship takes place, according to Collins, apprentices derive many cognitively characteristics from the subculture in which the practice is embedded. Having many members involved in targeting the same skills gives the apprentice not only access to different models of expertise, but also shows different ways to carry out the same task and the possibility to observe different degrees of skill, reinforcing the concept of learning in an incremental way.

The three main differences between traditional and cognitive apprenticeship are defined as follows:

> *In order to translate the model of traditional apprenticeship to cognitive apprenticeship, teachers need to:*
>
> - *identify the processes of the task and make them visible to students;*
> - *situate abstract tasks in authentic contexts, so that students understand the relevance of the work; and*
> - *vary the diversity of situations and articulate the common aspects so that students can transfer what they learn* (Collins et al., 1991).

Identifying the process and making it visible to the student is paramount, taking into account that the task in hand is completely different in nature. It is not a matter of crafting any more but a process of making the thinking visible. The second point situates the learner in an authentic context as happens in the traditional model, and should be mimicked in the new model. The learners are more motivated if they understand the task in hand and see an utility to it in the real world. Transfer of skills is rarely sought in the traditional model of apprenticeship. In the new model the goal is to make the student be able to generalise the skill and identify other situations where it can be applicable.

The cognitive apprenticeship model is suited to teach complex tasks but would not fit every aspect of teaching. It also challenges other preconceptions as the teacher assuming the part of *expert* all the time, with the ultimate goal of encouraging the student to become an expert. Järvelä concludes that a technologically rich learning environment facilitates learning in social interaction when this is based in the principles of the model (Järvelä, 1995). It does not only promote higher order thinking skills, but also allows to start a transition from a traditional teacher lead class, to a joint student and teacher problem solving oriented sessions.
The artefact designed for this study will try to support the incorporation of the ideas of this theory.

2.7 Communal constructivism

The use of ICT opens new views to communication in a group of learners. The main focus of communal constructivism (Holmes et al., 2001), relates how ICT can be used to allow students to leave an imprint in the course itself, constructing knowledge within and for the community. Characteristics of this theory as group work, peer tutoring and mentoring, and project-based learning, will have an implication in the design of the artefact.

2.8 Summary

Teaching programming is reported in the literature as an isolated practice, and along with the many other problems described in this chapter, it is a challenging subject to teach. The use of an immersive environment could provide new solutions, not only to these problems, but also providing a ground for a situated and embodied learning experience through a simulation. Introducing good learning principles and recreating the epistemic frame of the software engineering community is the main goal of this study, aided by the ideas of the cognitive apprenticeship model. The pursuit of such a goal has been documented in the design and methodology chapters that follow.

Chapter 3 Design

This study is based on two main ideas coming from the literature. The first is the use of digital games for learning, more concretely the use of the epistemic games theory (Shaffer D.W., 2006). The second is the use of ideas taken from the cognitive apprenticeship model (Collins, Brown, & Newman, 1989) within the gaming environment. A series of secondary topics, as the challenges of teaching programming, or social constructivism, have also been considered during the design of the artefact. The following sections will provide a study of the main points from these topics, focusing on the implications that these issues may have in the design of an educational artefact. This exploratory research study will try to investigate whether the creation of a situated learning experience within a virtual world is appropriate to tackle issues associated with professional education in the field of software engineering.

3.1 The artefact

The definition of game as *an activity followed by a series of individuals taking on a role, and behaving by the rules of such role*, was the heart of the experience. In the game created for this study, the participants can take on the different roles in the simulation of a software development company. During the game they will write code, documentation, install servers, or any other actions needed as part of their role. To this end, the company will have its headquarters in a virtual world, and all its activities and projects will be focused on educational tools for the virtual environment itself.

3.1.1 Architecture of the artefact

The experience took place in two immersive virtual world grids simultaneously, the commercial solution SL and the open source platform OpenSim, connected to the OSGrid grid of servers. Additional pieces of technology were used throughout the experience.

The following diagram depicts some of them (explanation following):

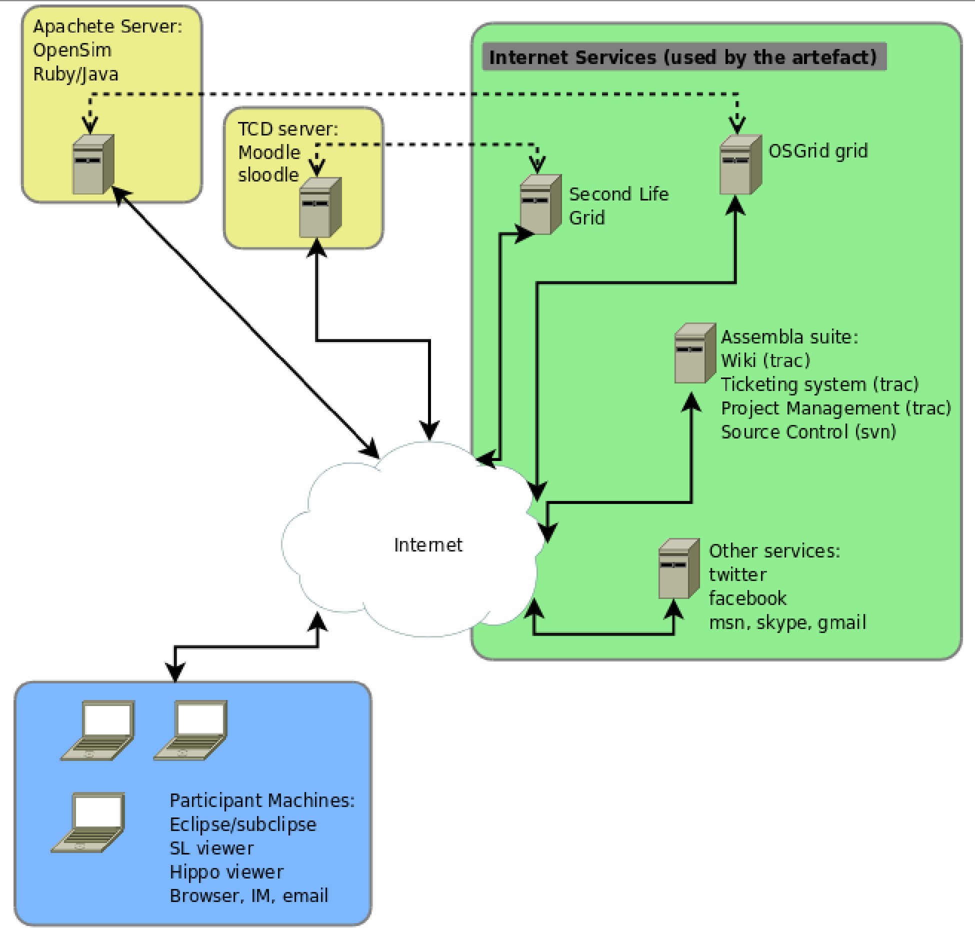

Figure 3.1: Architecture of the artefact

Inside the blue box, corresponding to the participant machines, a series of applications were used. As a minimum, the second life viewer to connect to the SL grid was needed, and although most of the participants used the set of tools exposed in the diagram, the choice was left open for them to decide the tools of their preference (programming IDE, virtual world viewer, source control client, IM tools, etc.). The two yellow boxes depict servers managed by the author. The TCD server is a college operated machine that hosts a Moodle site with a Sloodle module to interact with virtual worlds. The Apachete server is a virtual server that the author rents from a hosting company. The ruby and Java servers developed as part of this study were hosted in this machine, to be accessed from the virtual worlds. The dashed lines in the diagram show the dependencies of the services in these servers and the grids in the Internet. The green box wraps the Internet services used, two virtual world grids, the *assembla* suite of tools that provides all the functionality needed to manage a software team (assembla.com), and other services, mainly used for communication purposes, as twitter or gmail. The different parts of the system are all connected together by the Internet.

3.1.2 Configuration of the artefact

There were a number of tasks that the author needed to fulfil to get the environment ready for the study. Different servers are involved in the artefact, the OpenSimulator software being the most complex one, having to be configured to join a public grid. The installation involved other different services, as a database engine, a web server with support for ruby on rails and Java applications deployment, and all the configuration needed for such services. The moodle and sloodle modules also had to be configured. Regarding the assembla suite, the author needed to create and configure spaces for the different groups. All this was documented in the wiki before the participants joined the teams, including a selection of possible tools for the participants machines. Creation of other accounts as twitter and facebook, and connections through the assembla site were established. Inside the virtual worlds, the creation of work areas and the example projects were also needed. The following screen shot shows the appearance of the offices and community centre created in one of the grids.

Figure 3.2: Offices and social space in the grid.

3.2 Design considerations

There different topics from the literature affecting the design of the artefact will be discussed in this section.

3.2.1 Digital game-based learning

Literature concludes that games might be of value in education depending on how they are used, and they can provide a mechanism for situated cognition. Regarding game design, there is a lack of agreement in the available research about who should lead the design of educational games, professional game developers with no knowledge of education theories, or academics.

Based on the above ideas, this particular project utilises the affordances of modern virtual worlds (creation and scripting of objects, collaboration, and communication) to cater for that situated context, that will be appropriate based on the profiles of the target audience, computer science and engineering professionals and undergraduates. Regarding game design, in the authors opinion a balance is needed. That is why this project allows for the participants to be a creative part of the experience as consumers, but also as producers, by creating an authentic and real simulation, using the tools and values of the domain along with role-play.

3.2.2 Learning principles in good games

The following subset of principles refers to the work of James Paul Gee (2007), as discussed in the literature review chapter.

