

Lisa Maria Tomes

# Exploratory and Social Learning in 3D Virtual Worlds

Master's Thesis

submitted to

Graz University of Technology

Supervisor

Univ.Doz. Dipl.-Ing. Dr.techn. Christian Gütl

Co-Supervisor

Dipl.-Ing. Johanna Pirker, BSc

Graz University of Technology  
Institute for Information Systems and Computer Media



Co-Supervisor

Assoc. Prof. Vanessa Chang, PhD

Curtin University



Graz, March 2015



Lisa Maria Tomes

# Exploratives und Soziales Lernen in virtuellen 3D-Welten

**Masterarbeit**

eingereicht an der

**Technischen Universität Graz**

Betreuer

Univ.Doz. Dipl.-Ing. Dr.techn. Christian Gütl

Mitbetreuerin

Dipl.-Ing. Johanna Pirker, BSc

Technischen Universität Graz

Institut für Informationssysteme und Computer Medien



Mitbetreuerin

Assoc. Prof. Vanessa Chang, PhD

Curtin University



Graz, März 2015



## Abstract

Since the concepts of training and teaching exist, many different learning methods have been designed and tested, and were found to be applicable or discarded again. Every era has produced its own methods, depending on what ideas people had of education and knowledge transfer. In an age where virtually everyone has access to a computer and many things of everyday life happen in and on the internet, also learning with the help of these technologies is an issue. One of the latest developments in this field is learning in 3D virtual worlds. In this case, also the social component plays an important role. With access to the internet many learners at a time can be gathered in such worlds who can in succession learn or play together, and communicate with each other in various ways. Also, the exploratory character of these environments cannot be denied. Users can roam as avatars through the learning worlds and explore them. However, the creation of virtual worlds can be time-consuming and is no easy undertaking for the normal teacher. This work describes the design and implementation of software modules to enable teachers in the creation of virtual learning environments without any programming knowledge. As a basis for the development of the software tools the open-source platform Open Wonderland was used with which collaborative 3D worlds, e.g. for business meetings can be constructed. For this work, Open Wonderland has been extended with capabilities to also enable learning scenarios in virtual worlds. With the help of the implemented extensions an exemplary learning environment was created afterwards, where students can learn about ancient Egypt. Learning takes place in the form of a scavenger hunt, in which the students must work together to be able to solve a quiz at the end. The Egyptian world is one of many possible, exploratory learning scenarios that teachers can implement using the tools developed. To get first feedback, the Egyptian scenario has been tested and evaluated subsequently with several groups of students. The results show that the approach of exploratory and collaborative learning was well received among participants, although they had suggestions concerning the graphics and interactivity of the world. At the end of this work, therefore, recommendations for additions and improvements, which resulted from the evaluation are summarized and presented.



## Kurzfassung

Seit es Ausbildung und Unterricht gibt wurden die verschiedensten Lernmethoden konzipiert, erprobt und für anwendbar befunden oder wieder verworfen. Jede Epoche brachte dabei ihre eigenen Methoden hervor, je nachdem, welche Vorstellungen die Menschen von Bildung und Wissensvermittlung hatten. In einem Zeitalter, in dem praktisch jeder Zugang zu einem Computer hat und viele Dinge des normalen Alltags im und über das Internet passieren, ist auch das Lernen mithilfe dieser Technologien ein Thema. Eine der neusten Entwicklungen in diesem Gebiet ist das Lernen in virtuellen 3D-Welten. Dabei spielt auch die soziale Komponente eine große Rolle. Durch einen Zugang zum Internet lassen sich in solchen Welten viele Lerner auf einmal versammeln, die dann anschließend gemeinsam lernen, spielen und auf verschiedenen Wegen miteinander kommunizieren können. Auch der explorative Charakter dieser Umgebungen lässt sich nicht abstreiten. Benutzer können als Avatare durch die Lernwelten streifen und sie erforschen. Die Erstellung solcher Welten ist aber mitunter zeitaufwändig und für den normalen Lehrer nicht einfach durchführbar. Diese Arbeit beschreibt den Entwurf und die Implementierung von Softwaremodulen, die Lehrern das Erstellen von virtuellen Lernwelten ohne Programmierkenntnisse ermöglichen sollen. Als Basis für die Entwicklung der Software-Werkzeuge wurde die open-source Plattform Open Wonderland verwendet, mit der kollaborative 3D-Welten z.B. für Business-Meetings konstruiert werden können. Open Wonderland wurde für diese Arbeit um entsprechende Funktionalitäten erweitert, um auch Lernszenarien in virtuellen Welten zu ermöglichen. Mithilfe der implementierten Erweiterungen wurde anschließend eine beispielhafte Lernwelt erstellt, in der Schüler über das alte Ägypten lernen können. Das Lernen findet in Form einer Schnitzeljagd statt, bei der die Schüler zusammenarbeiten müssen, um zum Schluss ein Quiz lösen zu können. Die ägyptische Welt ist eines von vielen möglichen, explorativen Lernszenarien, welche Lehrer mithilfe der entwickelten Tools realisieren können. Um erste Rückmeldungen zu erhalten, wurde das ägyptische Szenario im Anschluss mit mehreren Gruppen von Studenten getestet und evaluiert. Die Ergebnisse zeigen, dass der verfolgte Ansatz des explorativen und kollaborativen Lernens bei den Teilnehmern gut ankam, sie allerdings Verbesserungsvorschläge bezüglich der Grafik und der Interaktivität der Welt hatten. Am Ende dieser Arbeit werden deshalb Empfehlungen für Erweiterungen und Verbesserungen, welche sich aus der Evaluierung ergaben zusammengefasst und präsentiert.





## Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, \_\_\_\_\_  
Date

\_\_\_\_\_  
Signature

## Eidesstattliche Erklärung<sup>1</sup>

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am \_\_\_\_\_  
Datum

\_\_\_\_\_  
Unterschrift

---

<sup>1</sup>Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008; Genehmigung des Senates am 1.12.2008



## Acknowledgements

At this point I have to express my gratitude to several people who have significantly contributed to the success of this project. First and foremost I would like to thank my supervisors Univ.Do. Dipl.-Ing. Dr. techn. Christian Gütl and Dipl.-Ing. BSc Johanna Pirker who guided me through this thesis with both, their expertise and their moral support. They helped me to see outside the box when I concentrated too much on a topic and to focus my attention when I was in danger of getting lost. Thank you for the countless hours you spent answering my questions and correcting the first draft chapters of this work. I could not have wished for better supervisors!

I would also like to thank the many people in Australia who made my research stay instructive as well as pleasant. There was, of course, Vanessa Chang, my supervisor at Curtin University, who was always willing to take a little of her already scarce time for me. There were Torsten Reiners and Erik Champion who provided lots of input on virtual worlds. Thank you for that. If you read carefully, you will find that some of your ideas have made it into this work. There were Joshua Hollick and Andrew Woods from the HIVE who immediately agreed to help me with 3D visualization. I am sorry, that this did not work out. I am sure, it would have been great! There were Lillian Lau and Veronica Goerke who were the nicest office colleagues I ever had and who made sure that I always had enough to eat. There were my friends from the karate club, Richard Kern and Flo Anderson who were not only fun to train with but also gave me a lift when I needed it and supplied me with very much needed kitchen utensils. And finally my friend and housemate Jinil Thomas who was always there for a good talk or an exciting sightseeing trip. All of you folks made my stay in Australia most pleasant and surely unforgettable. Thanks mates and cheers!

Last but not least I have to thank my parents for a quarter-century of love and support. I know, everybody thanks his parents at this point, but mine are really the greatest!



# Contents

<b>Abstract</b> . . . . .	<b>v</b>
<b>1. Introduction</b> . . . . .	<b>1</b>
1.1. Goals and Objectives . . . . .	1
1.2. Structure of this Thesis . . . . .	2
<b>2. Background and Related Work</b> . . . . .	<b>5</b>
2.1. Teaching Methods . . . . .	5
2.1.1. Lecture . . . . .	6
2.1.2. Demonstration . . . . .	7
2.1.3. Collaboration . . . . .	9
2.1.4. Learning by Teaching . . . . .	12
2.2. E-Learning . . . . .	14
2.3. Serious Gaming . . . . .	17
2.4. Virtual Worlds . . . . .	22
2.4.1. Terms and Definitions . . . . .	22
2.4.2. Virtual Worlds in Education . . . . .	24
2.5. Showcases . . . . .	28
2.5.1. Giza 3D . . . . .	28
2.5.2. The Egyptian Oracle . . . . .	30
2.5.3. The SEE Experience . . . . .	31
2.5.4. The Virtual TEAL World . . . . .	32
2.6. Summary . . . . .	34
<b>3. Design</b> . . . . .	<b>37</b>
3.1. Requirement Analysis . . . . .	37
3.2. Selecting a Virtual World Platform . . . . .	42
3.3. Conceptual Architecture . . . . .	48
3.4. Summary . . . . .	52
<b>4. Implementation Details</b> . . . . .	<b>53</b>
4.1. Introduction to the Open Wonderland Architecture . . . . .	53
4.1.1. General Overview . . . . .	53
4.1.2. Modular Design . . . . .	55
4.2. Item Module . . . . .	58
4.2.1. Description . . . . .	58

## Contents

4.2.2. Architecture . . . . .	63
4.3. Quiz . . . . .	73
4.3.1. Description . . . . .	73
4.3.2. Architecture . . . . .	75
4.4. Itemboard . . . . .	78
4.4.1. Description . . . . .	78
4.4.2. Architecture . . . . .	79
4.5. Summary . . . . .	82
<b>5. Showcase Scenario . . . . .</b>	<b>85</b>
5.1. Pedagogical Background . . . . .	85
5.2. Structure and Appearance . . . . .	86
5.3. Process . . . . .	90
5.4. Summary . . . . .	91
<b>6. Evaluation . . . . .</b>	<b>93</b>
6.1. Research focus . . . . .	93
6.2. Methodology . . . . .	94
6.3. Participants . . . . .	94
6.4. Procedure . . . . .	95
6.5. Materials . . . . .	96
6.6. Results . . . . .	98
6.7. Discussion . . . . .	107
<b>7. Lessons Learned . . . . .</b>	<b>109</b>
<b>8. Summary and Outlook . . . . .</b>	<b>111</b>
<b>List of Abbreviations . . . . .</b>	<b>114</b>
<b>List of Figures . . . . .</b>	<b>115</b>
<b>List of Listings . . . . .</b>	<b>119</b>
<b>Bibliography . . . . .</b>	<b>123</b>
<b>Appendix . . . . .</b>	<b>133</b>
<b>A. DVD Content . . . . .</b>	<b>135</b>

# 1. Introduction

The internet offers many opportunities for students to pass the time these days. Social networks and online games are designed in a way that attracts young people and keeps them motivated for long periods of time. At the same time, today's students find common school education more and more boring as it does not meet their expectations of highly stimulating multimedia input they are used to from surfing the internet in their free time. As a logical consequence, educators have been asking themselves in the past how they might be able to use the power of computer games and community platforms to deliver educational content. Unfortunately, many early attempts failed since they were often designed by academics who understood little or nothing of game design. Although the outcomes of these development processes might have been pedagogical valuable learning tools in some cases, they could not be called "games" any more in a sense that they were not fun to play and did not keep people engaged (Van Eck, 2006). Such "serious" games, as they have been called furthermore lacked the collaborative aspect that young people liked from their social networks. In recent years, however, some progress in the field of multi-user virtual worlds has been achieved. Virtual environments have successfully been used for learning languages (M. Ibanez, Kloos, Leony, Rueda, & Maroto, 2011; Gardner, Gánem-Gutiérrez, Scott, Horan, & Callaghan, 2011; Berns, Gonzalez-Pardo, & Camacho, 2013), physics (Pirker, Gütl, Belcher, & Bailey, 2013) and history (Manuelian, 2013; Jacobson & Gillam, n.d.; Nicoletta Di Blas, 2003). Rudman, Lavelle, Salmon, and Cashmore (2010) state that collaborative virtual worlds are particularly suited for group-based learning and learning in context, meaning that students can learn in an environment that corresponds to the learning topic. However, the creation of suitable learning environments is not an easy task and certainly exceeds the capabilities of an ordinary teacher as it cannot be assumed that a teacher has knowledge of programming. That is why we want to present in this work an approach that allows anyone to easily create virtual learning environments without needing any programming skills at all.

## 1.1. Goals and Objectives

To achieve the goal of making virtual world creation feasible for teachers, we have developed a set of universally applicable learning tools that can be added to an existing world, extending it by possibilities to add educational context and learning

## 1. Introduction

tasks. The tools can be used for various learning scenarios in every conceivable context. An exemplary learning scenario, the Virtual Egypt World (VEW) has been created eventually to demonstrate the functionality of the implemented learning tools and to provide a realistic environment for an evaluation.

For the creation of the VEW, the virtual world building software Open Wonderland (OWL) has been used. OWL is open source and facilitates the creation of multi-user environments by providing features like high-fidelity, immersive audio, text and voice chat, and the exchange of documents. In addition to that, its extensibility allows developers to easily add their own tools or new behaviors for avatars and objects (OWL-Foundation, 2014a).

The implementation of the VEW in OWL included:

- The design of valuable educational activities and processes in an immersive, virtual learning environment.
- The implementation of a set of learning tools for OWL which enable and support these learning activities.
- The creation of an exemplary virtual learning world in OWL to present and evaluate these tools.

The development of the Egypt world should represent a first exemplary showcase of how the implemented learning tools can be used for the creation of virtual learning environments. Therefore, the focus was placed on a historical topic that relates to ancient Egypt, but in practice the tools can be used for learning scenarios in other contexts as well.

## 1.2. Structure of this Thesis

Figure 1.1 shows an overview of how this thesis is structured. First, some theoretical background along with examples of related work is presented (Chapter 2). After that, the development of the learning tools and the VEW is discussed in detail (Chapters 3, 4, and 5). Finally, a preliminary evaluation with test users is described and first results regarding three different research focuses are presented (Chapter 6).

Chapter 2 comprises a literature review on the topic of learning and the various forms of online learning. After a brief excursus on traditional teaching and learning methods the classical forms of e-Learning as well as serious games and multi-user virtual worlds are treated in more detail. The development of different technologies for the use of learning is illuminated and their advantages and disadvantages are discussed. Furthermore, some virtual world examples from practice are presented and it is shown how these examples are used for learning.

In Chapter 3 the requirements for our learning tools are identified and listed. On this basis OWL is subsequently selected as the appropriate platform for building



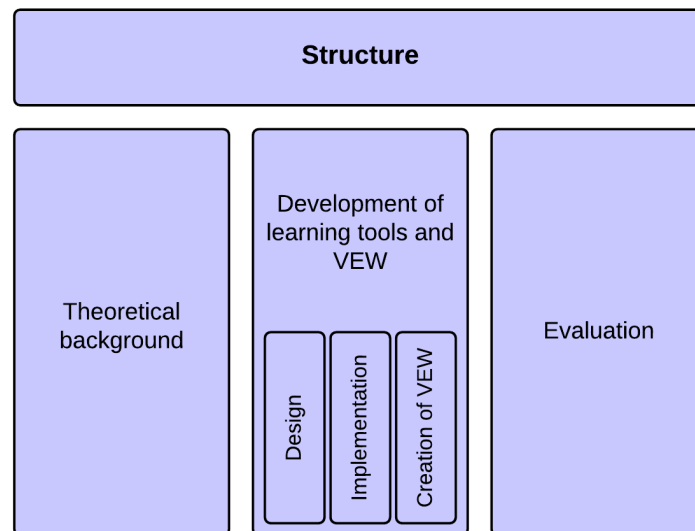


Figure 1.1.: Structure of this thesis. First, some theoretical background is presented, followed by the practical part of this work. There, the design and implementation of the learning tools are outlined and the creation of the VEW is described. Finally, the conduct and results of a preliminary user study are summarized.

our virtual world. Various reasons for this are given, including the extensibility of OWL and its built-in tools for collaborative work. To conclude the chapter, the conceptual architecture of the virtual learning experience and the tools developed for this are outlined. Special attention is paid to the exploratory and collaborative character of the learning experience.

Chapter 4 finally introduces the developed OWL modules, discusses how they were implemented according to the modular standards of OWL and describes their architecture in detail. For every developed tool a description of its functionality and an overview of its structure is given. Based on this chapter, a basic understanding should be established of how a continuous, collaborative, and exciting learning experience can be formed using the learning tools.

Chapter 5 deals with the design and creation of the VEW. For that purpose, the pedagogical background is explained, followed by a description of the structure and the appearance of the Egypt world. This description is also supported by numerous screenshots of the virtual environment. Also part of this chapter is an accurate depiction of the different stages of the learning experience students will undergo when they do the whole learning round-trip.

In Chapter 6, the conduct of a preliminary evaluation of the learning scenario in the VEW using the developed learning tools is described. The methodology and procedure are outlined and the selection of appropriate learning tasks is explained. Furthermore, the questionnaires, which are necessary for the collection of the three research focuses, motivation, immersion and usability are presented. At the end of Chapter 6, the results of the user study are summarized and discussed.

## 1. Introduction

Chapter 7 collects all findings and states some lessons learned. Last but not least, Chapter 8 gives a brief summary of this work and a short outlook. This contains suggestions for future work as well as general recommendations for the development of interactive, virtual learning environments.

## 2. Background and Related Work

Pedagogical concepts are the foundation of every teaching method, no matter whether it is a traditional or a computer-assisted approach. This chapter will give an overview of different possibilities of multimedia supported online learning, ranging from conventional e-Learning methods over game based-approaches to 3D virtual environments. Eventually, some successful examples from practice will be presented. But first, let us dedicate ourselves to traditional teaching methods as they have been known and used for many decades.

### 2.1. Teaching Methods

Teaching methods, there are a dime a dozen. One could probably spend a whole life and never find out about all of them. Nonetheless, there are a few core principles of how teaching can be designed. Figure 2.1 gives an overview of them. The different layers of the pyramid represent the amount of information that is retained by the students after a certain time, using the indicated activity or teaching method. However, the pyramid is only a theoretical model. The exact retention rates are disputed and Strauss (2013) even argues that the order of the methods could be shifted, depending on a variety of factors such as the learning material, the age of the subjects, or the method how memory was tested. In spite of that, most people would agree that active teaching forms tend to be better in terms of student understanding and memorization (Prince, 2004; Phillips, 2005; E. Wood, 2013). The order in which the teaching methods appear in the pyramid ranges from totally passive (lecture) to active (teaching others) and furthermore from activities done alone to activities done cooperating with others which is a good way to engage students and therefore enhances understanding (D. W. Johnson, Johnson, & Smith, 1991). In this section we will take a closer look at the following teaching methods:

- Lecture
- Demonstration
- Collaboration
- Learning by Teaching

As one can see we will follow the order of the pyramid and deal with the different teaching methods in the order in which they appear in the diagram. This does not mean, however, that we uprate one method but is simply done to have a

## 2. Background and Related Work

common thread. In this case, teaching methods will go from passive to active. Further evidence for the effectiveness of some of the methods mentioned can also be found in Bligh (1998).

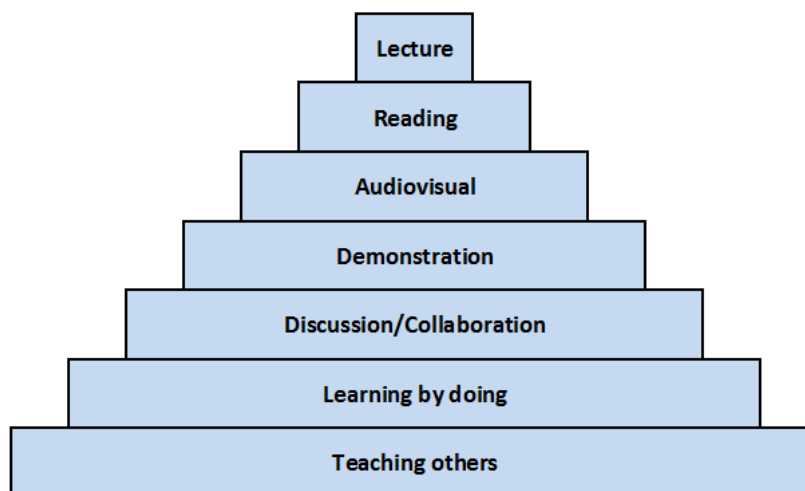


Figure 2.1.: The Learning Pyramid. Originally, the graphic comes from the National Training Laboratories (NTL) for Applied Behavioral Science where it was developed in the early 1960s. A similar diagram appeared in a book by Edgar Dale in 1954.

### 2.1.1. Lecture

The term "lecture" comes from a french word and means "reading". This form of teaching has its origins in times when there were no printed books for students. So in order to learn something, they had to listen to "lecturers" reading from their publications and try to write everything down. This was usually done on a grand scale in large auditoriums or lecture halls so that as many people as possible could benefit from the knowledge of the lecturer. Today this method is still used widely in schools and at universities, although many people seem to think that there are better ways to learn (Prince, 2004; Phillips, 2005; Bach, Haynes, & Lewis Smith, 2006). Many approaches to more active and student-centered teaching methods exist, but the problem is that they mostly work with small groups of students (see e.g. E. Wood (2013)). A practical alternative to lectures in a university environment with thousands of students has yet to be found.

The obvious benefits of lectures are that a large audience can be taught by one person and that the content, pace, and the way things are presented is entirely under the control of the lecturer. Furthermore, a skilled speaker might be able to arouse interest in his subjects (CIRTL Network, 2013). Bligh (1998), however, argues that lectures are ineffective for attracting interest in students. The author is of the opinion that a lecturer who might inspire one student does not necessarily have to have the same effect on other students and that a variety of different

presentation methods is necessary to keep students interested. That brings us right to one disadvantage of lectures, namely the need to have talented lecturers willing to invest time in diversified presentations. And while some schools or universities might have a bunch of them at hand, most institutions do not and this makes sitting in lectures a dull thing for students.

Maybe the most criticized thing about lectures is the one-way communication from lecturer to listener and the passive way in which information is received (Phillips, 2005; E. Wood, 2013; CIRT Network, 2013). Bligh (1998) further notices in his studies that attention of students declines after approximately 25 minutes. A short break was found to have a recovering effect on the level of attention but it will decrement quickly again as soon as the lecture resumes.

One approach to overcome the lack of interactivity and feedback that is inherent to lectures are so-called Classroom Response Systems (CRS), often also referred to as "clickers". Their purpose is to increase the interactivity between speaker and audience by allowing the audience to anonymously answer questions of the teacher. Therefore, the students simply click a button on small remote control devices which in turn send their answers directly to the teacher. The name of the students is not sent to the teacher which is supposed to raise student participation. Ioannou and Artino Jr. (2008) describe in their work the assessment of such a CRS.

The usual reluctance of students to participate in such activities was also noted by Elliott (2005). He mentions in his work the so-called "social loafing" effect which means that students believe their non-participation will not be noticed due to a large number of people. Another effect he found in lectures with large audiences is "free-riding". If the teacher asks the students for participation - either by asking questions directly or using a CRS - students tend to hand over their responsibility because they think that there are enough other students who will make contributions and theirs are therefore not needed. The author assumes that the reluctance to answer questions of the teacher without a CRS, but personally, might also be caused by fear of negative evaluation by the other students.

All in all there are many problems to overcome in traditional lecturing and it is certainly not the best method for teaching practical things such as behavioral skills (Bligh, 1998). Nonetheless, the benefits for teaching a large amount of students are indisputable and because of that, lectures are still part of university teaching.

### 2.1.2. Demonstration

*Showing* people how to do something rather than *telling* them how to do it is a powerful and often underused teaching method (Petty, 2004). Maybe it is even the most efficient that exist. Surely it is the oldest. Right from the beginning of time, before language exists at all, animals and people learned by imitation. So this teaching method does not require abstract language processing from students but

## 2. Background and Related Work

allows them to immediately implement what they have seen. Among others, McKee, Williamson, and Ruebush (2007) have studied the positive effects of the demonstration teaching method and have found that it enables students to understand new material more quickly and effectively.

As the name - demonstration means to "clearly show" - suggests, the technique is used to show off procedures. It is a direct way to illustrate concepts (Chamberlain, Kelly, & O'Neill, 1980). As a mixture of theoretical teaching and practical work it is supposed to have a more "lively" character than a simple lecture (World Health Organization, 1988). According to Eley and Norton (2004), a demonstration as a teaching method is most commonly used in science and technology courses. Petty (2004), however, is of the opinion that it can be used to teach any subject, be it physical or intellectual skills. For intellectual skills one may show good examples, for instance a well-written essay or a neatly solved mathematical equation. For practical skills the classical demonstration is applicable, where the teacher shows some kind of practical work to his students with the purpose to help them develop new abilities.

Before a demonstration is carried out, it is important that the teacher prepares his students accordingly. They should be familiar with the topic as well as with the theoretical background. Only then the demonstration should be started. To make sure that everybody is able to see properly, small group sizes are recommended. Another point to consider is making sure the students understand what is demonstrated. This can be done by explaining each step verbally, pointing out important things and difficulties that might occur. The students can be given observation sheets where they can write down what seems necessary to them. In any case the students should be able to imitate afterwards what they have seen. Before letting the students experiment for themselves, the teacher could repeat the demonstration but this time letting himself guide by his students e.g. by asking questions like "*What comes next?*" or "*Why is this important?*". That way, he can make sure that his students have completely understood the whole process. It is also a good way to involve students in the demonstration. Involvement engages the students more effectively in the process allowing them to easier remember the steps (Kim & Kellough, 1974; Petty, 2004).

It is commonly found that children learn better through imitation. Therefore, it is very important to really *do* what you are saying because students tend to remember the things you *do* rather than the things you *say*. The fact that students rather watch movements than listen to explanations is also often used in sports education. The teacher - or in this case the trainer - shows a specific move he expects his students to imitate. If he cannot do the movement himself, he may also ask a talented student to demonstrate the skill. Students can imitate the movement better after they have seen it (Child, 1981).

Naturally, not all classroom activities can involve demonstrations. They take time to prepare and carry out. Furthermore, not every school or university may have appro-

priate rooms or equipment for e.g. science experiments (Farooq, 2013). Nonetheless, demonstrations are a useful addition to school routine and can be used together with other teaching methods.

### 2.1.3. Collaboration

Collaboration describes the process of two or more individuals working together to achieve a common goal. Consequently, collaborative learning occurs when two or more individuals work together to achieve a mutual learning goal. This could be the understanding of a complex mathematical process, the ability to interpret two of Shakespeare's most popular plays or memorizing thirty new words of a foreign language (D. W. Johnson & Johnson, 1991).

The important aspect is that rather than one person acquiring the knowledge on their own, two or more people contribute to the same goal by sharing knowledge and resources, discussing possible solutions, explaining concepts and helping each other and thus developing a deeper understanding and longer memorization of the learned subject. Articulating and defending their own ideas and constructively criticizing those of others enhances the logical reasoning and critical thinking skills of students along with providing them with the valuable experience of having one's ideas criticized without being criticized as a person (D. W. Johnson & Johnson, 1991). Furthermore, because of the mixture of solution approaches of different people the outcome of collaborative learning often exceeds what would have been possible for one individual alone (Dillenbourg, 1999).

Collaborative learning is a student-centered approach. That means the role of the teacher changes from being the only source of knowledge to a facilitator who guides the students in acquiring knowledge and finding solutions on their own by working together with fellow students. For this kind of learning to be most effective all participants should have sufficient social skills. But collaborative learning tasks can also provide an opportunity to train these skills in a pressure-free environment where nothing more than one's grade are at stake. In today's workplace, social skills are implicitly expected of a new employee since most companies have discovered the enhanced productivity of groups working together to solve a problem, develop new ideas or work out a plan.

People tend to be more productive when working in groups. This is because collaboration comes naturally to human beings (D. W. Johnson & Johnson, 1991). Millions of years ago hunting groups turned out to be more successful than people hunting alone. People organized in settlements and villages had better chances to fend off attackers and keep everyone fed than individuals. Today, as an example, families are a still highly cooperative construct: Raising children proves to be much easier when the father and the mother share the responsibility and take different, complementary roles to ensure the well-being of their offspring.

## 2. Background and Related Work

In addition to the benefits and the natural disposition of humans to collaborate, students welcome the approach of collaborative learning as a change to the normal school day where competitive structures are still omnipresent. In a competitive learning environment students try to best each other and see their benefits in the failing of others. When working in a group quite the opposite is the case: Since the goal can only be reached by working together students learn to rely on, listen to, and help each other. In groups where the diversity of gender, race, and skill level is high collaborative learning leads to greater diversity understanding and improves racial attitudes (Swing & Peterson, 1982) as cited in (Laal & Ghodsi, 2012). Often, learning groups continue to talk and work together even after the task is finished (Bean & Weimer, 2011). This leads to greater cohesion and a better climate in the classroom.

Collaborative learning tasks can also help to reduce problems with troublesome students (D. W. Johnson & Johnson, 1991; Messick & Mackie, 1989). Shy or disliked students are embedded in an environment where they feel liked and respected. In a learning group they experience the feeling of trust and belonging. Even disruptive students develop responsibility for each other in a group (Bonoma, Tedeschi, & Helm, 1974). When there is a good group dynamics, the intrinsic motivation is high, which means students are highly motivated by themselves, they want to learn the subject and additionally feel the urge to perform well in order to please their group members. Moreover, student-student conflicts are reduced as students learn to solve problems among themselves on their own (D. W. Johnson & Johnson, 1991). On top of that, students perceive collaborative learning as more enjoyable than working alone. Even repetitive tasks become more fun when they are preformed together with fellow students (Panitz, 1999).

To ensure that competition among students does not occur, reward structure must change from a competitive reward structure (where students are graded on a curve) to a norm-based one (D. W. Johnson & Johnson, 1991). Of course, the grade of individual students can also be linked to the overall performance of the group. This is called positive reward interdependence and will be discussed later.

Collaborative learning is most effective in small groups. D. W. Johnson and Johnson (1991) have found out that the best size for learning groups is between two and six, with two or three members if the students are not so experienced with group work yet and five or six students when they already have adequate experience with working in groups. Another finding of their research is that heterogeneous groups perform better than homogenous ones with greater learning benefits for low-achievers, as well as high-achieving students. Forming heterogeneous groups also seems to be a good idea in terms of reducing social and/or racial problems within the class.

A common mistake of teachers is to believe that collaborative learning will happen automatically when they put students together in groups. Most of the time this is not the case (Kreijns, Kirschner, & Jochems, 2003). What is needed for students



in order to collaborate constructively towards a goal is positive interdependence. Positive interdependence denotes the "swim or sink together" feeling of students, that is the perception that they can either achieve their goal as a group or fail to do so. A single student can only succeed if the group does and vice versa. In other words, the work of a single student benefits every other one in the group and the work of all the other students benefits a single one (D. W. Johnson & Johnson, 1991).

There are several ways to create positive interdependence (D. W. Johnson & Johnson, 1991; Slavin, 1980). The most important of them are:

1. **Positive goal interdependence** Exists when students perceive that they can reach their goal only if all other group members reach their goals as well. To support this the teacher could for example ask for a single product (e.g. a report all group members sign or an answer all group members agree to) or he could give the group the task to make sure that all group members learn a set of predefined vocabulary and then check by querying a random student of the group.
2. **Positive reward interdependence** Exists when the overall achievement of the group influences the rewards (grades) of a single student. A possibility to structure this could be individual tests for the students after the group work but with the possibility to get bonus points if all group members reach a certain number of points on their tests. Such joint rewards also contribute to goal interdependence.
3. **Positive resource interdependence** Successful collaboration can also be ensured by not giving every student access to the complete information necessary to achieve the group goal. Instead, each student gets only a part and is subsequently responsible for teaching his part to the other group members. This procedure is often referred to as "Jigsaw" procedure (Aronson et al., 1978).
4. **Positive role interdependence** Finally, one can promote collaborative learning by using roles. That is every student gets a specific responsibility necessary to complete the group work. Some of these roles could be: *Writer, Explainer, Encourager, Timekeeper*, and so on. For more information see D. W. Johnson and Johnson (1991). Roles should be switched in every new group work so that each student gets the possibility to gain experience in different roles.

Many authors agree that collaborative learning can be more effective than individual work when it comes to critical thinking, social skills and depth of understanding (D. W. Johnson & Johnson, 1991; Slavin, 1980; Panitz, 1999; Gokhale, 1995). Therefore, certain conditions must be fulfilled. First, positive interdependence has to be created.

Second, promotive interaction has to be structured. Promotive interaction takes place when students encourage and assist each other, exchange needed resources and provide each other with feedback. Positive interdependence provides the ideal context within which promotive interaction can take place.

## 2. Background and Related Work

Third, individual accountability has to be established. Every student should feel responsible for the success of the group and no student should be able to rely on his group members and let them do all the work. In order to obtain this, either each or a random student of the group could be tested after finishing the task. This prevents free riders and makes sure that every student of the group has mastered the subject. The smaller the group size the higher the individual accountability is.

The fourth essential element of cooperative learning are social skills. If students do not have the skills to communicate effectively, make decisions collaboratively and manage conflicts constructively, groups cannot function productively. In that case, students have to be taught these small group skills and given time to practice before assigning them any group task. As mentioned before, social skills are indispensable in today's working life and should be trained in school and university.

Last, it has to be ensured that group processing occurs. Group processing is the term for reflecting how well the group worked together after finishing an assignment. After every group work some time should be taken for reflecting in order to improve the group's future effectiveness. Some things for group members to discuss during this phase could be: describe which actions were helpful and which were unhelpful in order to achieve the group goal, in which cases the communication worked and in which cases it did not, if there were any conflicts and how they were dealt with, and in general what actions should be continued or changed next time.

### 2.1.4. Learning by Teaching

Learning by teaching is not a new concept. It has been applied throughout history, even though it was not seen as an innovative teaching method back then, but rather used out of necessity because there were too many students who wanted to learn and too few teachers for all of them. It turned out to be not very successful though, which might be due to the fact that there was no supervision by an educator. This is, of course, not the case today. The modern form of learning by teaching has been shaped by the German Professor Jean-Pol Martin (Martin, 1985). The term he uses in his work is "LdL" which stands for "Lernen durch Lehren" and is German for "learning by teaching". He tried out the method in his French classes and found it worked very well. Since then he worked on refining it and went to great lengths to make this way of teaching popular in Germany. Although Martin used the learning by teaching method originally for language teaching, he is convinced that it can be used for any subject.

One of the key concepts of learning by teaching, which also distinguishes it from pure student presentations, is that the method of knowledge transfer is left to the students. The teacher can suggest some methods they may possibly use but in the end it is up to the students how they teach the assigned topics to their fellow classmates. Of course, the teacher should not start by letting them hold whole

lessons when he introduces the concept of learning by teaching for the first time. Instead, especially younger students should be introduced slowly to the method so that they can get used to working individually or in small groups and speaking in front of the whole class. In general, group sizes should be held small so as not to overwhelm the students but on the other hand teach them that work can be done more efficiently if it is divided among several people (Barnbeck & Neumann, 2005).

Many teachers are suspicious about this "new method of teaching". They fear that they lose control over their students and that the lessons held by them will have a poor quality. While this might be true in some classes, a large majority of teachers that work with the learning by teaching approach report very positive developments in their classes (Barnbeck & Neumann, 2005). Martin and Oebel (2007) stress that it is very important for the teachers to understand that their role changes in the learning by teaching approach. It changes from the teacher being the only source of knowledge to a teacher who only organizes the transfer of information to knowledge but leaves the rest to his students. This has the consequence that the students deal much more actively with the learning material to construct their own knowledge and therefore gain a deeper understanding than what would have been possible by simply listening to the teacher. And, of course, understanding the learning material is crucial, when you want to teach it to others.

