Peer Assessment in Computer Science and Modern Technologies to Build a Flexible E-Learning System around It

Theoretical Part of Bachelor's Thesis
at
Graz University of Technology
submitted by
Dominik Kowald and Joachim Maderer

Supervisor:
Dipl.-Ing. Dr. techn. Christian Gütl

Institute for Information Systems and Computer Media (IICM)
Graz University of Technology
A-8010 Graz, Austria

02.10.2009

Abstract

Actual E-Learning Systems usually only provide assessments of natural language answers by teachers or tutors and do not take notice of other assessment types like self assessment or peer assessment although these variants would have a great benefit, not only for teachers but also for students.

Students would have the chance to get into the role of the teacher and correct the answers of their colleagues. Furthermore they have to get deeper into the topic to make good corrections and so they have a higher learning success. Teachers and tutors would have the great advantage to save the time they would have needed to correct all the answers by themselves and to test a new assessment method in computer science.

This paper should give an overview about peer assessment in general and also about the most important aspects of peer assessment in computer science. This also includes the required technologies and frameworks to build a flexible E-Learning System which uses peer assessment as one of its assessment types.

Content

1 Introduction

1.1 Motivation

The importance of the lifelong learning process is increasing from year to year. This means that there has to be a way of providing feedback for learning activities in an easy, significant and also correct way. The classic assessment process in which a teacher or tutor provides feedback and grades for a group of students satisfies these needs but there is also an aspect to think about: If students need more significant feedback from year to year to get the know-how that is necessary in economy, also more teacher and tutors are needed which would lead to higher costs.

This problem tends to the idea of peer assessment which is the assessment of students by other students to provide feedback and grades. With this assessment method it is possible to reach two goals: The workload of teachers and tutors decreases because students provide their feedbacks among each other and the second advantage is that students also increase their educational skills while doing this. On the contrary, a high amount of student feedbacks is needed to get an evaluation which is very close to a teacher or tutor evaluation and so this means lot of administrative effort. To solve this problem, peer assessment could be included in e-learning systems where the calculations of these evaluations can be made in microseconds.

1.2 Structure of the Paper

This paper consists of two main parts: First, the theoretical part describing the ideas of classic peer assessment and computer based peer assessment and second, a more practical part dealing with frameworks and technologies that are needed to build a flexible e-learning system around this way of assessment.

The first part consists of chapter 2 to 4. Chapter 2 describes the general and practical aspects of classic peer assessment including the definition, the history, the usage and the advantages/disadvantages; also some differences to other assessment types like self assessment or group assessment are shown.

On the other hand, chapters 3 and 4 look at the technology enhanced parts of peer assessment and learning. This includes e-learning, the peer context of e-learning, e-assessment in general and of course the computer based peer assessment with its uses, its problems and its actual state of the art.

Chapter 5 acts as the link between the first and the second part and deals with the requirements that are needed to build an e-learning system using peer assessment; a possible system architecture for such a purpose is also given.

The second part consists of chapter 6 to 8 and the first of its chapters covers the data model including the database management system MySQL, the idea of object-relational mappings and the

Hibernate framework. Chapter 7 and 8 deal with the environments needed on the client and on the application side which includes frameworks and technologies like Microsoft .NET, the C# programming language, asp.NET, the role of the web sever, Java script and some more. The 9th and last chapter is the conclusion and acts as a summary and an outlook into the future.

2 Peer Assessment

2.1 Educational Assessment

The traditional view on the term assessment is that one or more teachers assess the work of students to provide a grade for their effort. An advanced view on it is that the purpose of the assessment process is not only to provide marks but rather to play a vital role in the learning process itself. There are even views on the term assessment that do not look on the summative aspect of it but see assessment as a special form of learning which is as important for the student as the learning process itself (Roberts, 2006).

Earl (2003) defined three main approaches of the assessment process: The assessment of learning, the assessment for learning and the assessment as learning. The first one, assessment of learning, can be seen as the classical assessment which deals with giving marks. When teachers are doing assessment for learning, they use the collected assessment data to modify the teaching process on the basis of this. The third definition, assessment as learning, means that students critically monitor what they are learning and use the feedback to improve their learning skills.

2.2 General Aspects

Peer assessment plays a high role in both types of assessments, formative and summative. This means that students can develop educational skills like giving feedback to and receiving feedback from other peers which gives them a higher feeling of responsibility in the sense of formative assessment (Kim, 2008).

On the other hand, reports have shown that also the summative assessment purposes are fulfilled because the contribution between the grades produced by teachers and peer assessment is very small (10 – 30%) (Kandlbinder, 2009).

2.2.1 Definition

Peer assessment is the assessment of students by other students to provide feedback and grading and can be seen as a subset of peer tutoring. This way of assessing should improve the quality of learning and should also decrease the workload of teachers and tutors because student involvement can not only be included in the evaluation procedure but also in the prior settings of the exams like the selection of the relevant topics (Bostock, 2000).

2.2.2 History

The history of peer assessment can be divided into two main parts, the first part which deals with the classical idea of it without the use of computer science and the second part which describes the uses of its approach in computer science. This chapter gives a short overview of the former part.