Following the *multimodal principle*, the artefact makes use of different visual and communication capabilities of the virtual worlds through scripted and interactive objects, but also allows for communication through different external tools, as the wiki, tickets or IM tools.

The application of the *discovery principle* involved participants in two ways, by making the artefact interactive, allowing users to inspect the objects themselves, and by organising the sessions as hands-on participative experiences.

The *Concentrated*, *Subset*, and *incremental principles* imply that participants should learn in a subset of the whole topic, showing paths and patterns to deal with the rest of the experience. The first stage of the study was guided by the author and other professionals involved, designed as training sessions, each building on top of the previous project and increasing in difficulty.

Any role-playing experience is a good context to experiment with identities (*identity principle*). In this study, students were treated as software engineers from day one, and were given the opportunity to play other roles during the game, being in charge of sessions or projects.

The *Psychosocial moratorium* principle implies that risks can be taken with limited implications in the real world. That was part of the artefact too, affecting positively in tasks as meeting clients confidently.

The *achievement principle* is a fundamental part of every game. Literature concludes that participants have to be rewarded at the appropriate level. In this study, the first reward was given to the first person to register valuable information in the wiki, one of the external tools. This also allowed to reinforce the importance of using the whole set of tools available. Other smaller rewards happened after certain tasks were finished.

As all the subjects are participants in the same community, the *semiotic domain*, *affinity group* and *insider* principles are part of the implementation per se.

3.2.3 Epistemic games

Within gaming literature, the epistemic games theory (Shaffer, D.W., 2006) was the one that inspired the artefact for this study. A proper setting should create the epistemic frame around the social community at hand. The main characteristics of the epistemic game theory could be evaluated in such a setting, that should allow for construction of knowledge, skills development, identity experimentation, application of values, and framing of the epistemology of the community.

Based on the above theory the epistemic frame of the software development community was the one chosen for this artefact. To this end, the fictitious companies were created, using real tools that allow for skills development (source control, programming environments, etc.), creation of knowledge (wiki, collaborative sessions), application of values regarding business models, and experimentation with identities through role playing.

3.2.4 Cognitive apprenticeship

Cognitive Apprenticeship is a model of instruction that goes back to apprenticeship but incorporates elements of schooling. This should inform the way in which the sessions are held. Modelling, scaffolding, fading, and coaching should be facilitated by the artefact throughout the experience, and responsibilities should be shifted towards the students. More than one model of expertise is essential to apprenticeship.

To this end, the artefact provides a model for a more personal manner of instruction and student/ professional interaction. The creation of meeting areas and open spaces for hands-on sessions eased the interaction. More than one model of expertise is reflected by the work of the professionals. Also Schoenfeld (1995) modelling activities were inserted in the experience by allowing the students to bring their own problems and go through all the process with them, to make the thinking process visible.

The integration of external tools allows for resource sharing and documentation of links and procedures, allowing for scaffolding and fading, and pushing the student to explore the whole setting, fostering problem solving, critical thinking, reflection, and social interaction.

Shifting responsibility to the students is another integral part of the implementation, delegating tasks as soon as possible, giving students not only responsibility, but the possibility to experiment with identities.

3.2.5 Other relevant factors

The following paragraphs discuss other ideas that have impacted the design of the artefact. These include challenges of teaching programming, teaching digital natives, and communal constructivism.

Traditional approaches to teaching programming reduce the *visibility* of the piece of code the student is working on with respect to the whole system. There is a need to introduce abstraction in this *vision*. Literature also reports the isolation in which programming courses are held in most institutions. These issues are catered for in the artefact by immersing the participant in a real and authentic environment, shifting visibility and abstraction to a wider angle. Furthermore, the social possibilities of such environment can ease the negative effects created by isolation issues.

According to the literature, digital natives bring a new set of variables and lifestyle choices into play into education. The artefact, as an educational tool, *speaks* the language of the digital natives, providing multiple tools that properly used can enhance learning. Availability of the resources are not confined to a classroom and a specified and fixed time. Instructor availability does not need to be synchronous all the time. Tasks were designed with parallel processing an interactivity in mind. Reflection practices are promoted both inside the virtual world and also making use of external tools.

Following the ideas of the communal constructivism theory, this project aims to foster learning as a social and facilitated, rather than individual and taught experience. The knowledge should *stay* and enrich the experience for new students to come. Learners should also be listened to. To this end, the physical virtual spaces in the artefact never resembled a classroom, using open spaces for social interaction. The use of external tools and social networking sites is intended to create comfort within the group members. The space allows for personal instruction, peer mentoring and project based learning. Students are welcome to bring their own projects, and also to participate in the activities that they find more appropriate.

3.3 Summary

The construction of the experience in a virtual world was the main idea of the project in order to enhance the learning experience. The resultant artefact is a quite complex set of tools and services used together. The novelty of the environment provides the perfect space for a situated experience, also catering for the needs of the new generations of learners. The social and communication aspects are well represented, especially with the possibility to extend and integrate other ICT tools. This was paramount to create an *authentic* and *professional* role-playing experience, in other words, an epistemic game. The possibilities of recording the interactions, not only in the virtual settings but also within the external tools, provided the perfect ally for a researcher in search of transcripts and other datasets to analyse. The next chapter will discuss the methodology of the research and the implementation of all the ideas discussed in here, within the particular experience created for this study.

Chapter 4 Methodology

4.1 Research Design

The characteristics of this project make it suitable to develop a *Qualitative* research exercise, in the form of a **case study**. This chapter will provide an outline and justification of the chosen research design, along with the data sets identified as appropriate for this study. The main datasets are the log files of the sessions, the subjective informed observations and reflection work from the author, the records of the external tools used during the project, pre-questionnaires, and individual interviews.

4.1.1 Case Study

The definition of case study is not and easy one to give (Bassey, 1999). A number of authors have proposed different definitions in recent years. Robert Yin characterised a case study as "*an empirical enquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident*" (Yin, 1994). Within the context of educational research, a case study has been defined as "c*ritical enquiry aimed at informing educational judgements and decision in order to improve educational action*" (Bassey, 1999).

4.1.2 Appropriateness of a case study

This project is based on the *observation of a group of people*, characterised as an Ethnographic Study in qualitative research terms. A certain number of characteristics or restrictions in this project, probably **time** being the most important of them, makes it a good candidate for an *Instrumental Case Study* as defined by (Creswell, 2007). Other characteristics of the project are the focus on a concrete activity with a limited number of participants, and the look for insight in the demonstration of the validity of specific issues, in this case the main research questions as stated in the next section of this document.

4.2 Research questions

The research questions for this study have been formulated as follows:

RQ1: **To what extent are the elements of the epistemic games theory supported by the game-based learning experience within the virtual world settings?**

RQ2: **To what extent does the artefact created within the virtual world setting allow for the application of the cognitive apprenticeship model through the interaction among students and professionals?**

The questions are based on the needs observed in the area of interest, focusing on two main problems, the isolation in which most programming teaching is carried on, and the gap between college practices and the workplace. Some of the problems that junior engineers face when joining the workforce may be fostered by teaching practices (Garlan et al., 1997; Lahtinen et al., 2005; Machanick, 1998).

4.3 Ethical concerns

Due to the nature of this dissertation, and its short duration in time, a revision and approval from the Review Board of Trinity College Dublin was not deemed necessary. The participants were all informed that the chat conversations were being recorded, and in some cases the participants recorded the sessions themselves to send to the researcher after the session. This can be verified in the log files. Other data collection tools as the wiki are open to the Internet, and this was previously agreed with the participants. No anonymising of data was required as the participants were never asked for their real life identities. The participants were at all time aware of what the data produced was going to be used for, and the public disclosure of that data would mean a very low risk. The three ethical principles of respect for persons, beneficence, and justice (Creswell, 2007), are safeguarded in this study.

4.4 Research biases

The researcher has no particular affinity for video games or virtual worlds. It was only when he realised the potential that this kind of media has in educational settings that he decided to investigate the field. The author has a broader belief in the potential on such an environment to enhance research in education. Being a computer scientist, the researcher might feel more comfortable in this environment than potential participants would, but in this case all the participants are related to the computer science area.

4.5 Rigour and validity

Due to the characteristics of this short project, no *formal* methods have been used to ensure rigour and validity. The automated mechanisms applied to collect data was deemed enough for the nature of the study at hand. Regarding validity a relatively similar approach to *Content Validity* (Creswell, 2007) has been followed naturally in this study, as some of the participants are experts in the field, and they have been validating and participating in answering the questions of this research project. As data has been collected from different sources, triangulation provides a certain amount of reliability and validity.

4.6 Datasets

This section lists the datasets gathered during the implementation of this study.

4.6.1 Logs from sessions

The chosen immersive virtual worlds provide several ways of communication as voice or text chat. Text logs can be automatically stored. The complete texts of every session with more than two people were recorded by the author or one of the participants. Other tools such as skype or msn were also used during the project, and the activity was recorded in the same way as described above. The benefits of this dataset towards answering the research questions come from the nature of the sessions and the collaborative activity itself. The author has tried to get as much information from the sessions as possible, following sometimes a disguised semi-structured interview approach.

4.6.2 Reflection work

The author recorded all the work during the project in a diary, in the form of a text document. This piece of data was intended primarily as an administrative tool, but it also provides insight about the environment, the perceptions of all the participants, and the reflection work of the author.

4.6.3 External tools

The author received several emails sent by the participants, recording their thoughts about the activities. Other *off-world* communication used a variety of external tools to the virtual environment. A group in the social network '*facebook*' was created to this end, as was an account in the social network '*twitter*'. The external tools provided by the Assembla suite proved very valuable when communication had to be off-world. The wiki and the ticketing system, where all the discussions about the projects were held, and tasks were created and assigned to the participants, recorded the activity in the projects. The source control system can also be used as a tracking mechanism when data needs to be analysed.