This learner-centered alternative teaching method which actively involves students in the design of lessons has celebrated many successes with the teachers who use it in their classes. Especially in language classes, learning by teaching was found to improve the performance of students (Martin, 1985; Pfeiffer & Rusam, 1992). This is most likely due to the fact that the students' role switches from passive to active and that they speak up to 80 percent more than they would normally do in a typical lecture. But this is only one of many benefits Barnbeck and Neumann (2005) have found when they questioned 90 teachers about their experiences with learning by teaching. Beneath the already mentioned intense engagement with the learning topics, students are also most of the time more motivated when they are able to design their own lessons. It is an alternative to their everyday learning routine and they enjoy having responsibility. In fact, there might not be another possibility of assigning more responsibility to the students (Martin, 2002).

It is, however, not enjoyed by all students and teachers. Especially the more introverted students may not like the thought of speaking in front of the whole class and also the students are well aware that preparing a lesson for their classmates can result in more work than they would normally have. The teachers, too, have to spend more time preparing the lessons. Besides introducing the topic in general, they have to split it into subtopics suitable for a group of students, prepare teaching materials, think of pedagogical hints they can give their students, consider their social skills, and so on. Finally, in some classes the approach might not work at all, due to their social structure, their unwillingness - or inability - to work individually

## 2. Background and Related Work

or their reluctance to speak and present in front of their classmates.

While in some small percentage of classes this might be a real obstacle when introducing the learning by teaching concept, in most classes students will adapt quickly and develop according social skills. If teachers take their time and train their students in group work first, they can learn to work independently, read out small texts in front of the class and be introduced slowly to speaking freely and developing their own presentation techniques. This takes some time, but is worth the effort when students can exploit the full benefit of the learning by teaching method afterwards. When the time for grading comes, the teacher must consider two different things: (1) how well has the student understood the assigned topic and (2) how well has he taught it to his classmates. So one could say that with the learning by teaching approach not only knowledge but also social competences are conveyed and assessed. Already Seneca the Younger (c. 4 BC – 65 AD) had realized that "*by teaching, we learn*". And although he certainly was not familiar with the words "soft skills", "team player", or "leadership", it is exactly those social skills which learning by teaching, among others, conveys and which are a real benefit in today's job market.

### 2.2. E-Learning

The teaching methods presented in the previous section are all field-tested and work well in schools and at universities. But the problem with these conventional approaches is that in order to learn one has to be present at a certain time at a certain location. In today's networked time, it cannot be taken for granted that this is possible for everyone. Maybe someone wants to learn in the evenings, when he comes home from work. Or someone wants to learn with his online friend from India who happens to have the same interests. That's where e-Learning comes into play. E-Learning describes all forms of learning where the learner is supported by some kind of electronic technology or digital media. This includes the use of the internet for example for the distribution of learning content or to learn together with friends. In fact, the internet is the technological foundation of e-Learning, which has its roots in the early days of computer conferencing. Maybe for this reason e-Learning is also often called "distance learning". However, Garrison (2011) is of the opinion that the term "distance-learning" is misleading and that e-Learning does not *always* have to involve great distances between learners and lecturer. He prefers to distinguish between two different types of e-Learning: (1) online learning and (2) blended learning.

Online learning takes place exclusively via the internet and can therefore rightly be called "distance-learning". It is characterized by asynchronous communication and interactivity, making use of e.g. e-Mail and discussion forums. Blended learning on the other hand is a mixture of online and face-to-face experiences. The author

believes that this is a much better approach because personal interaction either with the lecturer or with fellow students enhances commitment and motivation for a course. Furthermore, it is harder to slip out of the back door when being physically present in a lecture than to close a browser window.

One approach of blended learning are the so-called "flipped classrooms" (Treeck, Himpsl-Gutermann, Robes, et al., 2013; Fowler, 2013). This means that students get the material to prepare themselves for classes at home and the lecture time is used for activities and practical exercises. The benefit of this method is that the teacher can answer questions immediately and assist the students with the practical work, which is often the most difficult part. Of course, this approach only works if students really prepare in advance. For preparation, the teacher can for instance provide video recordings of last year's lectures. This has the further advantage that students can go through the material at their own pace, stopping or rewinding at certain points.

The advantages that Garrison (2011) sees in e-Learning is the formation of so-called "groups of learners". Students in such groups actively engage with and reflect on the learning material which deepens understanding. E-Learning allows learners to be completely independent - something not heard of in the past - while at the same time being highly connected with others and able to communicate any time.

In addition to Garrison's theoretical classification of e-Learning approaches, Taraghi, Ebner, Schön, et al. (2013) have looked at some practical implementations. In their work they mention, among others, the following four types of e-Learning:

- Web-Based Training (WBT)
- Learning Management Systems (LMS)
- Personal Learning Environments (PLEs)
- Massive Open Online Courses (MOOCs)

**Web-based training** Usually, web-based training (WBT) consists of several self-contained learning units, which can be worked on by a learner. Nowadays, it is not uncommon that units also include multimedia content like video or audio files to enrich the learning experience. But apart from the visual enhancement there is not much difference to textbook learning.

**Learning management systems** Learning Management Systems (LMS) on the other hand support the organization of learning processes by not only providing learning content but also evaluation and assessment tools, administrative support of teachers, as well as communication tools for teachers and students. The market for LMS is huge; nearly every educational institution is in possession of one. One famous example is *Moodle*<sup>1</sup> with over 50.000 registered sites (Moodle.org, 2015).

---

<sup>1</sup><https://moodle.org/>

## 2. Background and Related Work

Nonetheless, the full potential of these platforms is often not exhausted as they are often only used for uploading learning materials.

**Personal learning environments** A more individual form of e-Learning are personal learning environments (PLEs). PLEs are systems in which learners can integrate and manage distributed online resources, tools from other systems, or personal contacts centrally while having a lot of freedom on the content. PLEs are all about self-directed and individual learning with no predefined learning content. This makes them an ideal "management system" not for teachers, but for students themselves, since they can choose their own sources, tools and ways of learning. On the contrary, students must be competent in the use of computers and the internet to be able to use such a system and it is not easy for teachers to assess the learning progress when the student uses a PLE.

**Massive open online courses** Finally, the "major trend" of the last few years have been massive open online courses (MOOCs), meaning free online courses offered by various universities for the general public. Anybody can participate, as long as he is registered at the online platform of the course provider. Teaching material is often presented in form of short video sequences, followed by multiple-choice tests. Sometimes also some simple course management software and communication tools are included. The courses follow strict time schedules, dates for lectures are fixed and the courses are over after a predefined number of weeks. Many courses offer some kind of unofficial certificate to those who passed. The popularity of MOOCs reached its peak in 2012 when sometimes over 100.000 listeners sat in the virtual lecture halls. Currently, *Coursera*<sup>2</sup> is the most popular MOOC platform on the internet. However, more and more people criticize MOOCs, especially because of their low completion rates of less than 10 percent (Colman, 2013; Lewin, 2013; Catropa, 2013; Jordan, 2013; Fowler, 2013). A reason for that might be that many people who register for a course are only "window-shopping" without really planning to finish it (Jack Smith IV, 2015). As further shortcomings of MOOCs the amount of time and work needed to finish a course, the fact that some courses are simply boring, bad pedagogical concepts, and the claim that MOOCs are just another form of e-Learning were listed. However, there are also benefits such as the fact that learners can be reached which would otherwise not be able to attend a course at university and that - as it is the case with any form of e-Learning - learning is possible independent of time zones and physical boundaries (Dräger, 2013; Kolowich, 2013; MoocGuide, 2011). Levy (2011) presents four "lessons learned" from his experiences with a MOOC, namely (1) that learning is possible in such a course, (2) that learning often occurs through discussions with other users, (3) that learning is possible even when there is no assessment at the end, and (4) that a daily reminder in form of e-Mails or something similar really helps with motivation.

---

<sup>2</sup><https://www.coursera.org/>

To sum it up, e-Learning in general has some real benefits which took learning to a new level regarding communication, collaboration and interactivity. Disadvantages like the need for a computer with an internet connection or the time it takes some students to get used to the systems are increasingly disappearing in today's time. Even solutions for individual accountability of students have been found by using identity-verification programs (Fowler, 2013). What remains, however, are problems with engagement, which might be due to the lack of interpersonal communication (Knowledge@Wharton, 2012; Fowler, 2013; Jack Smith IV, 2015). And if people do not feel engaged, they lose interest and stop learning. In an attempt to overcome this problem, Brown and Cairns (2004) identify engagement as "*[t]he first stage of immersion*" and Kickmeier-Rust et al. (2007) take it even further and suggest using immersive computer games to make learning more engaging. Therefore, the next sections will introduce the concepts of serious games and collaborative virtual worlds to overcome the shortcomings of traditional e-Learning.

## 2.3. Serious Gaming

According to Muratet, Torguet, Jessel, and Viallet (2009) and Susi, Johannesson, and Backlund (2007), a serious game is a computer game whose purpose differs from pure entertainment. This definition makes perfectly sense regarding the fact that "normal" computer games are usually played for pleasure. Especially the young generation enjoys playing computer games for recreation in the evenings and on weekends. In these situations, parents often find themselves in the position where they have to remind their children to stop playing since they seem to be carried away by the game having completely lost track of time.

In contrast to that, students often find the classical lectures at school or at university "boring", no matter how hard the teacher or professor tries to arouse interest. Computer games are known to have a motivational and engaging effect due to their immersive and challenging character in combination with their short learning curve and an instant feedback/rewarding system (Siang & Rao, 2003). The challenge, that Siang and Rao (2003) identify is to merge learning and games so that the benefits of games can be used to learn about particular topics or gain practical insight into a specific domain. Likewise, Corti (2006) has emphasized in his article that serious gaming "*is all about leveraging the power of computer games to captivate and engage end-users for a specific purpose, such as to develop new knowledge and skills*".

Prensky (2003) noticed that in computer games learning seemed to occur naturally and that players do not have to be forced to learn but rather enjoy the process. Following this line of thought, Greitzer, Kuchar, and Huston (2007) suggest to use computer games for learning purposes by infusing instructions into game play. According to them, this has to be done via pedagogy. But in contrast to edutainment they as well as Zyda (2005) believe that the instructions have to remain subordinate

## 2. Background and Related Work

to story and that entertainment has to come first. There has to be a fixed storyline of the game and this has to be used to pursue the learning objective (Muratet et al., 2009).

In addition to the story, also the art and the software are important for a computer game. A developer team has to have different groups of specialists such as programmers who are responsible for implementing the game logic, the database connection, networking, an Artificial Intelligence (AI), and so on, and artists who are responsible for the look and feel of the game. In the end, it all has to blend together to form the finished game. To create a serious game, instructional scientists as well as subject matter experts have to work closely with the developer team. Too many games have been created that claim to be innovative e-Learning applications but suffer from passive learning paradigms and linear teaching methods. This is usually due to non-experts designing the learning tasks (Zyda, 2005; Greitzer et al., 2007).

The mistake of many of those systems is that despite sophisticated multimedia features, in the end they again often only present students with information they have to read through and memorize because later on they are tested on it. To avoid this pitfall, Greitzer et al. (2007) promote in their article the use of active learning paradigms and student-centered approaches when developing a new game. Therefore, they present a set of instructional design guidelines, some of them are summarized here:

- Students should be able to link new learning material to existing experiences. That way, it can be stored more effectively in the brain and can also be retrieved more easily.
- To facilitate information absorbance and, again, storage in the brain, learning material should be organized in small chunks. Furthermore, complexity should rise gradually from simple to difficult topics.
- Learners should have the feeling that they are working on a realistic task that is relevant for their every-day life. Small, constructed examples with a strictly limited set of possible solutions tend to not being taken seriously by the learners, who subsequently lose interest. Also, tasks have to be immersive and interactive to enable active processing of the information and therefore a more lasting learning experience.
- Taking into account that the learner is both, easily overwhelmed as well as unchallenged, the goal is to provide a learning experience that is in the "narrow zone" between those two extremes. In other words, the learner has to be perfectly engaged, finding the tasks neither too easy, nor too difficult.

The first and most successful serious game was *America's Army*<sup>3</sup> developed in 2002 by the Naval Postgraduate School on behalf of the United States Army. It is a first person tactical shooter game providing insight in the work of a soldier in the U.S.

---

<sup>3</sup><http://www.americasarmy.com/>



Army with the purpose of arousing interest in children and young people and to encourage them to consider such a career. Originally, America's Army was intended only for recruitment purposes but now the free multi-player game is sometimes also used for training (Zyda, 2005) and, more importantly, played by people all around the world even receiving the title of "*Most Downloaded War Game*" in 2009.

A game review from the year when the game came out describes America's Army as "*the most realistic portrayal of weapons and combat of any game*" (GamesFirst! 2002). Numerous versions were rolled out since then, including versions for Xbox and Xbox 360, arcade machines, and smartphones. The great success of the game led to a rethink in the scientific community concerning the usefulness of games and their benefits for learning new skills (Zyda, 2005; Susi et al., 2007).

In line with their proposed instructional design guidelines, Greitzer et al. (2007) describe in their article their work on a serious game called *CYBERCIEGE* in which players have to learn all about the world of cyber security to protect a fictional network. Different levels of varying difficulty should keep the learner motivated and help him understand one concept at a time. They range from very easy ones where learners only have to make a few adjustment to an otherwise already solved scenario to levels where they have to protect the network of a big company introducing all policies from scratch. Through an evaluation of their game, Greitzer et al. (2007) learned that such a layered approach is very important in order not to overwhelm the player in the beginning. For a satisfying learning experience, they further recognized that individual feedback for every step the learner does and a certain degree of adaptability of the system is necessary. Once familiar with the system, players also desired a multi-player capability to play the game against friends.

Another serious game successfully tested in a university environment is *Prog & Play* which was developed by Muratet et al. (2009). After an extensive literature research the authors found out that there was no serious game on the market which would help their students learn programming. Existing games were either not customizable by the teacher or did not include the programming in the game play. So they decided to develop their own game, using the Spring Real Time Strategy (RTS) engine. They extended the existing game *Kernel Panic* in a way that their students were able to program their own little "bots" which were supposed to support the students by taking over routine tasks in the game leaving the students more time to concentrate on important things. The more complex and intelligent the bots were, the more tasks they could take over and the higher was the possibility that the student would win. In addition to being fun for the students, the game was also completely adjustable by the teacher regarding the programming language and the tasks the students had to perform. A survey the authors undertook later showed that the game was widely accepted by students given that they had enough time to experiment and find their own solutions (Muratet, Delozanne, Torguet, & Viallet, 2012).

## 2. Background and Related Work

Finding your own solutions, immediate feedback, and the possibility to challenge friends appear to be among the most important factors when it comes to making learning fun. Siang and Rao (2003) further identify the fact that games usually do not require long training periods but can be played almost immediately as a significant motivational factor. Players like to find out the right approach by trial and error. This is called "Constructivism" and requires just the right amount of hints so that on the one hand the right solution is not revealed immediately but on the other hand the player also does not get frustrated by making no progress over a longer time period. The key is to challenge the player to keep him motivated. Prensky (2003) formulates it this way: "*A motivated learner can't be stopped.*" And to keep the motivation up should be the goal of every serious game.

To formalize what "motivation" in games really means, Siang and Rao (2003) adapted Maslow's pyramid of needs in order to show what game developers have to achieve to maintain constant motivation of players. Their version of the pyramid can be seen in figure 2.2.

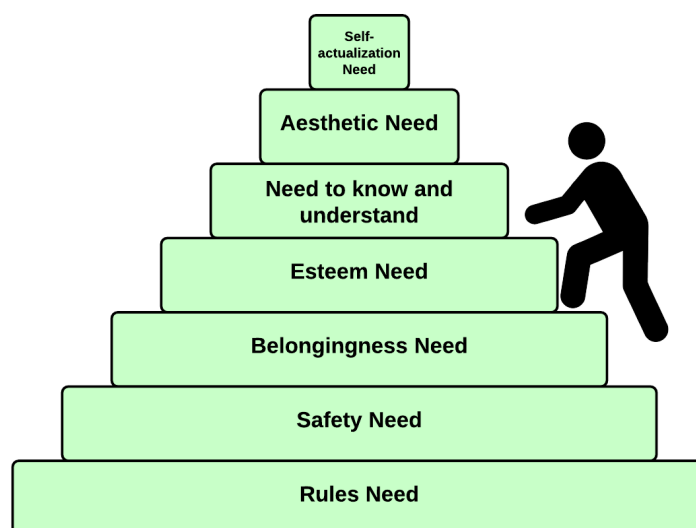


Figure 2.2.: Hierarchy of the player's needs by Siang and Rao (2003)

At the very beginning, players must understand the rules of the game. Without this basic requirement, they will lose interest within a few minutes and stop playing. Once familiar with the rules, it is important that players are able to remain in the game without losing. They have to have enough information about the game to feel secure in a certain position and to avoid being knocked out immediately. The next factor which has to be taken into account is a need to belong which means that the players have to feel comfortable playing the game knowing they are eventually able to win it even if it takes a long time.

While "being able to win" is the goal of the *Belongingness need* layer, the *Esteem need* layer is all about mastering the game. Players want to have full control over the game to win as "elegantly" as possible. After that, they want to know even more

about the game in the next layer such as what the special weaknesses of different opponents are, where secret items are hidden, and so on. As they become more familiar with the game, players start searching for special challenges apart from the normal game play. Only then the graphical appearance of the game plays a role for them. *Aesthetic need* means they strive for realistic graphics and animations as well as a good soundtrack.

Last of all, players want to realize themselves in the game. They want to be able to do anything as long as it is within the rules of the game. An example for that is a special mode that Tetris players invented in the 80's to challenge themselves. The goal is to create a completely intentional diagonal pattern of holes in the stack of Tetris game pieces which requires great skills at the game. See figure 2.3 for how this looks like.

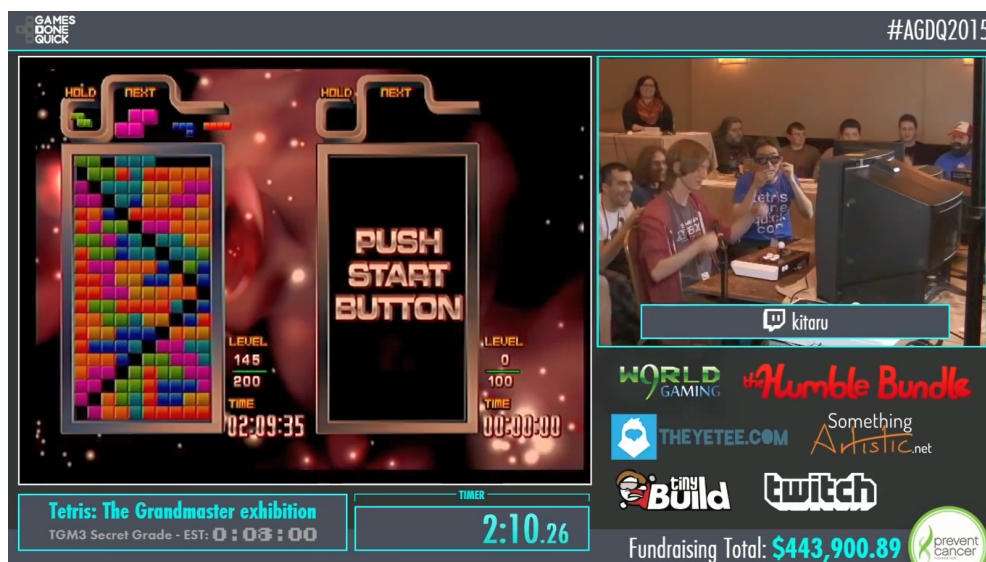


Figure 2.3.: Screenshot from the "Awesome Games Done Quick" (AGDQ) 2015 Charity Marathon Event where a player named "kitaru" creates the diagonal Tetris pattern

The research on serious games is not over yet and so far no satisfactory answer to the question of what makes a good serious game has been found. Greitzer et al. (2007) emphasize the need to create a "science of games" - a set of scientific and technical methods to develop games and get an understanding of them. The goal is to focus on pedagogical approaches to be able to create motivating learning experiences. In addition to the challenge of a good design, Schell (2014) identifies further difficulties when designing games for students. Firstly, implementing an interactive, visually attractive game is certainly very difficult for ordinary school teachers and to hire a company for it is expensive and may not be affordable for every institution. What is more is that playing a game can be time-consuming and may not fit in the time frame of a traditional lesson. Finally, most scripted learning games have a fixed procedure that is designed to be worked through by one student. Though the demand for social functionalities is high, possibilities to

## 2. Background and Related Work

collaborate with others are often rare in learning games. It might be that this type of learning requires a different environment. Virtual worlds, which have been found to provide a good platform for collaborative group work, will be introduced in the next section.

### 2.4. Virtual Worlds

Interest in virtual worlds used for education has been high in recent years (Bainbridge, 2007; Berger, 2012; Pirker et al., 2013; Riedmann, 2014). Berger (2012) for instance has found that *"virtual worlds are closely linked to collaborative learning as well as to simulations and video games."* He states in his thesis that virtual worlds can be used as a tool for group-based learning and problem-solving. However, he also mentions that virtual worlds with all their features have to be used properly to be beneficial to education. This section attempts to clarify how virtual worlds are defined, which properties are peculiar to them, and how they can be used for learning.

#### 2.4.1. Terms and Definitions

In the past, many authors have tried to find a comprehensive definition for virtual worlds (Bell, 2008; Bartle, 2004; Reis, Escudeiro, & Escudeiro, 2010; Holmstrom & Jacobsson, 2001; Bainbridge, 2007). Among these definitions certain terms seem to appear repeatedly. These are for instance:

- **Persistent**  
This term means that the whole world with all its objects will stay the same even if no users are logged in. All the data exists somewhere on a server, independently of any user. If someone logs in again after some time, he will find the world in the state he has left it, given that no other user has manipulated it in the meantime.
- **Synchronous**  
It is important that all users see the same representation of the world and that events which affect this representation happen for all of them simultaneously, otherwise they would not be able to communicate and collaborate.
- **Multi-user**  
A virtual world is characterized, among other things, by the fact that many users can be logged in at the same time. That way, people can work together feeling that they are "in the same place", even though they might be thousands of kilometers apart.
- **Avatars**  
To get the feeling of really "being there together with other", users in a virtual world are represented by avatars. Those are graphical representations of a real

person. When the user moves in the virtual world, the view of the world as it appears to him changes. At the same time, other users see his avatar change the position according to where the user has moved.

- **Network**

The last important term which appears in many definitions of virtual worlds is "network". Multiple users can only be logged in together and meet in the same virtual world if some kind of network connection exists between their computers. Today, this connection will most of the time come in form of the internet so as not to forfeit the benefit of working together in spite of being geographically dispersed.

Bell (2008) summarizes these scattered concepts together in his definition of virtual worlds, which reads as follows: "*a synchronous, persistent network of people, represented as avatars, facilitated by networked computers*". Novak (2012) identifies in his work further principles that in his opinion characterize virtual worlds: (1) a graphical user interface, either in 3D or in 2D, (2) the possibility of interaction with other users as well as with objects, and (3) personal user-experiences which come from being *immersed* in the virtual world.

While all these terms and definitions give a rather good impression of what a virtual world is and what it is not, some readers may not have a clear concept of them yet. In fact, the definition of virtual worlds does not include what purpose they should serve. One application which should be known to most readers are online games. Massively Multiplayer Online Role play Games (MMORGs) like *World of Warcraft (WoW)*<sup>4</sup> or *Guild Wars*<sup>5</sup> enjoy great popularity with WoW having over 10 million subscribers in November 2014 (Kollar, 2014). They fulfill all criteria for virtual worlds being a persistent, synchronous online platform where multiple users can meet in form of avatars and interact with each other or the environment e.g. by accomplishing quests. Their focus lies of course on entertainment and online role-playing. The users follow given storylines and act in accordance with progression-based systems like leveling up their characters, which is the reason why some authors do not consider MMORGs virtual worlds (Barnes, 2010; Novak, 2012). Others see in such games the foundations of all other kinds of virtual worlds (Sivan, 2008; Messinger et al., 2009). Messinger et al. (2009) sees the roots of today's virtual worlds also in social networks due to their capability to create a "profile" (similar to an avatar) with information about oneself and to define a circle of "trusted friends" to communicate with. Supporting this opinion, De Freitas (2008) claims that the boundaries between gaming worlds and virtual worlds used for other purposes are blurring. Marcus (2007), Novak (2012) use the property of a given storyline and a progress-based accomplishment system to distinguish between "*leveling games*" which are driven by some goal and virtual worlds for social purposes, where a goal might be present but is not in the foreground and users have much more

---

<sup>4</sup><http://eu.battle.net/wow>

<sup>5</sup><http://www.guildwars.com>

## 2. Background and Related Work

freedom to socialize and be creative. Examples for such worlds are *Second Life*<sup>6</sup>, *Open Wonderland*<sup>7</sup>, or *OpenSimulator*<sup>8</sup>. Especially *Second Life*, which was launched in 2003 was very popular among its users. In general, there was a big hype about virtual world platforms from 2003 to 2008 (De Freitas, 2008), but interest has stagnated since then (OECD, 2011). There were, however, in recent years some attempts to use virtual worlds as learning environments. The following subsection elaborates on the use of virtual worlds for learning and tries to identify some advantages over conventional e-Learning approaches.

### 2.4.2. Virtual Worlds in Education

Virtual worlds have been used for educational purposes in the past. For instance, M. B. Ibanez et al. (2011) present in their work an approach for foreign language learning by using situated and collaborative learning in a setting which resembles the city of Madrid in order to immerse the students and that way improve the learning results. Moschini (2010) states, "*Communication and social interaction are at the centre of virtual world social experience. Virtual worlds therefore present an ideal platform for the engagement of learners in constructivist-focused educational practice.*" Likewise, D. Wood, Hopkins, et al. (2008) are convinced that "[*simulated learning experiences*] can prepare students for future employment without the constraints of 'real world' industry placements". In the following some attributes which make virtual worlds suitable for learning are listed.

**Immersion and Presence** A great advantage of virtual worlds over 2-dimensional learning environments is an increased perception of *presence* and *immersion*, which means the user has the feeling of really "being there". Immersion and presence are two very related concepts, although some authors like to distinguish them. For instance Slater (2003) and Dalgarno and Lee (2010) define immersion as a measurable characteristic of the world, dependent on the technical capabilities to render sensory stimuli, whereas presence in their opinion is the subjective reaction of an individual to immersion, dependent on the personal perception of the environment. That means two different people can experience a different level of presence in a virtual world, although the property of immersion is the same.

**Socialization and Collaboration** Strongly related to the feeling of presence is the individual perception of *social presence* (Kreijns et al., 2003), *awareness* (De Lucia, Francese, Passero, & Tortora, 2009; Gütl, 2010), or *co-presence* (Dalgarno & Lee, 2010) in a virtual world. All three terms (for the remainder of this section *social presence*

---

<sup>6</sup><http://secondlife.com/>

<sup>7</sup><http://openwonderland.org/>

<sup>8</sup><http://www.opensimulator.org/>

will be used) refer to the feeling of "being there together with others". This feeling is supported by the use of avatars so one can actually locate the other users in the world and sometimes even see what they are doing, if the animations are realistic enough. To feel the presence and the awareness of others allows users to feel a sense of *belonging to a social group*, in contrast to many conventional e-Learning approaches where users cannot really see and "feel" the other users around them. The sense of belonging to a social community is enforced through the provision of various *communication channels* (De Lucia et al., 2009; Gütl, 2010). Conventional, 2D learning environments lack the visual channel and users can only use text or voice chat to get to know each other. This delays or even prevents the relationship building process which naturally relies heavily on non-verbal cues. 3D virtual worlds can solve this problem by also providing one or two non-verbal channels (mostly visual, sometimes additionally haptic) which greatly facilitates communication among group members. Group forming and relationship building is much easier that way. Therefore, virtual worlds are ideally suited to communicate, socialize and work collaboratively on a project (D. Wood, Hopkins, et al., 2008; Dalgarno & Lee, 2010; Berger, 2012).

**Intrinsic Motivation** Dalgarno and Lee (2010) list in their work a number of things that should be made possible by a virtual learning environment. One of them is increased *intrinsic motivation* which is a great benefit when it comes to learning. Lloyd P Rieber (2005), De Lucia et al. (2009), and Gütl (2010) ascribe the fact that virtual worlds lead to greater motivation to the user being able to make choices of his own and achieve his personal learning goals within the environment. In addition to active participation, game-based approaches can be used to further increase the intrinsic motivation (Garris, Ahlers, & Driskell, 2002).

**Active and Exploratory Learning** The greater effectiveness of active learning compared to passive learning has already been mentioned briefly in section 2.1. Active learning refers to an education model in which students engage with the learning material in many different ways. Besides reading, this can be listening to it, discussing it with peers or trying to apply it in practice. Very important is the fact that students actively engage instead of just passively absorbing information. Bonwell and Eison (1991) emphasize in their work that with this learning approach, the learning success is within the responsibility of the students. They suggest different ways of promoting active learning in classrooms, such as discussions, collaborative group learning or games. Collaborative virtual worlds take the same line by actively engaging their participants in learning activities and providing numerous possibilities to collaborate and socialize. Furthermore, they enable a "hands-on approach" even for things that would be impossible or too dangerous in real life. So virtual worlds provide a great platform to implement active learning concepts. For instance Schiller, Goodrich, and Gupta (2013) used SecondLife in an

## 2. Background and Related Work

undergraduate marketing course and found that "[w]hen 'doing' is incorporated in learning activity, enjoyment and learning outcomes improve significantly".

Exploratory or *discovery learning* is well in line with active learning because it also requires learners to actively construct their knowledge. The idea is that students are not given structured information from the beginning but rather a pool of unstructured learning material from which they have to try to draw conclusions for themselves. The exploratory learning approach was invented by Bruner (1961) and has gained popularity in traditional education and computer simulations. Njoo and De Jong (1993) state that "[c]omputer simulations are well fit for exploratory learning because they can hide a model that has to be discovered by the learner". In spite of its popularity, there are also some critical voices, according to which the learning approach brings no obvious benefits (Mayer, 2004). One reason for that might be the lack of structure and guidance that leads to errors and misunderstandings which is frustrating for many students (Alfieri, Brooks, Aldrich, & Tenenbaum, 2011). Marzano (2011) suggests an enhanced exploratory learning approach in which the teacher first introduces the students to the topic and explains the most important principles to them before letting them discover the rest for themselves. Furthermore, it is the task of the teacher to provide help and guidance throughout the exploratory tasks. Done correctly, exploratory learning can be a successful learning model. According to Kolb (2014) learning takes place in four steps which are constantly repeated: It starts with (1) a concrete experience, from which (2) insights are drawn. Then, (3) concepts and theories are formed and finally (4) an attempt is made to apply the knowledge gained to new situations. Figure 2.4 depicts this cycle. The author stresses its importance for exploratory learning because it illustrates that engagement, social interaction, and experience drawn from practice are important for a good learning outcome.

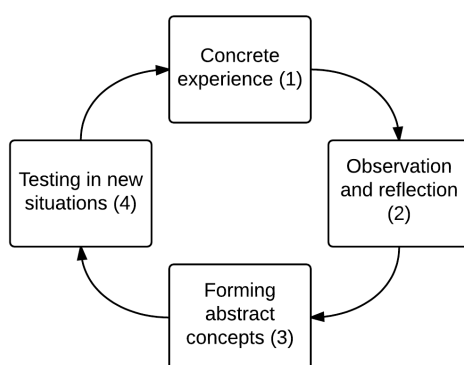


Figure 2.4.: The experiential learning cycle from Kolb (2014)

As a successful example for implementing the exploratory learning approach in virtual worlds the "infection control game", a hospital training simulation, should be mentioned (Jarvis, Smith, Hallam, & Knight, 2007). In this graphical simulation hospital staff, e.g. nurses can take on a third-person perspective in a hospital



environment and go through different hygiene protocols like correct hand washing. The goal of the game is to cause a real change in the behavior of hospital employees through realistic, immersive experiences.

In conclusion, then, it is clear that virtual environments have several advantages which make them valuable learning environments. They allow learning independent of time and place, so even geographically dispersed students can meet online and learn together, without having to be in the same place at the same time (Messinger et al., 2009; M. Lee, 2009; Dalgarno & Lee, 2010; Duncan, Miller, & Jiang, 2012). But that was also already the case with conventional e-Learning methods. What is unique to virtual worlds is the large number of opportunities to communicate and to socially interact with others via the visual channel. That holds great promises for constructivist learning scenarios because things are no longer experienced from third-person perspective, but from first (Dickey, 2005). Interaction with the environment allows users furthermore to try out things that would sometimes be too costly or dangerous in the real world (Bailenson et al., 2008; Warburton, 2009; Duncan et al., 2012). Workplace training could be taken to the next level that way (Nebolsky, Yee, Petrushin, & Gershman, 2003). Learning and training in virtual worlds also seems to get more and more popular also in today's workforce, especially regarding group-based problem solving. Davies, Fidler, and Gorbis (2011) even name *virtual collaboration* as "one of ten key skills for the future workforce". Lastly, the inherent immersion and sense of presence virtual worlds evoke, surpasses every 2-dimensional type of e-Learning. While an increased sense of presence in the virtual space by itself does not necessarily lead to better learning outcomes, it can lead to increased motivation and commitment, especially if problems are to be solved, which require several participants to work together (J. F. Chen, Warden, Wen-Shung Tai, Chen, & Chao, 2011). Therefore, virtual worlds could solve problems with engagement from which conventional e-Learning approaches suffer.

But there were also some disadvantages identified in various studies, most of them related to technical shortcomings or usability issues, such as difficulties in navigation (Chittaro & Ranon, 2007). As further drawbacks the problems teachers can experience when creating a virtual environment and the lack of acceptance of new learning methods among students were identified (Gütl, Chang, Kopeinik, & Williams, 2009; Gütl & Pirker, 2011). Also, a good design and a sound pedagogical approach are needed. None of these points can be neglected when wanting to make use of a virtual world as learning environment.

The next section will present some showcases where learning scenarios in virtual worlds have successfully been implemented.

## 2. Background and Related Work

### 2.5. Showcases

The previous section has outlined some benefits of virtual worlds over conventional e-Learning approaches, such as immersion and the ability to socially interact with others through the visual and the auditory channel. The goal of this section is to present some practical examples of virtual environments which were developed in recent years, and their application in real-world learning scenarios. Since our first prototype will be developed in Open Wonderland (OWL) and will deal with a topic relevant to ancient Egypt, we want to take a closer look at showcase examples which either use the same toolkit for creating virtual worlds or have a historical learning topic as well to get an idea of how such worlds are designed and where potential problems may be.

#### 2.5.1. Giza 3D

The Giza plateau west of Cairo is known for its large pyramids - Khufu, Khafre, and Menkaure - as well as for the famous Sphinx statue. Numerous excavations in the last 200 years have led to tons of documentation, photographs, sketches and drawings of the great Giza pyramids and their surrounding limestone tombs and temples. This data was collected and converted into electronic form by the Giza archives project<sup>9</sup> to form the world's largest collection of archaeological data from the Giza plateau. The "Giza 3D" project<sup>10</sup> is an ambitious undertaking of Harvard University which aims at combining the Giza archives with a realistic 3D visualization of the site (Manuelian, 2013). Figure 2.5 from Giza 3D (2013) shows the reconstruction of the Giza plateau at a glance.

Because there are so many documents and photographs from different eras of the site, the 3D visualization allows to switch back and forth in time between them. Users can see the excavation site as it appears nowadays or they can go back in time to see how the buildings looked like when they were yet undamaged from time, grave robbers, and over-motivated scientists. Whenever a user clicks on a building or an object, information in form of meta data, notes and comments from archaeologists, photographs, and drawings can be retrieved. The markings which show that some information to an object is available can be seen in figure 2.6(a). Thanks to the Giza archives, this material is also cross-referenced and can for instance lead to information about deities or objects depicted on the specific object.