It is clear that the usage of the peer assessment idea has increased a lot since the introduction of e-learning systems, especially web based e-learning systems, because of the high effort that is needed to evaluate the feedbacks and grades of a large group of people which have evaluated each other. Nonetheless, the mentioned peer assessment idea is very old.

One of the first examples goes back to the years 1774 – 1826 to the University of Glasgow where Professor Georg Jardine included methods and rules of peer assessment into his lectures and exams. Approximately 150 years later Bob Wayne Ford has written his doctoral thesis on the effects of peer editing and grading on writing skills of college freshmen and Christopher Orpen has compared the assessment performances of teachers and students which is still a very interesting topic. The next step in the history of peer assessment was its introduction in computer science but this issue will be discussed in a later topic (Guetl, 2009).

2.3 Practical Aspects

The main question that raises when thinking about the practical parts of peer assessment is if the students are able to give feedback which is helpful for other students and which can also increase the assessment performance of themselves. Of course there will always be a few students who do not respect this assessment principle and give unfair and inappropriate markings. To deal with such cases also tutors are needed who maintain the assessment process and also give ratings to increase the quality of the summative aspect of peer assessment.

Another important point is that the assessment process has to be anonymous to make sure that likes and dislikes among students have no effect on the grading. This also assures that the students are not afraid of being discriminated by other students if they give bad marks. It should also be considered that the students have to know the purpose of the peer assessment idea to take it serious and to produce useful feedbacks in succession (Kay, 2007).

2.3.1 Usage

As mentioned above, peer assessment is ideal if the principle of formative assessment can be useful for the students. This is really important because it would be very time consuming for the staff to provide all the feedback, which is needed to practice the student's skills, on their own.

One good example for this is the use of peer assessment to improve the reporting skills of journalism students. This is because journalism students have increased requirements for feedbacks

on their own work and so the students can provide them among each other. Furthermore the students get insights on the writing techniques of their colleagues and so they also gain opportunities to improve their writing skills seen from a different angle (Mathews, 2009).

2.3.2 Approaches

There are different approaches in using the peer assessment idea. Basically it can be used for each type of question, no matter if they are answered in oral or written form because the assessors are human beings that can deal with both inputs. Most of the time peer assessment is used for short natural language questions to give formative and summative assessment or for long natural language reports to provide feedback on the writing style of the author. In most cases there is no need to use peer assessment for single choice or multiple choice questions because they can be evaluated nearly automatically and so there is no possibility to provide formative assessments (Bhalerao & Ward 2001, Guetl 2009).

2.3.3 Advantages

Bostock (2000) describes the following potential advantages of peer assessment:

- Improving motivation because a sense of ownership of the assessment process is given
- Encouraging students to take responsibility for their own learning, developing them as autonomous learners
- Assessment is treated as a part of learning and so mistakes are opportunities rather than failures
- Practising the transferable skills, especially evaluation skills, needed for lifelong learning
- Encouraging deep and not surface learning

To sum this up it can be said that students have the possibility to see the other side of the usually strictly delimited learning and assessment process which improves the motivation of both sides. This also means that the mistakes of other students are opportunities for the assessors because in these cases they get an even deeper understanding of the topic. They have to carefully read a lot of answers for the same question, so the chance that they will keep this knowledge in mind for a longer time increases with each one. Furthermore the students will need assessment skills and also the ability to take responsibility for their feedbacks in their job career and so peer assessment can be a good practice.

2.3.4 Disadvantages

Based on Kandlbinder (2009) the following disadvantages can be found for peer assessment:

- Students may lack the ability to evaluate
- Students may not take it seriously
- Students may allow friendships, entertainment values etc. to influence their marking
- Students may fear the possibility of being discriminated against, being misunderstood etc.
- Students may misinform each other without any lecturer intervention
- High amount of administrational effort is needed

Summing up it has shown that the quality of peer assessment depends on the evaluation skills of the students. The possibility is quite high that there are students who think that peer assessment is only a game and do not set value on giving serious feedbacks and marks. So the administrational effort to find such false assessment can be very high. Another thing is that the peer assessment process has to be completely anonymous because usually the students know each other and they might be afraid of giving bad feedbacks. Further, checking such assessments involves a great administrational effort.

2.4 Differences to Other Assessment Methods

Peer assessment is often compared to other assessment methods because it is a quite new assessment principle and teachers are interested in its advantages and disadvantages and if it can be used in collaboration with other assessment methods. One example is self assessment because it has the same purpose as peer assessment, just from a different angle (Kandlbinder, 2009).

2.4.1 Teacher/Tutor Assessment

This is the classical way of assessment where the students are marked by teachers and/or tutors and are not able to influence the assessment process. There are also improved variants of this assessment type like the classroom assessment technique. This means that the assessment process is aimed at course improvement, rather than at just giving marks. To use this kind of assessment it is necessary to know about how and what students are learning (Rhem, 2003).

2.4.2 Automatic Assessment

Motivated by the classical teacher assessment idea, another modern assessment type has been evolved, namely the automatic assessment. It differs from normal teacher assessment in the important fact that no teacher is needed for it because it is done automatically by a computer system.