4.6.4 Pre-questionnaires and interviews

A pre-questionnaire was organised to gather information about the initial status of the participants and their experience, also recording facts about their level of knowledge and skills with certain tools. Due to the amount of data provided by the other datasources, only two semi-structured individual interviews were held at the end of the data gathering stage.

4.7 Implementation

This section describes a series of concrete aspects of the project as they happened in the context of this study. The architecture of the system has already been exposed in the design chapter.

4.7.1 Participants

There were three main subsets of participants in the project, classified as *students*, *professionals*, and *clients*. The student group, a total number of 9 undergraduate students in the school of computer science and statistics at TCD, voluntarily participated in the study, playing the part of *software engineers*. The professional group, a total of 5 ex-colleagues of the author of this study, played the part of t*echnical leads* and p*roject managers*. The clients group was formed by two members of the computer science department at TCD who are actively involved in virtual worlds and pedagogy research. The different cultural and educational backgrounds of the sixteen participants should be noted, coming from at least eight different countries.

4.7.2 The fictitious companies

In first instance, the students and professionals were divided into two companies, the open source managed community ***Apachete Foundation***, and the commercially orientated company ***MancoSoft Inc.***. This was decided with a view to better management practices and to experiment with different organisational cultures. Due to organisational theory ultimately falling outside the scope of the present project, the two companies were merged into the *Apachete* community towards the end of the study.

4.7.3 The sessions

The experience consisted in a series of sessions inside the virtual worlds (orientation, basics, and team work sessions), and multiple *offworld* asynchronous discussions through the wiki, project management tool, and IM services.

The number of orientation sessions was seven, and all the students participated in groups of two or individually. The length was less than one hour in all the cases and produced logs of about

17.000 words. The professional and client groups did not need orientation.

The were three 'basics' sessions that were attended by all the students and three of the professionals. The length was over two hours for each session, with an average of 6.000 words in the logs and 4 participants in each of them. After this, all the participants were ready to start with some 'real' work. With a blank canvas of two air platforms in SL, and a whole island in OSGrid, the participants were given freedom to build and start working in the initial projects. From this point on, 15 sessions were organised by the author, and an additional 2 sessions were organised by other participants, accounting for an average of 6.000 words more. There were a few more spontaneous meetings, but their activity was not logged, hence not used for analysis purposes. The final two interviews held through IM amount for an average of 2.000 words in text logs and about 20 minutes long each. All the asynchronous communication was recorded through the external tools.

4.7.4 The projects

The initial four projects were designed with increased complexity in mind for training purposes.

The first project was an ***avatar radar*** that shows the names of all the avatars in a distance of 96 meters. The source code of this project was facilitated to the students through the source control tool. They were challenged by the use of the external tool, and asked to create objects to host the given script. The following picture shows some of the objects proposed to host the radar script:

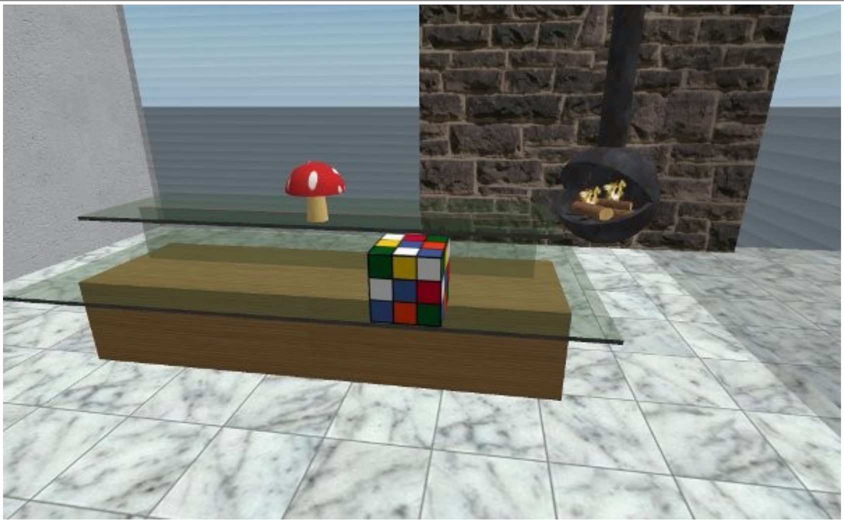

Figure 4.1: Avatar radar in the grid

The second project consisted in the development of a ***slide viewer*** similar to a *powerpoint* presentation but to be used within the virtual worlds. The students were asked to do some research on the topic and to add all their findings to the team's wiki. The picture shows works during a session dedicated to this project:

Figure 4.2: Slide viewer building session

After that it seemed a good idea to also have a ***pointer*** to use with it. The work took place in a similar way to the previous project.

The last project, the ***attendance radar***, extends the avatar radar by sending the information to a server in the internet. The complexity of this project supposed a real collaboration among professionals and students.

The virtual environment was very well suited as a platform for educational purposes. It provided a great vehicle to the implementation of the case study, being the use of external tools paramount to the development of the project. The next chapter will detail the outcomes of the analysis of the datasets previously described.

Chapter 5 Findings

The methodology chapter provided a description of the methods and deployment of the artefact in a real life environment, as informed by the design and conditions of this study. The implementation of the project with participants provided the author with a set of data. The current chapter describes a series of facts taken from the analysis of the recorded data, in an attempt to provide evidence that help to answer the research questions.

5.1 Findings

This section has been structured as follows; a general observation is given first, supported by facts taken from the data sources, triangulating among different sets when possible. An argument follows trying to show the contribution to answering the research question.

The following three subsections correspond to the analysis of epistemic games (first research question), cognitive apprenticeship (second question), and unexpected results and other observations. There is generally a very fine line between most of the concepts explained in the following subsections. Boundaries among elements are blurred. One example from the logs can show a different set of elements. In each case, even if the same example is used, the particular connotations for the element at hand were pointed out. As the some of the data sources were recorded live during the sessions, the typographical errors or use of slang in the text have not been corrected.

5.1.1 Epistemic games

The epistemic games theory (Shaffer, 2006), introduces a series of concepts in order to create a game-based environment able to reproduce in a “real” and “authentic” manner the processes by which members of a professional practice group develop the skills, knowledge, identity, values and epistemology of their community. Each of the following subsections will target the development of one of these elements within the artefact created for this study, and the close relation among all of them.

5.1.1.1 Skills

It was noted that participants gained a number of skills related to the profession. Important tools in any software setting, such as wikis, source control, or ticketing systems, were used throughout the project.

Source control

Source control is one of the most important tools in any software organisation and its use is a valuable skill that is generally only gained when joining the workplace. In this study, the version control skill and process was only known to three students according to the pre-questionnaires (appendix), but as it can be seen in the following extract of a session, it is used in a quite familiar way.

> Pixy: can someone send me the server script again? lost it [...]
> NewkRichard: bu u can get it on the svn Pixy
> NewkRichard: http://my-trac.assembla.com/apachetes/browser/AttendanceWS/trunk/client/collisionradar.lsl

In this case the source control application used is the *Subversion* system, generally abbreviated as '*svn*'. The example shows familiarity with its functioning, and also the use of domain specific terms and specialised language, illustrating good knowledge in the field. One of the students already using source control, stated the following in an interview:

> josmas: oh, one questoin, u had used svn before this project, hadn't you?
> Richaeeee: yeah, and i am using it actively for a project in college also
> josmas: ok, cool, i just wanted to put it as a skill gained, but i cannot use you for that
> Richaeeee: and i must say my experience in this project made me the "sole coordinator" in the svn part of our project:S […]
> josmas: u use it in college coz u want or coz they asked u to?
> Richaeeee: they said it would be suggested, bu its really useful, we are using it constantly, using code.google.com
> Richaeeee: bu we shud hav used assembla

The student felt confident to go a bit further in the use of the tool, and not only use it in a passive way, but actively coordinating other class mates in class assignments, out of this study. As seen in the fragment of the interview, the use of source control was suggested by the lecturer, but no infrastructure was provided, hence the use of a Google free service. This is a skill that will stay with the students for the rest of their lives, being very useful in any team work setting. The fact that the students are using a suggested tool '*constantly*' and qualifying it of '*very useful*' demonstrates that point.

Ticket systems and wikis

As can be contrasted in the pre-questionnaires, only three of the students had used a ticketing system before, and only one of them was not used to wiki technology. During this study, all of them were able to follow the phases of the ticketing process within the project. Both tickets and wikis share many functions in common, as the editing and update processes, formatting, etc. The following screenshot shows they use of tickets during the study. The tasks were developed and coordinated using this functionality.