The visualization can, among other things, be used for interactive instruction in class. However, it is not always easy for schools and universities to get hold of appropriately large or even immersive displays which would improve the experience

---

<sup>9</sup><http://www.gizapyramids.org>

<sup>10</sup><http://giza3d.3ds.com>

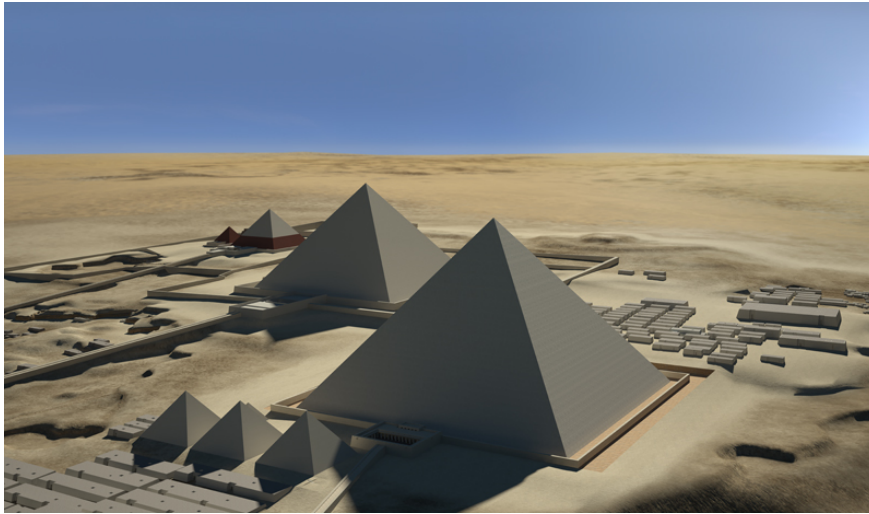


Figure 2.5.: 3D visualization of the Giza plateau. Image from Giza 3D (2013)

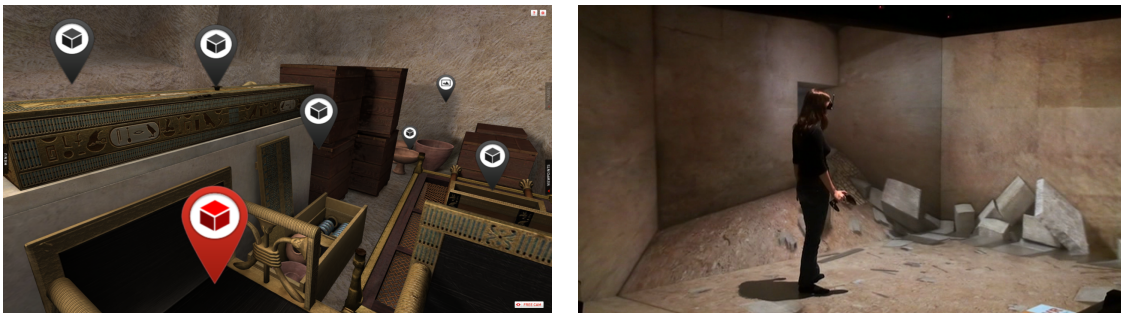


Figure 2.6.: Left: A grave chamber with different objects which have information attached to them; Right: A student immersed in a 3D projection of a tomb on an edge-shaped display

for students. Figure 2.6(b) shows a student wearing a pair of 3D glasses exploring a tomb which is projected on a large, edge-shaped 3D display. But the Giza 3D project can also be accessed via the internet with a normal browser and even viewed in 3D on an ordinary computer screen using an inexpensive pair of 3D glasses.

Besides being an interesting teaching tool, the Giza 3D visualization can also provide new insight for scientists. As figure 2.7(a) shows, all buildings and objects are reconstructed historically accurate which is made possible by modern 3D modeling techniques. Researchers can gain insight into the buildings from all angles and can even view them as they have originally looked like in the past.

One drawback of the Giza 3D project is that although students can explore the Giza plateau virtually, they cannot do so together with friends or fellow students. They are always alone in the environment, not able to ask questions or discuss matters immediately. Instead, they have to wait until everybody has finished their explorations and eventually discussions are started in the classroom. For that

## 2. Background and Related Work

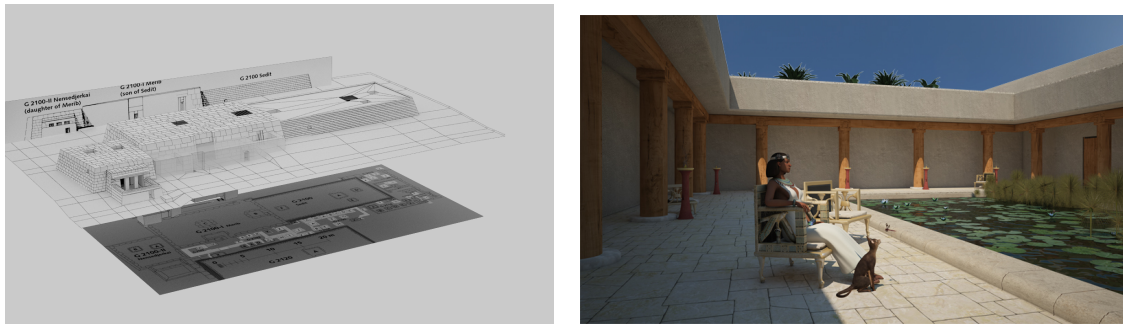


Figure 2.7.: Left: A Plan for a 3D wireframe model to accurately remodel a building; Right: An animated character which was added to a scene to make it more lifelike

reason, as an extension to the existing visualization, the introduction of user avatars is planned. So far, animated non-player characters have been included in the environment, as can be seen in figure 2.7(b). That way, students get a feeling for the size of the individual objects and buildings.

### 2.5.2. The Egyptian Oracle

An attempt to combine physical experiences with virtual environments and objects is the Egyptian Oracle (Jacobson & Gillam, n.d.). In this project, the 3D replica of an Egypt temple<sup>11</sup> is projected on a screen and serves as a stage for a religious, Egyptian ceremony. The actors are on the one side some computer-animated characters on the screen and on the other side a real person, who is dressed like she belongs into the Egyptian environment. The show is intended to be watched in groups in front of a large screen. The main actor is the "high priest", who leads the ceremony. He is also an animated character in the virtual environment, but is controlled by a digital puppeteer. That way, interaction with the audience is possible, which often leads to unexpected and funny moments. The puppeteer is able to see the audience via a web cam and can address specific people. However, the audience is often really surprised by that and does not react accordingly, which is one reason why additionally a real actress is needed to mediate the whole experience. Her task is to further encourage people chosen by the high priest to speak, to explain certain things to the audience and to animate them to participate in the religious ceremony. By combining virtual images and real actors, an immersive and interactive experience is created. According to the authors it is best suited for children between 8 and 12 years, although some success with larger groups of adults were obtained as well. Figure 2.8 shows an adult standing in front of the projection of the Egyptian ceremony.

<sup>11</sup>[http://publicvr.org/html/pro\\_egypt.html](http://publicvr.org/html/pro_egypt.html)



Figure 2.8.: Visualization of a religious, Egyptian ceremony with live audience participation. Image from PublicVR (2008)

The temple and the ceremony were modeled in Unity<sup>12</sup> which is a development engine for games and interactive 3D graphics applications. The drawback of the Egyptian Oracle Project is that the environment is not explorable by students themselves, but only accessible by attending the live show at a museum or university which organizes it. For the future it is planned to broadcast the shows live over the internet, so that people can watch it regardless of their location.

### 2.5.3. The SEE Experience

The Shrine Educational Experience (SEE) project is a somewhat older project that was conducted by the Israel Museum and the Politecnico di Milano to increase the popularity of the Israeli "Shrine of the Book" building and to allow students and other interested people to learn about the scrolls, which are stored in this building (Nicoletta Di Blas, 2003). The Shrine of the Book is depicted in figure 2.9.

The project team had already gained experience with virtual worlds in previous projects. They have found that collaboration does not happen automatically in multi-user environments, but that the users must be given appropriate tasks. So they have developed a virtual learning environment for high school and junior high school students and also simultaneously devised tasks which have to be completed in this environment to motivate the students and keep them engaged. Most of the tasks require students to collaborate with each other. Because of this the authors also claim their project has an integrative character.

A first test run was performed with four classes from different parts of the world. Together, they had to complete three sessions in the virtual world, each lasting approximately one hour. For all of the sessions the project team had designed a set of activities, consisting of (1) an introduction, (2) a knowledge acquisition part, (3) a number of educational games, and (4) a wrap-up in which the topics of the session were recalled briefly. With this approach they made sure that students did

<sup>12</sup><http://unity3d.com/>

## 2. Background and Related Work



Figure 2.9.: Exterior view of the Shrine of the Book. Image by Berthold Werner - Own work. Licensed under Public Domain via Wikimedia Commons

not linger around idly in the virtual world when they were supposed to learn. Interaction was provoked with games and quizzes in the third part of every session. The authors found that students liked to work together so they designed the games and quizzes in a way that teamwork was required in order to win points or answer questions correctly. The fact that in the end the team with the most points wins is intended to add a competitive character to the educational experience and to rise the students' commitment.

One thing that turned out to be quite problematic was the delivery of educational content in the virtual world. Virtual environments are not suitable to represent large amounts of theoretical knowledge. In this project the problem was solved in a way that the students were given the background material on the different sessions in advance so they could have a look at it before going into the virtual world. The materials largely had the form of interviews, so that students were taught the different views on a subject.

SEE has been successfully tested in Italy and Israel and the authors mention the great advantages of immersion and presence for learning. C.-H. Chen, Yang, Shen, and Jeng (2007) identify the following aspects that the SEE project provides: interpersonal communication, information browsing, as well as hand-on experience features. Nevertheless, the project has never gained fame with a wider audience beside some scientific mentions.

### 2.5.4. The Virtual TEAL World

As a final practical example in this section, the Virtual TEAL World (VTW) (Pirker, 2013; Pirker et al., 2013) should be mentioned. In contrast to the previous examples, it does not deal with historical themes, but is designed to teach physics concepts.

What makes it interesting nonetheless for our project is the fact that it uses OWL for creating virtual learning worlds.

A rather successful approach of teaching freshmen physics, the Technology-Enabled Active Learning (TEAL) approach (Dori & Belcher, 2005), was developed at the Massachusetts Institute of Technology. It is all about group activities and hands-on experiments, but needs many physical resources and a lot of time to prepare. The VTW tries to transfer that approach into the virtual world so that resources can be minimized and preparation has to be done only once because a virtual world can be reset and reused infinitely many times. A further benefit is that students from all around the world can learn together without having to be in the same place. For transferring the collaborative group tasks into the virtual world usage is made of the built-in collaborative and pedagogical tools of OWL such as the Whiteboard, the PDF Viewer, the Sticky Notes or the Video Player. Additionally, the *TEALsim* software (Massachusetts Institute of Technology, 2010) was adapted for OWL to visualize the effects of different physical experiments directly in-world (Berger, 2012), as can be seen in figure 2.10 from Pirker (2013).

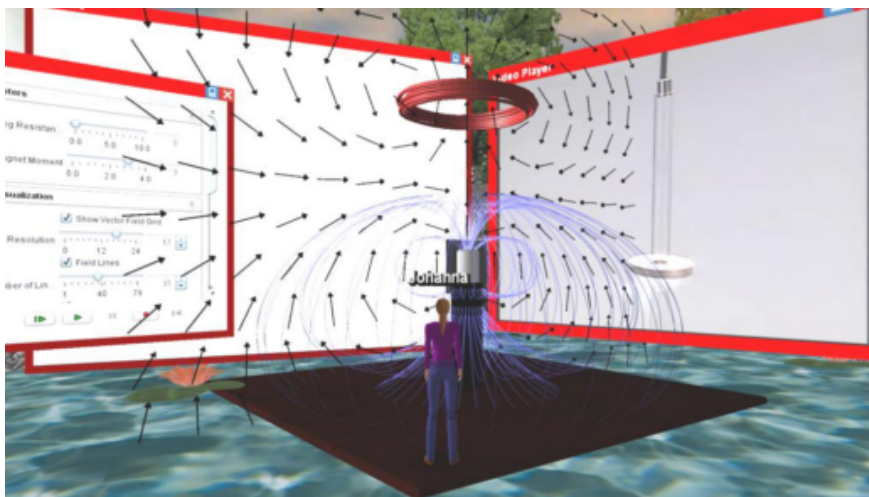


Figure 2.10.: Immersive physics experiments. Image from Pirker (2013)

The whole learning experience the VTW provides consists of five different stages: (1) an overview of the course plus instructions of what to do during the online learning session, (2) a first concept question to assess to what extend the students are already familiar with the topic, (3) explanations and collaborative learning using the PDF Viewer, the Video Player, and a discussion area where a whiteboard is provided, (4) experiments and simulations with the *TEALsim* module, and (5) the concept question again to assess the learning process of the students. An example setup with a group of students can be seen in figure 2.11 from Pirker (2013).

In addition to the design of the learning experience and the development of various learning and assessment tools, Pirker (2013) also developed a set of usability guidelines for the design of learning scenarios and activities in virtual worlds. The

## 2. Background and Related Work

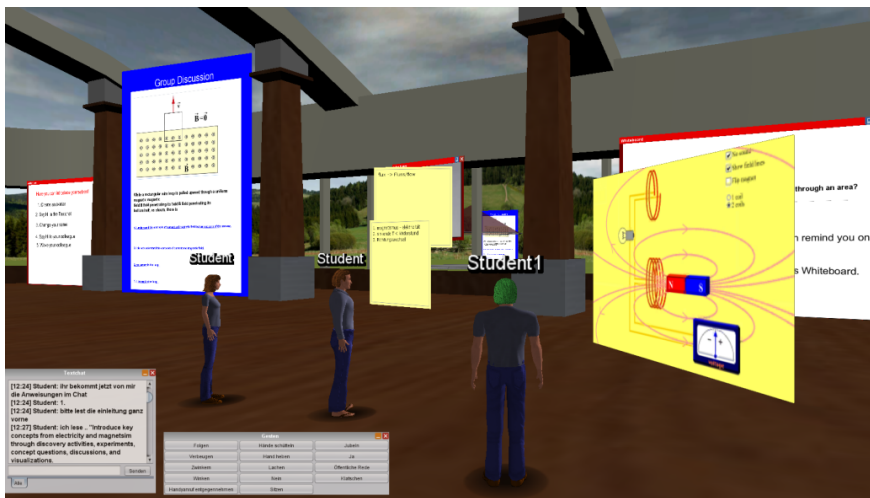


Figure 2.11.: Student group doing a learning round trip together in the VTW. Image from Pirker (2013)

goal was to optimize the user experience and to minimize difficulties the students would have due to technical limitations or not user-friendly interfaces. As a result the VTW has done very well in an initial evaluation, experiencing good acceptance by the users. The only things the students complained about were performance issues and the rather outdated graphics, which are both problems of the virtual world building software OWL. Other than that, most of the test users were satisfied with the usability and positively mentioned the interactivity and possibilities for active participation and collaboration in the VTW.

## 2.6. Summary

In this chapter, some teaching methods have been presented. The methods "Lecture", "Demonstration", "Collaboration", and "Learning by Teaching" were described here in more detail. Following this, the concept of e-Learning has been introduced. E-Learning offers people from all over the world the opportunity to learn together in online courses. But although methods to deliver content can be varied and rich in e-Learning, e.g. by making use of live broadcasts of lectures or video recordings, engagement is often low, which might be due to the anonymity and the feeling of distance people experience when sitting in front of their computers at home. Serious games and virtual worlds may be able to overcome this lack of commitment and motivation by providing immersive spaces where users can meet and complete tasks together. Virtual worlds in addition usually provide a great number of communication channels, including voice chat and avatar gestures, which allow the users to feel a sense of presence in the world. It has been found that this heightened feeling of immersion and presence in 3D virtual worlds can lead to



greater motivation and engagement than in conventional 2D learning environments, which in turn benefits the learning outcome. Apart from that, virtual worlds also offer great opportunities for socializing and collaborative group work. To conclude this chapter, some examples of virtual learning environments that have been used successfully in practice were outlined. The next chapter will deal with the design of our 3D learning world, starting with a requirement analysis, followed by the selection of our virtual world platform, and finishing with the presentation of the learning experience itself, including a short description of the implemented software modules and the temporal organization of the learning round-trip.



## 3. Design

The goal of this project is to develop a set of tools for a virtual 3D world where students and interested people from all over the world can meet and perform learning tasks together. Like in a classroom, students should be able to benefit from collaborative and social interaction, but without the need to be physically present. Instead they fulfill tasks together with others in the virtual world finding an authentic environment they can explore freely and thus ideally becoming involved with the learning topic. An appealing 3D environment as well as a high degree of interactivity are in these cases supposed to raise motivation and commitment.

The learning tasks should be held relatively simple. The implemented tools should include information retrieval from various objects along with a means to store this information somehow for later usage. Furthermore, a quiz is planned in order to check the acquired knowledge at a given point in time. The special features of virtual worlds are taken into account by integrating an exploratory character into the learning experience, coming in form of a scavenger hunt. This is also supposed to enhance excitement and bring a sense of adventure to the usually unpopular task of learning.

In this chapter the methods used and considerations made in order to create a suitable learning scenario where the focus lies on exploratory and collaborative learning are described. First, requirements will be identified in order to successfully implement learning concepts. Then, a suitable software will be chosen to create our virtual environment. Finally, the implemented software modules, as well as the design of the scavenger hunt will be outlined.

### 3.1. Requirement Analysis

Many demands are placed on learning scenarios in virtual worlds and even more if these scenarios are intended to be collaborative. One requirement is certainly that a need to work in groups arises, which goes beyond simple chat. Learning activities must be carefully designed in order for the students to benefit from a virtual environment (M. Lee, 2009).

It is clear that there are different stakeholders involved in this project, namely the teachers, who want to create learning content as conveniently as possible, and the students, who are supposed to consume this content, but additionally want to be

### 3. Design

entertained in that process. As a consequence, there are different requirements which have to be met. Most of the time these requirements will contradict each other since on the one hand things should be held as simple as possible while on the other hand complex virtual adventures are desired.

The main advantage of virtual worlds is probably their interactive character which can be easily used to enhance motivation. Another one is that abstract things and situations that would be impossible in real life can be simulated. That can be used to create an appealing learning experience which is exciting for the students and not too hard for teachers to create and maintain.

In the following some basic requirements which have to be fulfilled to satisfy both, teachers and students will be outlined. Since the learning scenarios should be collaborative, a Multi-User Virtual Environment (MUVE) is needed. This in turn requires the virtual world platform to have a client/sever-architecture in order to allow multiple users to connect to a single world. Once a suitable virtual world platform has been identified, some tools have to be developed so that a learning round-trip in a virtual environment can be created. The tools should, among other things, enable students to draw information from the environment and to somehow store it for later usage. Finally, there are some requirements on the learning experience itself regarding the used learning model. To conclude, therefore, it seems that there are five main points on which we impose our requirements:

- Server
- Virtual World Software
- Environment
- Learning Tools
- Learning Experience

#### **Requirements regarding the Server**

The task of the server is to manage the content of the virtual worlds/learning environments created. It is the central point from which everything is administered and must therefore fulfill strict requirements.

**Accessibility and Availability** To guarantee location-independent, collaborative learning, the server on which the virtual world software runs has to be accessible from those regions the students come from. In addition to that, the server has to be available when the students are supposed to be taking their learning round-trip. If the students should be able to decide for themselves when they do the learning tasks, the server must be reachable for them 24/7.

**Performance and Scalability** Since it has to be expected that at any time multiple users are logged into a graphically complex world, the server capacity must be

designed in a way that nobody experiences lags or delays. The more users are expected the more powerful the server hardware has to be, respectively the more server spaces have to be offered.

## Requirements regarding the Virtual World Software

The virtual world software is the platform on which we create our virtual environments. It represents the software environment that makes creating virtual worlds possible.

**Multi-User** The software we are going to use has to support multiple users because implementing such a feature is not in the scope of this work. Nonetheless, allowing groups of students to explore the learning world together is a basic prerequisite for creating a collaborative experience. Luckily, multi-user is a feature that can be taken for granted in most virtual world systems since they are usually designed for social group activities (Bell, 2008).

**Graphics** The software has also to be a 3D graphical system since in contrast to textual systems these are more likely to provoke the feelings of immersion and presence which in succession lead to motivation. Furthermore, they are more comfortable to use which is ascribed to the fact that visual representations are intuitive, whereas text has to be read and interpreted (Holmstrom & Jacobsson, 2001).

**Extensibility and Flexibility** In addition to being a graphical virtual world, the platform we are going to use has to support the creation of our own virtual worlds using 3D models in the most common file formats so that already existing models e.g. from the Google Warehouse (Google Warehouse Website, 2015) can be used. This is important so that teachers, who will not be 3D designers in most cases, are able to create new scenarios and learning environments easily as well as modify existing ones.

Extensibility is supposed to include not only the extension possibilities of content creators, but also those of programmers implementing new features or tools for the learning world. Perfect for our purposes would therefore be a flexible, extensible toolkit for creating virtual worlds, which is ideally also written in a conventional programming language.

### 3. Design

#### **Requirements regarding the Environment**

The created virtual environment is very important for the learning experience. With the structure, realism and richness of details of the built world the teacher can mainly influence the students' motivation.

**Graphical Appearance** It is essential that the created learning world is designed in a way that invites students to explore. It must be possible for the teacher to easily import and arrange 3D models. The objects should evoke associations with the real world and the students should be able to use them as they are accustomed to, e.g. sit on chairs or climb staircases.

**Interactivity** To really be appealing to students the environment should be as interactive as possible. This means the user is able to actively design his own learning experience, e.g. having different selection options when clicking on objects or being able to make his own decisions which in succession affect the outcome of the learning round-trip. The interactivity of the system should raise motivation and allow for individualized learning up to a certain extent.

**Maintainability** Once created, teachers must be able to modify or extend their learning worlds by adding or removing objects or rearranging them within the world. Worlds should be able to be saved and reloaded when needed. Furthermore, it should be possible for programmers to extend the implemented functionality anytime.

#### **Requirements regarding the Learning Tools**

The main part of this work is the implementation of suitable learning tools for the virtual world platform to be used in a virtual learning environment. These little pieces of software should be universally applicable in any world as long as it is created with the chosen virtual world software.

**Functionality** First of all, the programmed tools must meet the requirements that are provided in section 3.3. Error states are to be intercepted and must not lead to crashes of the overall system. Moreover, all functionality has to be tested with the latest version of the used virtual world software.

**Usability** In addition to being functional, the implemented tools should also be user-friendly and easy to use for both, content creators, respectively teachers and

students. It is further contemplated to enhance the user experience by gamification approaches like different colors for tasks already done and still remaining as well as progress indicators and the like.

## Requirements regarding the Learning Experience

In this subsection, requirements concerning the way the students should learn in the virtual world are imposed. To achieve success in learning, a sound pedagogical background to justify the used learning method is required.

**Collaborative Learning** For social interaction it is necessary to provide appropriate communication tools like text or voice chat. But although this is required, it is not sufficient to enforce collaboration. A feeling of belonging to a community, along with the perception of the presence of other community members and, consequently, a feeling of social presence is also crucial for people to work collaboratively.

As literature research (D. W. Johnson & Johnson, 1991; Kreijns et al., 2003) has shown there are five ways to encourage collaborative problem solving:

- Positive interdependence
- Promotive interaction
- Individual accountability
- Social skills
- Group processing

For this project, it is assumed that participants already have sufficient social skills necessary for working collaboratively. Furthermore, the task of group processing is left to the individual teacher who divides the class into groups and guides them through the virtual learning experience. The remaining requirements, namely positive interdependence, promotive interaction, and individual accountability are to be met by the learning environment.

As with a high degree of interactivity, the possibility to collaborate with others is expected to enhance the motivation among the students. To make collaborative learning in virtual worlds possible by means of implementing appropriate tools should be one of the objectives of this project.

**Exploratory Learning** In order to fully exploit the advantage of creating fantastic worlds within the virtual context the students should get the possibility to learn while exploring those worlds.

Exploratory learning is a teaching and learning approach which motivates learners to experiment. In that process they can also gain unexpected insights or reach the expected learning goals on ways unintended by the teacher. The use of computers

### 3. Design

as a communication tool for students to learn together with friends, or as a way to simulate dangerous situations that would be unsafe in the real world, is a characteristic of exploitative learning (Foraker Labs of Boulder, 2014).

L. P. Rieber (1996) lists in his article four basic principles of exploratory learning:

- Learners control their own learning
- Knowledge is rich and multidimensional
- There are diverse ways to approach the learning task
- Learning feels natural and does not have to be enforced

In combination with a collaborative approach, exploratory learning, too, should be one of the objectives in the design of the tools for learning in a virtual world.

**Assessability** Measuring the learning progress is another important requirement of the project. Teachers should be able to monitor the activities of the students individually and also the students themselves should have a way to check their knowledge during the learning experience. That way, it can be verified if the desired learning goals have been met and where potential problems lie.

## 3.2. Selecting a Virtual World Platform

Different studies (Pirker et al., 2013; Riedmann, 2014) have shown that in terms of collaborative virtual worlds the open-source toolkit Open Wonderland (OWL) (OWL-Foundation, 2014a) is a good choice. OWL has been developed at Sun Microsystems and was continued as an open-source project when Oracle acquired Sun in January 2010 (Kaplan & Yankelovich, 2011; Parsons & Stockdale, 2010).

With OWL it is possible to build customized virtual worlds for multiple users and make them available for people all around the world via the internet. Therefore, one has to simply set up an OWL server, which is free in contrast to other virtual world platforms like SecondLife (Linden Research, 2014) and gives you the benefit of having complete control over the created worlds and the connected users. Only some firewall configurations are necessary, for instance opening an UDP port for audio communication, but that is basically all that is needed for the students to be able to connect from their computers at home. Since OWL is written in Java the students can start it directly from their browser and do not need an external viewer software. Prerequisite for this is of course that they have a recent version of Java installed.

One big benefit of OWL and also one of the main reasons why it was chosen for this project is that OWL is built in a modular style that is completely extensible. The main installation only comes with a few core modules whereas the rest can be installed when needed. Programmers can develop new modules altering and



### 3.2. Selecting a Virtual World Platform

extending the functionality of OWL in every conceivable way. To give an overview of the modules that already exist and make them available publicly, a so-called "warehouse" exists to which the OWL website provides access (OWL-Foundation, 2014c). OWL also has a very vibrant community that gladly and quickly assists with programming problems.

In addition to being completely extensible by programmers it is also very easy even for non-experienced computer users to create worlds. Thanks to a Collada loader (Collaborative Design Activity; [www.khronos.org/collada](http://www.khronos.org/collada)) supporting the drag and drop of 3D models, one can simply take an existing file and drop it into the OWL client window to place the model into the world. A simple tool to move the objects and edit their sizes and positions further lowers the technological hurdle and allows the intuitive design of new worlds.

The second great advantage of OWL is that it is built for educational and business contexts and thus for collaborative tasks. For that reason, lots of tools to support group work already exist. Some of them have to be installed separately but most collaborative modules come with the basic installation of OWL. Those include:

1. **User list**

A textual overview of all users who are currently online.

2. **Chat**

It is possible to start a conversation with one or more users.

- (a) **Textchat**

For textual conversation an input field at the lower left corner can be used.

- (b) **Voicechat**

Voicechat is activated automatically so that all other users can hear what is spoken into the microphone. It can be deactivated by clicking on the symbol in the lower right corner.

3. **Whiteboard**

The Whiteboard serves as communication tool for collaborative brainstorming or representing ideas graphically.

4. **Sticky notes**

A small notepad that can be placed anywhere on the screen for writing down short memos.

5. **Document viewer**

Documents in pdf format can be shared with other users by simply dragging them into the OWL window.

The modules in action can also be seen in figure 3.1. All of these tools have been made for use by multiple users. Due to the graphical representation of users as avatars in the virtual world it is easy to see which users are working with certain tools like the Whiteboard, since one has to be near the board in order to see what is written on it.

### 3. Design

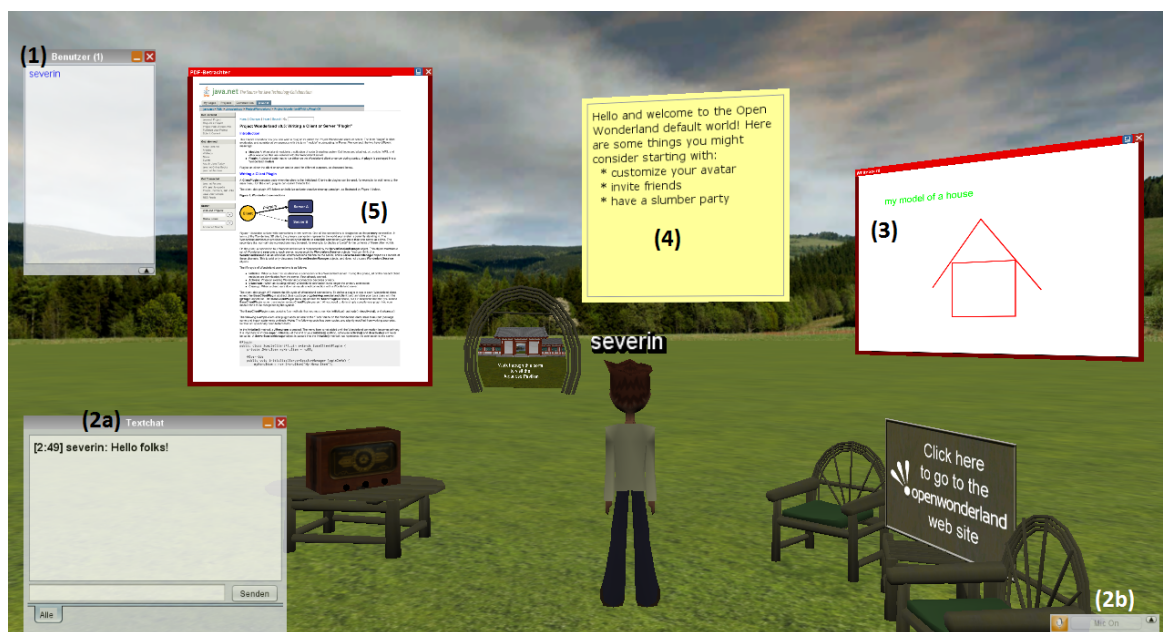


Figure 3.1.: Collaborative modules that come with the basic installation of OWL

There is also one core functionality of OWL called *shared applications* which is the name for an X11 application sharing feature where one user at a time can take control to e.g. edit a text document, while other users who are nearby can watch. The control can then be given to another user. All kinds of Linux applications, like Firefox or Open Office can run directly in-world that way. Although it was not used for this project, this feature can come very handy and really distinguishes OWL from other avatar based group meeting applications like Second Life or Active Worlds. Figure 3.2 from OWL-Foundation (2014d) shows a picture of a meeting in OWL where some people collaboratively create some piece of software.

Another thing that supports collaborative work is the built-in high-fidelity immersive audio capability of OWL. It allows for stored audio tracks to be played in CD quality and furthermore, and more importantly, users can also hear the voices of others in high quality. That way, users can use natural speech for communication as if they were in the same room together. Along with the visual avatar system it enhances the feeling of presence in the virtual world (Andreano et al., 2009).

Now that some reasons for choosing OWL as our virtual world platform of choice were given the next section will describe some of the modules used for this project.

## OWL Modules Used

For the creation of an exemplary learning environment, a variety of already existing OWL modules was used. The sample learning environment was created to provide the possibility to try out the envisaged learning round-trip. It also serves as a base



Figure 3.2.: OWL event where people collaboratively create some software. Screenshot from OWL-Foundation (2014d)

for a first evaluation. The idea was to use an Egypt setting since this is usually a topic which can be made interesting for both, younger and older students. More details on the created environment can be found in chapter 5.

In contrast to those modules which were newly created and which will be discussed in detail in chapter 4, the modules described in the following subsections either came with the basic installation of OWL itself or were downloaded from the OWL module warehouse. Another way to obtain modules is the community SVN server (OWL-Foundation, 2014b). There, the modules are classified into "stable" and "unstable" according to whether they have been tested thoroughly or are somewhat newer, respectively still in development.

Three of those freely available modules created either by the project developers or members of the community which were used for this work are listed in the following subsections.

**Textchat** The *Textchat* module comes with the standard OWL installation and provides users with a means to exchange written messages. It comes in form of a small rectangular HUD Component at the bottom left side of the client window which is usually visible when OWL starts. Else, it can be found in the 'Windows' menu in the menu bar. Figure 3.3 shows the *Textchat* window.

### 3. Design



Figure 3.3.: The OWL *Textchat* HUD component

**Chatbot Module** This module, aiming at making NPCs more interactive, was implemented by Riedmann (2014) and is used to give students hints about what they have to do in order to successfully complete the learning experience. An example of this can be seen in figure 3.4.

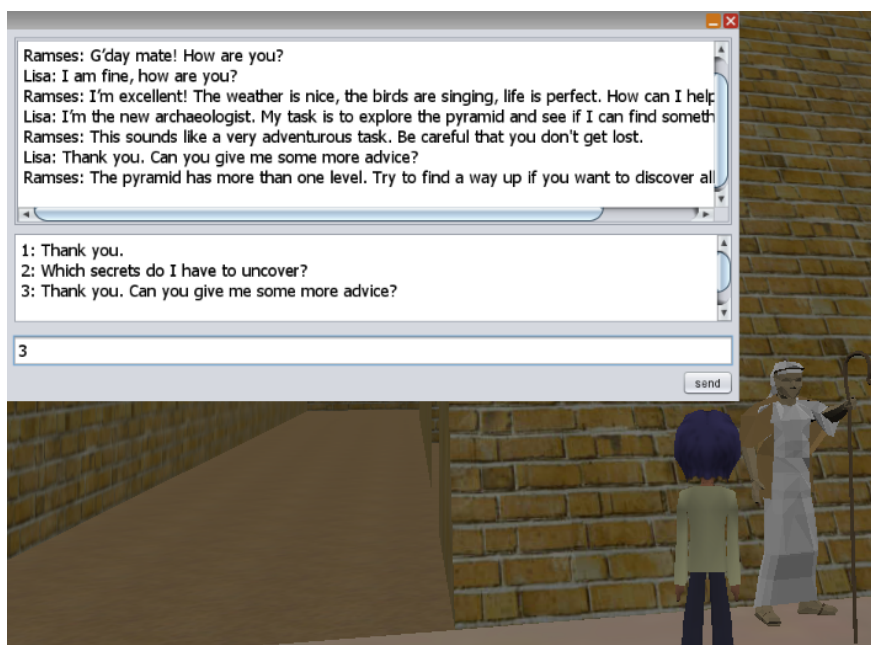


Figure 3.4.: Example of a conversation with a *Chatbot*

The *Chatbot* comes in two forms, managed and with an A.L.I.C.E. (Artificial Linguistic Internet Computer Entity) bot in the background (Wallace, 2009). For this work, the managed chat was sufficient, meaning that there was a script file containing all possible questions the user could ask and the according answers on the server serving as input resource for the *Chatbot*. Part of this script file can be seen in listing 3.1.

Listing 3.1: Part of an exemplary script file for the *Chatbot* module

```

<questionNode nodeNr="2">
  <question>
    <questiontext>I am fine, how are you?</questiontext>
    <answertext>I'm excellent! The weather is nice, the birds are
      singing, life is perfect. How can I help you?</answertext>
    <nextsos>4</nextsos>
  </question>

  <question>
    <questiontext>Not so good.</questiontext>
    <answertext>Oh, why that?</answertext>
    <nextsos>3</nextsos>
  </question>
</questionNode>

<questionNode nodeNr="3">
  <question>
    <questiontext>I have to complete this difficult
      task...</questiontext>
    <answertext>Which task? Can I help you somehow?</answertext>
    <nextsos>4</nextsos>
  </question>
</questionNode>

```

**Stickynote** The *Stickynote* module can be added to the world via the cell palette after having been installed, resulting in a little yellow pad on which text can be written. It is typically used to write down ideas during a brainstorming session or short notes for other users, as one can see in figure 3.5.