Gütl (2007) reports on the experiences that were made while implementing a prototype of a fully automatic knowledge assessment tool. This tool was based on ROUGE (Recall-Oriented Understudy for Gisting Evaluation) which makes it possible to assess short free-text answers by comparing these answers with the reference answers. Of course this technology will take time until its assessments will be as good as teacher assessments because of the complexity of the human language.

2.4.3 Self Assessment

In contrast to peer assessment, in self assessment students reflect and suggest grades for their own learning and not the learning of other students. So the students have the great benefit that they have a higher engagement with their own work which gives them the ability to determine their weaker areas to train them. On the other hand most students find it very hard to assess their own work and there are often little correlations between the marks they give themselves and the marks which are given by the teachers but this improves with practice (Roberts, 2006).

2.4.4 Group Assessment

There are different definitions for group assessment which go from the assessment of groups as a whole to the assessment of group members by other group members. The second definition is very similar to the definition of peer assessment and in most cases peer assessment is used within the context of group work to determine individual marks for each group member. The introduction of peer assessment into it should solve the problem of the free-riders. These are students who sit back and do not involve into the group work because they want to benefit from a shared group mark (Roberts, 2006).

2.5 Conclusion

Peer assessment can be very useful for students because not only the summative but also the formative assessment approach is given. This means that it can be used to determine grades for the student's work and to provide feedback to increase their educational skills. Furthermore the teacher's workload is decreased because the students create their feedbacks themselves. On the other hand it is necessary to maintain the peer assessment process and to define some guidelines for it to make sure that it will deliver useful results. This can lead to a higher administrational effort.

3 Technology Enhanced Learning

3.1 Definition

Technology enhanced learning can be defined as any learning activity which is accomplished through technology. This includes e-learning but also computer presentations that are shown by the teacher (TLRP, 2009).

3.2 E-Learning

When talking about modern assessment principles, also online e-learning systems have to be mentioned because they can improve face-to-face methods in many ways: Reducing the resources needed for evaluations, saving the given evaluations in a database, saving time and decreasing costs in succession (Roberts, 2006).

3.2.1 Definition

There are lots of definitions for e-learning because it is a widespread domain. Some people think that e-learning is only the process of sitting in front of the computer and study the teacher's materials instead of sitting at the university or in school and learn in the classical way. Actually it can be seen as any type of learning where technology is used beyond the pedagogical process itself (Nichols, 2008).

3.2.2 History

Aranda (2009) states that the very beginning of e-learning can be equalized with the starting points of distance learning although there are some differences between those two types because distance learning in its classical sense is only defined as the physical separation of the student from the teacher. In 1840, some educational programs where offered in England which were held by correspondence through mail. This can be seen as the first form of e-learning/distance learning because technologies like television, radio and the computer of course have only made this approach easier and better but have not changed the core idea of it.

Around 1960 the first teaching machines were developed and so the gap between pure distance learning and e-learning could be closed. The next milestone was the introduction of hypertext and

hypermedia in the 1990s which made it possible to use the internet for e-learning purposes. The Web 2.0 idea even improved this approach because of its social aspect (About e-learning, 2009).

3.3 Peer Context in E-Learning

The idea of the Web 2.0 term put the social context into the World Wide Web. This means that users have the ability to work together and share their knowledge among each other. The principle of sharing knowledge can also be found in the context of e-learning in various ways which are described in the following chapter.

3.3.1 E-Learning 2.0

Like the term Web 2.0, the term e-learning 2.0 refers on the social aspect of software but in this case the social aspect is used to socially create knowledge. In other words we use social learning instead of the old fashioned way of creating and learning a knowledge base only. In the e-learning context new technologies like blogs, wikis and even virtual worlds like Second Life can be used for this (Brown, 2008).

3.3.2 Peer Learning

Peer learning is the way of cooperative learning which uses student-to-student interactions to handle the learning process. To make peer learning successful it is necessary that all participating students have enough knowledge of the actual topic and that the size of the student groups is also aligned to the actual topic. Another important aspect has already been mentioned in the chapter about group assessment: The problem of the free-writer which can be solved with individual peer rating of the team members (Christudason, 2003).

3.3.3 Peer Tutoring

Peer tutoring is the process in which one student, the peer tutee, is being tutored by another student, the peer tutor. This approach has lots of benefits for both sides: The tutor sees the learning process from the other side and can train his evaluation skills and the tutee has the ability to get one-on-one feedback from a peer who can take time to deliver useful formative and summative assessments. Furthermore, peer tutoring can be compared with peer assessment because one student is going to be

assessed by another student which is the definition of peer assessment; indeed it is even a subfield of peer tutoring (DuPaul, 1998).

3.3.4 Peer Reviews

In opposition to peer tutoring, peer reviewing is the process in which the work of one student is reviewed by one or more experts and not by other students who have approximately the same level of knowledge. Peer reviews can be a great benefit for students because it is very difficult to find every mistake in a piece of work and they will also get a different, professional view on it. On the other hand, peer reviews are also very important in the field of publications and papers because papers that were not peer reviewed seem to be suspicious by professional readers (Rowland, 2007).