Slide Viewer

Ticket	Summary	Component	Status	Resolution	Version	Type	Priority	Owner	Modified
#5	slide viewer research	Research	new	None	1.0	task	high	rawb	03/30/09
#9	Trac Permissions	Team work improvements	closed	fixed	1.0	task	high	martaK	03/25/09

Pointer

Ticket	Summary	Component	Status	Resolution	Version	Type	Priority	Owner	Modified
#2	Research on Pointer tools in virtual worlds	Research	assigned	None	1.0	task	high	francy	04/07/09

Avatar Radar

Ticket	Summary	Component	Status	Resolution	Version	Type	Priority	Owner	Modified
#4	Avatar radar look and feel	Building	new	None	1.0	task	highest	rawb	03/30/09
#6	Avatar radar in OSGrid	Scripting	assigned	None	1.0	task	high	francy	03/31/09
#8	Trip to OSGrid	Team work improvements	closed	wontfix	1.0	task	high	josmasflores	03/30/09
#7	Trac communications	Team work improvements	closed	wontfix	1.0	task	highest	martaK	03/22/09
#1	Creation of a mailing list for communication purposes	Team work improvements	closed	fixed	1.0	task	high	brokendwarf	03/19/09

Figure 5.1: Tickets system

The picture shows different tasks in the form of tickets, assigned to different people, being students or professionals. The students used the tool confidently, having almost no problems understanding its functionality. In a similar fashion, The following screenshot shows how the tracking of activities and sharing of resources was done through the tickets functionality, that also links directly to the wiki. The image show an extract of a conversation a professional (MartaK, project manager), with a student (Francy, software engineer), where they exchange information and also the functionality of the ticketing system and wiki.

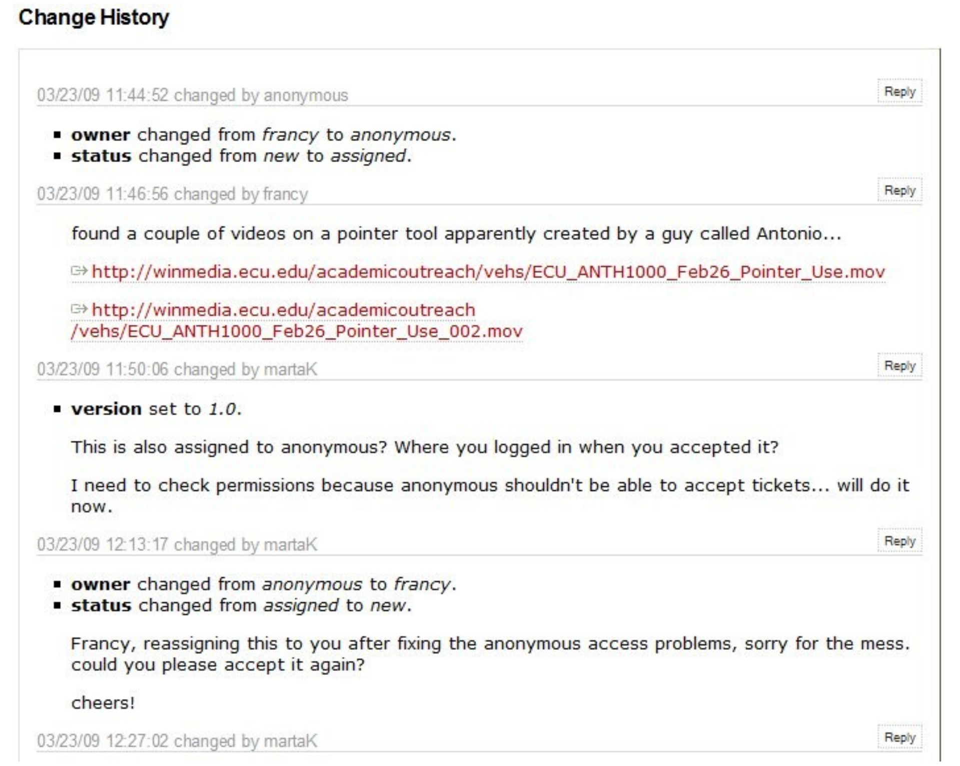

Figure 5.2: Tracking activities using tickets

Other Skills

In a very similar way, support for the claim of students gaining or expanding other skills can be found in the data sets. As an example, seven of the students had never used the ruby programming language before, but they were able to interact with a server written in ruby from inside the virtual world. The same happened with a server written in the Java language by one of the professionals. Seven students had never heard of the state machine programming paradigm. All the code produced for the virtual world was laid out in that fashion, giving in this case not only to the students, but also to the professionals a new skill, and a new way to look at the programming field.

All the examples exposed show how the skills element of the epistemic games theory is supported by the artefact. In the case of source control, it was a topic that occupied multiple sessions, and some of the outcomes of them were documented in the wiki by the students (electronic appendices). Source control is a quite complex system that needs an accompanying certain amount of knowledge (client-server architecture, IDEs and plugins, etc.), which is the topic of the next section.

5.1.1.2 Knowledge

Any skills gained will always go hand in hand with new knowledge.

States machine paradigm

Connecting to a fact stated in the previous section, to create code for state machines, the participants need to gain the knowledge associated to the states machine paradigm. If they did not have it before, as recorded in the questionnaires, they have probably gained it during one of the sessions as evidenced by the fact that they were able to create code within the artefact settings. The concepts of the state machines theory were exposed in the scripting basics sessions, and naturally reinforced in any later coding session as it is an integral part of the programming language used. They were also documented in the wiki previous to the session, and the students were asked to read about it beforehand. The professional focused on showing the differences between the programming that the students are used to produce in college, and the new approach at hand.

The following is an extract from one of these session, where 'You' is the professional, and all the other avatars in the conversation are students:

> You: i'll give you 2 scripts and you will tell me whats the difference, ok? u can create 2 boxes and test them if you want [...]
> NewkRichard: one uses states and the other uses a if and else [...]
> You: ok guys, what'#s better? states of if/else? [...]
> NewkRichard: i presume states as if and else would become impossible if there was many states? […]
> You: good answer newK
> You: plus LSL was conceived as a state machine [...]
> You: guys, u all get the difference between states and if/else?
> NewkRichard: yip
> Rawb: yeah
> Vinnie: yeah
> Pixy: yes ti see the difference between them

The concepts are worked on during the practical session, and they have to be clear to all the students if they want to do any work inside the virtual world. It is factual knowledge that was deemed important enough to be reinforced by a practical session. The artefact allows for this to happen.

From factual to procedural knowledge

During the sessions, there was a natural transition from factual knowledge to the use of the facts to develop problem solving skills and procedural knowledge. Connecting to the previous exposed discussion about the source control skills, the conversation finished as follows:

> josmas: u use it[source control] in college coz u want or coz they asked u to?
> Richaeeee: they said it would be suggested, bu its really useful, we are using it constantly, using code.google.com
> Richaeeee: bu we shud hav used assembla

There is an important meaning to this piece of information, as the student has gained enough confidence and knowledge to be able, not only to use the tool at an advanced level (as discussed before), but also to evaluate the piece of software used in his college work (code.google.com) against the one used in this study (assembla.com), to the point that he can show a preference for the latter. This preference can be based in many factors, as ease of administration, ease of use, reliability or any other attribute of the system, which is not as important for this study as the fact that the student has gained enough knowledge to be able to make a comparison.

Expertise

As in the previous example, a more specific kind of knowledge, at a deeper level, can be observed in the last few sessions, when students start building on facts that were dealt with in previous occasions. The professional BDS is explaining how the client that the students are developing will interact with the external server, when the student NewkRichard asks a question related to the code:

> [2009/04/04]
> BDS: and then it will record events when people step on it, and step off
> NewkRichard: ok so the person will have to be "in the object"
> NewkRichard: and u no the TODO thing
> NewkRichard: is that lik jus a simple loop, or is it more complex that we hav to have some sort of way of knowing if the same user is in the num detected
> NewkRichard: mayb that doesnt make ne sense:P
> BDS: it might work with more, would be best to try with one first

The question that the student newKRichard asks is perfectly sensible, although he doubts its relevance. He is actually thinking ahead of the professional who has left some code marked as *TODO (*a common practice in programming used when some part of the code might not be as simple as expected, and will have to be dealt with at a later stage). The student is right with his hunch that more than one user at a time interacting with the object could cause trouble, as he has seen in a previous example, a couple of sessions beforehand:

> [2009/03/31]
> NewkRichard: em, wat does total_number do?
> Vinnie: yeah, that's what i was wondering[...]
> You: it counts the number of people doing the touch
> Rawb: ohhh

But this knowledge is quite specific to the task at hand and to the current virtual environment, so the students are gaining not only factual knowledge during the exercise, but also developing problem solving skills and expertise in the area.

Active learning

Reflection-in-action was an observable element of the sessions where students were delegated certain tasks as orientation or training. They were forced to go back to all the facts that they were taught when they went through the session themselves as students. This definitely reinforces all the concepts that they were exposed to. The student Sasha was asked to organise the orientation session for a new member of the group, Bazooka. The replication of the two logs would not be appropriate in this space (appendix), but a comparison with one of the sessions led by a professional yields a very much extended and specific accountability of facts. During Sasha's orientation session, he simply was given an object to be forced to learn how to use the inventory. When the student organised the session himself, he gave the same object to the new member, but also the following set of instructions that were never given to him:

> Sasha: If you want to give this item to someone else
> Sasha: "right-click" and chose copy
> Sasha: And then paste in the "objects " tab
> Sasha: You can copy many times
> Bazooka: where do i paste?
> Sasha: But there are certain items in SL which are "locked" so you cannot copy it
> Sasha: Paste it in the "objects" tab
> Bazooka: I get it

This is only a small example of how the learning exercise can be enriched by the knowledge gained by the students, by their desire to learn, and by the reflection-in-action that they are forced to produce when given the responsibility to organise part of the activities by themselves. This was observed in several occasions. Related to this, the following extract of and interview with Sasha exposes this interpretation in his own words:

> Sasha: (...) It got me from passive thinking into active thinking: I thought about what was going on in the other person's head. I tried to anticipate problems, and solutions to them
> me: what i eman is, did you get something in return?
> Sasha: I think the experience is comparable to public speaking. You're anxious to do it before you do it, you think about what to say next while you're doing it, and you're glad you did it when you've done it
> me: that's a very good comparison :)
> Sasha: What I got in return was self-confidence It made my knowledge more consciously accessible
> Sasha: So I got from knowing how to do something, to being able to communicate what I know

The experience has provided, at least for this particular participant, more than a passive acquisition of factual knowledge, and the artefact and the delegation of tasks are accountable for it.