As one can see, a large variety of collaborative modules is already available in OWL. Nonetheless, for our purpose it is necessary to implement some more tools which meet our requirement of creating a virtual world where students can meet and learn together. So far it is for instance not possible to provide students with any kind of information besides writing it on a Whiteboard or Stickynote which is both not really appealing when it comes to exploratory learning. Instead, it should be possible to provide objects themselves with information to enable students to obtain information directly where they are of interest. Furthermore, a quiz to check the acquired knowledge would be helpful for teachers and students to assess the learning progress. Some OWL modules already exist in that area but none easy enough to use for teachers who are not so familiar with the computer and none which are marked as "stable". Consequently, such a module also has to be developed from scratch.

### 3. Design



Figure 3.5.: Three users standing around a *Sticky*note cell

In the next section some pedagogical considerations for learning in a virtual environment, as well as the tools that are planned to be implemented are presented and the envisaged learning scenario is outlined in more detail.

### 3.3. Conceptual Architecture

For the virtual learning experience several OWL modules will be developed and implemented to create a scenario which ensures that exploratory and collaborative learning can take place. The goal should be a joint acquisition of knowledge by the students in an environment that they find stimulating. The design of the virtual environment itself will be the subject of chapter 5, whereas the current chapter is going to describe the functionality of the developed modules based on the requirements imposed by the concept of collaborative learning.

**Item Module** The *Item* module is intended to extend OWL in a way that it is possible for objects to be provided with information that can be obtained and stored in a private *Inventory* by the students. Objects with this capability, called *Items*, should be clearly visible to students by being visually distinctive from objects without information. As a result, students should be able to use a virtual 3D world and the objects in it as a resource of information. They should strive to collect as many information texts as they can, learning about the topic in that process. With their *Inventory* they should have a way to check which information they have already gathered.

**Quiz Module** The *Quiz* module should provide teachers and students with a means to assess the acquired knowledge, and, in case of the teacher, assign a grade to the students. It should be freely designable for the teacher in terms of the number and content of the questions. Additionally, a teleport function shall be implemented, to move students who have successfully answered all questions out of the world. The quiz should start when the students click on a specific object or come near a certain location in the world.

**Itemboard Module** The purpose of the *Itemboard* is to facilitate the exchange of information between the students after they have collected all *Items* intended for them. The *Itemboard* works like any other whiteboard, but with the addition that students can select an *Item* from their *Inventory* and add it to the board with just one click. Of course, only *Items* they have collected before are available. After one student has added one of his *Items* to the *Itemboard*, the information text as well as the image is visible to all other students, even if they were not able to obtain the information before.

As emerged from the requirement analysis, there are some techniques which should be used to promote collaborative learning, namely positive interdependence, promotive interaction, and individual accountability.

**Positive Interdependence** First and foremost, interdependence is very important when it comes to collaboration. The learning scenario should be designed in a way that the tasks can only be completed with the help and support of other students. So there has to be a resource that is not available to all participants, but has to be shared in order for everybody to succeed. In our case, this resource is information. The approach of this work to achieve collaboration among students is based on the concept of *positive resource interdependence*, described by D. W. Johnson and Johnson (1991).

When dividing the students into groups, the teacher also assigns them a role. In consequence, certain information can only be acquired by students with a special role while it is not available for others. These pieces of information have to be shared at the end of the learning round-trip so that every student is able to complete the final assessment, for instance a quiz. It is expected that in the process of sharing the student in possession of the information further engages with the content and therefore consolidates his knowledge.

For the prototype, it is planned to implement four predefined roles. A later version of the program could contain a mechanism for the teacher to customize the roles and add new ones.

To facilitate communication among the students, the *Inventory* plugin is intended to show a hint in the description window of non-available *Items* containing the

### 3. Design

name of a student who might be able to pick up the information or of a student who already has. That way, the students know who to turn to in order to obtain a certain piece of information. Of course, the learning experience should be designed in a way that every student is able to pick up some information while in turn every piece of information should be accessible for at least one student.

The approach of dividing the information is similar to the *Jigsaw* approach Aronson et al. (1978) describe in their work. Each student gets only part of the information that is necessary for solving the task, and is then responsible for sharing his part with his group members. This approach makes sure that each student takes responsibility for reaching the group's goal and nobody stays in the background letting the others do all the work. However, if the teacher deems it necessary, he can also assign a certain role more than once to make sure no single student can jeopardize the success of the group by not giving away his information.

If the teacher knows his students well and trusts them not to sabotage the learning experience by not sharing a piece of information, collaboration can be further enhanced by allowing every *Item* to be picked up only once. So as soon as one student adds the *Item* to his *Inventory*, it is not available for other students any more, even if they have the right role. Of course, there should still be a hint in the *Inventory* of the other students of who owns the *Item* and the owner furthermore should have a possibility to give it back if he feels, someone else should have the *Item*. That way, the teacher could for example implement some kind of "race" where it is about who collects the most things in a certain period of time. This approach should be implemented as optional, not mandatory possibility of designing the learning experience for the teacher.

**Promotive Interaction** Promotive interaction describes the act of discussing, sharing knowledge, explaining things to each other, as well as motivating and encouraging other students. It can only take place in cooperative group work situations where the group members possess sufficient social skills (Kreijns et al., 2003). Although most effective face-to-face, it is assumed that the envisaged mechanisms to create positive interdependence, which were mentioned before, will form a base for promotive interaction in the virtual environment as well. The graphical representation of other users by avatars is expected to create sufficient social awareness among the students. For discussion there will be a separate group area where the *Itemboard* stands, so that students have a distinct place to meet when they are finished with collecting the *Items*. The *Itemboard*, voice chat, or text chat can in further succession be used to exchange and discuss the collected information.

**Individual Accountability** Although they are allowed and encouraged to share and discuss information beforehand, the final quiz has to be completed by each student individually. That way, every student can get his own grade, regardless of



the grades of his group members. The fact that students cannot rely on their group members, but have to engage and work for their own success prevents the so-called "free-rider-effect" (Kerr & Bruun, 1983).

Upon successfully answering all the questions of the final quiz, it is planned to implement a teleport function to give the teacher the opportunity to gather all his students who succeeded in the quiz in a specific place. This could be for instance a world where the students get instructions on a new task or some kind of information they can use as a "coupon" for receiving a good mark in the "real world".

Figure 3.6 gives an overview of how a typical learning experience in a virtual environment with the help of the implemented tools could look like.

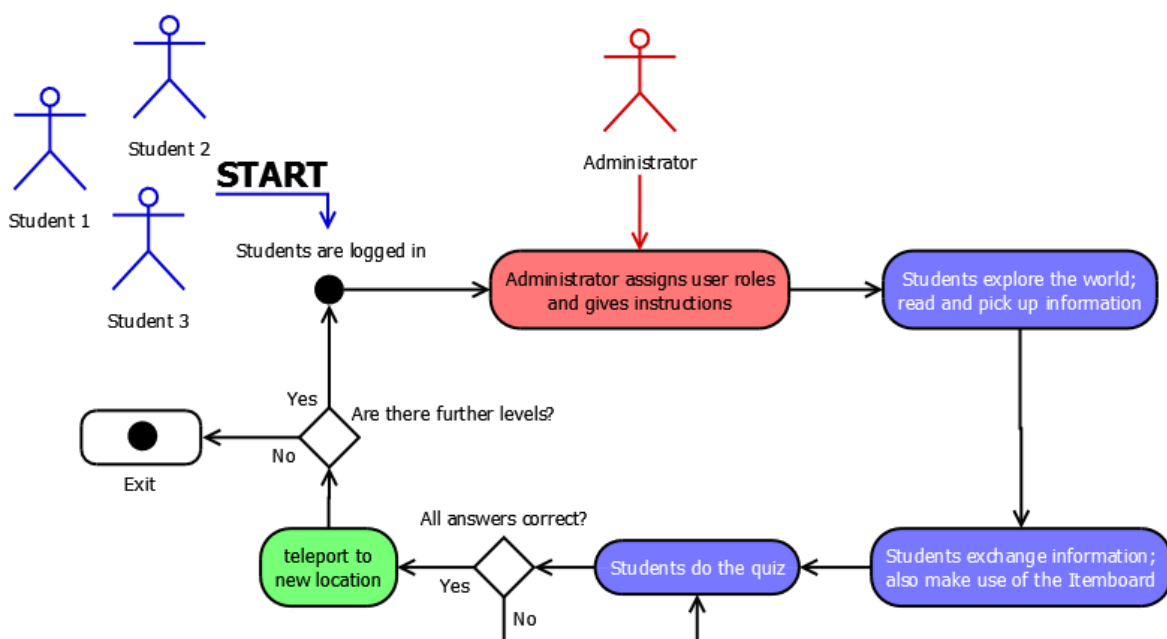


Figure 3.6.: Schematic process of a learning round-trip

First, students will meet in a specially designed area where they can get to know each other. The teacher is also present in this area. He is responsible for explaining the task to the students and assigning a role to each group member. Thereafter, students have time to discuss how they are going to solve the task at hand and then can start the scavenger hunt.

After they have collected all the *Items*, or after a certain period of time, they meet again in the discussion area to share what they have found. Collaborative tools, like the *Itemboard* are provided in this area.

The students can attempt to answer the questions of the quiz anytime, but they will be most successful, if they have exchanged all information beforehand. As mentioned before, as soon as they have answered all questions correctly, they will be

### 3. Design

teleported into a new world, which could mean the end of the learning experience or the start of a new level, in which case the cycle starts anew.

## 3.4. Summary

Before developing the first prototype of the virtual world, different requirements have been defined. First of all, general design aspects like availability, performance, scalability, and extensibility have to be considered. In addition, conditions must be created to make the collaborative and exploratory learning possible. For this purpose, a closer look at the important pedagogical aspects (1) positive interdependence, (2) promotive interaction, and (3) individual accountability was taken. Also, the different focuses of teachers and students regarding a virtual learning environment were taken into account.

Subsequently, OWL has been identified as a suitable platform for the implementation of this project. It meets all the requirements, especially in terms of collaborative features and extensibility. Many tools already come with the standard installation. Nonetheless, some features for the collaborative learning round-trip have to be implemented from scratch. This is firstly the *Item* module, secondly the *Quiz* module and lastly the *Itemboard*. These modules will include various functionalities which allow the teachers to exactly define the learning content and the students to pick up the information intended for them while exploring the world. The quiz allows the verification of what was learned at the end. The *Itemboard* should serve as a communication aid when the students share their information after the scavenger hunt and should help students with the acquisition of knowledge for the final quiz.

A typical learning round-trip was identified to consist of five steps: (1) The students meet in the virtual world and the administrator, which will in most cases be the teacher, explains the learning process to them. Prerequisite for this is that the students were all able to log into the virtual world. (2) The students start the scavenger hunt by exploring the world and picking up the different *Items* they find. (3) The students share their knowledge since not everybody is able to pick up every information text. For this purpose, they can also use the *Itemboard*. (4) The students try to solve the quiz. If successful, they will (5) be teleported to a new world which can either mean the end of the learning experience or the start of a new level.

The next chapter will deal more specifically with the tools developed, namely the *Item* module, the *Quiz* module and the *Itemboard*. The final design of the virtual learning world will be the topic of chapter 5.

## 4. Implementation Details

The focus of this chapter is the development of three Open Wonderland (OWL) modules, which were created in order to fulfill the requirements and meet the goals discussed in the previous chapters. At first, the modular architecture of OWL in general is outlined. After that, the developed modules are introduced and their structure is discussed in detail.

### 4.1. Introduction to the Open Wonderland Architecture

In this section, the structure of OWL's client/server architecture and the components it uses for the operation of virtual worlds are described in detail. Furthermore, the module system with which it is possible to add new functionalities is illustrated. This knowledge is necessary in order to understand the structure of the self-implemented modules, which are presented in the remainder of the chapter.

#### 4.1.1. General Overview

Figure 4.1 shows the various components of OWL in a network diagram. It is intended to give an overview of the services that are necessary for the operation of OWL. The connecting lines there indicate how the individual components work together. The diagram as well as the subsequent description of the individual parts are based on Kaplan and Yankelovich (2011).

Basically, four services are required for OWL to run: A web server, the *Darkstar* game server, *JVoiceBridge*, and a shared application server. They can either run on the same machine or can be deployed on different machines for higher scalability.

The web server is realized with the *Glassfish* technology and provides a management interface for various services, e.g. authentication, user management, modules, or worlds. It also allows to stop or restart the other servers, regardless of whether they run on the same machine or not. The web server uses TCP to communicate with the *Darkstar* game server, *JVoiceBridge*, and the shared application server.

The *Darkstar* server is the game engine of OWL. It is based on *Project Darkstar*, an open source middleware for building online multi-user environments, which is

#### 4. Implementation Details

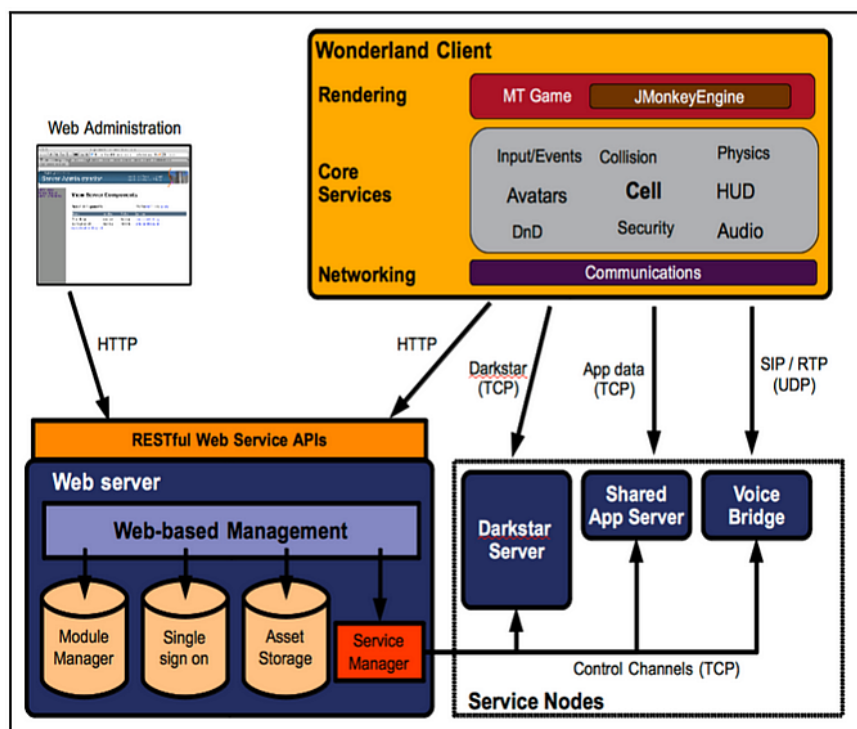


Figure 4.1.: Relationship between the various components of OWL. Diagram from Kaplan and Yankelovich (2011)

now called the *RedDwarf Server*<sup>1</sup>. As a game server it is responsible for managing the virtual world states. For this purpose, it is important that the individual states get synchronized as fast as possible in order for the users to have a consistent view of the world. To meet this demand, the latency is optimized in favor of a high throughput.

For providing high-fidelity immersive audio, *JVoiceBridge* is used. The audio is mixed separately for different users and locations on the server side, while constant communication with the *Darkstar* server takes place to keep informed of the current positions of the users. In order to allow voice chat, a port for UDP communication has to be opened on this server.

Given the need of application sharing, a shared application server which runs the Linux operating system has to be set up. It allows for desktop applications, for instance Open Office or Firefox to be shared directly in-world. Of course, only one user can operate these applications at a time, but others can watch and control can be easily passed on. This server is not necessary if no desktop applications have to be shared. Multi-user applications written directly for OWL can be executed regardless of the presence of this server. Most of the time, it is preferable to use OWL applications because they do not need a control-passing system.

<sup>1</sup><https://github.com/reddwarf-nextgen/reddwarf>

On the client side, the *JMonkeyEngine* is responsible for rendering the world. It is a popular engine for rendering and scene graph creation but can only work with a single thread at a time. Because of that, the *MT Game Engine* is used to add the needed multiprocessor capabilities. Core services include the positioning and moving of objects, collision detection, as well as the calculation of physics.

The next section deals with the modular design of OWL which makes it easy to add new features, like *cells* or *capabilities* through several well-defined extension points.

### 4.1.2. Modular Design

OWL is described as a flexible, completely extensible toolkit. A lot of this extensibility is due to its elaborate module system. OWL itself comes only with the most important core functions, like authentication, networking, content management, and client rendering. Aside from that, all other functionalities are implemented via modules that can be added to the basic installation.

A module can either be a new object - or *cell* in OWL, since the word *object* is already a keyword in Java and cannot be used as an identifier - or a so-called *capability*. A *cell* is the term for a single 3D volume in the world, whereas a *capability* has no visual representation itself, but can extend existing cells by new behaviors. In any case, a module in OWL is a set of Java classes, packed into a JAR archive, with well-defined interfaces OWL can call. Developers can extend OWL's functionality in many ways using new *cells* or *capabilities*. For example adding new menu items or new authentication mechanisms, or providing new services like the A.L.I.C.E chatbot, which is a chatbot for natural languages. It was used by Riedmann (2014) to implement a functionality for interactive communication with a Non-Player Character (NPC) and is only one example of the many forms in which OWL can be extended.

One major benefit of the modular approach in form of JAR archives is that OWL does not have to be recompiled and redistributed every time a developer adds a new feature. The archives can simply be uploaded using the web interface of the server, in which case the modules will be installed during the next restart, or even deployed to a running system when developing with an IDE like NetBeans<sup>2</sup>. Furthermore, this modular structure allows to change existing modules OWL comes with or completely replace them with new ones (Finnigan, 2009).

#### Module Content

Modules, *cells* as well as *capabilities*, have to be built following a certain scheme, so that OWL can call their corresponding interfaces and work with them. For instance,

---

<sup>2</sup><https://netbeans.org>

#### 4. Implementation Details

if a module should display some kind of visual object, it has to contain a subclass of OWL's `Cell` class, representing the object. A `Cell` object can draw objects using the *jMonkeyEngine*, or load an art resource (Slott, 2009). On the other hand, if the module should not have any visual representation but instead be something general that other objects should be able to "perform", a *capability* and a subclass of `CellComponent` is needed.

That everything in connection with *capabilities* has the word "component" in its name, has something to do with the history of OWL. Originally, the *capabilities* were called components, but that proved to be somewhat confusing for the users. So the name was changed on the user's side but kept internally because the according classes and interfaces were already named that way.

The server side counterpart of a *cell* or *component* is a "managed object", or "MO". The managed object is a kind of "master-copy" of the `Cell` or the `CellComponent`, making sure that all client copies are synchronized and every user has the same view of the world. If one user changes an object in his world, for example by moving it or changing its size, the client sends a message to the server, which updates the managed object and sends the updated data to all other clients, so they can update their objects. A diagram of the relationship between `Cell` classes and their `Cell MO` can be seen in figure 4.2 from Finnigan (2009).

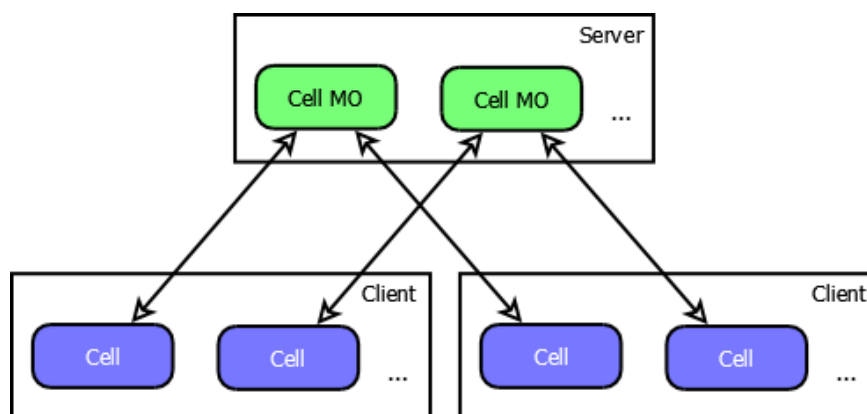


Figure 4.2.: Relationship between `Cell` objects on the different clients and the `CellMO` object on the server. Diagram taken from Finnigan (2009)

For sending status information to the server and back to the clients, and also for initializing all the different objects on the server when OWL starts, the classes `CellServerState` and `CellClientState`, respectively `CellComponentServerState` and `CellComponentClientState` have to be subclassed in every module.

The server state class stores, as the name suggests, the state of the *cell* or *capability* on the server. For example, if the *cell* is to display a brick, the *cell* server state could hold information about the size and the texture of the brick. It is annotated with

JAXB<sup>3</sup> annotations, so that the server state can be saved into an Extensible Markup Language (XML) file and loaded the next time OWL starts to initialize the managed object. The server state class is also used to send the current status of the *cell* or *capability* on the server to all clients to ensure synchronization in case the object was updated on one client.

For sending information about updates to the server, the clients use an instance of `CellClientState` or `CellComponentClientState`. The client state class can hold the same information as the server state class or not, depending on the module.

Another thing that a module that displays an object must contain is a *renderer*, telling OWL how to display the object. The renderer is responsible for actually drawing the *cell*. It must be a subclass of `BasicRenderer` (Finnigan, 2009). A *capability* on the other hand has no graphical representation and does not have to include a renderer. Instead, it has to provide a so-called *properties panel*, where the preferences of the *capability* can be set.

The last important part of a module is a *factory* class. It has to be implemented in order for the module to appear in the cell palette, which is a list of all available modules or for the *capability* to appear in the list of all components. The factory class defines among other things a method for returning the default server state, which shall be used to create the new managed object on the server. Once the MO was created, an instance of `CellServerState` or `CellComponentServerState` is used to inform all other clients about the new module.

In addition to *cells* and *capabilities*, a module can also contain so-called *plugins* for the client or the server side. These pieces of code exist independently from any objects, which have to be added to the world and subsequently take a fixed position in the virtual environment. They can be used for instance to add menu items or for functions like tooltip texts which should be present regardless of the current location of the user.

Finally, a module may include additional resources such as preview images or textures. These should be placed inside a folder called *art*. Then they can be accessed using a special URL (Finnigan, 2009).

### Module Structure

There are three main packages in which the classes listed above should be classified into: *client*, *server*, and *common*. As the name suggests, the *client* package contains everything relevant for the client, that is for instance the `Cell` class, the `CellFactory` class, and the `BasicRenderer` class for any new *cell*. The *server* package holds the MO class, as it is the server side representation of the *cell* or the *capability*. The *common* package contains the classes relevant for both, the client and the server. The

---

<sup>3</sup><https://jaxb.java.net/>

## 4. Implementation Details

client state class and the server state class belong in this category. Of course, there can be other packages relevant for the module, but OWL requires these three.

In the remainder of this chapter, the architecture of the OWL modules developed for this project is discussed in detail. The modules were all implemented with respect to multiuser accessibility, performance, and usability. Because the modules were consistently written in Java, some knowledge of this or a similar programming language is certainly helpful in order to understand the following sections.

### 4.2. Item Module

The developed *Item* module is intended to be a useful tool for exploring large worlds with lots of things to see and information to absorb. One can think of many scenarios where having the possibility to "store" objects in an inventory could become handy. Not without reason, most RPGs and adventure games implement this feature. In case of learning scenarios, it should be used for looking up information about an object one has already seen before, for instance to answer questions or explain something to a fellow student.

#### 4.2.1. Description

One can imagine the *Item* module as consisting of three parts. Firstly, the *Itemize!* component which allows all kinds of 3D objects to be tagged with an information text as well as with a thumbnail image so as to provide information for the students. The *Itemize!* component also adds a menu item to the object's context menu which enables the user to pick up the information text and store it in an *Inventory*. The code for the *Inventory* is the second part of the *Item* module. Lastly, the *Student Manager* module allows the teacher to assign different roles to the students and define for each *Item* which roles are necessary to pick up the information text. All in all, the *Item* module allows teachers to turn an otherwise inanimate world into an information resource for their students. The following subsections represent the three-part structure of the *Item* module:

- *Itemize!*
- *Inventory*
- *Student Manager*

#### **Itemize!**

To turn an object into an *Item*, one has to add the according component in the object's properties window. Figure 4.3 shows the properties panel which is accessible



once the *Itemize!* component has been added and where it is possible to enter an information text and upload a 2D image describing the object. There, it can also be specified which roles the users must have in order to be able to pick up the information text. The concept of roles will be explained later in this section. Another thing that can be specified is whether the *Item* information should be 'available only once'. 'Available only once' means that the *Item* can be picked up only by one student and is not available any more for the other students after that.

It is important that the teacher is registered in OWL as an administrator if he wants to create a new *Item*. Otherwise, he will only see an empty window instead of the *Itemize!* properties panel and will not be able to enter any information.

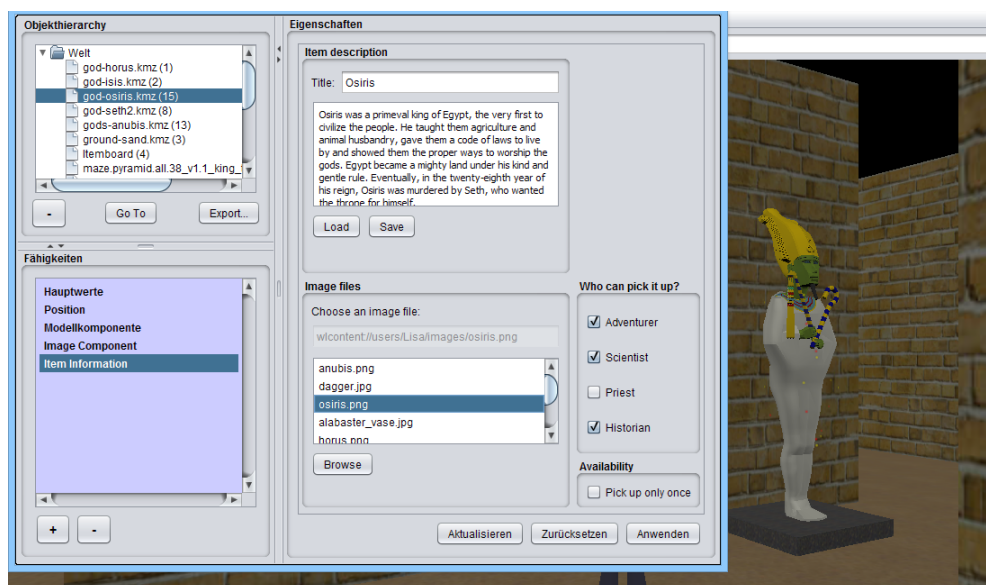


Figure 4.3.: The *Itemize!* capability's property panel

The *Item* information text gets stored on the server along with the image. It is possible to store and load a description text file to prevent having to retype the text for a similar *Item* in another learning world. The description gets stored in form of a XML file. Listing 4.1 shows the format of such a file.

The moment the teacher adds the *Itemize!* capability to an object, the students are able to read and pick up the information. In order for the students to recognize that a certain object has information for them, objects with the *Itemize!* capability are marked with a glitter effect. This comes in form of small yellow and red dots rotating around the 3D model, as can be seen in figure 4.4. The effect appears as soon as the capability is added. In case the teacher has selected the 'available only once' option, the effect disappears when a user picks up the *Item*.

In any case, the students can see the information provided for them as a kind of tooltip text when they hover over the object with their mouse cursor. Figure 4.5(a) shows what that looks like. The students can pick up the information by

#### 4. Implementation Details

Listing 4.1: Exemplary *Item* description xml file

```
<item>
  <title>Isis</title>
  <content>Isis was the sister and wife of Osiris. On his side she
    ruled over Egypt. When she heard of her husband's murder, she
    was grief stricken. She tirelessly searched for his body so that
    he may be properly buried and rest in peace in the Underworld.
    When she finally found him, she and her sister Nephthys used
    their magic to revive him just long enough, so that Isis could
    conceive a child with him. The child was a boy named
    Horus.</content>
  <date>2014-06-10</date>
</item>
```



Figure 4.4.: Glittering objects signaling that there is some information to pick up

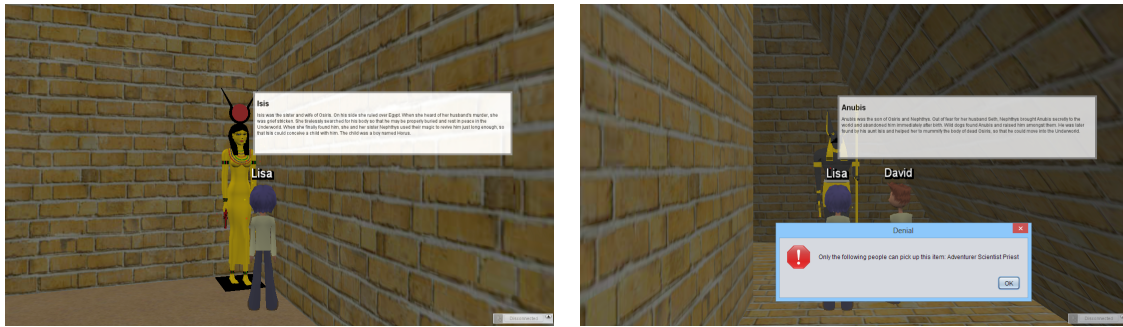


Figure 4.5.: Left: The *Item* information text of an Egypt statue; Right: Student does not have the right role for picking up the information

right-clicking it and choosing 'Pick up'. If they do not have one of the roles the teacher specified for this object, the students are not able to pick it up and an error message appears as in figure 4.5(b). Otherwise, the *Item* information gets stored in their *Inventory*. The process of assigning roles to students will be discussed later in this section.

## Inventory

Items picked up by a student appear in his *Inventory*. The *Inventory* can be accessed via the 'Tools' menu and contains an overview of all objects available in the world. Objects which the student has already picked up appear in blue color and if the student clicks on them the information text and the associated image are shown. The *Inventory* is depicted in figure 4.6.

The other object names are either red or gray. Red means the object is still available and has yet to be found by the student. If an object name appears in gray it means the *Item* cannot be picked up by this student either because of his role or because someone else has already picked it up and the teacher has marked it as 'available only once'. The color coding was implemented to enhance the motivation of the students through the approach of gamification. Furthermore, in the case of a grayed out *Item*, a hint is shown to the user providing him with the name of the user who has picked up the *Item* or might be able to pick it up because he has the right role. This is supposed to increase collaboration among the students. In figure 4.7 such a hint can be seen.

Students can and should use the content of their *Inventory* to answer the questions of the *Quiz* at the end of the learning experience. Information they were not able to obtain by themselves should be given to them by their fellow students.

In figure 4.8 it can be seen that there is a button left of the object's thumbnail image. This button is to return the *Item*, e.g. to forfeit the *Item* information. If the *Item* was

#### 4. Implementation Details

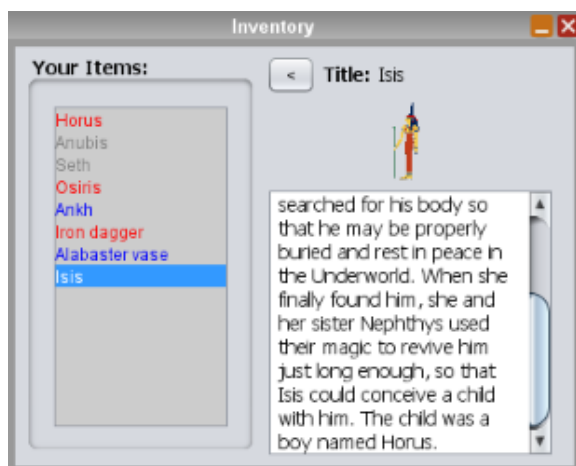


Figure 4.6.: *Inventory* showing a picture and some information of the Egyptian goddess Isis



Figure 4.7.: *Inventory* showing an *Item* the user cannot pick up plus the hint given

'available only once', it can be picked up by other students again after it has been returned by the student who picked it up first.

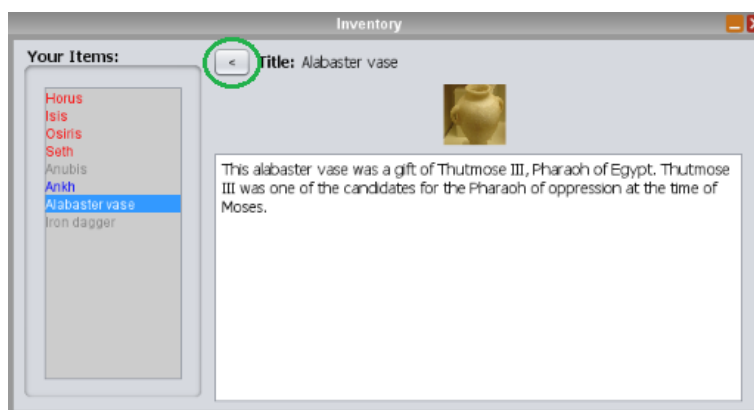


Figure 4.8.: Button to return an *Item* left of its image in the *Inventory*

#### Student Manager

With the help of the *Student Manager*, the teacher can assign one of four roles to the students. The *Student Manager* shows a list of users who are currently online and the teacher can choose a role for each of them. The list of users is obtained in this case from the OWL presence manager API. Figure 4.9 shows how the *StudentManager* window looks like.

The *Student Manager* panel is accessible via the 'Tools' menu. As with the *Itemize!* properties panel, the teacher has to be administrator to be able to call up this panel. If a student who is not an administrator clicks on the according menu item, an

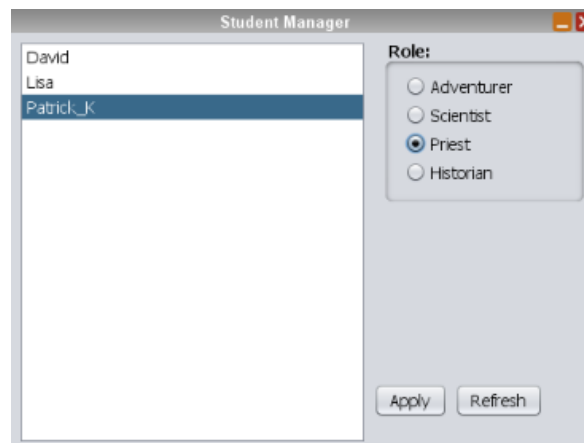


Figure 4.9.: *Student Manager* panel where one student has been given the role 'Priest'

information is shown to him containing his current role but he will not be able to see the list of all users and change his role or those of others.

For the showcase scenario described in chapter 5, there are four roles to choose from: Adventurer, Scientist, Priest, and Historian. For each *Item* the teacher can specify which role one must have in order to pick it up. So to be able to obtain any information at all it is crucial to have been assigned one of these roles. The names of the roles are predefined and cannot be changed by a teacher. Further development of this work could include a dynamic management of roles by the person creating the new learning environment.

### 4.2.2. Architecture

Like most OWL modules, the *Item* component consists of three packages, namely client, common, and server. In the client package all classes responsible for the client side logic and rendering can be found. The package common contains classes which are necessary for both, the client and the server side of the module. Lastly, the server package consists of those classes necessary to keep all clients synchronized (the "managed objects", or "MO classes") as well as any server plugins.

#### Client

In the following section, class diagrams and descriptions of the most important classes on the client side of the *Item* module can be found. For the sake of simplicity and readability, some classes and relations have been omitted from the class diagrams and others might not be described in full detail. Figure 4.10 shows the first part of the client side classes.