Furthermore, peer reviews can be used as a type of software review which is very important in the software development process because it is not only possible to find early design errors with it but also possible security errors later in the process. One method to achieve this is the process of pair programming in which two persons develop software together on one computer (Cockburn and Williams, 2000).

3.4 Conclusion

Using peer assessment in computer science one has to be aware of at least two facts: The definition of e-learning and the impact of the peer context within computer science. E-learning can be seen as every type of learning where technology is used for it and the peer context has a high influence on computer science especially since the introduction of the Web 2.0. Furthermore, modern peer assessment has many similarities with the peer-techniques because each of it also uses the evaluations of one or many peers to build new knowledge with it.

4 Peer Assessment in Computer Science

4.1 General Aspects

In the last decades the primary movement was only to transfer existing methodology to computer implemented systems. But this also limited the variation of assessment types because of a desire to evaluate the data automatically. In order to adapt to society changes which raised the importance of lifelong learning, new methods for assessment with focus on comprehensive skills are required.

E-assessment as a general term describes all methods around the assessment process that can be achieved with computer systems. Basically E-Assessment has to be separated into Computer Based Assessment (CBA) and Computer Assisted Assessment (CAA). While the first concept refers to the actual test situation whenever a student is interacting with a computer that also provides automatic evaluation and feedback, the second concept rather describes a broader process that includes evaluation, analysis and reporting (Al Smadi, 2009).

4.2 History

After using peer assessment in its classical way at universities and colleges (see section 2.1.2) the first ideas raised to include it into computer based systems to automatically evaluate the grading and feedbacks of the students. One of the first systems which had used the peer assessment approach was MUCH (Many Using and Creating Hypermedia). This was a tool for collaborative learning.

At the end of the 1990s network based and web based systems were developed like NetPeas (networked peer assessment system) and tools which performed the assessment process via email. Some newer systems at the beginning of 2000 were OASYS and OPAS (Online Peer Assessment System). OASYS made it possible to use automatic assessment for multiple choice questions and peer and self assessment for free text answers and OPAS was the first system which offered the functionality for involved persons to comment, discuss and present the results of the assessments (Guetl, 2009).

4.3 State of the Art

Web researches showed that there are a lot of peer assessment tools available today, some commercial, some open source and free and some only accessible via web after registration. The most

important of this wide range of tools is WebPA, an open source tool available via Source-Forge and CATME, a peer rating service available via their homepage.

4.3.1 WebPA

WebPA (Web-based Peer Assessment) is an open source online peer assessment tool which is used for a fair assessment of group work. The tool reaches this goal by using peer assessment inside the group to make sure that each student will get a fair and individual mark because in classical group ratings it is common practice that each team member receives the same team mark.

WebPA was developed by a group of academics at the Loughborough University. These academics have already used an online peer assessment system (PASS) since 1998 but they were not satisfied with its functionalities and so they rewrote this system based on the feedback of tutors and students and called it WebPA. One main advantage of WebPA is that it gives the ability to individually change the way the scoring is working (WebPA, 2009).

4.3.2 CATME

CATME (Comprehensive Assessment of Team Member Effectiveness) is a web service that allows people to use peer and self assessment to evaluate how effectively each team member works for the team and was developed based on extensive university researches.

The tool works as a web service and so the registered students provide their comments and ratings about their team members via the web site after logging in. This happens in five categories (CATME, 2009):

- Contributing to team‘s work
- Interacting with teammates
- Keeping the team on track
- Expecting quality
- Having relevant knowledge, skills and abilities

4.3.3 Online Self and Peer Assessment Tool

This tool offers a web interface where users can subscribe to teams and can negotiate roles and responsibilities for the semester. These resulting contracts can be seen by the monitoring tutor and the team. During the semester the team members evaluate the articles according to their

responsibilities with self and peer assessment. At the end of the semester tutors use the peer assessment information and personal observations to form marks with the team. The tool requires an Apache web server running PHP and a MySQL database. The software can be downloaded for free but its use is limited to educational purposes only (Luca & Dunbar with Oliver, 2002).

4.3.4 OASYS

OASYS (Online Assessment System) is a peer assessment system which features two basic process steps. At first a test phase is started where all participants have to answer questions. Some of these questions are multiple choice questions, which can immediately get evaluated automatically by the system. The remaining questions are natural language answers. Afterwards, if all tests have been completed, each participant gets a set of anonymously selected test sheets to evaluate. Based on the multiple choice results the system classifies the test sheets in three categories from worst to best results. Now the system approximately distributes one test from each category to each participant. After the peer assessment has been completed the results can be moderated by tutors in case of high variances between the assessments per test. The system was implemented with Apache, PHP and MySQL (Bhalerao & Ward, 2001).

4.3.5 Mini-PAT

Mini-PAT (Mini Peer Assessment Tool) is a peer assessment tool provided by the Intercollegiate Surgical Curriculum Programme (ISCP). It is based on the Sheffield Peer Review Assessment Tool (SPRAT) and was customized to fulfill the needs of surgical training. Further, it is designed for workplace-based assessment. The participant has to give an answer with self-confidence and can then select co-workers which will rate the answer. As this can be done at any time, the participant probably has to wait until the rating has been completed. Communication is carried out via E-Mail and the questionnaires are anonymously. The assessment tool is supported by a detailed guidance document in order to achieve reliable results (ISCP, 2009).