In the same light as the examples presented, the students are now happily using two different virtual world grids when, as recorded in the questionnaires, seven of the students had never heard of the concept of grid.

All these examples show how the knowledge element of the epistemic games theory is supported by the artefact, and it is connected tightly with the skills element. The case of the delegation of tasks also gave the opportunity to play a different role in the game, experimenting with a different identity, which is the topic discussed in the next section

5.1.1.3 Identity

Identity is one of the main characteristics of the power of games in education. As registered in the questionnaires, most of the students engage with different identities while playing video games, and that can also be applied to a virtual world setting. In this study, students were asked to be software engineers from the moment they joined a team. They were confronted with different situations during the study, not only in a pure learning sense during taught sessions, but also in different real life situations, as meeting real clients. The delegation of tasks, as accounted for in the previous section, also allowed them to experience more than one identity during the study.

A change in attitude was noted when the students had to face 'real' clients. In this case, as opposed to their open disposition to learning during the sessions, they wanted to appear as more knowledgeable. In the next fragment this situation can be observed (sleepy is a client, Josmas is the author, Rawb and Duinn are students):

> Sleepy: could you guys tell me about some of the other projects you've been working on in SL?
> Sleepy: Rawb and Duinn
> You: as u know sleepy we've been in for only a cuple of weeks

> Sleepy: sure
> Rawb: well, so far Ive done some scripting and designing and such on houses and objects... nothing too complex as of yet.
> Sleepy: what type of scripts?[...]
> Rawb: well the house is a standard house that can open doors etc, kind of like these doors here. But as josmas, said, we have only started a couple of weeks and so we havent gone into anything major
> You: we've been looking at the radar stuff, a slide viewer
> Rawb: my major projects involve other computer languages
> --- At the same time the author was having private conversations with the students:
> Rawb: just trying to sound like i know what im doing :D
> Duinn: i'm ok not saying anything?
> josmas: sure, no prbs
> Duinn: very afraid of revealing my ignorance

In the conversation above, the attitudes are different to other sessions. One of the students does explain his activity and tries to excuse it with the fact that he has not been using the environment for a long time, which is totally true. The other students opt for not speaking at all, after checking with the professional if the attitude is feasible. It is a very different situation to any other session had so far, and it forced the students to behave differently and show a more professional side of themselves, in other words, a different identity.

A number of aspects included in the welcome email to the group members spawn certain positive reaction in some of the members. This is an extract from the first mail sent to the group:

> Email from Josmas(the author) to the group: "*Please note that there are no students in the team. You guys are joining the team as software engineers and due to the open source nature of the company you will be included in all the discussions, meetings, ideas for projects, design exercises, etc. We are all equal in this team and the titles only reflect the professional experience of the individual. During the project every individual will be able to try different positions, or assume different roles in different specific projects.* "
> Reply from Sasha(student) to Josmas: "*my initial impression was that what you said in your email sounded really awesome. Especially the part where we're all equal and get to try various roles in different project. I'm really looking forward to that :)* "
> Reply from Pixy to Josmas: "*I am glad to receive the title 'Software Engineer', sounds good*"

Most of the students would be happy to be treated as software engineers(as recorded in the pre-questionnaires). The welcoming email makes a clear point, *no students in the team*. Being an engineer entails a great deal of responsibility that it would not be associated with a student role, even within a gaming setting. The students are happy about it and welcome this rule in the game.

They also welcome the possibility of trying out different roles. The delegation of tasks gave them the opportunity to explore new identities. When asked about his experience in a delegated task, the student Sasha relates:

> Sasha: I would say I was happy to do it. It gave me a chance to demonstrate responsibility. […]
> Sasha: I would say that it made me act more responsibly than usual, but not by a significant margin.
> I would say that I was more detail-orientated than usually. In other words, I didn't want to appear sloppy, and put my 100% into the job [...]
> Sasha: Yes, it did put me in your shoes for a while. It got me from passive thinking into active thinking: I thought about what was going on in the other person's head. I tried to anticipate problems, and solutions to them.

Sasha is happy to be given the opportunity to play a different part, and to be able to step into the professional side for a while, with more responsibility and being in charge of the running of the session. He experimented with a different identity and he liked it, and the artefact has provided the support needed for this to happen.

Asking the students to behave as software engineers from the beginning of the study had contributed to broadening the experience for them. This happened not only in a passive mode but also actively, forcing them to reflect on the acquired knowledge, and putting them in situations that they had not encountered before. The fact that most of the students took on their roles not only happily, but also glad to be treated as equals within the team, takes the discussion to a new element in the theory and in the artefact: values

5.1.1.4 Values

Values characterise every social group or environment. What is important, what might be interesting, or what has a valid meaning and what that meaning is, are directed by the rules of a particular group, and learning how to value and care about all this is part of the process of becoming a professional. The welcome email to the group members contained also a number of values that spawned certain positive reactions among the participants. This is an extract from the first mail sent to the group, outlining the content that relates to values:

> Email from Josmas(the author) to the group: "*...due to the open source nature of the company you will be included in all the discussions, meetings, ideas for projects, design exercises, etc. We are all equal in this team and the titles only reflect the professional experience of the individual...*"
> In the reply from Sasha(student) to Josmas we can also see his reaction to the values: "*my initial impression was that what you said in your email sounded really awesome. Especially the part where we're all equal...* "

It is difficult to isolate values and fostering of identities. Being treated as an equal and being assured that their ideas are as important as any other member of the team, no matter the level of knowledge or years of experience, was sought after in this project to allow the students to participate and express themselves freely. This was reflected during the sessions, where different participants would expose their ideas during work sessions, without any problems of 'status', but also accepting their own limits. In the following conversation, the professional BDS and the students NewKRichard and Pixy are working on a complex server related project in a collaborative way, where everybody have their chance to speak their minds:

> BDS: i have some ideas, but if you have some too we could discuss here :)
> NewkRichard: ok wel, em wats ur ideas
> NewkRichard: mayb we can help to implement
> BDS: one thing is that we should modify the server to do proper logging of all the events
> NewkRichard: yeah, so like every post request it logs the users?
> BDS: yes [...]
> Pixy: like creating a session? [...]
> NewkRichard: exactly so, like a new session as pixy says, a certain "Checkin time interval" , and then the log is uploaded to the websire [...]
> NewkRichard: tbh, man, i dont even no where to start, bu im willing to learn:d

All these ideas expressed by both students or professionals are collected to make improvements to the project. The students are aware of their own limitations ("*i dont even no where to start, bu im willing to learn*"), and are happy to be guided by the professional. Everybody is considered in the process and acknowledge for their effort. The intentions expressed from day one in the project could have affected positively to this kind of participation, demonstrating that values play an important part in the artefact. All the participants were aware that every piece of code or documentation created was being posted to a publicly accessible website and source control system. This was one of the premises of the project and the students were happy accepting it.

Four important elements have been examined so far, but their meanings might differ in different settings, depending on the social community within they are examined. In this study, the targeted community is software engineering. The epistemic frame that this kind of social group creates to empower its members to develop knowledge, skills, identity, and values, is the topic discussed in the next section.

5.1.1.5 Epistemology

Continuing the idea from the previous section, epistemology wraps all other elements in the theory, framing and giving them a domain to inhabit. In this sense epistemology in this project, and in the theory applied in it, is domain specific, stating the rules of what is true or not in the domain at hand. Unlike previous sections it would be really difficult to identify an extract of a log that shows support for this element. In fact the whole artefact demonstrates it by recreating the processes that allow the community members to develop all the other elements of the theory. It is embedded in the learning experience and that, instead of being proved by a couple of sentences in a log, it is demonstrated by the data sets themselves. Having a project management tool, a wiki for documentation purposes, a ticketing system, a source control repository, diverse ways of communication, a programming environment, different clients for several services, using an scripting language that uses a states machine paradigm, or stating that all the members of the team are equal and share every outcome that will be publicly published, demonstrates the point that the artefact supports the epistemology element of the theory.

The sections above show clear evidence that all the elements of the epistemic games theory are supported by the artefact within the virtual environment. The process of learning to think and care as a professional can be facilitated by the use of the cognitive apprenticeship model. This project aimed to allow for the model to be used within the artefact. The next section describes the observations about this topic during the running of this study.

5.1.2 Cognitive Apprenticeship

The following subsections provide examples of the observations regarding the processes in cognitive apprenticeship, modelling, scaffolding, fading, and coaching, as found in the data sets collected in this study.

5.1.2.1 Modelling

The following subsections provide examples of the observations regarding the processes in cognitive apprenticeship, modelling, scaffolding, fading, and coaching, as found in the data sets collected in this study.