## 4. Implementation Details

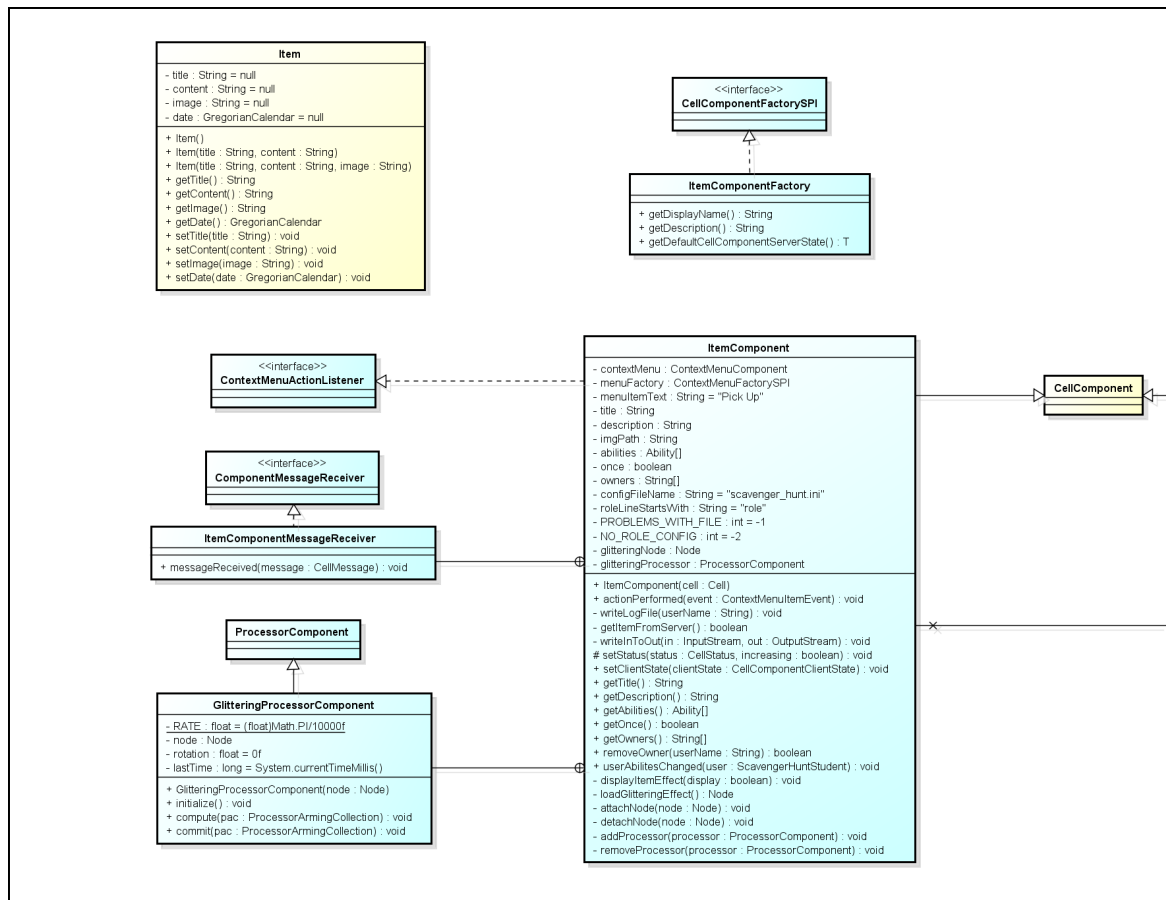


Figure 4.10.: Class diagram showing the client side of the *Item* component - Part 1. Classes written especially for the *Itemize!* capability are colored in blue.

- ItemComponentFactory**  
 For the users to be able to see the *Itemize!* capability in the list of available components so they can select it and add it to objects, it is necessary to implement the *CellComponentFactorySPI*, which *ItemComponentFactory* does. This class contains methods to display the component correctly in the object editor. Additionally, the method *getDefaultCellComponentServerState()* returns an instance of *ItemComponentServerState*, which holds the state of the server side representation of the component and tells the server, which *CellComponentMO* to create.
- ItemComponent**  
 This class represents the component itself on the client side. An instance of it is created for each object with the *Itemize!* capability. It stores the title, description and image of the *Item*, its roles, and whether it can be picked up only once. It also creates an additional context menu item 'Pick up' for all objects with the *Itemize!* capability and implements the interface *ContextMenuActionListener*

to be able to react to action events on this menu item.

When the 'Pick up' menu item is clicked, the role of the current user is loaded and compared with the roles of the clicked *Item*. The roles of the students are stored in files on the server in the corresponding user directories. If it is decided that the student has one of the roles specified for the *Item* and is therefore able to pick it up, it is further checked whether nobody else has already picked up the *Item*, in case it may be picked up only once. To keep track of all owners of an *Item*, they are stored in an array. Each time a user picks up or returns an *Item*, the array is updated and a `CellMessage` is sent out to all other *Item* cells.

Given the checks show that the role of the user matches and that the *Item* is still available, an XML file with the *Item*'s content is created on the user's computer and the image file gets downloaded from the server and stored in the same directory. On the basis of the contents of this directory the *Inventory* panel displays the different *Items*. For reading and writing XML files, the open source program interface JAXB is used.

The class `ItemComponent` is further responsible for displaying the *Item* glitter effect. To achieve this, a special `ProcessorComponent` is made use of. It loads a KMZ model and rotates it every couple of milliseconds by a fixed rate to create the sparkling effect. The owner array is used again in this situation, together with the variable `once`, to determine whether the *Item* effect should be activated or not.

In figure 4.11 the second part of the client side classes is displayed.

- `ItemComponentClientPlugin`

The window with the *Item* information text, which appears when a user hovers over an *Item* with the mouse cursor, was realized with a `HUDComponent`. The class responsible for this is `ItemComponentClientPlugin`. It uses an `EventListener` which listens to `HoverEvents` to detect whenever the mouse cursor enters an object. In that case, the primary cell which triggered the event is checked and if it is found to have the *Itemize!* capability, a `HUDComponent` is created and filled with the title and the description text of the *Item*. The information for that is obtained directly from the `ItemComponent` class which can be accessed via the method `getComponent()` of the triggering cell.

The `HUDComponent` in succession gets displayed by the main HUD. HUD stands for "Heads-up display" and refers to a 2D region of the OWL client window where HUD components can be displayed. A HUD component can be any 2D or 3D visual object. In our case it is a nicely formatted `JPanel` which is passed to the `HUDComponent` instance at creation time.

## 4. Implementation Details

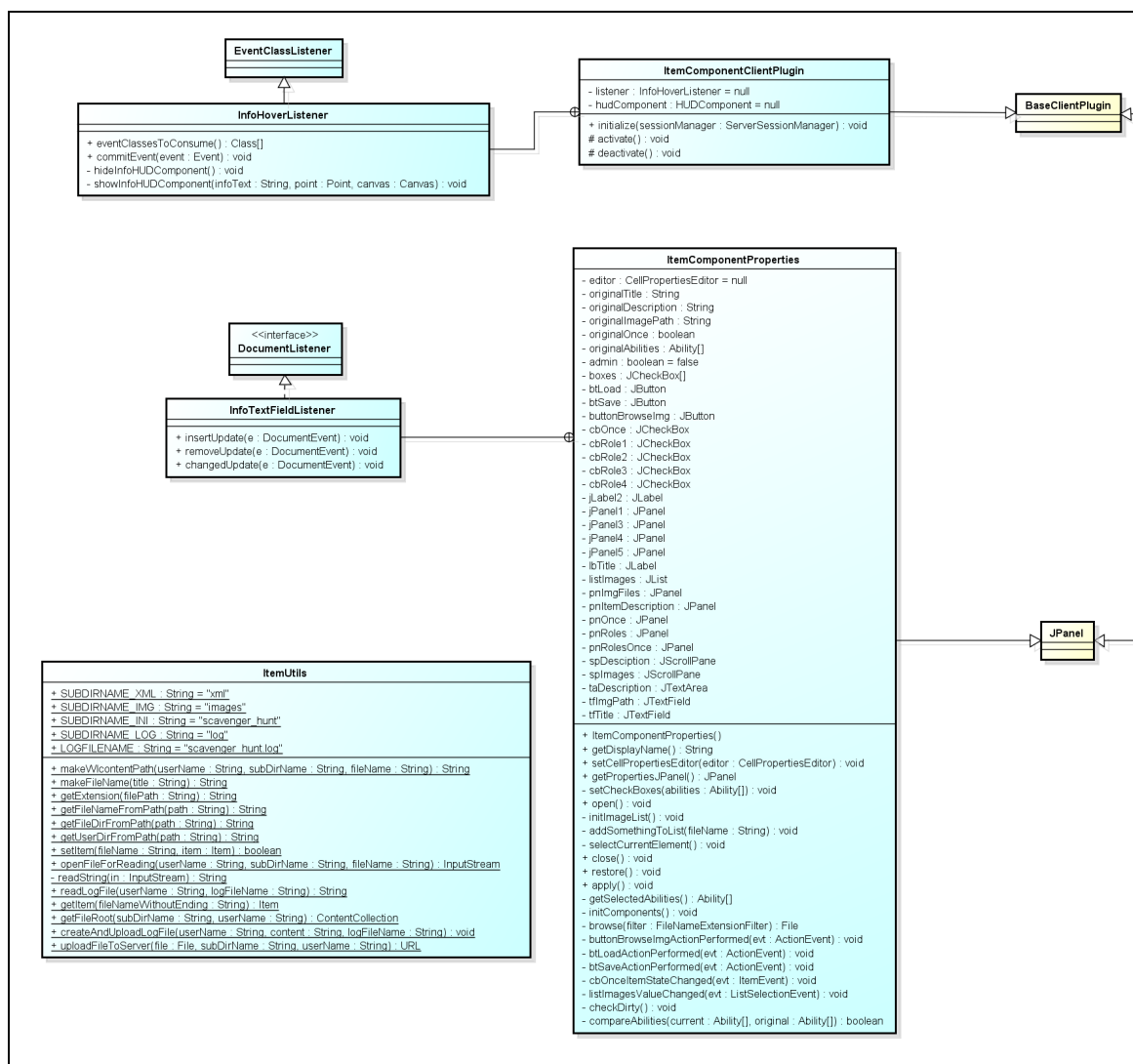


Figure 4.11.: Class diagram showing the client side of the *Item* component - Part 2. Classes written especially for the *Itemize!* capability are again colored in blue.

- **ItemComponentProperties**

The question remains, how the *Item* information text and the image, as well as the other *Item* information, get to the server in the first place. This is the task of the class *ItemComponentClientPlugin*. It represents the panel where the *Item* information can be entered by the teacher. First, the information gets loaded from the *ItemComponentServerState* which is accessible via the *CellPropertiesEditor* member variable. If the teacher enters something into the text fields or chooses another value for the image file or the roles, a *DocumentListener* reacts to the changes and enables the 'Apply' button with which the changes can be taken over and stored on the server. This happens



via the set methods of the class `ItemComponentServerState`.

Once a title or description text has been entered in the *Item* component properties panel, it is possible to store it on the server for later usage. Similarly, it is possible to load an existing information text which was stored on the server previously. Furthermore, if the teacher wants to use another image for his newly created *Item*, he can simply upload one from his computer. For all those file manipulations on the server the class `ItemUtils` is used.

As students should not be able to change the *Item* information text or, even worse, the roles of an *Item*, the `ItemComponentProperties` panel was made only accessible to administrators, that is, member of the "admin" group on the server. To ensure this, the current user's avatar cell is checked whether it has the *StudentManager* capability, which only administrators have. More on that can be found in the description of the `StudentManagerServerPlugin`.

- `ItemUtils`  
All methods to interact with files on the server can be found in the class `ItemUtils`. It has methods to read files, as well as create and upload them. For this purpose, it uses the OWL Content Repository. Methods such as `getUserRoot()`, `getChild()` or `createChild()` allow to navigate the directory structure on the server. With the method `put()` of the class `ContentResource` content can be uploaded. The method `getInputStream()` is used for opening a stream in order to read files.

The classes for the *StudentManager* functionality can be seen in figure 4.12.

- `StudentManagerComponent`  
The `StudentManagerComponent` is used to verify whether a user has admin rights or not. Therefore, a server plugin was written which adds this component to any avatar cell where the user is member of the "admin" group. On the client side it then only has to be checked, whether the primary view cell (the avatar) of the current user has the *StudentManager* capability. The class is only used for this purpose; it does not store or manage any additional information.
- `StudentManagerClientPlugin`  
Here, the 'StudentManager' menu item is created and added to the 'Tools' menu. When it is clicked by a user, the verification described before takes place in order to decide, whether a HUD with the `StudentManagerPanel` should be displayed. It should be displayed, of course, only if the current user is an administrator. Otherwise, the role of the user is loaded from a file and shown to him in a simple `JOptionPane` window.

In case the user has admin rights, the `StudentManager` is used to load all users and their roles from the respective files and the `StudentManagerPanel` is

## 4. Implementation Details

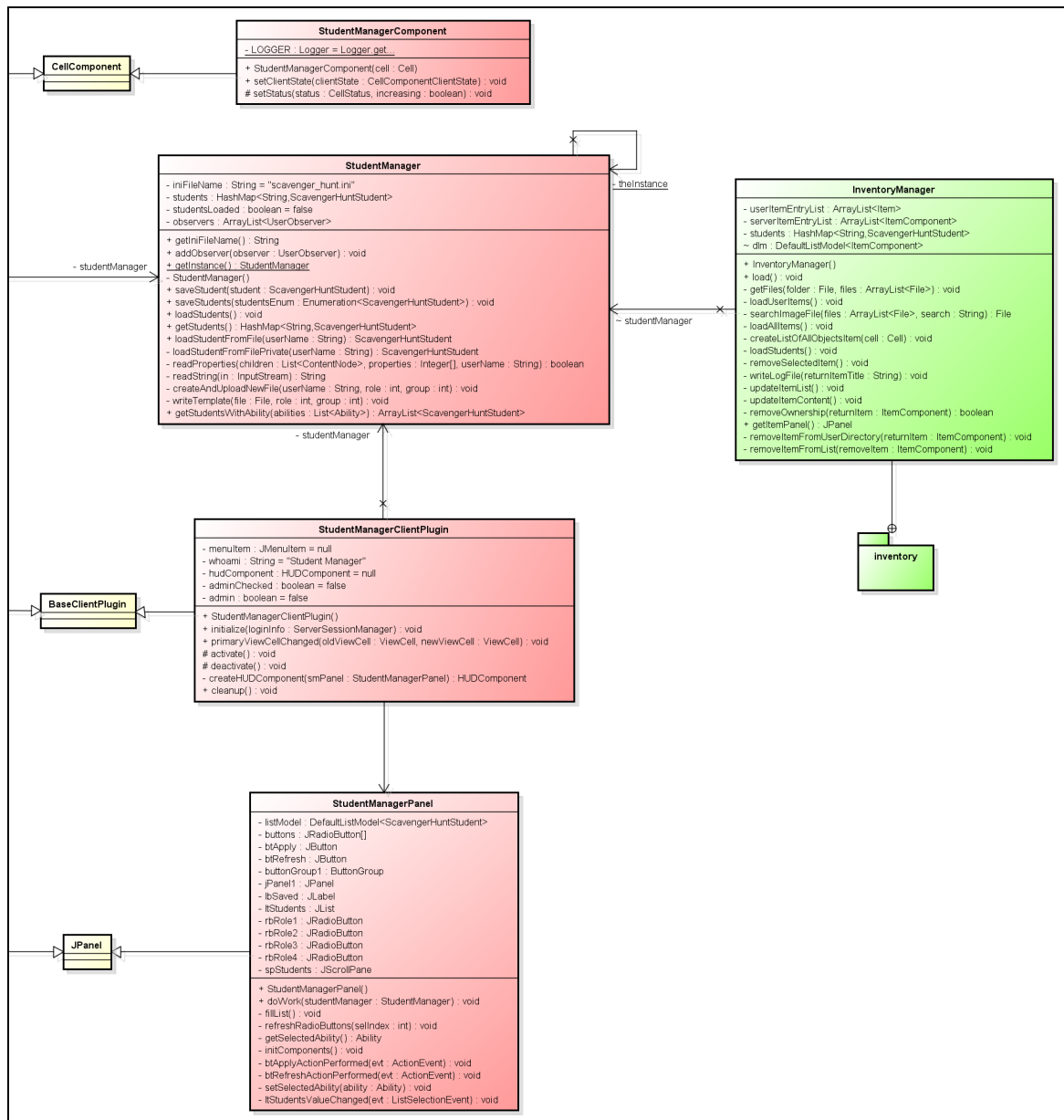


Figure 4.12.: Class diagram showing the *Student Manager* part on the client side of the *Item* component in red. Colored in green are those parts which already belong to the *Inventory* plugin but are relevant for the *Student Manager* part as well to understand how the two parts are connected.

displayed in a HUDComponent. When a new role for one of the users is selected, the 'Apply' button has to be clicked and the user's file on the server is updated.

- **StudentManagerPanel**

The StudentManagerPanel is used to display all users and their roles. As soon as the users are loaded by the StudentManager, the panel's JList is

filled and the JRadioButtons are initialized. The panel then is displayed in a HUDComponent by the StudentManagerClientPlugin.

- **StudentManager**  
This class contains all methods regarding the management of the user's configuration files on the server. The configuration files currently only store the role of each user but can be extended by further attributes any time. The StudentManager class has methods to load students and read configuration files whereas new, empty files are created for users who do not have configuration files yet. Changes in students' roles can be stored back to the server. For actual file manipulations, again, the methods of the helper class ItemUtils were used.

Figure 4.13 shows the classes which implement the *Inventory* function on the client side of the *Item* module.

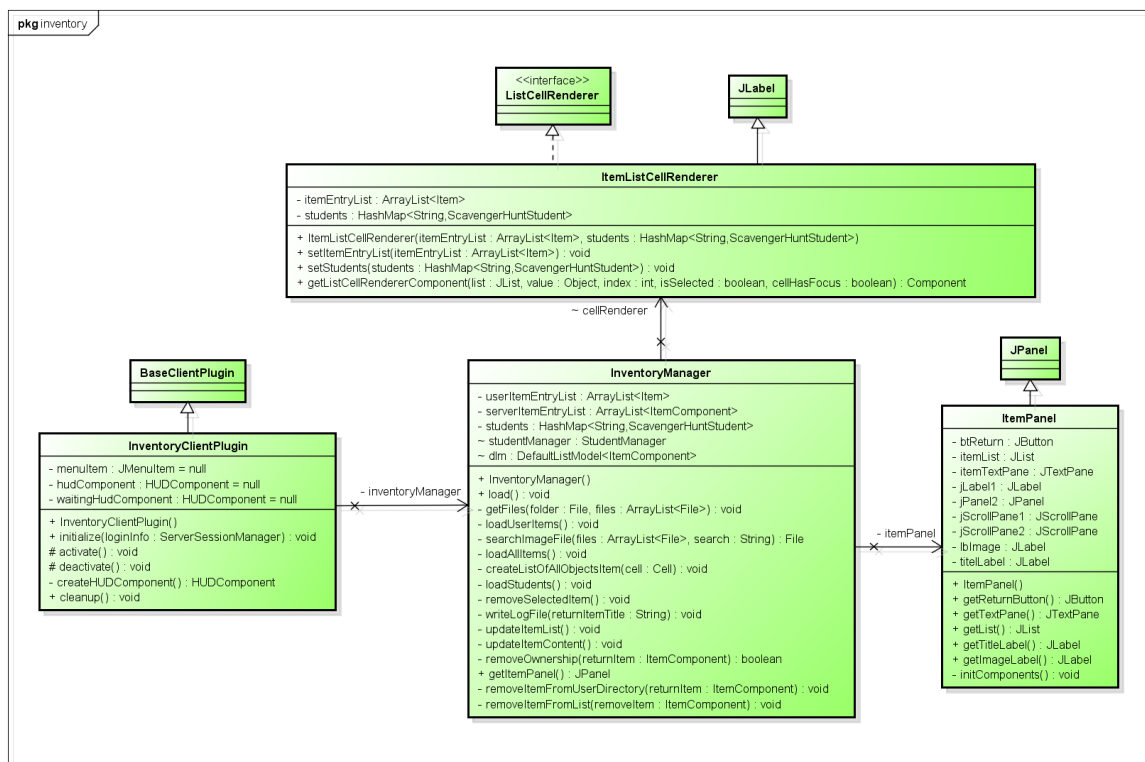


Figure 4.13.: Class diagram showing the *Inventory* part on the client side of the *Item* component in green color.

- **InventoryClientPlugin**  
This client plugin adds the 'Inventory' menu item to the 'Tools' menu. When the item is clicked, the InventoryManager is called to load all necessary information and initialize the ItemPanel. After that, a HUDComponent which

## 4. Implementation Details

displays the panel is constructed.

- **InventoryManager**  
The class `InventoryManager` loads all *Items* available in the world, as well as all *Items* stored as XML files in the user directory to display the description text and image of those *Items*, the user has already found. It furthermore manages the `ItemPanel`, which it passes back to the `InventoryClientPlugin`, and the `ItemListCellRenderer`.
- **ItemPanel**  
The `ItemPanel` shows a list of all available *Items*, along with a windows for the *Item* descriptions and images. A return button to give back *Items* already found was also implemented. The panel gets managed by the `InventoryManager`, which listens to click events on the *Item* list and updates the content window accordingly.
- **ItemListCellRenderer**  
For coloring the *Item* titles in the list of all *Items* in the `ItemPanel`, the `ItemListCellRenderer` is used. First, it identifies the name of the current user and his role in order to be able to compare it to the roles of the current *Item*. That, along with the owners of the *Item* and the information if it should be picked up only once determines the color of the *Item* title.

### Common

Next, the common-package is described, which contains all classes relevant for both, the client and the server side of the module. The client state, the server state, and an arbitrary amount of cell messages are usually part of this package. Figure 4.14 shows most important common classes implemented for the *Item* component.

- **ItemComponentServerState**  
This class is responsible for delivering status updates to all clients. Therefore, it holds the *Item* title, description, image path, abilities, owners, and if it should be picked up only once. In addition to that, the method `getServerComponentClassName()` returns the fully-qualified name of the MO class telling the system which server side class to instantiate.
- **ItemComponentClientState**  
Similarly, the class `ItemComponentClientState` is used to communicate changes of an *Item*'s values on the client side to the server MO. It also has all the member variables necessary to store the information of an *Item*.
- **StudentManagerComponentClientState**  
The class `StudentManagerComponentClientState` is responsible for communi-

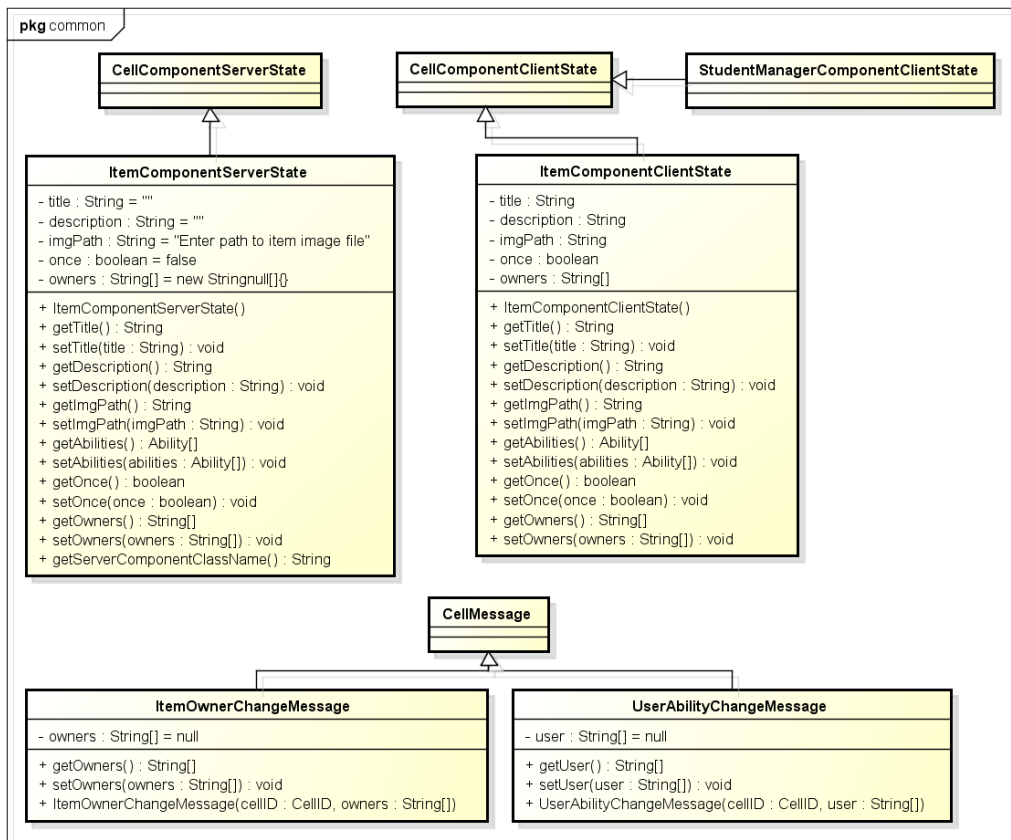


Figure 4.14.: Class diagram showing the package common of the *Item* component

ating changes of the *StudentManager* attributes to the server MO. Since the *StudentManager* capability does not have any additional attributes beneath its existence, this class does not have any member variables.

- **ItemOwnerChangeMessage**  
If someone picks up or returns an *Item*, a message is sent to all other clients so that they can update their *ItemComponents*. For that reason, the class *ItemOwnerChangeMessage* is used. It notifies the other clients whenever there was a change in the ownership of an *Item* for them to refresh e.g. their *Item* glitter effect immediately.
- **UserAbilityChangeMessage**  
In case the user's configuration file does not get written fast enough when the teacher changes the role of a student in the *StudentManager*, an additional cell message, the *UserAbilityChangeMessage*, is sent so that the list of all students containing their roles can be updated on all clients.

## 4. Implementation Details

### Server

Located on the server side of the module are the Managed Objects (MOs) as well as any server plugins. The MOs contain a server side representation of the components and, if necessary, some logic for synchronizing the clients. Typically, this involves, but is not limited to, receiving and redistributing of cell messages. In figure 4.15 the server side classes of the *Item* module can be seen.

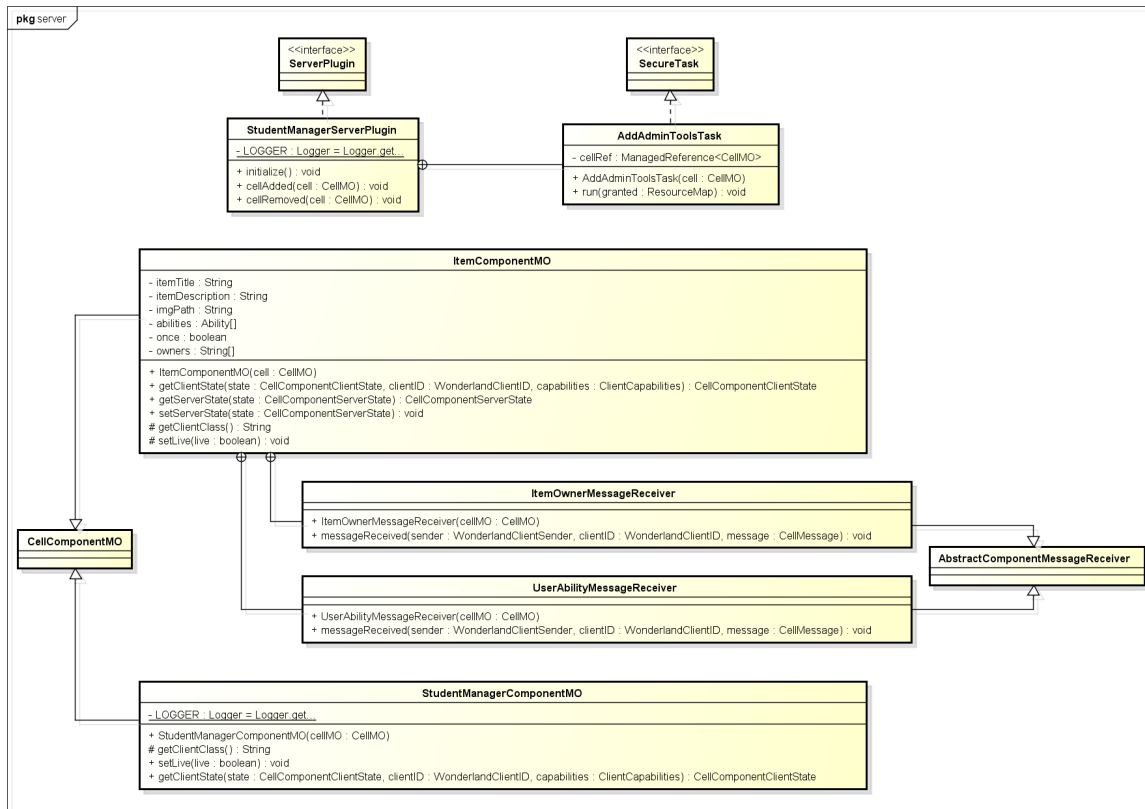


Figure 4.15.: Class diagram showing the server package of the *Item* component

- **ItemComponentMO**  
The class `ItemComponentMO` represents the *Item* component on the server side. It can be seen as a "master copy" and serves as reference in order for all clients to be in synch. Additionally to containing all the necessary member variables to save the information of one *Item*, it also manages the reception of `ItemOwnerChangeMessages` and `UserAbilityChangeMessages`.
- **StudentManagerComponentMO**  
This class represents the *StudentManager* capability on the server. Since only the existence of this capability is checked on the client side and no additional attributes are needed, this class does not have any member variables.

- `StudentManagerServerPlugin`

A `ServerPlugin` is loaded in the very beginning when the Darkstar server first starts up. Thus, code can be executed without the user being able to influence it. In this case, the `StudentManagerServerPlugin` is used to mark every avatar cell where the user is a member of the "admin" group on the server. So every time a user calls up the *StudentManager* it can be checked whether the user's avatar has the *StudentManager* capability to determine if he is allowed to see the list of all users and change their roles or if he should only be able to see his own role. The idea of identifying administrators that way was taken from the *Admin Tools*, an already existing OWL-module.

## 4.3. Quiz

The *Quiz* component is another capability that can be added to any object in the virtual world. Similar to the *Concept Question* module described in Pirker et al. (2013) it teleports students to a predefined place depending on whether they answered a set of questions correctly with the slight difference that the teleport action is executed only on correct answers and no action for wrong answers can be defined. The questions that have to be answered can be specified by the creator of the virtual learning experience. For example, a teacher can ensure that way that his students really deal with the information they obtain and do not only pick up the information texts without reading them.

### 4.3.1. Description

As with any capability in OWL, the *Quiz* has to be added to a 3D model via the object editor. In the according properties window the teacher then can add, delete and modify questions on the learning topic to his liking, as can be seen in figure 4.16. For every question, one to six different answers can be defined, whereas for every answer it can be selected if it is correct or not. In the current version of the *Quiz* module only multiple-choice questions are supported. For the next version also questions where students can enter free text are thinkable.

To facilitate the creation of new learning experiences, the *Quiz* capability further allows for quizzes to be stored on the server so that they can be loaded and used at another time. Similarly to the *Item* component, XML files are used for that purpose. Listing 4.2 shows an example of how such a file could look like.

At the bottom the *Quiz* properties window contains four text fields to enter a destination where users should be teleported to when they have answered all questions correctly. This location is specified by three OWL coordinates, x, y, and

## 4. Implementation Details

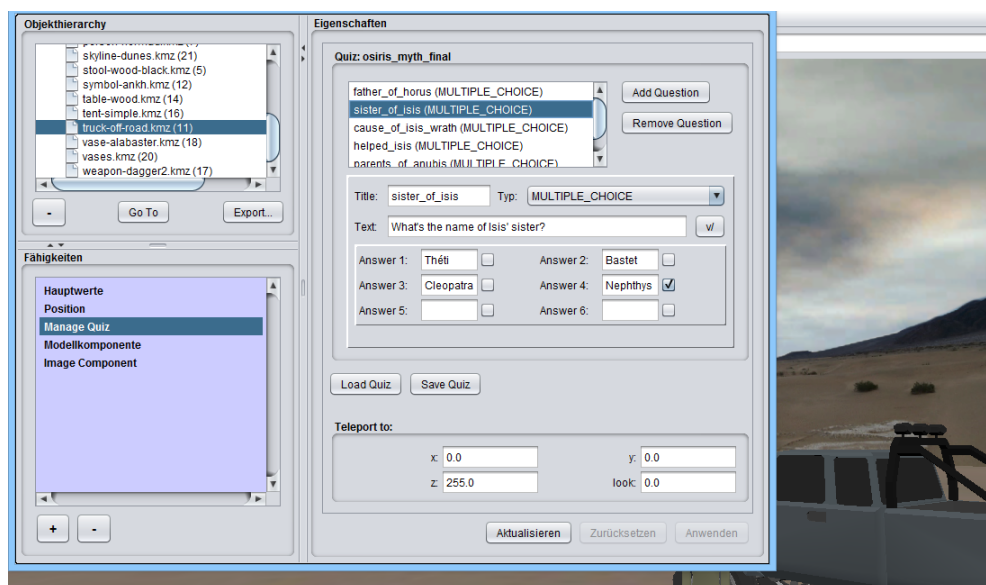


Figure 4.16.: The Quiz capability's property panel. At the top, a list of already created questions is displayed. Currently, only the type "MULTIPLE\_CHOICE" is available. For each question the title, text, and up to six answer possibilities can be specified. Additionally, there are checkboxes beneath each answer with which the teacher can mark an answer as correct. Quizzes can be stored to modify/reuse them at a later time. Finally, at the bottom, the teleport location where the students are transported to can be entered using OWL coordinates.

Listing 4.2: Exemplary Quiz xml file

```

<Quiz>
  <name>osiris_myth_final</name>
  <questions>
    <title>father_of_horus</title>
    <type>MULTIPLE_CHOICE</type>
    <text>Who was the father of Horus?</text>
    <answers>
      <entry><key>Osiris</key><value>>true</value></entry>
      <entry><key>Seth</key><value>>false</value></entry>
      <entry><key>Anubis</key><value>>false</value></entry>
      <entry><key>Imhotep</key><value>>false</value></entry>
    </answers>
  </questions>
  <!--...more questions...-->
</Quiz>

```

z, as well as a look direction. The look direction should be set to zero in order to position the avatar in a forward-looking way.



Once added to a 3D object, the *Quiz* capability pops up a window containing all the questions defined by the teacher as soon as the avatar of the user collides with the object. Beneath the questions, a set of comboboxes is displayed where the students can choose their answers. According to what the teacher has specified, one, many or even no answer at all can be correct. Figure 4.17 shows what that looks like.