4.4 Usage

In winter term 2007 a controlled peer assessment experiment has taken place at the University of Technology Graz. Beside 29 participants of the regular course “Information Search and Retrieval”, five PhD students were present to perform the tutor-role. Supported by a self-written prototype called *Peer-Assessor* the experiment included the following steps:

- Introduction to the experiment, especially peer assessment and consideration that the peer assessment process is also important for the learning process
- First, the participants had to study the prepared topic "Document classification" based on articles from Wikipedia. The available time frame was 45 minutes.
- Directly after the learning session, a test containing five questions was performed with the prototype system. Study material was not allowed. In addition to the natural answer a self confidence value was required (15 minutes).
- After a break the peer assessment session started. All students had to evaluate all students' answers in random order in comparison with a reference answer created by the tutors.
- Finally, students as well as tutors had to answer a questionnaire. The results for knowledge acquisition and peer assessment performance were sent by email.

Furthermore, the five questions have been selected to cover different kinds of answer types including definition, enumeration, explanation of concept (two times) and name of an abbreviation. Best overall results were achieved with enumeration in correlation with tutor assessment results (Guetl, 2007).

4.5 Conclusion

Since experiments like Guetl's (2007) have been performed in controlled environments with only a few participants, it was possible to get clean results without missing evaluations. However, when peer assessments will be used on a regular basis, maybe with a huge amount of participants in home work, nobody can guarantee that each one will do the assessment conscientiously. Sure there will be negative results on the assessment performance on those participants, but it is also possible that someone does not finish the whole process. As a consequence average result calculation will not be correct for other students and that will of course increase the amount of administrative work.

5 Building a Flexible E-Learning System around Peer Assessment

5.1 Introduction

The first four chapters have given an overview about peer assessment in the classical sense, e-learning, peer-learning and peer assessment in computer science as a combination of these three topics. The next step is to use the ideas and requirements of these approaches and add the knowledge about modern technologies and frameworks to think about a flexible e-learning system which uses peer assessment as one of its assessment methods.

5.2 Requirements

Beside the usual requirements of a modern e-learning system like usability, security and extensibility to support different exam and question types, a peer assessment system has a the special need to provide easy and efficient methods for creating useful formative assessments.

Guetl (2009) stated the following ways to fulfil these needs in a personal communication:

- Coloured markings for annotating which parts of the given answers are correct, wrong or irrelevant
- Additional textboxes to mention comments and missing parts
- Showing all individual comments and annotations in the exam feedbacks
- Implementing an answer distribution algorithm that makes sure that enough student and tutor evaluations are created and distributed fairly

5.3 System Architecture

5.3.1 Layer vs. Tier

While in language terms there is hardly a difference between a layer and a tier, software paradigms define a multi layer application as a design that features different components that interact via interfaces, consequently within the same process space. The aim of separated layers is to divide an

application into logical parts, ideally exchangeable, for example a data access layer. Also pluggable components are possible. Today most enterprise applications rely on three layer designs. These layers are usually for presentation, control and data access. Tiers, on the other hand, define a distribution pattern. They usually communicate via network connections, maybe with named pipes if running on the same operating system (MSDN Library, 2009).

5.3.2 Three-Tier-Application

Based on the definition before, a web application could be seen as a typical three-tier-application environment. Although the client is far away from the server it is also a part of the application that is responsible for presentation. The web server is responsible to host the primary application domain which executes the business logic. While it would be possible to store the data with an integrated component, common practice is to use a separate database management system (MSDN Library, 2009).