5.1.2.2 Modelling

Modelling and personal instruction have been observed as a very positive influence on the students, also showing the ways in which the artefact supports various kinds of interaction. Problems outside of the projects at hand were welcome during the sessions. This provided the opportunity to collaborate and also to make the process of looking for a solution 'visible' to the students, as opposed to a prefixed solution, or to the one a only path that students generally find in a text book . In the following interaction taken from an msn log (appendix), both the professional and student actively look for a solution to a quite complex problem. The student is trying to install a copy of the OpensSimulator server in his personal laptop:

> Josmas: ok, hold on... what the exact [error] message? can u copy+paste
> Richaeeee: install the Windows version of Mono to run .NET executables; there is no windows version of mono is ther?? ; i taut that was the linux copy of the .net framework or something along them lines:p
> Josmas: yup, it is... but it might be the package... let me cehck that
> Josmas: ok, i think i got it, the package i gave u is not source code
> Richaeeee: oh ? :)
> Josmas: it is an executable for windows!
> Richaeeee: yeah its executables:p, hahah i think we both mite need some coffee

In trying to help with the installation, the professional forwards the wrong file to the student. Seeing that the professional is also making mistakes, and seeing how they deal with the problem is important for students to develop a belief in their own capabilities. The paragraph has many interpretations and could be linked to the reinforcement of identity as a software engineer. There is also a clear link with the concepts of skills and knowledge that are needed in order to accomplish such a difficult task, and probably gained during the process. The example shows how the modelling process of a challenging task for both the student and professional affects positively their interaction, and the final outcome of the session, as opposed to a traditional lecture.

A further interpretation of the collaboration among students and professionals is the fine line that separates the modelling and the scaffolding process. It is a natural progression, after having modelled the activity, the students should be brave enough to start taking on responsibility, even though they know that the professional will be available to continue guiding them in the task. Observations about the scaffolding process are introduced in the following section.

5.1.2.3 Scaffolding

Scaffolding provides the students with a somehow eased environment where they can continue to develop their way into mastering a task. It is a natural progression from the modelling stage, when a part of the task at hand will try to be achieved by the students on their own. But sometimes further help is needed:

> Richaeeee: em.. ive got all the packages, bu for some reason my linux wont allow me to svn out ??, ive set up my http proxy and all... its soo wierd
> Josmas: that can be the network, it's probably closed; wget goes through port 80 i think and svn doesn't

In the conversation above, the student tries to replicate what he is reading in a piece of documentation, but something is not working, even following all the steps in the model. The professional knows where the problem might be, but does not give the full answer, but enough clues to get the student back on track. The advantages of effective modelling were noted in different occasions, providing for the start of scaffolding. At this stage the students were able to work on their own, just asking for help when needed. In some cases, further scaffolding was deemed appropriate:

> Richaeeee: ive checked the "svn --help"
> Richaeeee: i dont see ne crazingly obvious way
> Richaeeee: ill google
> Josmas: cool; here, have a look at /etc/subversion/servers; at the end of that file
> Richaeeee: that looks vv promising doesnt it?
> Josmas: yup

In this example the effect of previous modelling is observable. The student does not ask how to do something until he has already tried to find a solution by using the '--help' command, as he has seen the professional doing in other occasions. He also decides to use a search engine, another solution technique observed before. In this particular case, the professional knowing that the answer is not in the help file of the command, and it might be difficult to find in the web, points the student in a different direction but still without giving the answer, and then fades his help and lets the student keep on with his exploration.

In both examples the professional will help in a certain way, not giving the answer directly, but pointing at possible paths for exploration, that added to the previous modelling of the task at hand, should be enough to solve any problem the student may be facing. The natural progression of taking away the support and shifting the responsibility to the student side is recorded in the next section

5.1.2.4 Fading

The best example of fading during the study are the delegated sessions of orientation, where a student leads the session without the professional being present. The responsibility is shifted to the student side, that is now acting himself as a professional. It is still an act of fading as it was agreed that a professional would be available during this type of sessions. The artefact allowed for them to be logged in a different location, and available all the time through private chat. In an interview with the student leading this session he declares:

> Sasha: I would say I was happy to do it. It gave me a chance to demonstrate responsibility.
> me: good, so you were happy having more responsibility, but you knew that you could do a good job
> Sasha: Yes. I would be anxious if I was unable to do the job to a personally-set high standard

During this session the student was helped in several occasions. This was possible thanks to the capabilities and the organisation of the artefact, and it resulted in very positive results in the students.

The three processes seen so far, modelling, scaffolding, and fading, are wrapped together in an attempt to follow the whole process of getting the students to become the experts, always providing the needed support. This wrapper is the coaching process as introduced in the following section.

5.1.2.5 Coaching

As in the case of the epistemology element in the previous theory, coaching is an ongoing process that cannot really be reflected in an extract from the logs. The whole artefact in itself, by the use of the external tools and the communication tools, cater for the coaching aspect of the project. Many examples can be seen where the professionals try to foster the use of the external tools, communication, and show themselves always available to help. From an early point, students were assigned tasks in the ticketing system and they were in charge of the research work that any of the projects needed. The main goal of the study was to gradually shift the responsibility of the tasks to the students, to finally get them to become experts.

The last two main sections collect observations regarding the two main theories in this study. Added to these, a series of events, some of which were unexpected, also occurred during the facilitation of the learning experience. Some of them are related in the next section.

5.1.3 Other observations

Although the following issues are not related to any of the research questions, they play an important part in the design and implementation of the artefact, and can effect the running and the flow of the exercise. This is why they were deemed sufficiently important to be noted.

5.1.3.1 Engagement and enthusiasm

The appeal of the virtual world and gaming environment played a big part in this project. Students did not only act as users of complex technology but also dealt with the installation and administration of complex servers. The following extract from a log shows this point, when during a long session, the professional (Josmas) thinks that is time for a pause:

> Richaeeee: yeah jus one a sec
> Josmas: no worries, leave it for later if u bored already :D i'll be around
> Richaeeee: haha im not bored, no :D
> Richaeeee: learning learning:D

Even though they have been installing a complex system for about two hours, the student explicitly says that he is not bored and he is enjoying the learning. This particular session was a command line intense and complex job.

5.1.3.2 Rewards

Rewards is one of the mechanisms used in games to grab attention and create motivation and engagement. In the course of this study a number of rewards were hidden in the process. This seems to have had an effect in some of the participants. The following extract of one of the logs during a conversation with one of the students , proves this point.

> francis08: yea, yeah. the idea of reward see to trigger some people to start working :D
> francis08: especially me. :)
> You: :D that was part of the plan

As explained in a previous section, this reward was designed to reinforce some of the values of the group. The first person to add valuable information to the public wiki was prized with a cash sum of virtual currency.

5.1.3.3 Zone of proximal development

During one of the first sessions mixing professionals and students, the following situation was noted. At some stage one of the professionals, unaware of the level of knowledge of the students, started to talk about his idea for a Java server, how he would implement it, and looked for feedback among the students. The students were being bombarded with a series of advanced concepts that they have not seen in college yet. The other professional realises that the students are not following and decides to contact privately the other professional:

> josmas: u are going to scare them to death with java
> BDS: sorry, i do not know what is their experience

The students did not see the lines above, but in any case they expressed their concerned about the situation:

> NewkRichard: i seem a little out of my depth bu im sure i can jus catch up yeah???
> You: sure guys, no prbs, u will get there
> Sasha: I second that newk
> You: this is part of it, and we are here to help

It is a complex conversation the one reported, and the students were out of their comfort zone. But even in that case they felt that with the help from the professionals, they could tackle a problem that they would not go into on their own. It seems that the mix among different levels of expertise could stretch a bit the zones demarcated by Vygotsky in his work. Personal instruction and the cognitive apprenticeship model might expand the learners views of their own capabilities.

5.1.3.4 Other factors

There are several issues regarding technical problems that can sabotage a session. Hardware issues, connectivity problems or access restrictions inside the college network, needed to be overcome in many occasions during the study. Timing was also a an issue, all the participants volunteered to the project and to be able to put them together in groups and manage times that they could all be online at the same time was a concern and hard work. On a positive note, the fast visual response that virtual worlds provide when scripting affected very positively the experience.

Other observations are related to such topics as collaboration (less prevalent when professionals were not involved), peer to peer mentoring, project based learning, distracting animations and other internal and external noise, and the fact that students perceptions of the projects and tasks are quite different from the professionals. Students tend to quantify work load and skills and effort required at a much lower scale that the professionals.

5.2 Summary

This chapter reported observations regarding the two main theories in the study, epistemic games and cognitive apprenticeship, as taken from the analysis of the data sources recorded during the learning experience with the artefact designed and implemented in this study. The artefact supports the main elements of both theories, creating an environment where the epistemic frame of the software engineering community can be developed. The next chapter will analyse the findings reported here in greater depth and in the context of the literature, and along with the reflection work of the author will try to find a valuable answer to the research questions.

Chapter 6 Discussion

The findings chapter described a series of facts taken from the analysis of the recorded datasets, in an attempt to provide evidence that help to answer the research questions. The current chapter provides reflection and discussion about those questions, drawing from the results previously exposed.

6.1 The research questions

The questions for *this* particular study were formulated as follows:

RQ1: *To what extent are the elements of the epistemic games theory supported by the game-based learning experience within the virtual world settings?*

RQ2: *To what extent does the artefact created within the virtual world setting allow for the application of the cognitive apprenticeship model through the interaction among students and professionals?*

The following discusses each of the questions in some detail, in the light of the data presented in the *previous* chapter, along with important observations during this study. Discussion of the first research question will be done through an analysis of epistemic games, whilst the second question will be discussed through analysing the cognitive apprenticeship model.

6.1.1 Epistemic games

The findings outlined in the previous chapter regarding the elements of the epistemic games theory will be dealt with in the subsections of skills, knowledge, identity, values and epistemology.