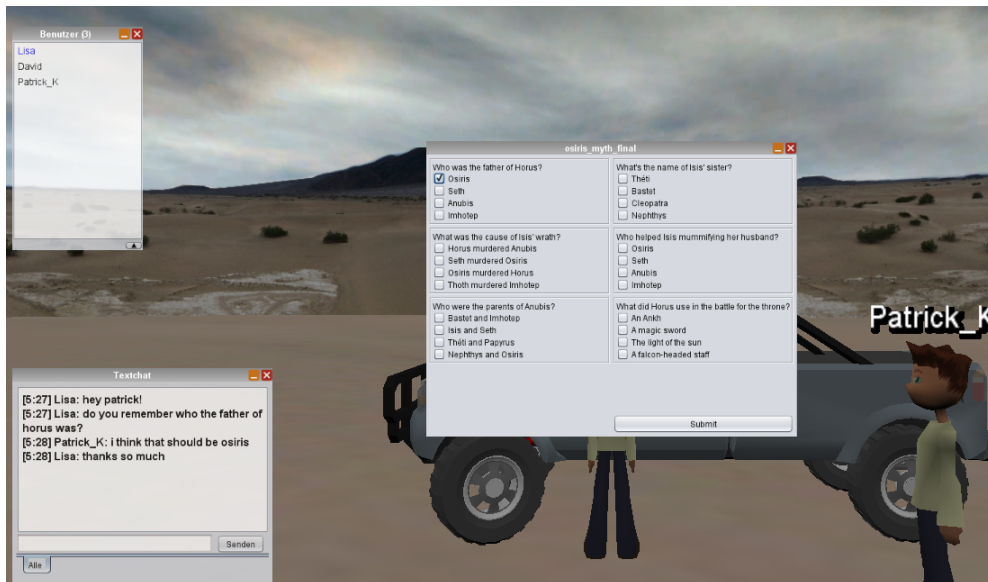


Figure 4.17.: Student attempting to solve the *Quiz* with a second student helping

When the students believe they have answered all questions correctly, they can press the 'Submit' button to have their answers checked. If it is found that all answers are correct indeed, the teleport function is executed. If not, an according message is displayed. Students can answer quizzes as often as they like. Each time they press the 'Submit' button, their attempt gets written to a log file. Thus, the teacher has a way to check if any learning progress took place among the students.

To prevent students from right clicking on a *Quiz* object and look up the correct answers, the objects should be made non-clickable before the learning round-trip begins. Alternatively, a mechanism similar to the *StudentManager* component could be implemented which allows only administrators to edit the *Quiz* properties panel.

### 4.3.2. Architecture

In this section only the client side of the *Quiz* module is described since the packages common and server very closely resemble those of the *Item* component, containing client and server states, cell messages, and server MOs. As in the previous section, some classes are shown in simplified form to maintain the readability of the

## 4. Implementation Details

diagrams and to be able to reduce the descriptions to the essentials. Figure 4.18 shows the classes on the client side of the *Quiz* module.

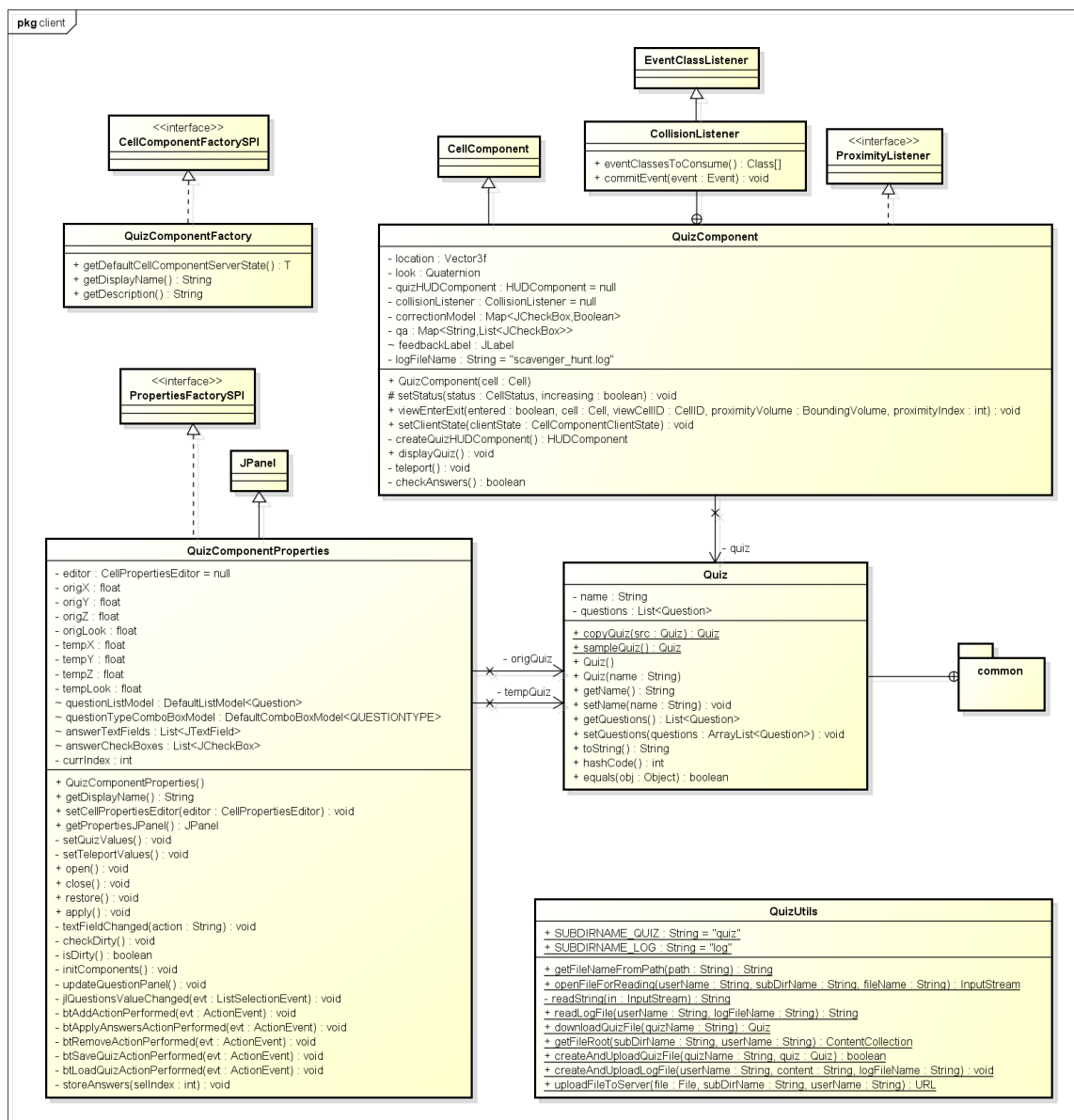


Figure 4.18.: Class diagram showing the client side of the *Quiz* component

- **QuizComponentFactory**

Just as in the *Item* module, the factory class of the *Quiz* module is responsible for correctly displaying the component in the object editor. For that purpose it has exactly the same methods to return the name of the module and a short description of its functionality, as well as a method with which the default server state can be retrieved so that the server knows which cell MO to create.

- `QuizComponent`

The aim of the `QuizComponent` class is to show the window with questions and answer options whenever a user is in the proximity of the *Quiz* object. It implements the class `ProximityListener` and adds a `CollisionListener` to the object cell for that reason. Both react to the respective events with the creation of a HUD component containing the questions specified for the *Quiz* component and the 'Submit' button.

A click on the button starts the routine for checking the selected answers. To facilitate this, a map is created when constructing the *Quiz* window in which all the radio buttons and the truth values of the associated answers are stored. When checking the correctness of the given answers at the same time also a log file is written to the server containing all the answers of the user along with the information if the answers are correct. The teacher can use this log file to monitor the learning progress of the students.

If it is found that all questions have been answered correctly, the teleport function gets executed. The code for this function has been taken from the already existing OWL *Portal* component.

- `QuizComponentProperties`

To be able to manage the attributes of an object with the *Quiz* capability there has to be a class which implements the interface `PropertiesFactorySPI`. In our case this is the class `QuizComponentProperties`. It ensures that whenever a user adds the *Quiz* capability to an object, there is the `QuizComponentProperties` panel where the questions and coordinates of the teleport destination can be managed.

When first opened, the according text fields are loaded with the values from the server or initialized to zero, respectively empty strings, if there is no managed object for this component on the server yet. Upon entering or changing some values using this text fields, the 'Apply' button is activated and the changes can be communicated to the server via the server state object.

The functionality to save and load whole quizzes is similar to the *Item* component. For the *Quiz*, too, XML files storing questions and answer options get written to the server. For file manipulations the helper class `QuizUtils` is used.

- `QuizUtils`

This class contains various methods to read and write files that reside on the server. Like the *Item* module it uses the OWL Content Repository for that. The open source API JAXB was included and the methods `marshal()` and `unmarshal()` were used to create an XML file out of a *Quiz* object, and vice versa. In order to make working with JAXB possible, the class *Quiz* had to be

## 4. Implementation Details

provided with the according annotations.

- Quiz

The class `Quiz` represents a quiz instance entered into the properties panel of a *Quiz* object. It contains the name of the quiz and a list of questions, whereas a question in turn consists of a title, the question type, a question text, and a data structure mapping a number of answer strings to boolean values indicating if the answers are correct or not. The class is annotated with JAXB annotations so that it can be stored on the server easily.

### 4.4. Itemboard

The *Itemboard* is an extension of the existing OWL *Whiteboard* module, which provides the users with a simulated canvas where they can write and draw on, using different tools known from various graphics painting programs. It can be used like a whiteboard in the real world meaning everyone is able to immediately see what the other users are writing or drawing. This makes productive meetings and brainstormings in the virtual world possible.

#### 4.4.1. Description

For our virtual learning experience discussions among the students and the exchange of information after the scavenger hunt should be facilitated. Therefore, they should be able to place *Items* they already found on the board for other students who have not been able to pick this information to see. While the original *Whiteboard's* toolbar only has buttons to draw geometric figures and insert one short line of text, the *Itemboard* complements these capabilities by an additional button to "pin up" the text and the image of those *Items* one has stored in the *Inventory*. The process of placing an *Item* on the *Itemboard* can be seen in figure 4.19.

The button to pin up an *Item* is located at the rightmost side of the toolbar. The user has to first select the button and then click on a place inside the white space to open the dialog to select an *Item* from the *Inventory*. As a result, the *Item* information text and image are placed at the exact point where the user has clicked before. Of course, this position can also be changed in order to arrange a set of *Items* on the board. Therefore, the user has to click on the selection tool marked by the white mouse cursor and drag the image or the text to the position where he would like to have it. The picture and text can be moved independently.



Figure 4.19.: Student adding one of his *Items* to the *Itemboard*

#### 4.4.2. Architecture

Most of the functionality to add things to the drawing space and to synchronize the canvas between the clients was already provided by the OWL *Whiteboard* module. So here only the modifications needed to implement the placing of *Items* are described. These mostly concern the client side of the module whereas on the server side only minor changes were necessary. Figure 4.20 shows the first part of the classes implemented or modified for the *Itemboard* module.

- **ItemboardControlPanel**  
The first task was to add a button with an icon for placing an *Item* on the canvas to the *Itemboard* toolbar. The toolbar is represented by the class *ItemboardControlPanel*. In addition to the button also a tooltip text was added to improve usability.  
  
Whenever a button on the toolbar is pressed, a set of classes which implement the *ItemboardCellMenuListener* is notified of the event. The interface defines that all implementing classes must have a method for each of the buttons on the toolbar. In case of the new *Item* button the method *item()* is called.
- **ItemboardToolManager**  
The class *ItemboardToolManager* implements the interface *ItemboardCellMenuListener* to get informed whenever a user presses one of the buttons on the *ItemboardControlPanel*. According to which button has been clicked, a method is called which sets a member variable to represent the selected tool. The method *item()* sets this variable to the element *ITEM* of

## 4. Implementation Details

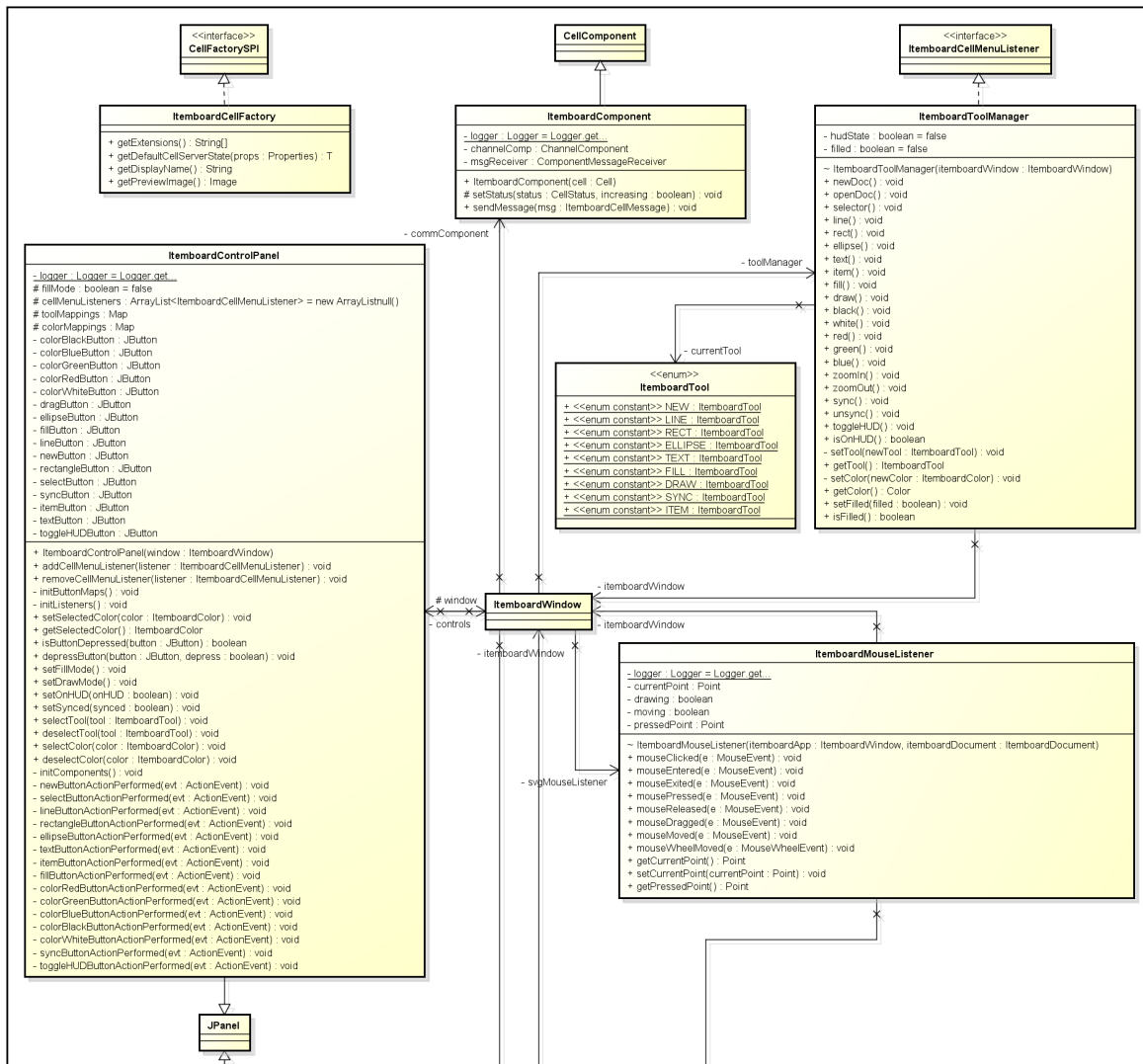


Figure 4.20.: Simplified class diagram of the client side of the *Itemboard* module - Part 1

the enum `ItemboardTool`. If necessary, the member variable can be retrieved via the method `getTool()`.

- **ItemboardMouseListener**

This class listens to mouse clicks on the drawing canvas. When it detects a click it checks which tool is selected by retrieving the current tool from the class `ItemboardWindow` which in turn gets the currently selected tool directly from the `ItemboardToolManager`. The information then gets passed on to the `ItemboardDocument` where the *Item* element finally gets added to the board.

In figure 4.21 the second part of the most important client side classes of the *Itemboard* module can be seen.



Figure 4.21.: Simplified class diagram of the client side of the *Itemboard* module - Part 2

- **ItemboardDocument**

The process of creating an *Item* element on the *Itemboard* is executed in a separate thread, the *ItemGetter*. There, the *ItemChooserPanel* is instantiated and inserted into a HUD component.

Once the user has selected an *Item* from the *ItemChooserPanel*, an SVG "image" element is created. The the image of the selected *Item* is resized to a maximum width and height. In order for the image data to be displayed, a

## 4. Implementation Details

base-64 binary string is constructed and added to the SVG element. Lastly, a multiline text element is created for the *Item* description text making use of the SVG element "tspan". Correct line breaking is ensured with the help of the `StringTokenizer` class and some basic `String` functions.

To make sure that the newly created SVG elements are also able to be selected and moved around the board, the method for dragging had to be extended to also recognize "image" and "tspan" elements. The movement itself as well as the calculation of the drawing overlay are done by the class `Itemboard Window`.

- `ItemChooserPanel`  
The `ItemChooserPanel` is displayed by the `ItemGetter` thread so that the user can choose an *Item* to be added to the *Itemboard*. It consists of a list filled with all the *Items* of the *Inventory* and two buttons to confirm the selection or to cancel the process.
- `ItemboardUtils`  
The aim of this class is to provide helper functions for the work with the Document Object Model (DOM) and XML Strings. In addition to that, it contains methods to calculate the boundaries of the different shapes (line, rectangle, ellipse) and a method to wrap text.
- `ItemboardLogFileUtils`  
While the class `ItemboardUtils` was created by the OWL community and was only extended for the *Itemboard* module, the class `ItemboardLogFileUtils` was newly created and is responsible for managing the log file. As in the *Item* and the *Quiz* module, the OWL Content Repository is used for accessing files on the server, for reading an input stream is retrieved and for uploading files the method `put()` of the class `ContentResource` is used.

### 4.5. Summary

In this chapter a brief overview of the structure of an OWL module was given and the modules that were developed for this work were presented. In addition, some already existing OWL modules that were used as they were or slightly adapted to fit the needs of a collaborative learning environment were mentioned.

First, there was the *Itemize!* capability which allows to provide any 3D object with an information text and a preview image turning it into a source of information for the users. The information text can be picked up and stored in an *Inventory* given that the user has the right role to do so. Roles of users can be managed in the *StudentManager* by any user who is member of the admin group. For each *Item* it



can be defined, which roles are necessary to pick it up and if it can be picked up only once or by various users.

Second, the *Quiz* module was presented. It is intended to deepen the knowledge of the students by querying it at the end of the learning round-trip. Students have to answer a set of multiple-choice questions pre-defined by the teacher and only if they got everything correct they are teleported to a new location, which could be for instance the next level. To monitor the learning progress, a log file with the given answers of the students is written to the server and can be retrieved by the teacher.

The last module introduced was the *Itemboard*. As an extension of the OWL *Whiteboard* module it functions similar to a whiteboard in the real world. Sketches can be made and short text fragments can be written onto it. Of course, it can be used collaboratively. In addition to that, on the *Itemboard* also those *Items* one has already found can be placed. The aim of this module is to facilitate the exchange of knowledge with those users who were not able to pick up certain *Items* in order for everybody to be able to do the *Quiz*.

The three modules were developed to form a continuous, collaborative, and exciting learning experience. To demonstrate that, an exemplary virtual world where all of these modules come together was designed. The next chapter outlines the design of this world and describes the various tasks the students can perform in it.



## 5. Showcase Scenario

To demonstrate the functionality of the implemented learning tools and to provide a realistic environment for an evaluation an exemplary learning scenario, the Virtual Egypt World (VEW) has been created. The idea was to use an Egypt setting since this is usually a topic which can be made interesting for both, younger and older students.

### 5.1. Pedagogical Background

In the VEW, students are supposed to learn about the **Osiris myth** which is a story from ancient Egyptian mythology. It tells about the murder of king Osiris by his brother Seth and the subsequent fight of Osiris's son Horus for the throne<sup>1</sup> (Felske, 2014).

Learning takes place in the form of a scavenger hunt in which students must find all parts of the story. These parts are hidden somewhere in the virtual environment so students have to search for them and bring them all together in order to learn the whole story. The exploratory character as well as the 3D environment and the chance to interact with different objects and avatars should increase the motivation of the students to learn. Furthermore, as it is typical for exploratory learning approaches, the students are able to control their own learning speed (L. P. Rieber, 1996).

Nonetheless, group work is important. The whole learning experience builds on the fact that students collaborate and share the information they found. To enforce this, the concept of roles is introduced. Teachers have the possibility to assign a role to each of their students which restricts the set of possible information texts the student can collect. So in order to learn everything about the Osiris myth the students have to work together. One could say they are interdependent due to the limited information resource.

For the VEW prototype, the following four roles were implemented:

- **Adventurer**  
Is able to pick up information about dangerous or hard to reach objects.

---

<sup>1</sup>For more information on the Osiris myth see <http://www.egyptianmyths.net/mythisis.htm>

## 5. Showcase Scenario

- **Scientist**  
Is able to understand complicated issues and solve riddles.
- **Priest**  
Is able to pick up information about religious items.
- **Historian**  
Is able to pick up information about objects with an interesting history.

The goal is that in the end every student knows everything about the myth. It would be desirable that one student gives away a piece of information he owns in exchange for some information he was not able to obtain. In the process of sharing it is expected that the students further engage with the story and therefore deepen their knowledge. In the unlikely case that a student does not know who to contact regarding a specific piece of information he is given a hint with whom he must talk.

To meet the requirement of assessability and to provide a means for the teacher to grade his students, a knowledge test at the end of the learning experience is included. This test has to be done by each student individually and therefore requires the knowledge of the complete story. That way it can be ensured that every student of the group has mastered the subject and no one is able to rely on his group members for successful completion of the course.

## 5.2. Structure and Appearance

This section describes how the VEW was created. The 3D models used for building the world all came from the Google 3D Warehouse (Google Warehouse Website, 2015). The models there are free to download, use, modify, and distribute, even for commercial purposes.

At the heart of the learning world is a pyramid the students can explore. It consists of three levels, which are built like a maze. Each level is a bit smaller than the level before, due to the pyramid-shaped structure. Each upper level is also only accessible via a ramp hidden in the level below it. In order to reach the top level, the students have to find their way through the mazes on the first two floors.

The parts of the Osiris myth mentioned earlier are hidden inside five statues of Egyptian deities which were distributed in the pyramid. Therefore, 3D models from the Google Warehouse were taken and provided with the *Itemize!* capability, so that the respective information texts could be attached to the statues. Figures 5.1 and 5.2 show two of them. Of course they were somewhat hidden around bends, so that they are not easy for students to spot.

## 5.2. Structure and Appearance



Figure 5.1.: The statue of Isis hidden somewhere in the pyramid maze



Figure 5.2.: The statue of Anubis hidden even better

The most important statue - the statue of Horus - can be found on the top layer of the pyramid. There, a room was designed in the style of a grave chamber, as can be seen in figure 5.3. The statue of Horus gives away the most valuable piece of information, so if students reach this chamber they have a good chance of successfully completing the *Quiz* at the end of the learning experience.



Figure 5.3.: The statue of Horus in the chamber at the top level of the pyramid

Besides the five Egyptian statues also three other objects with information texts attached to them were hidden somewhere in the world. They also give information to the students but not in connection to the Osiris myth. In the *Quiz* at the end all of the questions refer to the ancient Egyptian story but some of the answers contain information from the three additional objects so students really have to read carefully.

Outside the pyramid, students can find the "archaeologists' tent" with the *Itemboard* where they are supposed to meet after exploring the pyramid and exchange the

## 5. Showcase Scenario

information they found. The *Itemboard* can be seen in figure 5.4. Besides the *Itemboard* also more conventional tools like the text or the voice chat can be used to share information.



Figure 5.4.: The "archaeologists' tent" containing the *Itemboard*

After making sure everybody knows the whole story, the students can move on to the *Quiz* component which comes in form of an all-terrain vehicle as can be seen in figure 5.5. The truck was chosen in analogy to "driving away" after having learned everything about the ancient Egyptian myth.

An additional feature which has been integrated to help students by giving them hints about what they have to do is a so-called *Chatbot* as described in Riedmann (2014). It comes in form of a nomad figure and is located at the entrance of the pyramid. The chat is managed, which means it is guided by a script. At every stage of the conversation the user has different options of what he can say. He can choose between them by pressing the according number on the keyboard. If the right options are chosen, the nomad tells about the Egyptian statues and the structure of the maze. Figure 5.6 shows the *Chatbot*. The students can abort the chat at any time and can restart it by clicking on the figure again.

If the students manage to answer all the questions of the *Quiz* correctly, they get teleported to a so-called "Congratulations World". In the current prototype this is only an empty world with a note congratulating the students and saying that they have successfully completed the learning experience. The "Congratulations World" is shown in figure 5.7. In practice, this could be another level with a more difficult maze and another piece of Egyptian history to learn about.

## 5.2. Structure and Appearance

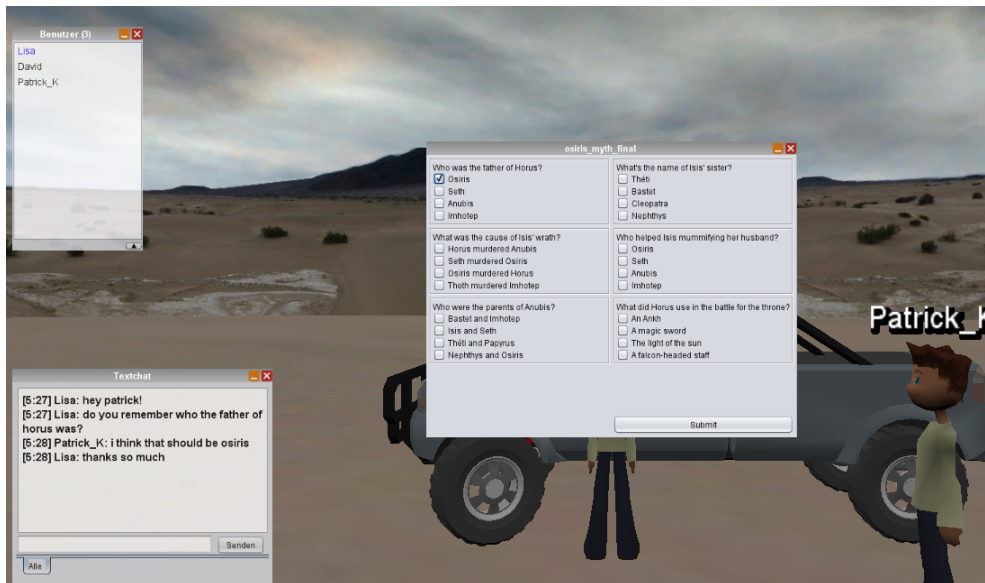


Figure 5.5.: The truck where the *Quiz* panel pops up

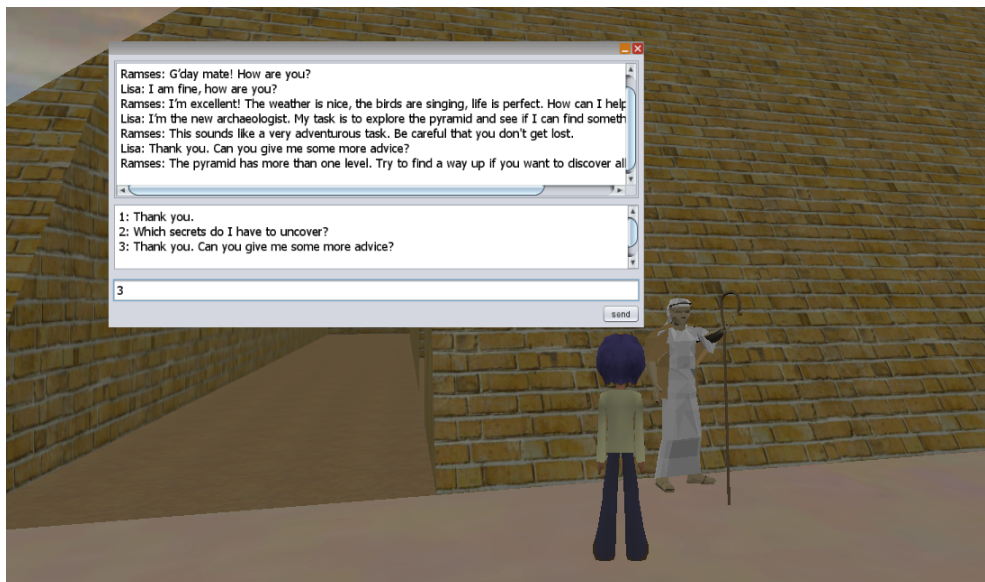


Figure 5.6.: *Chatbot* giving hints at the entrance of the pyramid

## 5. Showcase Scenario



Figure 5.7.: Three students having reached the "Congratulations World"

### 5.3. Process

The diagram of how a learning round-trip in the VEW can proceed can be seen in figure 5.8. The first prerequisite for starting the learning round-trip is that all students are logged into the world the teacher has created for them. As soon as they have managed to do this the teacher will see their avatars with their names above their heads appear out of nowhere. The world is designed for groups of 3-5 students but there can be more, depending on the scope of the learning material.

There can be a phase where students get to know each other, get familiar with the controls and customize their avatars. However, the next important step for the scavenger hunt is that the teacher assigns each of the students a role so that they can start searching for things and picking them up. As mentioned before, the teacher has four predefined roles to choose from. Each student can have only one role but in case there are more than four students it is possible to assign the same role to two students.

After that the students are free to explore the pyramid and find their way to the top level, finding all statues and other information objects that have been prepared. Before that they can talk to the Chatbot at the entrance of the pyramid if they want to, respectively recognize that there is some chat function for them there. The Chatbot gives some additional information about the structure of the pyramid and how to find a way through the maze.

If possible, the students should collect all information texts and share them in the end. Therefore, a meeting place at the archaeologists tent with the *Itemboard* has been designed. The board facilitates the exchange of information since found



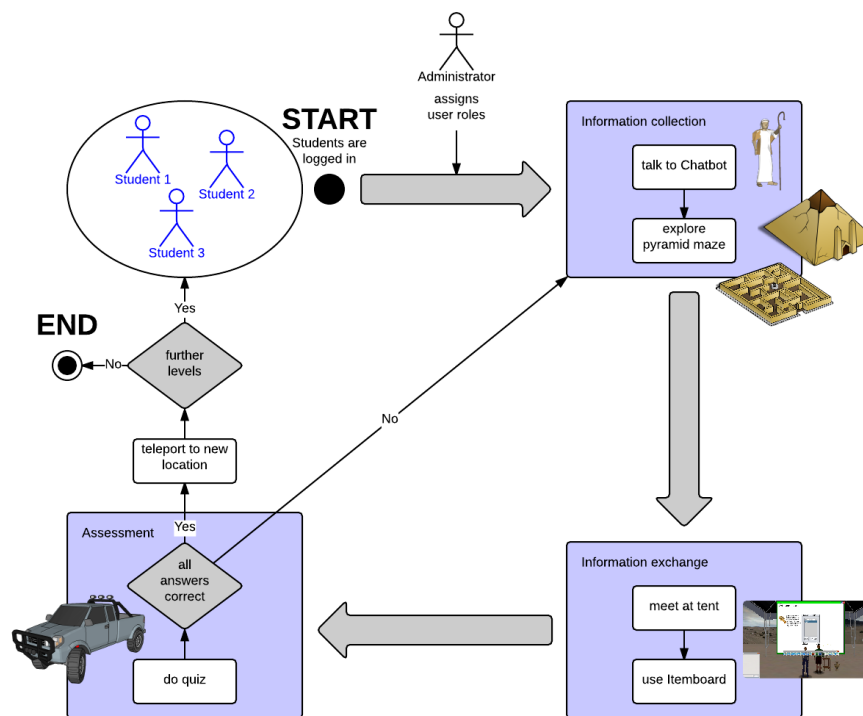


Figure 5.8.: Process of the learning round-trip in the VEW

objects can be pinned up directly.

The last station for the students is the *Quiz truck*, which they can activate upon going near it. The quiz can be repeated as often as desired. Communication between students is possible while answering the questions. Nevertheless, each student must solve his quiz individually. Only when all answers are correct, the system teleports the user to a new location specified by the teacher upon creating the quiz. In case this is a new level, a new learning round-trip starts as soon as enough students are logged in.

## 5.4. Summary

This chapter has described the design of a showcase learning scenario. It was decided to remodel an Egypt environment, using 3D models from the Google 3D Warehouse. The learning environment consists of a maze pyramid, where five statues of Egyptian gods are hidden. Each statue tells a part of the Osiris myth, students should learn about. In addition to the statues, three other objects which give information to the students are hidden inside and outside the pyramid. In front of the pyramid, there is a meeting point in form of an "archaeologists tent", where an *Itemboard* is provided for discussion. When the students are finished, they can leave the world with the all-terrain vehicle which pops up a *Quiz* teleporting them

## 5. Showcase Scenario

into the "Congratulations World" as soon as they answer all questions correctly. Lastly, there is a Chatbot located at the entrance of the pyramid. Students can talk to him and receive valuable information.

The created 3D environment should give an example of what types of learning scenarios are possible with the implemented tools. It is, of course only a first prototype and many more applications than the Egypt setting are thinkable.

In the next step we have evaluated the VEW scenario with a user study in which we have collected data on three research focuses with the help of a set of standardized questionnaires. The next chapter will discuss this evaluation and its results.

## 6. Evaluation

To evaluate the implemented learning tools and how they work together, a study with test users was conducted. The users were divided into groups and asked to perform a set of tasks together in the Virtual Egypt World described in the showcase scenario. Given the broad scope of functionalities of the learning tools, it was not possible to test everything in one test session, which was scheduled for approximately 70-80 minutes. For that reason, only the students' point of view was tested, not taking into account the features the implemented tools provide for teachers and lecturers, such as creating new information objects and quizzes.

The remainder of this section will discuss the methodology and results of this study. It is to say that, especially when it comes to immersion and usability, the outcome of this study is almost certainly influenced by the overall immersion and usability of Open Wonderland itself, since the tools build on OWL features and the whole learning experience includes other OWL functions as well.

### 6.1. Research focus

For the development of the learning tools the goal was to design them in a way that not only enables effective reception of knowledge, but also leads to more intrinsic motivation, personal commitment and a sense of wanting to solve the task at hand. Therefore, three research focuses were set for this evaluation:

- The motivation of students measured by factors such as loss of self-consciousness, or transformation of time
- The immersion the students experience while they are working on tasks in the Virtual Egypt World
- The usability of the developed learning tools and if the students would use them in practice

To collect data on these topics, a set of standardized questions were accumulated and presented to the users as part of the post-questionnaire after they used the system and completed the learning tasks. For all three research focuses, the users' answers were evaluated to result in a score ranging from 0 to 100, reflecting the degree of motivation, immersion, and usability of the system.

Gathering information about the learning process, which provides information about how much knowledge the students actually gained, was not part of this

## 6. Evaluation

evaluation and will be done in a follow-up study.

### 6.2. Methodology

The study was conducted with groups of students at university level. Group sizes ranged from two to four in order to simulate a typical group work and to make use of the various tools for communication and collaboration OWL provides. Since it was not possible to use the built-in voice chat, due to firewall restrictions, a group call with the internet telecommunication software Skype<sup>1</sup> was set up to simulate the built-in OWL tool.

Therefore, of course, it was necessary to use separate rooms for the test users to prevent them talking to each other directly and circumventing the communication software that way. In each of the test rooms a PC or laptop pre-setup with OWL and Skype was provided to accelerate the test procedure. Additionally, to improve the call quality, every student was asked to bring a headset or earphones.

In one of the test rooms, the main room, where everyone would be meeting at the beginning and at the end of the test session to fill out the questionnaires a camera set-up was installed. This was done so that the learning round-trip of one test user could be filmed and the expression of the user while facing certain difficulties or unexpected situations could be captured.

### 6.3. Participants

Overall, 18 students took part in the evaluation, their ages ranging from 20 to 31 years (Mean (M) 25.5, Standard Deviation (SD) 3.15). 13 of the students were male. The majority of the participants were students at Graz University of Technology and their fields of study therefore mostly technical (Computer Science, Software Development), although two of them had a pedagogical background as they were studying to become teachers. In terms of profession, eight of them already worked part-time as software engineers and one was a PhD student. They received no payment for participating in the study but were promised to get some debriefing information about the results.

In the following section the terms "student", "test user" and "test participant" will be used synonymously.