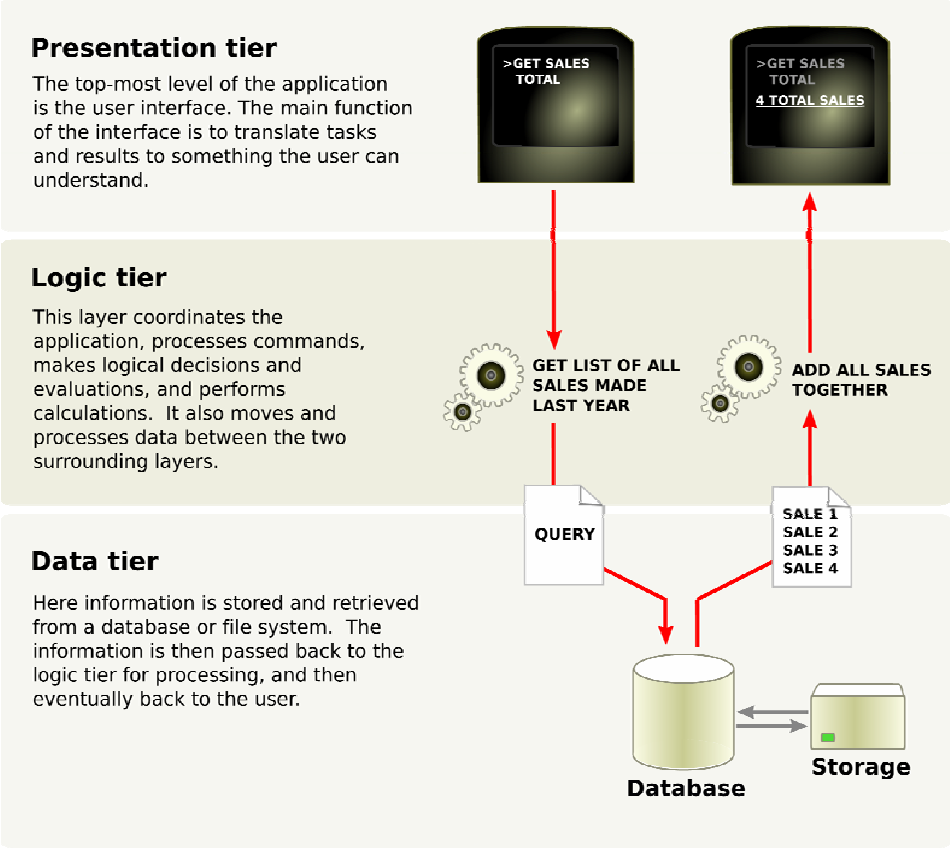

Figure 1. Three-tiered application model, taken from Wikipedia[1]

[1] http://en.wikipedia.org/wiki/Multitier_architecture (l. v. September 27, 2009)

5.4 Conclusion

The architecture of a flexible online peer assessment system can be described as a classical three-tier application model, as seen in figure 1, including the data, the logic and the presentation tier. To implement such a model and to fulfill the mentioned requirements it is necessary to take a deeper look at the components within these three tiers which will be done in the next three chapters.

6 Data Model

6.1 Introduction

This section describes MySQL as an open source database management system, Hibernate as an open source persistence framework to use all the advantages a modern object oriented programming language is offering and object-relational mappings as a modern and easy data access concept to combine these two technologies.

6.2 MySQL

MySQL is the world's most popular open source relational database management system. Its primary application area is in conjunction with Apache and PHP running web servers. This system configuration is typically installed on Linux servers and is also known as LAMP. Beside its open source edition which is released as MySQL Community Server it is also possible to purchase a commercial more advanced edition with professional support (MySQL, 2009).

Most language features described by ANSI SQL 3 are implemented by the newest MySQL version. As nearly all database management systems it provides some mentionable language extension like the "limit" clause that makes select queries capable of paging (MySQL, 2009).

6.3 OR Mappings

While programming languages enhanced their features with focus on object based models, DBMS mostly rely on relational algebra due to performance benefits. There have been several studies and prototypes for object based storage solutions since relational databases were designed for procedural programming paradigms. There are only a few established object database solutions on the market but lately new object database projects have been founded since reflection features have raised in modern high-level programming languages. But these approaches are designed for layer-internal operation, which means the data store must be located on the same machine. For example, the German car manufacturer BWM uses DB4o for an embedded car electronics prototype. Nevertheless object databases are hard to configure and refactoring the data model is much more complex in comparison to relational databases (Patterson, 2009).

However, since companies have already invested in expensive relational database solutions and a vast amount of legacy solutions exist, it was necessary to invent solutions that make it possible to use object oriented features of newer languages effectively with relational database management systems.

Object-Relational Mapping Services fill the gap between those two approaches. They offer functionality to save and instantiate objects without the need for writing additional lines of code. Due to the circumstance that SQL based relational databases can only store and manipulate scalar values, it is necessary to convert objects to scalar values and vice versa. This is usually done by a mapping layer. The interface that is provided for the remaining application is normally similar to native object database APIs (Barry, 2009).

6.4 Hibernate

As a part of the JBoss Enterprise Middleware System (JEMS), Hibernate belongs to Red Hat and is therefore an open source project that offers reliable professional support. Its core service consists of an exchangeable object persister interface but with primary support for mapping objects to SQL based relational databases. It is seen as powerful feature – in contrast to concurrent products – that it does not hide the SQL interface from the programmer (Hibernate, 2009).

Hibernate handles the greatest part of common SQL operations for the programmer. The remaining complex and specialized queries can either be written in DBMS specific SQL, object-oriented criteria API or in Hibernate's own query language HQL (Hibernate, 2009).

6.4.1 Definition of Class Mappings

The traditional way for defining persistent classes is to create an XML file for each entity. Hibernate will analyze those files which are contained in the executing package. By using these files hibernate can even create an appropriate database structure on each supported database. To achieve this, Hibernate uses providers called dialects. A dialect defines all DBMS specific language differences, especially for DDL statements but also for all other kind of SQL queries (Hibernate, 2009).

A newer way for defining mappings is based on Java Annotations, respectively .NET Custom Attributes for NHibernate, which allows embedding the mapping configuration directly into the entity class definition (Hibernate, 2009)

6.4.2 Subclasses

One of the most exciting features of object-oriented programming is the ability to define derived classes. Hibernate offers different approaches to achieve this functionality. The Hibernate documentation defines them namely as follows:

- Table per class hierarchy

- Table per subclass
- Table per concrete class

While "Table per class hierarchy" is obviously the simplest method, it fails in being well extendable. With increasing amount of subclasses more and more disk space will be wasted. All fields which are not used for one concrete type will be null for each other type that exists. Furthermore, it is likely to affect other subclasses while extending the table definition.