6.1.1.1 Skills

The acquisition of skills is not easy to demonstrate. To support a claim like that, the author will base his argument in the level of use of certain tools, and the familiarity with which the participants can talk about the number of systems that formed the artefact. In the case of the source control, the fact that the system records the credentials of its users can constitute evidence that most of the students were able to use it. They were able to download all the information contained in an external repository, modify the files in it, and commit the files back to the system. The findings in this section support a further level of skill in at least one of the students, who reported during an interview that, thanks to the confidence gained in using the system through this project, he is no longer using it passively, but also coordinating college work for his course team mates. What is important is that the students did not learn how to use a specific tool, but gained a skill. This claim is supported by the fact that this particular student uses a different solution for source control that the one used in this project. All the participants were freely allowed to choose the concrete client application that they wanted (sometimes more than one), and helped throughout the set up process.

The use of the ticketing system within the artefact also records all the interactions from the different users. It can be considered in a similar way as the source control example, as per the evidence shown in the findings chapter. These two systems are widely used in the majority of software development settings, and an early introduction to them can be beneficial for any participant in their near future. This is only a small example of the kind of systems that the participants dealt with during this study, and supports the claim that regardless of their level of knowledge or expertise, the artefact provided space for skills acquisition and further development. This fact also highlights the connection between skills and the topic of the next section, knowledge.

6.1.1.2 Knowledge

As stated in the findings, it is almost impossible to separate the acquisition of new skills from the topic of knowledge. The findings reported demonstrate both the acquisition of factual knowledge during the study, and also to the derivation of the facts acquired into procedural knowledge. In the example of the *states machine* paradigm, it is a condition of its use to be able to understand at least some of the concepts associated with it. The recording of facts associated with this paradigm in the external wiki helped to this end, not only as 'class material' developed by an expert, but as a source of information that could be worked on and further developed, collaboratively or individually, and edited to this end.

The artefact provided the opportunity, not only to access the information at any time, but also allowed for user created content, providing also for further reflection given that interactions took place in a public space. The artefact also provided the appropriate context for the organisation of group sessions, where application of the acquired concepts and collaboratively construction of knowledge was also possible. As in the case of skills, the acquisition of knowledge can be found at different levels. The participants were able to use specialised language in a familiar way, as recorded in the different datasets. Students were also empowered to test different tools and systems, and were confident enough to make critical judgements and express preferences for one or the other. The artefact also gave participants the opportunity to build on knowledge acquired throughout the different sessions, due to the projects being laid out in such a fashion, allowing for the development and growth of areas of expertise.

The reporting of *reflection-in-action* during sessions where important tasks were delegated, also supports the claim that the artefact allowed for acquisition of knowledge. As reported by a student after a session in which he was in charge of doing orientation with a new team member, the tasks "*got me from passive thinking into active thinking: I thought about what was going on in the other person's head. I tried to anticipate problems, and solutions to them*". He also reports that his self confidence was boosted by the responsibility, and that he was pushed to go from *knowing* something to *being able to communicate* that something, reinforcing what he already knew. The authenticity and relevance of the task played a fundamental part of it, as the appropriateness of the context provided by the artefact, an almost 'real' software company simulated setting. This shift of responsibility was designed around the concept of identity in gaming theories, the topic of the following section.

6.1.1.3 Identities

Most of the students recorded their thoughts in the pre-questionnaires about how they engage with different identities when playing video games. This is an important element in a situated learning experience, where the environment is designed to be as authentic and real as possible. In this study the targeted community is software engineering, and the students were asked to play the part of an engineer from the beginning. According to the epistemic games theory, engaging in roles helps the participant to think and care about the community like professionals do, and the artefact provided for this. Also recorded in the questionnaires is the fact that most students would welcome the opportunity to be called *software engineers*. The chance to play different roles was also a feature of the artefact that some of the students felt attracted to. There were two main deviations from the role as a software engineer as reported in the findings, a situation when students attended meetings with 'real' clients, and a different situation where the running of sessions was delegated to students. During the meetings with clients, a different attitude was observed, where students showed themselves less open to learn and more worried about hiding what they did not know. During the orientation delegated tasks, a different process altogether happened, when the students passed from a passive to an active attitude, being faced with more responsibility than usual. These deviations were an integral part of the artefact, and were possible according to the values by which the fictitious companies were laid out. The notion of values is discussed in the next section.

6.1.1.4 Values

The values embedded in the core of the fictitious companies presented through the artefact affected the learning experience as reported in the findings chapter. The students were not only happy to experiment with their new identity as software engineers, but were also positively pleased by the fact that they were treated as 'equals'. The level of participation in the sessions was positively increased, regardless of levels of knowledge or skill. The students exposed their limitations but also demonstrated their willingness to learn and be guided. They demonstrated a positive attitude towards the open source values embedded in the artefact, and they all agreed to publish their work and share their experience in a publicly accessible system, both in the virtual world grids and the external tools that are part of the artefact.

The four elements discussed so far will be brought together in the next section as part of the epistemic frame that this study provided through the creation of the artefact.

6.1.1.5 Epistemology

Epistemology wraps and situates all the other elements in the theory within the settings and beliefs of a social community regarding a professional practice. The artefact allows the participants to develop the other elements of the theory in a software engineering context.
It aims to give the students the possibility of acting like a professional, using the tools professionals use, and learning to think like them. It enables students to see professionals reasoning and solving problems, finding solutions or even compromising. Data from the pre-questionnaires was also taken into account, especially the sections concerning how students feel about assignments, and how free they are to propose their own ideas. It was deemed valuable to provide for the possibility to enable students to bring their own projects into the game. The virtual environment gave the opportunity to simulate a software company. The role-playing game set the imagined business domain in the minds of the participants and provided the ability to

focus on the situation and the different roles and identities. This added to the use of external tools, a different programming paradigm, and a set of values, demonstrates that the artefact supports the epistemology element.

6.1.2 Cognitive apprenticeship

The four elements of apprenticeship, modelling, scaffolding, fading, and coaching will be the topics of the next four subsections.

6.1.2.1 Modelling

Modelling was a main part of the guided sessions during orientation, and scripting and building basics. The environment is very adequate for this kind of interaction being highly visual, giving a quick response, and providing multiple communication channels. A more interesting kind of modelling happened when both professionals and students tackled together problems that they had not encountered before. As demonstrated in the findings, the collaboration made it possible to make the thinking "*visible*", in the sense that the students were able to see the professionals trying to find solutions to problems, and were able to see all the reasoning and application of factual knowledge in a real problem solving session within the context provided by the artefact. Students also availed of different models of expertise from different professionals, widening their views of the process.

The natural progression from modelling is scaffolding, which is the topic of the next section.

6.1.2.2 Scaffolding

The scaffolding process was supported by the artefact in multiple ways. The combination or the virtual world grids with other external tools facilitated this work by giving a chance to further develop resources to both professionals and students. The fact that the professionals could check on all the information published, providing support and even correcting any misunderstandings, gave the students more confidence in their use. As reported in the findings, during guided sessions, professionals tended to point out at possible solutions instead of giving answers to problems. This also contributes to scaffolding in a more human sense of the process, unrelated to the technology component of the artefact itself, but embedded within. The professionals were not seen as teachers but as equals, so collaboration was more open. This also means that the responsibility for the tasks was shared, allowing for the next step in the process, fading the support.

6.1.2.3 Fading

In the process of helping students to become the experts, the next natural step after scaffolding would be fading the support and shifting the responsibilities of the task towards them. As reported in the findings, the delegation of tasks had a big impact in the participants. This activities also allowed for fading, shifting responsibility and taking off some of the support giving to date. During these sessions, at least one of the professionals was online all the time, but in a different location within the virtual world, out of the view of the participants. Should students run into trouble they were reassured that support was available, but in a more faded fashion than in previous sessions. The environment and the design of the artefact allowed for this to happen.

6.1.2.4 Coaching

As coaching is the thread running through all the apprenticeship process, it can be said that the artefact itself, through its design, allows for this process to happen as recorded in the three previous sections. The main three differences regarding traditional apprenticeship (as stated in the literature) were catered for in the artefact by:

- making the thinking “visible” through modelling new problems in a collaborative way
- situating the tasks in and authentic context so students understand the relevance of the work they are asked to do, and

- varying the diversity of situations so students can see commonalities and transfer what they learn, as in the case of the source control client applications.

Although the research questions have been discussed already, the following section deals with a series of observations that are related to the learning experience and to the answering of the questions themselves.

6.1.3 Other observations

Other observations that were deemed important during the analysis of the datasets are recorded in the next subsections.

6.1.3.1 Engagement and enthusiasm

The author was surprised by the level of engagement and involvement of all the participants. The affordances of the virtual world and the novelty of working in such a different environment, reinforced their commitment to the project. Engagement and motivation are proved by solely the fact that volunteers, with apparently nothing to gain, participated in the project. The group of professionals could have been driven by loyalty or friendship towards the researcher, but the group of nine students had no relation at all with any of the other participants, and all the interaction with them was online, even the final interviews. The virtual environment was actually not needed or used in all the sessions. The integration and parallel use of external tools facilitated communication. Some of the interactions were on plain chat systems, but the theme of 'gaming and virtual worlds' was needed to start up the project, and to get people interested.

6.1.3.2 Rewards

The rewards system introduced in the artefact had a positive impact over some of the students, reporting to feel more motivated to do more work and unlock the next reward.

6.1.3.3 Zone of proximal development

As reported in findings, there were occasions where students exposed their concern when advanced topics that they had not seen before were exposed, but they also felt confident about the possibility of stretching their own limitations and capabilities with the help of the professionals.