---

<sup>1</sup><http://www.skype.com>

## 6.4. Procedure

Six test sessions were carried out over the course of one week. In each test session one group of users was invited to try out the learning environment together. Four of the user groups consisted of three students each. There was also one test group with four students and one group of two.

First, the participants were greeted and informed about the purpose, duration and structure of the study. After that, they were asked to fill out a consent form in which they agreed that audio and video recordings of them would be made and used for teaching and research purposes. Having completed the consent form, the students were given the pre-questionnaire where they had to fill in demographic data and give information about their experience with virtual worlds and computers in general. After a short training to show them the basics of navigation and communication in OWL, they were taken to their test rooms to begin the learning tasks.

Each task was given to the students only after they have completed the previous one. Furthermore, all students received their tasks at the same time so that they could start working on the new task together.

### Tasks

**Task 1** The first task was simply to wander around and explore the place around the pyramid. This was also to get used to controlling the avatar. The students were asked not to enter the pyramid yet.

**Task 2** The second task consisted of introducing themselves using the group chat, as well as the built-in text chat of OWL.

**Task 3** In the third task, the students were told to examine their user role, which had been given to them in the meantime by the administrator. According to their role they would be able to pick up certain items or not.

**Task 4** The test participants were allowed to enter the pyramid at last in the fourth task. The task description told them to go searching for the 'mysterious statues' which held some information for them. It was also explained to them how they could pick up the information and look it up in their *Inventory*. The fourth task needed the most time for completion. Although there was no time limit for completing the tasks, in some cases the fifth task was given to the users even though they had not found all the statues yet in order to save time and keep the students from losing attention.

## 6. Evaluation

**Task 5** For the fifth task, the test participants had to meet outside the pyramid again to exchange the information they had found. They were also encouraged to use the *Itemboard* for that purpose.

**Task 6** Finally, the sixth task told them to go to the object provided with the *Quiz* capability to answer some questions about the story they were supposed to have learned. After answering all the questions correctly, they were teleported to the "Congratulations World" where they were told that they had successfully completed the learning experience.

During the test the participants were monitored by the administrator all the time. As soon as they reached the "Congratulations World" they were informed that they could leave their respective test rooms and meet again in the main room with the camera. There the students were interviewed about their experiences. Transcripts of the interviews can be found in the test sessions appendix on the DVD.

To get detailed information on motivation, immersion and usability, the test participants were asked to fill out a post-questionnaire after the interviews. This was the last part of the test session, after which the participants were thanked for their participation and showed out.

### 6.5. Materials

The materials described in this section, along with the transcript of the test user interviews are included in the appendix on the DVD.

#### Pre-Questionnaire

In the pre-questionnaire data such as age, sex, occupation, as well as experience with the computer and Egyptology were collected. It consisted of 25 questions about the user and his background knowledge, especially experience with computer games and virtual worlds. Most questions used a Likert scale between 1 (strongly disagree) and 5 (strongly agree) but there were also a few text questions about the students' expectations of a virtual learning world included.

#### Task List

A list with detailed task descriptions was prepared for every test user. Each task description was written on a separate piece of paper, to give it to the users separately so that they would not be overwhelmed by the amount of information.

## Post-Questionnaire

The post-questionnaire was composed of a general part, gathering the users' personal opinion on the learning round-trip, as well as on the different tools. Further questions concentrated on design and communication. Lastly, there was a set of standardized questions about motivation, immersion, and usability, which got evaluated to get a score from 0 to 100 in these areas. Altogether, the post-questionnaire contained 80 questions of which a large amount could be answered by selecting a value between 1 and 5 on a Likert scale.

### Motivation

The questions about motivation were taken from Jackson and Marsh (1996). They try to measure motivation by asking the test participant about *the nine dimensions of flow*, namely:

- challenge-skill balance
- action-awareness merging
- clear goals
- unambiguous feedback
- concentration on task at hand
- sense of control
- loss of self-consciousness
- transformation of time
- autotelic experience

Together, they have found, these nine dimensions give a good measurement of the amount of flow the test user experiences while completing the tasks. For the evaluation of the Virtual Egypt World a subset of 20 questions from the original questionnaire they suggest was taken. For the sake of simplicity, it has been refrained from assigning an individual weight to each of the questions but rather allowing them to equally contribute to the result.

### Immersion

In terms of immersion the questions came from Jennett et al. (2008). In this paper, a questionnaire consisting of 32 items which aim at gathering information about immersion was developed and tested in three different experiments. Again, for our post-questionnaire the amount of questions was reduced to 21 at first and later to 20, omitting another question about the graphics. This last question was left out on the one hand to get the same amount of questions as for motivation and on the other hand because it was obviously not clear to the test participants that they should rate the graphical appearance of the learning tools and not OWL itself.

## 6. Evaluation

### Usability

Lastly, the usability was assessed using the standardized System Usability Scale (SUS). The SUS is a ten-item questionnaire, developed by Brooke (1996). It also uses a Likert scale where users fill in how much they agree with a statement on a scale from 1 (= strongly disagree) to 5 (= strongly agree). At the end, with the ratings of all users, a score between 0 and 100 can be calculated representing the usability of the system. The SUS is somewhat subjective but nonetheless widely used in software development (Zviran, Glezer, & Avni, 2006).

To be able to also compare the motivation and the immersion to other systems, a score from 0 to 100 was calculated for these two areas as well. As for the SUS, all questions could be answered by choosing a value on a Likert scale from 1 (= strongly disagree) to 5 (= strongly agree). Both, for the SUS and for motivation and immersion, the users' answers were converted into values from 0 to 4. Negatively phrased questions were of course inverted. Then, the values were added for each participant and multiplied by a factor to get a result ranging from 0 to 100. For the SUS, this factor was 2.5. Because there were 20 questions for motivation and immersion instead of 10, the factor 1.25 was used for these questionnaires. Lastly, the mean value and standard deviation was calculated from the individual sums of the participants.

## 6.6. Results

The study pointed out some interesting results. It is noticeable that the opinions on usability, graphical appearance, and the usefulness for learning purposes diverged widely. Whereas some users loved the whole experience and had almost nothing to criticize, some other users were not that convinced and complained of clumsy controls, old-fashioned graphics and too little interaction and collaboration. Yet others stated in the interview that they did not feel engaged and that completing the learning tasks was not rewarding for them but the video recording showed them in intense discussions with the other test participants while hunting for the information and cheering at the end when reaching the "Congratulations World".

The remainder of this section will discuss the results of the evaluation in detail, attempting to draw a conclusion and give a prognosis in the final subsection.

### Pre-Questionnaire

The following subsections sum up the results of the questions of the pre-questionnaire regarding the users' experience with the computer and Egyptology, as well as their expectations on the Virtual Egypt World.



## Experience

Each of the 18 test users stated that they have no special knowledge in the field of Egyptology. However, 3 users reported to have heard of the Osiris myth, although none of them could remember the details.

In terms of computer usage 10 test participants rated themselves as experts (5 on the Likert scale), 6 stated they have good or medium computer skills (4 or 3 on the Likert scale) and only 2 thought of their skills as below average (2 on the Likert scale), resulting in a mean value of 4.28 (SD 1.02), as one can see in figure 6.1. Internet usage was rated even higher with a mean of 4.39 (SD 0.78). Here, nobody chose a value less than 3 on the Likert scale.

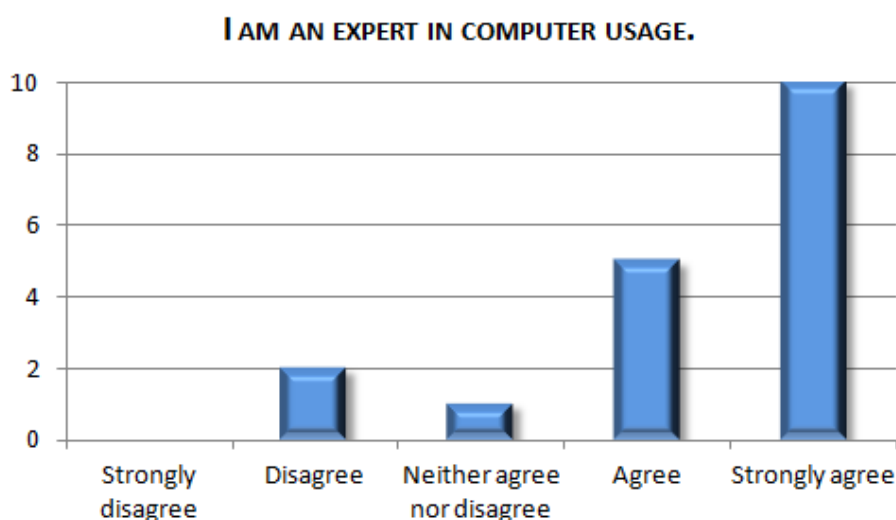


Figure 6.1.: Diagram showing the expertise of the test users in terms of computer usage

Although many users seemed to be experts in computer usage, only 7 thought their expertise in the usage of video games was very good (M 3.83, SD 1.1) and even fewer (2) felt comfortable with massively multiplayer online games (M 2.06, SD 1.26). So it is not surprising that no one claimed to be an expert in virtual worlds and only 2 evaluated their skills as above average (M 1.94, SD 1). The graph for this can be seen in figure 6.2. If the test participants stated to have used virtual worlds before it was largely for gaming. No one had any experience in the field of simulated experiences or training before. Also, the majority (13) has never used a virtual world to work collaboratively.

## Expectations

In the pre-questionnaire participants were asked where they saw advantages of virtual worlds used for learning. Typical statements were *"geographically dispersed*

## 6. Evaluation

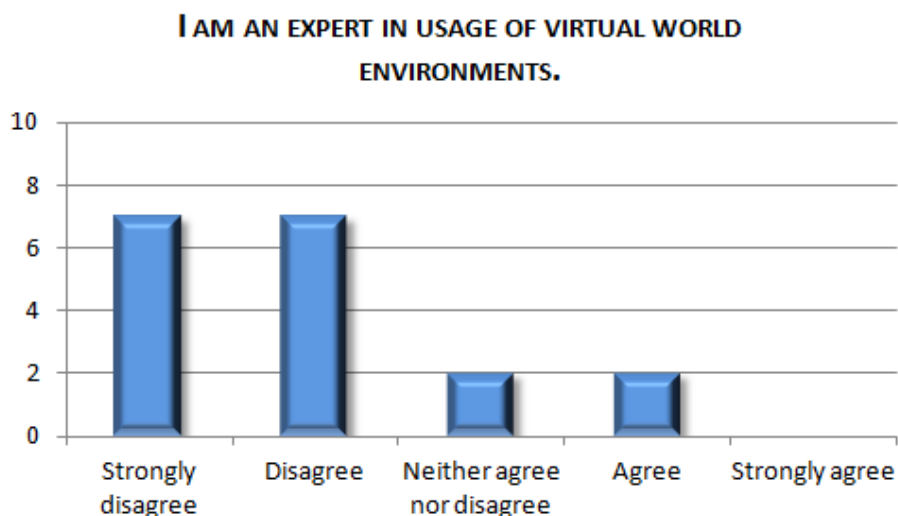


Figure 6.2.: Diagram showing the expertise of the test users in terms of virtual worlds

*people can learn together", "play-like environment", "you can create difficult situations as often as you want", "people/children are able to learn more individually", or "possibility to show and simulate things, safe working environment".*

When asked about disadvantages some mentioned distraction because of the play-like environment, loss of concentration, impersonal experiences, and time inefficiency. Surprisingly, a large amount of the test participants, although students of Computer Science themselves, stated that it would be bad for children to spend too much time in front of the computer at a young age.

Lastly, the students were asked what learning objectives and activities they would like to use in a virtual world. The answers on that question were somewhat divided, since many students did not seem to have concrete ideas of what virtual environments could do for them. *"a quest I can only achieve when increasing my knowledge/answering questions", "instant visual and interactive representation", and "close to reality, so that eventually learned skills can be applied in real life"* were three of the answers given. Others were *"learn about money and economics", "mathematics, grammar, languages", and "learning history"*. One can see that, lacking experience in that field, the students were having completely different expectations of learning in a virtual world.

### Test sessions

The first thing that could be observed was that the students had almost no technical difficulties once OWL was started. There was only one case where OWL had to be restarted. More difficulties were experienced by the test participants when it came

to controlling the avatar. Particularly the users who were not so experienced with computer games had problems navigating in the narrow corridors of the pyramid. The more experienced computer users seemed to have no problems but they also complained at the end that the controls were clumsy and not what they were used to from modern computer games.

Furthermore, it could be commonly noticed that once the test participants had the voice chat available, almost nobody used the built-in text chat any more. It has to be said however, that using Skype instead of the OWL voice-chat function might have led to a slightly different degree of acceptance among the users, since Skype ensures a constant communication and it is not necessary to press the space button as in OWL when you want to say something.

The *Itemboard* led to some controversy among the students. While some of them loved it and thought it very important for the learning experience, others hated it and found it completely unnecessary. However, what they all agreed upon was that the *Itemboard* module certainly needed improvements in terms of usability. Because although the module proved to be 100% bug-free in the learning experience, most test participants noted in the feedback questionnaire at the end that it was 'buggy' which might be due to the fact that some things did not work as they expected.

Some test users also complained about the limited space the *Itemboard* provided. There was just not enough room for all the items and if the students pinned up too many information texts the board quickly became cluttered. Figure 6.3 shows a situation with an overfull *Itemboard* during one of the test sessions.



Figure 6.3.: Limited space on *Itemboard*

Luckily, this problem can be solved easily by inserting another *Itemboard* for the next version of the learning world.

## 6. Evaluation

Another thing that led to different opinions among the students was the design of the pyramid itself. The fact that it was built like a maze seemed to be challenging and fun for some of the test participants, while others found it annoying and distracting. It even went that far that some test users stated it was the best feature of the 'game' while others thought it to be the worst.

One last discrepancy that has already been mentioned in the introduction and that should be included in this chapter was that some students who stated that they did not like the learning experience at all apparently had a lot of fun completing the learning tasks and doing the *Quiz*. To reconstruct this the video recordings of the test sessions have been included in the test sessions appendix on the DVD.

Lastly, as it is perhaps the case with any evaluation, some test participants did not seem to understand certain questions of the post-questionnaire or were too lazy to answer them properly. So it happened that the same question, once phrased positively and once negatively, was answered both times with either '(strongly) agree' or '(strongly) disagree' which is obviously not correct. But it is to be expected that the relatively large amount of test users compensates a small number of wrongly answered questions.

## Interviews

In the interviews, two points of criticism emerged clearly. On the one hand the test participants rated the experience as *"more interesting than a lecture about the topic or to learn the whole thing only theoretically"* and could imagine an application in elementary and secondary school. On the other hand some of them wished for more interactivity: *"The problem in the pyramid is actually a typical 'gaming stuff'. You simply approach the different objects and you know you have to pick them up. But apart from that you have nothing further to do with the objects. That was not enough for me."* Another student stated: *"You would have to force the student to occupy himself with the content of the information text after picking it up."*

Other suggestions such as animating historical scenes and let them take place in front of the user were also made but it was quickly realized that doing so would be a tremendous effort and would require the teacher to have programming skills.

The other thing participants criticized was, of course, the graphics. Especially the fact that you could look behind a wall if you place the avatar at the right angle in front of it bothered many. One test user mentioned: *"The problem is that you are used to computer games as they look nowadays and if you then have to work with something like that, you might lose the joy in it."*

Altogether, approximately a third of all test participants stated that they would use the learning environment in practice and another third could imagine its application in schools.

## Post-Questionnaire

In the post-questionnaire, participants were asked about their first impression and what they liked or disliked about the learning round-trip. While some test users were positively surprised by the whole experience (*"I like the idea of finding information. The statues were interesting and the learning goal has been fulfilled! Fun learning! Moreover, the constant communication was great!"*) others were a bit more sceptical (*"nice idea, but with technical issues that interfere with the experience"*) and again, many complained about the graphics (*"graphics looked a bit stale, not all that realistic or inviting to exploration"*).

What most of them liked was that the pyramid was built like a maze inside and that collaboration was necessary to solve the tasks. *"The riddle was nice, and I liked that teamwork was required to do the final quiz."* Two test users also positively mentioned the person in front of the pyramid that gave them hints and one student found the world conveyed a *"sense of adventure"*.

The points that the test participants disliked the most were the graphics and the cumbersome controls. *"The controls were not good, especially inside the pyramid. Everytime I turned I could only see a wall."* Moreover, the controls of the *Itemboard* did not seem to be intuitive for the users. But although they did not like the handling of it, more than half of the test users thought the *Itemboard* was 'important' or 'very important' for the learning experience (M 3.39, SD 1.2). The *Quiz*, however, was considered even more important with a mean of 3.94 (SD 1.26).

In terms of collaboration OWL was commonly found to make discussions with other users easy (M 4.11, SD 1.23). The users' opinions on that can be seen in figure 6.4. Also, the students seemed to really have enjoyed the experience and had fun going through the learning tasks (M 4.0, SD 0.9). *"I actually found it very funny. It had something adventurous."* *"It was fun. But you also had to be concentrated. It wasn't just folderol, but rather with focus."*

As expected, the graphical appearance was rated rather badly. Although the test participants had different views about whether a graphically rich learning environment is important (M 2.83, SD 1.29), 4 users complained about the graphics when asked *"What's your first impression of the virtual Egypt world?"*. *"Bad graphics"* or *"Graphics could be improved"* were typical answers given to that question.

Despite the graphics and the other points of criticism mentioned in this section, many of the test participants thought that the learning round-trip was a positive experience which expanded their knowledge. Figure 6.5 shows that only two users "disagreed" with this statement and nobody selected "strongly disagree". Nonetheless, a majority of eight test users chose to answer with "neither agree nor disagree" which is an indicator that there is room for improvement here.

The next subsections discuss the results in terms of motivation, immersion and usability in more detail and give the score that was calculated in these fields.

6. Evaluation

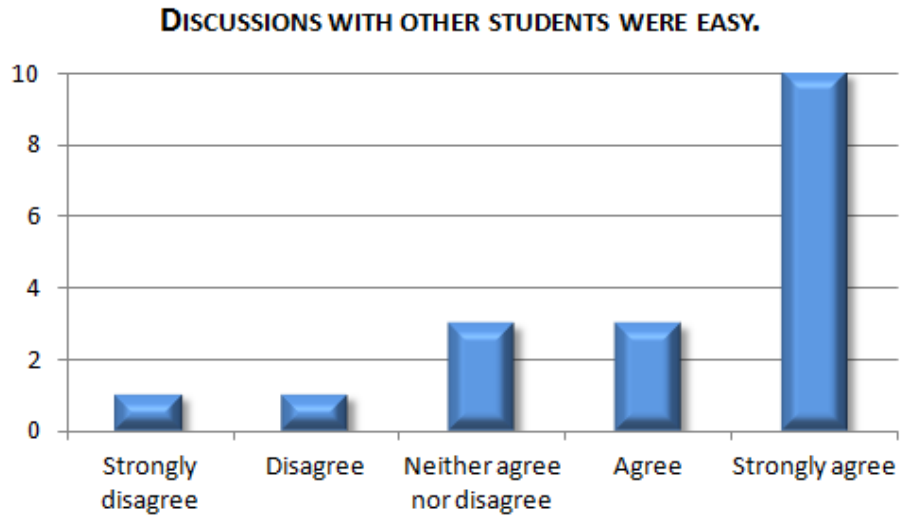


Figure 6.4.: Diagram showing the opinion of the test users on the ease of discussions in the learning world

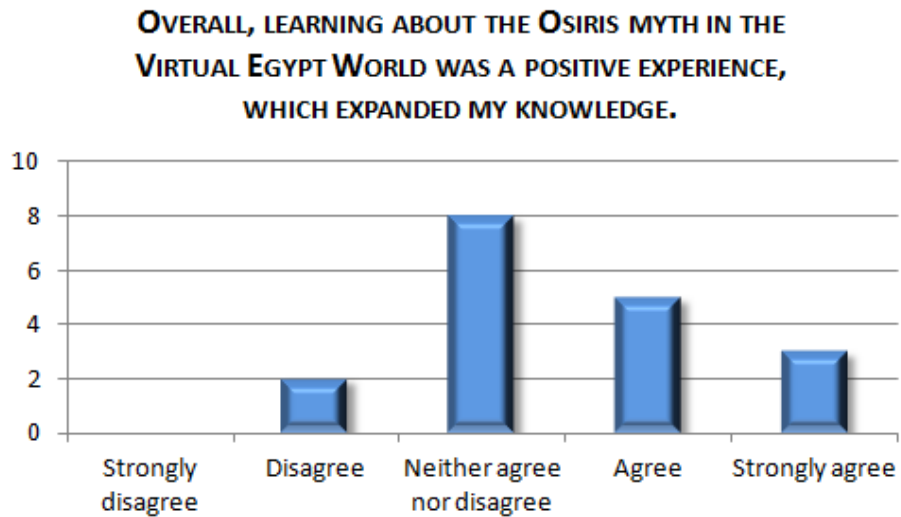


Figure 6.5.: Diagram showing the opinion of the test users on the expansion of their knowledge

## Motivation

On a scale from 0 to 100 the mean score for 'Motivation' was 63.68 with a standard deviation of 13.47 which is a good score but also shows that the test users were torn out of their flow by navigation problems and unintuitive controls. The group dynamics on the other hand may have contributed to an enhanced intrinsic motivation. Due to the fact that some team members were dependent on others when they did not have the right role to pick up some information, everyone felt motivated to do his part in order for the group to collect all necessary information texts. Furthermore, the game-based approach of coloring the names of the different objects in the *Inventory* according to whether they had already been found or not was used to further increase the intrinsic motivation.

Because intrinsic motivation can be linked to immersion (users are typically more likely to experience the feeling of 'flow' when the environment is highly immersive), the results for immersion might have influenced the score of this subsection.

## Immersion

The score for immersion was actually quite low with a mean value of only 50,66 (SD 11,42). It seems that the unrealistic graphics and the cumbersome controls did not allow the feeling of 'really being there' to come up. Figure 6.6 shows the ratings for the statement "*It was as if I could interact with the world of the game as if I was in the real world*". The arithmetic mean of only 1.76 (SD 0.9) might be characteristic for the problems the test users experienced in terms of controls.

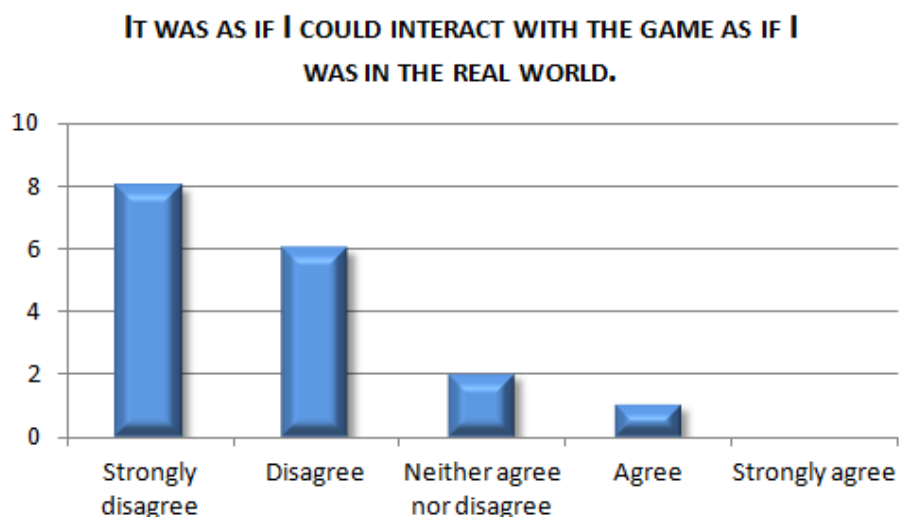


Figure 6.6.: Diagram showing the users' responses to the statement "*It was as if I could interact with the world of the game as if I was in the real world.*"

## 6. Evaluation

Nonetheless, when asked for their personal feeling of immersion the test participants stated '7' on average (on a scale from '1' to '10' where '10' meant 'totally immersed'). 2 users even chose the highest value and only 3 chose less than '6'. It also seemed that users were able to forget about their everyday worries. They rated the statement "Everyday thoughts and concerns were still very much on my mind" with a mean value of 1.82 (SD 0.73). Figure 6.7 shows the individual values for that statement.

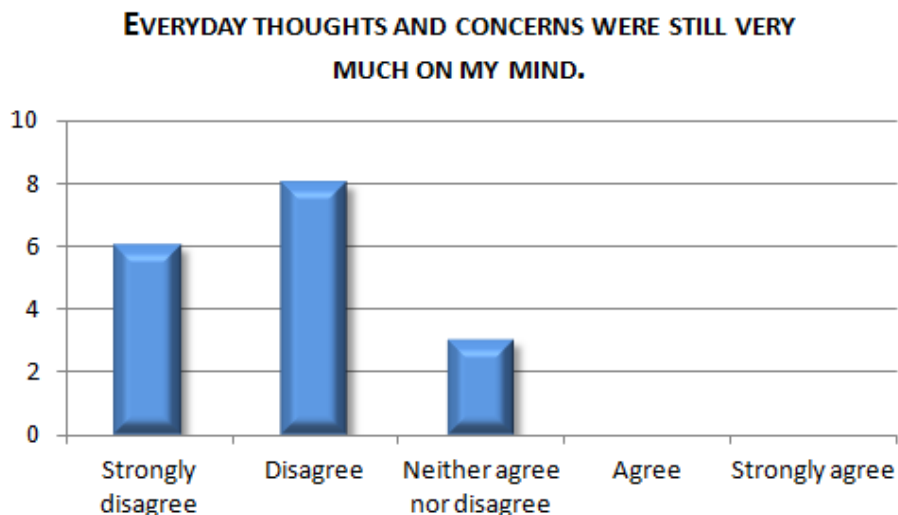


Figure 6.7.: Diagram showing the users' responses to the statement "Everyday thoughts and concerns were still very much on my mind."

One could conclude that the score for immersion might have been higher if more questions about navigation and controlling had been omitted from the original questionnaire from Jennett et al. (2008) making place for questions about involvement.

### Usability

User acceptance in general seemed to be good. The test participants claimed, that they were perfectly able to use the system without the help of a technical person and nobody needed to "learn a lot of things before they could get going with this system" (M 1.33, SD 0.49), as can be seen in figure 6.8.

The mean score of the SUS was 65,28 (SD 17,70). Compared to the Virtual TEAL World described in Pirker et al. (2013), a similar project in OWL, which has a mean SUS score of 73,89 (SD 22,81), the Virtual Egypt World is ranked at an average level. Especially the small amount of time people needed to learn to use the system was rated very positively.



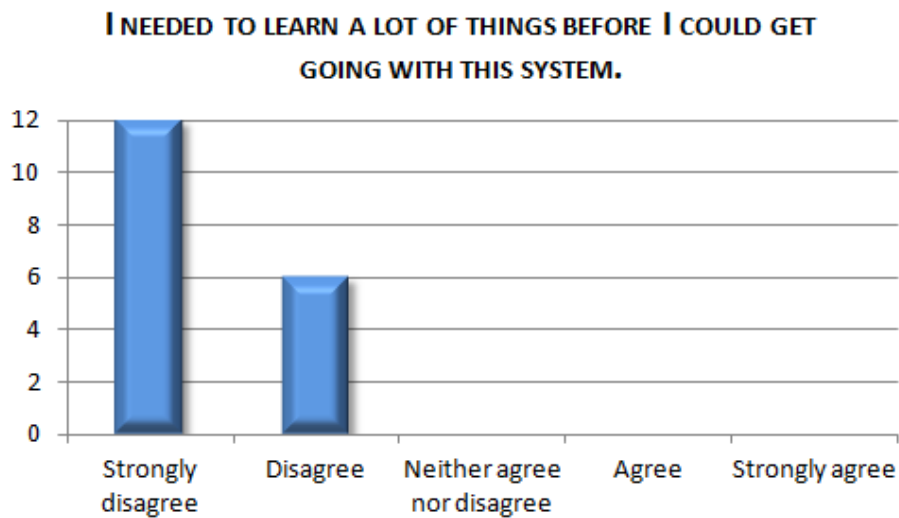


Figure 6.8.: Diagram showing the users' responses to the statement "I needed to learn a lot of things before I could get going with this system."

Figure 6.9 shows the results of the evaluation in terms of motivation, immersion, and usability at a glance.

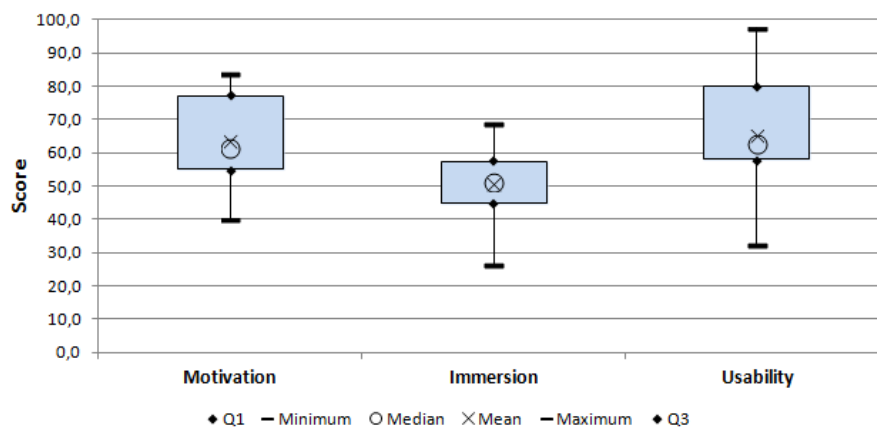


Figure 6.9.: Boxplot of the three research focuses motivation, immersion, and usability

## 6.7. Discussion

Many compromises had to be made while developing the system. On the one hand it should be easy for teachers to create new learning environments and on the other hand learning should be fun and engaging for students. It is clear that more interactivity, animation, and activities tailored especially for the current learning

## 6. Evaluation

topic on the student side lead to more work on the teacher side, perhaps even requiring programming skills. The problem there is satisfying students used to the look and feel of current computer games.

In the current version of the system some users stated that they did not feel motivated to read the texts in the first place but rather would have liked an additional activity after picking up the information text which forced them to deal with content of the text. A few also said they would have liked it if the historical stories the statues tell about had taken place before their eyes as some kind of graphical animation. Apart from exceeding the skills of most of the teachers, animating such a scene and displaying it is not what OWL was thought and built for. In the end, the test participants agreed that to assemble a 'real game' rather than a 'learning game', teaching would have to be more subtle and could not come in such large amounts as it was required for this project.

Altogether, it can be said that most of the test users seemed to enjoy themselves greatly while working on the tasks together. Most of them stated that they had fun and that the concept of exploratory and collaborative learning appealed to them. Besides the old-fashioned graphics and the problems with the controls they mostly agreed that the system would be a valuable addition to learning from textbooks in schools, if nothing else because through the virtual environment the students would have a reference to the topic they are supposed to learn about. This could be especially useful in the subject of history where historical buildings or locations can be recreated. Moreover, searching for the information in groups can bring a sense of adventure to the classroom and may benefit the learning motivation.

## 7. Lessons Learned

In the course of this work, I spent three months at Curtin University in Perth, Western Australia. It was my first long stay abroad and it was a valuable experience for me. Not only did I benefit from the expertise of the Australian researchers but I also learned to work in an international team where I had to discuss, explain and defend my ideas. This chapter will cover some of the aspects I learned from the research and implementation work which I did in Australia, as well as the main findings of the subsequent evaluation which was conducted back home at Graz University of Technology.

### Theory

Literature to traditional learning methods has been around for decades. The key principles have been known for many years and have been researched thoroughly by many different authors. Virtual worlds for education, however, are a relatively new topic that has been studied only in the last 10 years, give or take a few. Therefore, real insights are scarce and many things have not yet been investigated. For this work I had a look at many different approaches to the topic, some of them successful and some of them not. Many innovative teachers and researchers tried to use virtual worlds for learning in their schools and at their universities with different outcomes. Unfortunately, most experiments are described only by words with few formal experiments among them. It seems that a universal guide for the creation of successful virtual worlds has yet to be found and that one can only try and learn from the mistakes others made. Whether a virtual learning world is successful also depends, as with most things in school, strongly of the engagement of the teacher. However, all successful virtual experiences seem to have in common that they really immerse students and provide interactive and collaborative tasks for them. That should in any case be a clue for the creation of learning environments in the future.

### Implementation

Implementing the learning tools in OWL turned out to be very difficult in the beginning. The free virtual world building software is maintained by a small open source community, which is why only few tutorials and instructional guidelines

## 7. Lessons Learned

exist. Furthermore, because the software is under constant development many of these tutorials which are not large in number to begin with are also outdated. However, there is a very good discussion forum where many people are active and where questions are answered quickly. Once I understood the structure and modular architecture of OWL, the implementation of the learning tools was not that difficult anymore. Problems were caused mainly by the file transfers from server to client but could ultimately be solved by using a new content repository library which was hinted by one of the community members in the discussion forum. Emphasis was placed on the exact implementation of the pedagogical approaches identified in the design phase. All modules could be implemented as defined during this phase, although greater value could have been placed on avatar control issues. When implementing something that should be liked by people, the usability and simplicity of the controls should always be kept in mind.

## Evaluation

The user study showed that difficulties with the controls were indeed a reason why people did not like the created environment as much as they probably could have. The second reason were the admittedly not very pretty graphics of OWL. But that was a tradeoff known before when selecting OWL as a virtual world platform. OWL focuses on business applications and puts value on tools for communication and collaboration rather than good-looking graphics. The appearance of the world, however, turned out to be very important for students. They are used to the look and feel of modern computer games and are not easily satisfied with less. The learning experience itself was liked on the whole, although some students wished for more interactivity. In my opinion, the implementation of the learning tools as first proof of concept was successful. Nevertheless, future versions, which may be developed using another virtual world software, provide room for improvement.

The evaluation itself went relatively smoothly. Only minor problems with the brought along laptops of the test participants, especially with the MacBooks occurred. Next time when conducting such a user study, some extra time for preparing the user laptops should be allowed. A further hindrance was the fact that participants could not use the built-in voice-chat, but that was compensated for by the use of Skype. Altogether, thanks to the prepared test environment an easy supervision of the test sessions and a prevention of further technical difficulties could be ensured. As a consequence, the participants were also not distracted by problems caused by the infrastructure, such as firewall issues, but could concentrate entirely on exploring the virtual world.

## 8. Summary and Outlook

The goal of this project was to develop a set of tools for teachers to be able to use virtual worlds for learning in their classes. Interactivity and visualization are known to have a positive effect on the learning process. Virtual worlds offer these opportunities and in addition, they allow students to interact, socialize and work together collaboratively. So far, this technology was not easily accessible to teachers. With the developed tools, teachers should be able to create virtual worlds and turn them into learning environments for their students without any programming skills. Furthermore, the implemented features should benefit students by supporting teaching models like exploratory and game-based learning which is possible only to a limited extent with conventional e-Learning methods. That way, the needs of different types of learners can be met and the learning experience can be improved.

The learning tools were developed for the virtual world platform Open Wonderland (OWL) which was chosen because it is freely available and easy to extend, and because it has a lively community. After an exhaustive research on existing virtual worlds for learning and a consequent requirement analysis, three modules for OWL were created. The *Item* module allows the teacher to add information to specific objects in the virtual world which the students can discover. They can also pick up the information texts and store them in an inventory to look it up later. That way, for instance, a scavenger hunt could be created, where the students have to find as much information as possible. For that purpose, they can work together and share the information using text or voice chat or the second implemented module, the *Itemboard*. In the end, they can do a *Quiz*, also prepared for them by the teacher, to reach a next level. Collaboration and group work is promoted by assigning the students roles which limit them in their ability to pick up certain information objects.