In terms of well formed relational data models the subclass based method is the cleanest. A partial entity only exists if it is necessary. Although this approach is clear it leads to disadvantages because of intensive queries. Therefore, it is possible to add a discriminator column that determines the concrete class and is located in the root table.

Using one table for each concrete class differs from the subclass approach in that way that even the shared columns are separately described for each inherited class. This technique is similar to the internal storage procedure of PosgreSQL which supports a native table inheritance feature. The consequence is that the results for compound base class queries have to be connected with union statements (Hibernate, 2009).

6.4.3 Lazy Loading and Dynamic Proxy Classes

Given that it is not possible to load the whole database with all its associations to application's memory at once there must be a mechanism which allows associated entities being loaded on demand. This mechanism is called "Lazy Loading". Hibernate implements this feature by implementing the proxy design pattern (Hibernate, 2009).

The proxy pattern is defined as a class that acts as a surrogate between a client and the real target object. Beside some other specialized proxy types, a "virtual proxy" loads larger amounts of data if they are needed, i.e. if a getter is invoked for the first time. So this typically requires that the proxy implements an interface or extends a class which must not necessarily be defined with an abstract modifier (Gang of Four, 1995).

With modern programming languages like Java or the .NET Framework an application can use built in type reflection libraries to create such previously described proxies at runtime. Therefore, Hibernate expects all properties, respectively setters and getters, to be declared "virtual" which would be the default for Java but not for C#. Hibernate relies on the GCLib Library for dynamically compiling virtual proxies (Winkler, 2009).

It is not guaranteed that Hibernate instantiates the actual concrete persistent class. That means the program cannot rely on calling virtual methods neither can the "instanceOf" keyword be used. Polymorphism must be achieved by alternative methods, i.e. applying the visitor design pattern. However, it is possible to turn off this feature explicitly for a class hierarchy but this will also result in a missing virtual proxy support. Further, one has to be careful not to use private members when

working with class internal logic because they will never get initialized in the base class when the dynamic proxy overrides the super class getter. This also and especially regards the implementation of the "equals" method (Winkels, 2009).

6.4.4 Caching

Hibernate offers different cache strategies to reduce the amount of database server roundtrips. This regards the storage of loaded entities in hydrated form (arrays of scalar values) as well as newly created objects that should be saved. Pontarelli (2009) describes so called delayed SQL as a common pitfall for programmers who are new to Hibernate. Hibernate will not save entities until the session is flushed. Therefore, Hibernate seems to be an independent object database but certainly it is not. The best way to come over this problem is to flush the session after each data manipulation statement.

6.4.5 Advantages

Phutela (2006) describes advantages of Hibernate as follows:

- Transparent Persistence allows Hibernate to map objects automatically without the need to write code for each property of each object.
- Because of the integrated independent query languages (HQL and Criteria API) database queries can be written without targeting a specific database.
- Since the default approach to configure associations between objects and tables is to use XML files it is not necessary to modify the source code when the database model changes slightly. By editing the XML files an application can be adapted.

Regarding typical relational queries (joins) benchmarks have shown that Hibernate and object relational mappers can generally benefit from the underlying relational DBMS (Pieter van Zyl et al., 2006).

6.4.6 Disadvantages

According to Pieter van Zyl et al. (2006), who compared modern object databases (namely the db4o project) to Hibernate, the object-relational approach is about two times slower than the native ODBMS. The last one concerns the initial creation time of the Hibernate runtime environment as well as the performance of common persistence operations.

Further disadvantages according to Phutela (2006) are

- Although Hibernate is an open source product it will not be enough to only rely on the free internet support for enterprise applications.
- Hibernate can be an overhead for simple applications.
- It will consume more time to learn how Hibernate works. This can be a problem if a new programmer is responsible to maintain a Hibernate based application.
- When not bypassing Hibernate with database dependent SQL some operations are not possible or will consume more time.

6.5 Conclusion

The data model is an abstract model which describes the data storage and data retrieval layers in a system and exists beside the graphical user interface and the business logic components. In a modern e-learning system which uses peer assessment as one of its assessment methods, it seems ideal to use MySQL as an open source database management system and the open source Hibernate framework with its object-relational mapping technology to have easy object-oriented data access possibilities.

7 Application Environment

7.1 Introduction

As the most important part of a web application the application environment is responsible to accept client request and generate appropriate views for the user. While the data model and therefore the backend layer for data access was already described in a separate major chapter the following chapters will concentrate on web frameworks which represent the front-end to the client and so they are more relevant to security.

7.2 .NET

The .NET Framework is an open and standardized application development platform and was invented by Microsoft. It is comparable to Java and offers a virtual machine concept called Common Language Runtime (CLR) which interprets Microsoft Intermediate Language (MSIL), a pendant to Java byte code. Beside the runtime, the .NET Framework product family provides a wide range of development tools, services and an extensive class library. While the compiler itself was free since the beginning, the integrated development environment Visual Studio is sold like its predecessors. Meanwhile, in reaction to open source tools like Sharp Develop, Microsoft releases slightly limited Express Editions, which can also be used for commercial development. At this time the .NET platform is officially only supported on Microsoft based operating systems including a compact framework edition for mobile devices like smart phones (MSDN Library 2009, Mono 2009).