6.1.3.4 Other factors

A great feature of the virtual environment is the quick feedback that it provides when working with programming scripts. This might be thought as one of the premises to engage digital natives, but it also had an impact on the professionals, that are used to work with heavy weighted platforms where changes are difficult to introduce. The almost immediate feedback resulting from a change in a script inside the virtual world affected very positively their perception of the environment. The participants also considered '*great fun*' the whole experience, and some of them intend to continue using virtual worlds. Due to this interest the *Apachete Foundation* (founded in this project) will continue to operate as an open source group. The only concern with the environment is the one associated with proprietary technologies in the case of the SL grid. This is why in the project there was an interest to use open source solutions when possible. Other significant obstacles, as hardware or connectivity problems can not be left unmentioned, but they could be overcome in the concrete situation in which the project took place.

6.1.3.5 About the participants

Due to the diversity in the set of participants, their different cultural backgrounds (Ireland, Spain, Germany, Russia, Hungary, Panamá, etc.), different education, ages, and their different levels of experience, it may be appropriate to suggest that the results of this study could be generalised, and that this approach could be successfully applied to a wider subset of participants, although further work would be necessary to verify this suggestion.

6.2 Answering the questions

With respect to question one, all the elements of the epistemic games theory have been demonstrated within the context of the artefact, and with clear links to its design and implementation, specifically with respect to the cases of values and identity, and allowing for episodes of creation of knowledge and development of skills.

Question two may be similarly addressed. Based on analysis of the data, it has been clearly demonstrated that the combination of the different services in the artefact enable the processes in the cognitive apprenticeship model to be applied in the study, and more importantly for their translation from a traditional apprenticeship view to a cognitive approach within a technologically advanced environment.

These answers to the research questions seem to be substantially in the affirmative, so it would seem appropriate to conclude that virtual worlds, integrated with external services, provide an appropriate environment for a game based situated learning experience in the field of software engineering.

6.3 Limitations of the research

There were two main limiting factors in this project, one being time and the other the volunteering nature of the participants. The exercise lasted 6 weeks, with an expected commitment from the participants of only two hours a week. The fact that all of the students were volunteers, with nothing to gain but the experience itself, made the study challenging. As stated before, the author has never met any of the students in real life. The nature of the virtual environment also supposed a limitation during some of the sessions. The group was small in numbers but the participants were very committed all throughout the project, and their perceptions were very helpful and valid.

6.4 Significance of the research

In general terms, the diversity of the participants and their different backgrounds and education makes the group a good source of contrasted opinions and feedback. The main limitations of the project had obviously had an impact in the results, but the experience has been considered as very valuable by most of the participants.

6.5 Suggestion for future research

The project has a sense of continuity built within the experience. The students participating in a first run, could become professionals in a second or maybe third cycle of the project. Another interesting proposal would be to organise the experience sponsored by real companies, and having those companies bring the professionals into the project. This could be ideal in a masters program or even in an undergraduate cycle.

The present study focused on the support that the artefact, created in a gaming setting within a virtual world, can provide for the theories of epistemic games and cognitive apprenticeship. The experience proved so broad that a number of research paths were left open and could be considered for further research. Some fields as organisational theory, social constructivism, communities of practise, or more concrete aspects of teaching programming could have been analysed and studied within the set of data collected.

The limitations of the scripting language in the environment used could prove difficult to integrate the artefact in the current curriculum of a computer science degree. There are other environments (*open croquet* and *project wonderland*) that seem more suited to a project like this, but were left out due to lack of time and resources. These environments could also be considered as topics for future research.

Chapter 7 Bibliography

Bassey. (1999). *Case Study Research in Educational Settings*. Open University Press.

Berg, W., Cline, M., & Girou, M. (1995). Lessons learned from the OS/400 OO project. *Commun. ACM*, *38*(10), 54-64. doi: 10.1145/226239.226253.

Collins, A., Brown, J., & Newman, S. (1989). Cognitive Apprenticeship: Teaching the craft of reading, writing and mathematics. In *Knowing, Learning and Instruction: Essays in Honor of Robert Glaser* (LB Resnick, Lawrence Erlbaum.). Hilsdale, NJ.

Collins, A., Brown, J. S., & Holum, A. (1991). Cognitive apprenticeship: making thinking visible. *AMERICAN EDUCATOR*, *6*, 38--46. doi: 10.1.1.124.8616.

Creswell, J. W. (2007). *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research* (3rd ed.). Merrill.

Esteves, M., Morgado, L., & Fonseca, B. (2007). CONTEXTUALIZAÇÃO DA APRENDIZAGEM DA
PROGRAMAÇÃO: ESTUDO EXPLORATÓRIO NO
SECOND LIFE®. *Conferência IADIS Ibero-Americana*. Retrieved May 2, 2009, .

Garlan, D., Gluch, D., & Tomayko, J. (1997). Agents of change: educating software engineering leaders. *Computer*, *30*(11), 59-65. doi: 10.1109/2.634865.

Gee, J. P. (2007). *What Video Games Have to Teach Us About Learning and Literacy* (1st ed.). Palgrave Macmillan.

Holmes, B., Tangney, B., FitzGibbon, A., Savage, T., & Meehan, S. (2001). Communal Constructivism: Students constructing learning for as well as with others. *Proceedings of SITE 2001, Florida*.

Järvelä, S. (1995). The cognitive apprenticeship model in a technologically rich learning environment: Interpreting the learning interaction. *Learning and Instruction*, *5*(3), 237-259. doi: 10.1016/0959-4752(95)00007-P.

Kiili, K. (2005). Digital game-based learning: Towards an experiential gaming model. *The Internet and Higher Education*, *8*(1), 13-24. doi: 10.1016/j.iheduc.2004.12.001.

Kolb, D. A. (1984). *Experiential Learning: Experience as the Source of Learning and Development*. Financial Times/ Prentice Hall.

Lahtinen, E., Ala-Mutka, K., & Jarvinen, H. (2005). A study of the difficulties of novice programmers. *SIGCSE Bull.*, *37*(3), 14-18.

Lave, J., & Wenger, E. (1991). *Situated Learning: Legitimate Peripheral Participation* (1st ed.). Cambridge University Press.

Ludlow, P., & Wallace, M. (2007). *Second Life Herald: The Virtual Tabloid That Witnessed the Dawn of the Metaverse*. MIT Press.

Machanick, P. (1998). The abstraction-first approach to data abstraction and algorithms. *Computers & Education*, *31*(2), 135-150. doi: 10.1016/S0360-1315(97)00064-X.

Machanick, P. (2007). Teaching Java backwards. *Computers & Education*, *48*(3), 396-408. doi: 10.1016/j.compedu.2005.01.009.

Papert, S. (1998, June). Does Easy Do It? Children, Games, and Learning. *Game Developer magazine*, 88.

Pedroni, M., & Meyer, B. (2006). The inverted curriculum in practice. In *Proceedings of the 37th SIGCSE technical symposium on Computer science education* (pp. 481-485). Houston, Texas, USA: ACM. doi: 10.1145/1121341.1121493.

Prasolova-Forland, E., Divitini, M., & Lindas, A. E. (2007). Supporting Social Awareness with 3D Collaborative Virtual Environments and Mobile Devices: VirasMobile. In *Proceedings of the Second International Conference on Systems* (p. 33). IEEE Computer Society. Retrieved May 2, 2009, from http://portal.acm.org/citation.cfm?id=1260778.

Prensky, M. (2001a, October). Digital Natives, Digital Immigrants. *NCB University Press*, *9*(5). Retrieved May 2, 2009, .

Prensky, M. (2001b, December). Do they really think different? *MCB Uniiversity Press*, *9*(6).

Prensky, M. (2005). *Digital Game-based Learning* (Paragon House Ed.). Paragon House Publishers.

Ryoo, J., Fonseca, F., & Janzen, D. (2008). Teaching Object-Oriented Software Engineering through Problem-Based Learning in the Context of Game Design. In *Software Engineering Education and Training, 2008. CSEET '08. IEEE 21st Conference on* (pp. 137-144). doi: 10.1109/CSEET.2008.26.

Schoenfeld, A. (1985). *Mathematical Problem Solving*. Academic Press.

Shaffer, D. W. (2005). Epistemic Games. *Innovate*, *1*(6).

Shaffer, D. W. (2006). *How Computer Games Help Children Learn*. Palgrave Macmillan.

Sherrell, L. B., & Mills, D. L. (2008). Introducing software engineering processes via games and simulations: a Tri-P-LETS initiative. *J. Comput. Small Coll.*, *23*(4), 133-139.

Squire, K. D. (2005). *Replaying history: Learning world history through playing Civilization III*. Indiana University. Retrieved from http://website.education.wisc.edu/kdsquire/REPLAYING%20HISTORY.doc.

Starr, C. W., Manaris, B., & Stalvey, R. H. (2008). Bloom's taxonomy revisited: specifying assessable learning objectives in computer science. *SIGCSE Bull.*, *40*(1), 261-265. doi: 10.1145/1352322.1352227.

Van Eck, R. (2006). Digital Game-Based Learning: It's Not Just the Digital Natives Who Are Restless…. *EDUCASE review*, *41*(2).

Venners, B. (2005, June 6). Design Principles from Design Patterns. A converstaion with Eric Gamma, Part III. Retrieved May 2, 2009, from http://www.artima.com/lejava/articles/designprinciplesP.html.

Vygotsky, L. S. (1978). *Mind in Society: Development of Higher Psychological Processes* (New edition.). Harvard University Press.

Yin, D. R. K. (1994). *Case Study Research: Design and Methods* (2nd ed.). Sage Publications, Inc.

Chapter 8 Appendix

All data sources are available in the CD attached, in electronic format, including:

Logs from virtual worlds/other chat systems.
Export of the project management sites, and subversion repositories.
Dissertation diary.
Student pre questionnaires.
Emails (in html format).
Links to the artefact, twitter, and facebook accounts used.