To show a potential application, an environment resembling Ancient Egypt, namely the Virtual Egypt World (VEW) was created with 3D models from the Google Warehouse. A preliminary evaluation was conducted with the help of this world to get first opinions of students about the pedagogical approach. For the evaluation a scavenger hunt as described before was arranged. Participants in groups of two to four people had to hunt for statues of Egyptian deities in a pyramid maze. The statues told them parts of the Osiris myth, an ancient Egyptian story, which the test users had to assemble later on. Of course, there was also a quiz at the end of the scavenger hunt, where the test participants had to prove their knowledge.

## 8. Summary and Outlook

The learning experience was mostly liked by the students, especially the game-based approach and the exploratory character, although the graphical appearance of the world and the limited interactivity was criticized. Some users reported that they were not motivated to read the texts that were appended to the statues. Instead, they would have liked to undergo an additional activity after picking up the information texts that really forced them to deal with the content of the texts. Alternatively, some students argued that a graphical animation instead of the texts would have been beneficial for learning and more convenient for them than reading. Such animations are, however, difficult to create for most teachers and OWL does not make the task easier since this is not what it was built for. Other platforms like Unity<sup>1</sup> are certainly better suitable for this kind of task. In the future, a follow-up project will try to port the features for collaborative learning described in this work to Unity. The test participants also have largely concluded for themselves that for really having fun in the virtual environment, teaching would have to be more subtle and could not come in large amounts. Maybe the follow-up implementation in Unity can consider this argument and focus more on gamification and interactivity.

Other suggestions for improvements concern the implementation and include:

- **Dynamic management of roles**  
Currently, the four roles that can be given to students are predefined and cannot be changed by the teacher. A further development of the *Item* module could include a possibility for content creators to customize these roles and add new ones.
- **More question types**  
The *Quiz* module was implemented in a way that teachers can add any desired questions, although so far it is only possible to add multiple-choice questions. But there are many more question types thinkable, e.g. questions where the answer text can be entered freely.
- **Return all Items**  
In order to avoid that one student collects all information objects and does not share his information with others there should be a mechanism for teachers to take all information texts away from this student so that the other students are again able to pick them up. Furthermore, this can ensure that a student who has already logged off no longer appears in the hint in the inventory which says who has already picked up an Item.
- **Server logging**  
To provide further means of assessment, the server logging functionalities could be extended so that the individual student activities can be tracked more accurately. That way, it would be possible to also assess the social and collaborative behavior of the students.
- **Support for multiple languages**  
Finally, as it is beneficial for international usage, the learning tools should

---

<sup>1</sup><http://unity3d.com/>

be available in a range of languages. In the current version of the prototype the texts of the user interface are written in English. An additional resource file should be provided, so that OWL is able to display them also in other languages.

Although still a bit of effort on the part of the teacher is required to create a virtual learning environment that can be used in classrooms, the developed tools help a lot in this process and it is no longer necessary to hire a programming company. The evaluation showed that the developed tools are user-friendly and the created scenarios are indeed able to bring life and variety to everyday school life. For the future, a graphical improvement would be desirable so that today's generation that has grown up with more and more powerful computer games can enjoy the same visual experience they are used to from their free time also in their education.





# List of Abbreviations

<b>AI</b>	Artificial Intelligence
<b>CRS</b>	Classroom Response System
<b>DOM</b>	Document Object Model
<b>LMS</b>	Learning Management System
<b>MMORG</b>	Massively Multiplayer Online Role play Game
<b>MO</b>	Managed Object
<b>MOOC</b>	Massive Open Online Course
<b>MUVE</b>	Multi-User Virtual Environment
<b>NPC</b>	Non-Player Character
<b>OWL</b>	Open Wonderland
<b>PLE</b>	Personal Learning Environment
<b>RTS</b>	Real Time Strategy
<b>SEE</b>	Shrine Educational Experience
<b>SUS</b>	System Usability Scale
<b>TEAL</b>	Technology-Enabled Active Learning
<b>VEW</b>	Virtual Egypt World
<b>VTW</b>	Virtual TEAL World
<b>WBT</b>	Web-Based Training
<b>WoW</b>	World of Warcraft
<b>XML</b>	Extensible Markup Language



# List of Figures

1.1.	Structure of this thesis . . . . .	3
2.1.	The Learning Pyramid . . . . .	6
2.2.	Hierarchy of the player's needs by Siang and Rao (2003) . . . . .	20
2.3.	Screenshot from the "Awesome Games Done Quick" (AGDQ) 2015 Charity Marathon Event where a player named "kitaru" creates the diagonal Tetris pattern . . . . .	21
2.4.	The experiential learning cycle from Kolb (2014) . . . . .	26
2.5.	3D visualization of the Giza plateau. Image from Giza 3D (2013) . .	29
2.6.	Left: A grave chamber with different objects which have information attached to them; Right: A student immersed in a 3D projection of a tomb on an edge-shaped display . . . . .	29
2.7.	Left: A Plan for a 3D wireframe model to accurately remodel a building; Right: An animated character which was added to a scene to make it more lifelike . . . . .	30
2.8.	Visualization of a religious, Egyptian ceremony with live audience participation. Image from PublicVR (2008) . . . . .	31
2.9.	Exterior view of the Shrine of the Book. Image by Berthold Werner .	32
2.10.	Immersive physics experiments. Image from Pirker (2013) . . . . .	33
2.11.	Student group doing a learning round trip together in the VTW. Image from Pirker (2013) . . . . .	34
3.1.	Collaborative modules that come with the basic installation of OWL	44
3.2.	OWL event where people collaboratively create some software. Screenshot from OWL-Foundation (2014d) . . . . .	45
3.3.	The OWL <i>Textchat</i> HUD component . . . . .	46
3.4.	Example of a conversation with a <i>Chatbot</i> . . . . .	46
3.5.	Three users standing around a <i>Stickynote</i> cell . . . . .	48
3.6.	Schematic process of a learning round-trip . . . . .	51
4.1.	Relationship between the various components of OWL. Diagram from Kaplan and Yankelovich (2011) . . . . .	54
4.2.	Relationship between Cell objects on the different clients and the CellMO object on the server. Diagram taken from Finnigan (2009) .	56
4.3.	The <i>Itemize!</i> capability's property panel . . . . .	59
4.4.	Glittering objects signaling that there is some information to pick up	60

## List of Figures

4.5.	Left: The <i>Item</i> information text of an Egypt statue; Right: Student does not have the right role for picking up the information . . . . .	61
4.6.	<i>Inventory</i> showing a picture and some information of the Egypt goddess Isis . . . . .	62
4.7.	<i>Inventory</i> showing an <i>Item</i> the user cannot pick up plus the hint given	62
4.8.	Button to return an <i>Item</i> left of its image in the <i>Inventory</i> . . . . .	62
4.9.	<i>Student Manager</i> panel where one student has been given the role 'Priest' . . . . .	63
4.10.	Class diagram showing the client side of the <i>Item</i> component - Part 1	64
4.11.	Class diagram showing the client side of the <i>Item</i> component - Part 2	66
4.12.	Class diagram showing the <i>Student Manager</i> part on the client side of the <i>Item</i> component . . . . .	68
4.13.	Class diagram showing the <i>Inventory</i> part on the client side of the <i>Item</i> component . . . . .	69
4.14.	Class diagram showing the package common of the <i>Item</i> component	71
4.15.	Class diagram showing the server package of the <i>Item</i> component .	72
4.16.	The <i>Quiz</i> capability's property panel . . . . .	74
4.17.	Student attempting to solve the <i>Quiz</i> with a second student helping	75
4.18.	Class diagram showing the client side of the <i>Quiz</i> component . . . .	76
4.19.	Student adding one of his <i>Items</i> to the <i>Itemboard</i> . . . . .	79
4.20.	Simplified class diagram of the client side of the <i>Itemboard</i> module - Part 1 . . . . .	80
4.21.	Simplified class diagram of the client side of the <i>Itemboard</i> module - Part 2 . . . . .	81
5.1.	The statue of Isis hidden somewhere in the pyramid maze . . . . .	87
5.2.	The statue of Anubis hidden even better . . . . .	87
5.3.	The statue of Horus in the chamber at the top level of the pyramid .	87
5.4.	The "archaeologists' tent" containing the <i>Itemboard</i> . . . . .	88
5.5.	The truck where the <i>Quiz</i> panel pops up . . . . .	89
5.6.	<i>Chatbot</i> giving hints at the entrance of the pyramid . . . . .	89
5.7.	Three students having reached the "Congratulations World" . . . . .	90
5.8.	Process of the learning round-trip in the VEW . . . . .	91
6.1.	Diagram showing the expertise of the test users in terms of computer usage . . . . .	99
6.2.	Diagram showing the expertise of the test users in terms of virtual worlds . . . . .	100
6.3.	Limited space on <i>Itemboard</i> . . . . .	101
6.4.	Diagram showing the opinion of the test users on the ease of discussions in the learning world . . . . .	104
6.5.	Diagram showing the opinion of the test users on the expansion of their knowledge . . . . .	104

6.6. Diagram showing the users' responses to the statement *"It was as if I could interact with the world of the game as if I was in the real world."* . . . 105

6.7. Diagram showing the users' responses to the statement *"Everyday thoughts and concerns were still very much on my mind."* . . . . . 106

6.8. Diagram showing the users' responses to the statement *"I needed to learn a lot of things before I could get going with this system."* . . . . . 107

6.9. Boxplot of the three research focuses motivation, immersion, and usability . . . . . 107



# List of Listings

3.1. Part of an exemplary script file for the <i>Chatbot</i> module . . . . .	46
4.1. Exemplary <i>Item</i> description xml file . . . . .	60
4.2. Exemplary <i>Quiz</i> xml file . . . . .	74





# Bibliography

- Alfieri, L., Brooks, P. J., Aldrich, N. J., & Tenenbaum, H. R. (2011). Does discovery-based instruction enhance learning? *Journal of Educational Psychology, 103*(1), 1.
- Andreano, J., Liang, K., Kong, L., Hubbard, D., Wiederhold, B. K., & Wiederhold, M. D. (2009). Auditory cues increase the hippocampal response to unimodal virtual reality. *Cyberpsychology & Behavior, 12*(3), 309–313.
- Aronson, E. et al. (1978). *The jigsaw classroom*. Sage.
- Bach, S., Haynes, P., & Lewis Smith, J. (2006). *Online learning and teaching in higher education*. McGraw-Hill International.
- Bailenson, J. N., Yee, N., Blascovich, J., Beall, A. C., Lundblad, N., & Jin, M. (2008). The use of immersive virtual reality in the learning sciences: digital transformations of teachers, students, and social context. *The Journal of the Learning Sciences, 17*(1), 102–141.
- Bainbridge, W. S. (2007). The scientific research potential of virtual worlds. *science, 317*(5837), 472–476.
- Barnbeck, A. K. & Neumann, T. (2005). Lernen durch lehren (ldl) in theory and practice.
- Barnes, S. (2010). Virtual worlds come of age. *Journal For Virtual Worlds Research, 3*(3).
- Bartle, R. A. (2004). *Designing virtual worlds*. New Riders.
- Bean, J. & Weimer, M. (2011). *Engaging ideas: the professor's guide to integrating writing, critical thinking, and active learning in the classroom*. Jossey-Bass higher and adult education series. Wiley. Retrieved from <http://books.google.at/books?id=GUErs81BVEEC>
- Bell, M. W. (2008). Toward a definition of "virtual worlds". *Journal of Virtual Worlds Research, 1*(1).
- Berger, S. (2012, April). *Virtual 3d world for physics experiments in higher education* (Master's thesis, Graz University of Technology).
- Berns, A., Gonzalez-Pardo, A., & Camacho, D. (2013). Game-like language learning in 3-d virtual environments. *Computers & Education, 60*(1), 210–220.
- Bligh, D. A. (1998). *What's the use of lectures?* Intellect books.
- Bonoma, T. V., Tedeschi, J. T., & Helm, B. (1974). Some effects of target cooperation and reciprocated promises on conflict resolution. *Sociometry, 37*(2), 251–261. doi:<http://dx.doi.org/10.2307/2786379>
- Bonwell, C. & Eison, J. (1991). Active learning: creating excitement in the classroom aehe-eric higher education report no. 1.

## Bibliography

- Brooke, J. (1996). Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189, 194.
- Brown, E. & Cairns, P. (2004). A grounded investigation of game immersion. In *Chi '04 extended abstracts on human factors in computing systems* (pp. 1297–1300). CHI EA '04. Vienna, Austria: ACM. doi:10.1145/985921.986048
- Bruner, J. S. (1961). The act of discovery. *Harvard educational review*.
- Catropa, D. (2013). Big (mooc) data. Retrieved February 6, 2015, from <https://www.insidehighered.com/blogs/stratedgy/big-mooc-data>
- Chamberlain, V. M., Kelly, J., & O'Neill, M. (1980). *Creative home economics instruction*. Webster Division, McGraw-Hill.
- Chen, C.-H., Yang, J.-C., Shen, S., & Jeng, M.-C. (2007). A desktop virtual reality earth motion system in astronomy education. *Educational Technology & Society*, 10(3), 289–304.
- Chen, J. F., Warden, C. A., Wen-Shung Tai, D., Chen, F.-S., & Chao, C.-Y. (2011). Level of abstraction and feelings of presence in virtual space: business english negotiation in open wonderland. *Computers & Education*, 57(3), 2126–2134.
- Child, D. (1981). Psychology and the teacher.
- Chittaro, L. & Ranon, R. (2007). Web3d technologies in learning, education and training: motivations, issues, opportunities. *Computers & Education*, 49(1), 3–18.
- CIRTL Network. (2013). Lecturing: advantages and disadvantages of the traditional lecture method. Retrieved February 3, 2015, from <http://www.cirtl.net/node/2570>
- Colman, D. (2013). Mooc interrupted: top 10 reasons our readers didn't finish a massive open online course. Retrieved February 6, 2015, from [http://www.openculture.com/2013/04/10\\_reasons\\_you\\_didnt\\_complete\\_a\\_mooc.html](http://www.openculture.com/2013/04/10_reasons_you_didnt_complete_a_mooc.html)
- Corti, K. (2006). Games-based learning; a serious business application. *Informe de PixelLearning*, 34(6), 1–20.
- Dalgarno, B. & Lee, M. J. W. (2010). What are the learning affordances of 3-d virtual environments? *British Journal of Educational Technology*, 41(1), 10–32. doi:10.1111/j.1467-8535.2009.01038.x
- Davies, A., Fidler, D., & Gorbis, M. (2011). Future work skills 2020.
- De Freitas, S. (2008). Serious virtual worlds. *A scoping guide*. JISC e-Learning Programme, The Joint Information Systems Committee (JISC), UK.
- De Lucia, A., Francese, R., Passero, I., & Tortora, G. (2009). Development and evaluation of a virtual campus on second life: the case of seconddmi. *Computers & Education*, 52(1), 220–233.
- Dickey, M. D. (2005). Three-dimensional virtual worlds and distance learning: two case studies of active worlds as a medium for distance education. *British journal of educational technology*, 36(3), 439–451.
- Dillenbourg, P. (1999). *Collaborative learning: cognitive and computational approaches*. *advances in learning and instruction series*. ERIC.
- Dori, Y. J. & Belcher, J. (2005). How does technology-enabled active learning affect undergraduate students' understanding of electromagnetism concepts? *The Journal of the Learning Sciences*, 14(2), 243–279.

- Dräger, J. (2013). Maßgeschneiderte vorlesungen für alle. Retrieved February 6, 2015, from <http://www.zeit.de/2013/48/onlinekurse-personalisierung-bildung>
- Duncan, I., Miller, A., & Jiang, S. (2012). A taxonomy of virtual worlds usage in education. *British Journal of Educational Technology*, 43(6), 949–964.
- Eley, M. G. & Norton, P. (2004). The structuring of initial descriptions or demonstrations in the teaching of procedures. *International Journal of Mathematical Education in Science and Technology*, 35(6), 843–866.
- Elliott, D. (2005). Early mornings and apprehension: active learning in lectures. *Journal of Hospitality, Leisure, Sport and Tourism Education*, 4(1), 53–58.
- Farooq, U. (2013). Demonstration method of teaching meaning, advantages & disadvantages. Retrieved February 4, 2015, from <http://www.studylecturenotes.com/curriculum-instructions/demonstration-method-of-teaching-meaning-advantages-disadvantages>
- Felske, C. (2014). Der osiris-mythos. Retrieved April 28, 2014, from <http://www.selket.de/von-mythen-und-maerchen/osiris-mythos/>
- Finnigan, D. (2009). Wonderland development part 2: intro to modules. Retrieved December 18, 2013, from [http://openwonderland.googlecode.com/svn/www/wiki/beginner/Wonderland\\_Tutorial\\_02.pdf](http://openwonderland.googlecode.com/svn/www/wiki/beginner/Wonderland_Tutorial_02.pdf)
- Foraker Labs of Boulder, C. (2014, December). Explanation of "exploratory learning" in the glossary of "usability first". Retrieved from <http://www.usabilityfirst.com/glossary/exploratory-learning/>
- Fowler, G. (2013). An early report card on moocs. Retrieved February 6, 2015, from <http://www.wsj.com/articles/SB10001424052702303759604579093400834738972>
- GamesFirst! (2002). America's army preview. Retrieved January 23, 2015, from <http://www.gamesfirst.com/index.php?id=933>
- Gardner, M., Gánem-Gutiérrez, A., Scott, J., Horan, B., & Callaghan, V. (2011). Immersive education spaces using open wonderland from pedagogy through to practice. *Multi-user virtual environments for the classroom: Practical approaches to teaching in virtual worlds*, 190–205.
- Garris, R., Ahlers, R., & Driskell, J. E. (2002). Games, motivation, and learning: a research and practice model. *Simulation & gaming*, 33(4), 441–467.
- Garrison, D. R. (2011). *E-learning in the 21st century: a framework for research and practice*. Taylor & Francis.
- Giza 3D. (2013). Giza 3d website. Retrieved February 16, 2015, from <http://giza3d.3ds.com>
- Gokhale, A. A. (1995). Collaborative learning enhances critical thinking.
- Google Warehouse Website. (2015, December). Sketchup's 3d warehouse web site. Retrieved from <http://sketchup.google.com/3dwarehouse>
- Greitzer, F. L., Kuchar, O. A., & Huston, K. (2007). Cognitive science implications for enhancing training effectiveness in a serious gaming context. *Journal on Educational Resources in Computing (JERIC)*, 7(3), 2.

## Bibliography

- Gütl, C. (2010). The support of virtual 3d worlds for enhancing collaboration in learning settings. *Techniques for fostering collaboration in online learning communities: Theoretical and practical perspectives*, 278–299.
- Gütl, C., Chang, V., Kopeinik, S., & Williams, R. (2009). 3d virtual worlds as a tool for collaborative learning settings in geographically dispersed environments. In *International conference of interactive computer aided learning* (pp. 310–323).
- Gütl, C. & Pirker, J. (2011). Implementation and evaluation of a collaborative learning, training and networking environment for start-up entrepreneurs in virtual 3d worlds. In *Interactive collaborative learning (icl), 2011 14th international conference on* (pp. 58–66). IEEE.
- Holmstrom, H. & Jacobsson, M. (2001). Using models in virtual world design. In *System sciences, 2001. proceedings of the 34th annual hawaii international conference on* (9–pp). IEEE.
- Ibanez, M. B., Rueda, J. J. G., Galin, S., Maroto, D., Morillo, D., & Kloos, C. D. (2011). Design and implementation of a 3d multi-user virtual world for language learning. *Educational Technology & Society*, 14(4), 2–10.
- Ibanez, M., Kloos, C. D., Leony, D., Rueda, J. J. G., & Maroto, D. (2011). Learning a foreign language in a mixed-reality environment. *Internet Computing, IEEE*, 15(6), 44–47.
- Ioannou, A. & Artino Jr., A. R. (2008). Teaching educational psychology: using a classroom response system for summative group assessments and to enhance interactivity. Retrieved February 3, 2015, from [http://www.sp.uconn.edu/~aja05001/comps/documents/CRS-AERA08\\_FINAL.pdf](http://www.sp.uconn.edu/~aja05001/comps/documents/CRS-AERA08_FINAL.pdf)
- Jack Smith IV. (2015). Free online courses are still falling short of their ultimate promise. Retrieved February 6, 2015, from <http://observer.com/2015/01/free-online-courses-are-still-falling-short-of-their-ultimate-promise/>
- Jackson, S. A. & Marsh, H. W. (1996). Development and validation of a scale to measure optimal experience: the flow state scale. *Journal of Sport & Exercise Psychology*, (18), 17–35.
- Jacobson, J. & Gillam, R. (n.d.). The egyptian oracle; recreating egyptian religious ceremony in mixed reality. *unpublished.[Online]*. Retrieved from <http://publicvr.org/publications/JOCCH-7.pdf>
- Jarvis, S., Smith, S., Hallam, C., & Knight, J. (2007). Preliminary findings from research on the causes of failure to adhere to infection control in two nhs trusts. oral presentation. *Health protection*.
- Jennett, C., Cox, A. L., Cairns, P., Dhoparee, S., Epps, A., Tijs, T., & Walton, A. (2008). Measuring and defining the experience of immersion in games. *International Journal of Human-Computer Studies*, 66(9), 641–661. doi:<http://dx.doi.org/10.1016/j.ijhcs.2008.04.004>
- Johnson, D. W. & Johnson, R. T. (1991). *Learning together and alone : cooperative, competitive, and individualistic learning* (3rd ed.). Boston : Allyn and Bacon.
- Johnson, D. W., Johnson, R. T., & Smith, K. A. (1991). Active learning: cooperation in the college classroom.

- Jordan, K. (2013). Mooc completion rates: the data. Retrieved February 6, 2015, from <http://www.katyjordan.com/MOOCproject.html>
- Kaplan, J. & Yankelovich, N. (2011). Open wonderland: an extensible virtual world architecture. *Internet Computing, IEEE, 15*(5), 38–45.
- Kerr, N. L. & Bruun, S. E. (1983). Dispensability of member effort and group motivation losses: free-rider effects. *Journal of Personality and social Psychology, 44*(1), 78.
- Kickmeier-Rust, M. D., Peirce, N., Conlan, O., Schwarz, D., Verpoorten, D., & Albert, D. (2007). Immersive digital games: the interfaces for next-generation e-learning? In *Universal access in human-computer interaction. applications and services* (pp. 647–656). Springer.
- Kim, E. C. & Kellough, R. D. (1974). A resource guide for secondary school teaching. planning for competence.
- Knowledge@Wharton. (2012). Moocs on the move: how coursera is disrupting the traditional classroom. Retrieved February 6, 2015, from <http://knowledge.wharton.upenn.edu/article/moocs-on-the-move-how-coursera-is-disrupting-the-traditional-classroom/>
- Kolb, D. A. (2014). *Experiential learning: experience as the source of learning and development*. FT Press.
- Kollar, P. (2014). World of warcraft hits over 10 million subscribers as warlords of draenor launches. Retrieved February 11, 2015, from <http://www.polygon.com/2014/11/19/7250737/world-of-warcraft-warlords-draenor-10-million-subscribers>
- Kolowich, S. (2013). Researchers push mooc conversation beyond 'tsunami' metaphors. Retrieved February 6, 2015, from <http://chronicle.com/blogs/wiredcampus/researchers-push-mooc-conversation-beyond-tsunami-metaphors/48911>
- Kreijns, K., Kirschner, P. A., & Jochems, W. (2003). Identifying the pitfalls for social interaction in computer-supported collaborative learning environments: a review of the research. *Computers in Human Behavior, 19*(3), 335–353. doi:[http://dx.doi.org/10.1016/S0747-5632\(02\)00057-2](http://dx.doi.org/10.1016/S0747-5632(02)00057-2)
- Laal, M. & Ghodsi, S. M. (2012). Benefits of collaborative learning. *Procedia - Social and Behavioral Sciences, 31*, 486–490. World Conference on Learning, Teaching and Administration - 2011. doi:<http://dx.doi.org/10.1016/j.sbspro.2011.12.091>
- Lee, M. (2009). How can 3d virtual worlds be used to support collaborative learning? an analysis of cases from the literature. *Journal of e-Learning and Knowledge Society, 5*(1). Retrieved from [http://je-lks.org/ojs/index.php/Je-LKS\\_EN/article/view/300](http://je-lks.org/ojs/index.php/Je-LKS_EN/article/view/300)
- Levy, D. (2011). Lessons learned from participating in a connectivist massive online open course (mooc). In *Proceedings of the chais conference on instructional technologies research 2011: learning in the technological era* (pp. 31–36).

## Bibliography

- Lewin, T. (2013). Universities abroad join partnerships on the web. Retrieved February 6, 2015, from <http://www.nytimes.com/2013/02/21/education/universities-abroad-join-mooc-course-projects.html>
- Linden Research, I. (2014, December). Second life official site - virtual worlds, avatars, free 3d chat. Retrieved from <http://secondlife.com/>
- Manuelian, P. D. (2013). Giza 3d: digital archaeology and scholarly access to the giza pyramids: the giza project at harvard university. In *2013 digital heritage international congress (digitalheritage 2013)*. IEEE.
- Marcus, T. D. (2007). Fostering creativity in virtual worlds: easing the restrictiveness of copyright for user-created content. *J. Copyright Soc'y USA*, 55, 469.
- Martin, J.-P. (1985). *Zum aufbau didaktischer teilkompetenzen beim schüler: fremdsprachenunterricht auf der lerntheoretischen basis des informationsverarbeitungsansatzes*. Narr.
- Martin, J.-P. (2002, December). Lernen durch lehren (ldl). *Die Schulleitung - Zeitschrift für pädagogische Führung und Fortbildung in Bayern*. (4), 3–9.
- Martin, J.-P. & Oebel, G. (2007). Lernen durch lehren: paradigmwechsel in der didaktik. *Deutschunterricht in Japan, Zeitschrift des Japanischen Lehrerverbandes*, (12), 12.
- Marzano, R. J. (2011). Art and science of teaching. the perils and promises of discovery learning. *Educational Leadership*, 69(1), 86.
- Massachusetts Institute of Technology. (2010). Tealsim project at mit. Retrieved February 13, 2015, from <http://web.mit.edu/viz/soft/visualizations/tealsim/index.html>
- Mayer, R. E. (2004). Should there be a three-strikes rule against pure discovery learning? *American Psychologist*, 59(1), 14.
- McKee, E., Williamson, V. M., & Ruebush, L. E. (2007). Effects of a demonstration laboratory on student learning. *Journal of Science education and Technology*, 16(5), 395–400.
- Messick, D. M. & Mackie, D. M. (1989). Intergroup relations.
- Messinger, P. R., Stroulia, E., Lyons, K., Bone, M., Niu, R. H., Smirnov, K., & Perelgut, S. (2009). Virtual worlds—past, present, and future: new directions in social computing. *Decision Support Systems*, 47(3), 204–228.
- MoocGuide. (2011). Benefits and challenges of a mooc. Retrieved February 6, 2015, from <http://moocguide.wikispaces.com/2.+Benefits+and+challenges+of+a+MOOC>
- Moodle.org. (2015). Moodle statistics site. Retrieved February 9, 2015, from <http://moodle.org/stats>
- Moschini, E. (2010). The second life researcher toolkit—an exploration of inworld tools, methods and approaches for researching educational projects in second life. In *Researching learning in virtual worlds* (pp. 31–51). Springer.
- Muratet, M., Delozanne, E., Torguet, P., & Viallet, F. (2012). Serious game and students' learning motivation: effect of context using prog&play. In *Intelligent tutoring systems* (pp. 123–128). Springer.

- Muratet, M., Torguet, P., Jessel, J.-P., & Viallet, F. (2009). Towards a serious game to help students learn computer programming. *International Journal of Computer Games Technology*, 2009, 3.
- Nebolsky, C., Yee, N. K., Petrushin, V. A., & Gershman, A. V. (2003). Using virtual worlds for corporate training. In *Advanced learning technologies, iee international conference on* (pp. 412–412). IEEE Computer Society.
- Nicoletta Di Blas, S. H., Paolo Paolini. (2003). The see experience: edutainment in 3d virtual worlds. *Museums and the Web 2003*. Retrieved April 25, 2014, from <http://www.archimuse.com/mw2003/papers/diblas/diblas.html>
- Njoo, M. & De Jong, T. (1993). Exploratory learning with a computer simulation for control theory: learning processes and instructional support. *Journal of Research in Science Teaching*, 30(8), 821–844.
- Novak, T. P. (2012). Quality of virtual life. *TRANSFORMATIVE CONSUMER RESEARCH FOR PERSONAL AND COLLECTIVE WELL-BEING*, David Glen Mick, Simone Pettigrew, Cornelia Pechmann, Julie Ozanne, eds., New York: Taylor and Francis.
- OECD. (2011). Virtual worlds. immersive online platforms for collaboration, creativity and learning. *OECD Digital Economy Papers*, 184. doi:<http://dx.doi.org/10.1787/5kg9qgnpjmg-en>
- OWL-Foundation. (2014a, December). Open source 3d virtual collaboration toolkit. Retrieved from <http://openwonderland.org/>
- OWL-Foundation. (2014b, November). Open wonderland community svn server. Retrieved from <http://openwonderland-modules.googlecode.com/svn/trunk/0.5>
- OWL-Foundation. (2014c, November). Open wonderland module warehouse. Retrieved from <http://openwonderland.org/index.php/c-modules/module-warehouse>
- OWL-Foundation. (2014d, December). Wonderblog. Retrieved from <http://blogs.openwonderland.org>
- Panitz, T. (1999). Collaborative versus cooperative learning: a comparison of the two concepts which will help us understand the underlying nature of interactive learning.
- Parsons, D. & Stockdale, R. (2010). Cloud as context: virtual world learning with open wonderland. In *Proceedings of the 9th world conference on mobile and contextual learning (mlearn 2010), valetta, malta* (pp. 123–130).
- Petty, G. (2004). *Teaching today: a practical guide*. Nelson Thornes.
- Pfeiffer, J. & Rusam, A. M. (1992). Der student als dozent: die methode 'lernen durch lehren' an der universität. *Praktische Handreichung für Fremdsprachenlehrer*. Frankfurt/Main: Verl. P. Lang, 425–433.
- Phillips, R. (2005). Challenging the primacy of lectures: the dissonance between theory and practice in university teaching. *Journal of University Teaching & Learning Practice*, 2(1), 2.

## Bibliography

- Pirker, J. (2013). *The virtual teal world-an interactive and collaborative virtual world environment for physics education* (Doctoral dissertation, Master's thesis, Graz University of Technology).
- Pirker, J., Gütl, C., Belcher, J. W., & Bailey, P. H. (2013). Design and evaluation of a learner-centric immersive virtual learning environment for physics education. In *Human factors in computing and informatics* (pp. 551–561). Springer.
- Prensky, M. (2003). Digital game-based learning. *Computers in Entertainment (CIE)*, 1(1), 21–21.
- Prince, M. (2004). Does active learning work? a review of the research. *Journal of engineering education*, 93(3), 223–231.
- PublicVR. (2008). Publicvr website. Retrieved February 16, 2015, from <http://publicvr.org>
- Reis, R., Escudeiro, P., & Escudeiro, N. (2010). Comparing social virtual worlds for educational purposes. In *Advanced learning technologies (icalt), 2010 ieee 10th international conference on* (pp. 186–190). IEEE.
- Rieber, L. P. [L. P.]. (1996). Microworlds. In *Handbook of research for educational communications and technology* (pp. 583–603).
- Rieber, L. P. [Lloyd P.]. (2005). Multimedia learning in games, simulations, and microworlds. *The Cambridge handbook of multimedia learning*, 549–567.
- Riedmann, P. (2014, January). *Realistic immersive virtual agent-based learning environment (rivale)* (Master's thesis, Graz University of Technology).
- Rudman, P., Lavelle, S., Salmon, G., & Cashmore, A. (2010). Swift-ly enhancing laboratory learning: genetics in the virtual world. *Association of Learning Technology*.
- Schell, J. (2014). *The art of game design: a book of lenses*. CRC Press.
- Schiller, S. Z., Goodrich, K., & Gupta, P. B. (2013). Let them play! active learning in a virtual world. *Information Systems Management*, 30(1), 50–62.
- Siang, A. C. & Rao, R. K. (2003). Theories of learning: a computer game perspective. In *Multimedia software engineering, 2003. proceedings. fifth international symposium on* (pp. 239–245). IEEE.
- Sivan, Y. (2008). 3d3c real virtual worlds defined: the immense potential of merging 3d, community, creation, and commerce. *Journal For Virtual Worlds Research*, 1(1).
- Slater, M. (2003). A note on presence terminology. *Presence connect*, 3(3), 1–5.
- Slavin, R. E. (1980). Cooperative learning. *Review of Educational Research*, 50(2), 315–342. Retrieved April 26, 2014, from <http://www.jstor.org/stable/1170149?origin=JSTOR-pdf>
- Slott, J. (2009). Project wonderland v0.5: creating a simple shape (part 1). Retrieved December 18, 2013, from <http://openwonderland.googlecode.com/svn/www/wiki/backup/pdf/ProjectWonderlandDevelopingNewCell05Part1.pdf>
- Strauss, V. (2013). Why the 'learning pyramid' is wrong. Retrieved February 3, 2015, from <http://www.washingtonpost.com/blogs/answer-sheet/wp/2013/03/06/why-the-learning-pyramid-is-wrong/>
- Susi, T., Johannesson, M., & Backlund, P. (2007). Serious games: an overview.



- Swing, S. R. & Peterson, P. L. (1982). The relationship of student ability and small-group interaction to student achievement. *American Education Research Journal*, 19(2), 259–274. doi:10.3102/00028312019002259
- Taraghi, B., Ebner, M., Schön, S., et al. (2013). Systeme im einsatz. wbt, lms, e-portfolio-systeme, ple und andere. In *L3t. lehrbuch für lernen und lehren mit technologien*.
- Treack, T. v., Himpsl-Gutermann, K., Robes, J., et al. (2013). Offene und partizipative lernkonzepte. e-portfolios, moocs und flipped classrooms. In *L3t. lehrbuch für lernen und lehren mit technologien*.
- Van Eck, R. (2006). Digital game-based learning: it's not just the digital natives who are restless. *EDUCAUSE review*, 41(2), 16.
- Wallace, R. S. (2009). *The anatomy of alice*. Springer.
- Warburton, S. (2009). Second life in higher education: assessing the potential for and the barriers to deploying virtual worlds in learning and teaching. *British Journal of Educational Technology*, 40(3), 414–426.
- Wood, D., Hopkins, L. et al. (2008). 3d virtual environments: businesses are ready but are our 'digital natives' prepared for changing landscapes? Ascilite.
- Wood, E. (2013). Problem-based learning: exploiting knowledge of how people learn to promote effective learning.
- World Health Organization. (1988). *Education for health: a manual on health education in primary health care*. World Health Organization.
- Zviran, M., Glezer, C., & Avni, I. (2006). User satisfaction from commercial web sites: the effect of design and use. *Information & Management*, 43(2), 157–178. doi:http://dx.doi.org/10.1016/j.im.2005.04.002
- Zyda, M. (2005). From visual simulation to virtual reality to games. *Computer*, 38(9), 25–32.



# Appendix



# Appendix A.

## DVD Content

The following content can be found on the DVD attached to the printed form of this work:

- PDF version of the thesis
- **Development**
  - Installation file for used Java version (JDK 7u65)
  - JAR file to install an Open Wonderland server
  - Developed OWL software modules (Item, Itemboard, Quiz)
- **Evaluation**
  - *Materials*
    - \* Pre-Questionnaire
    - \* Task-List
    - \* Post-Questionnaire
  - *Test Sessions*
    - \* Filled in consent forms of test participants
    - \* Video recordings of all test sessions
  - *Results*
    - \* Statistical evaluation of the questionnaires including all text answers