7.2.1 Language Independency

One of the most important design criteria for .NET was its language independency. That means that all .NET compilations should be able to communicate, irrespective from which high-level language they were compiled to intermediate language. The goal is to move the choice of the right language and consequently all required libraries to the background and let heterogeneous developer teams work more efficiently together. But this entails that several protocols for object definition, memory representation and runtime type information are required. The .NET Framework achieves this with “Common Language Specification” (CLS) and “Common Type System” (CTS) which defines the existence of value based and reference based types. Type systems are usually and necessarily defined as a part of a programming language. The CLR moves this concept from the language to the runtime itself and guarantees type safe and secure interaction between all .NET assemblies (Kuehnel, 2008).

7.2.2 Mono

Although there is no official Microsoft .NET Framework, several open source projects have started to implement alternative runtimes and class libraries which cover the most parts of the official Microsoft class library. Supported by Novell, Mono is one of those projects and so far it is the most advanced open source implementation of the Common Language Runtime. At this time, Mono features all major parts of the .NET Framework 2.x class library and several parts of the newest Microsoft releases. There is even an emulated implementation for Windows Forms applications which normally rely on the message queue system of classic windows applications (Mono, 2009).

7.3 C#

C# is simple object-oriented programming language for general purpose and was defined by Microsoft and standardized at ECMA International in cooperation with Intel and Hewlett-Packard. Its primary design goals are source code portability, internationalization, suitable for hosted and embedded applications and developing components for distributed environments (ECMA Standard 334, 2006).

A definition that might best describe C# was given by Mössenböck (2003), who said that C# is a mix of 70 percent Java, 10 percent C++, 5 percent Visual Basic and 15 percent new features. Although it is true that most of the existing programming languages could be ported to compile for the .NET platform, C# is the default language which was completely new developed to serve all needs and features of the Common Language Runtime. Until now, whenever a new major CLR version was released, a new C# language specification and compiler were also released to make use of the new capabilities (MSDN Library, 2009).

7.4 asp.NET

As part of the new .NET Framework product family asp.Net was designed to displace the former Active Server Pages concept that has been shipping with Microsoft Internet Information Services (IIS). While normal ASP makes use of the controversial Windows Scripting Host, Asp.Net is based on the secure core of managed code and has access to the vast amount of common .NET Framework classes, including XML and XSLT processing, efficient file system access and many more APIs that can be imagined.

Asp.Net is far more than a compact server-side script execution engine. It defines an extendable infrastructure that features onboard services like classic dynamic html pages with integrated scripts as

well as pre-compiled code behind classes or XML Web Services for external API interfaces to be used by other websites or internal client side JavaScript or flash extensions. Starting with a simple web request and application context a programmer can completely implement a specific execution handler for a whole directory or file pattern (MSDN Library, 2009). As integral part of this new concept, asp.NET provides a framework called Web Forms that will be explained later.

7.4.1 Web Servers

Asp.Net applications require an external web server to deliver their content to the client. Following web servers are popular for ASP.NET:

- Microsoft Internet Information Server (IIS)
- XSP in conjunction with Apaches "mod_mono"
- Cassini web server

While Internet Information Server is the standard system for delivering Asp.Net applications on windows based operating systems and represents an entire HTTP 1.1 conform web server, XSP is only an Asp.NET compatible execution engine that ships together with an HTTP 1.0 compliant test server. Production systems will require the Apache module for Mono to provide reliable and consistent services. The Cassini web server is a sample application by Microsoft and offers insights on how ASP.NET hosting works. It is released under a shared source licence (MSDN Library 2009, Mono 2009).

7.5 Web Forms

Borrowed from WYSIWYG based window design solutions Microsoft established a controls based framework to describe web pages like a typical GUI window. XML Tags are used to extended the mark-up of the page and define the server-side and client-side behaviour of such a control. Event handler attributes can be defined and bound to the code behind file via runtime type information capabilities (MSDN Library, 2009).

7.5.1 View State Management

Since web requests are basically a stateless concept that follows the Input-Process-Output Model, Microsoft has developed an automatic system to persist relevant state information of a cyclic requested page within hidden form fields. Most of the controls that are part of the Web Forms

Framework support the View State concept and it is even possible to turn it on or off on each control to save bandwidth if state information is not needed. Although this concept looks very convenient it has produced several problems and leads to large amounts of post data that has to be submitted on each request. Therefore the feature is heavily discussed around experts (MSDN Library, 2009).

7.5.2 Validation

Common input situations can be validated with special validation controls. All of these controls implement the same interface so the page is able to check the integrity of the relevant input fields on each critical post back. Validation controls are written along with all other controls and refer to the validated controls by their ID. Further, it is possible to implement custom validation controls for all business logic scenarios that could not be covered with the default set of controls. It is possible to define both a server-side validation routine, which is very important for security reasons and also a JavaScript version on the client side, to reduce post backs and enhance convenience.

7.5.3 Advantages

Berardi (2008) describes advantages as follows:

- Great support for WYSIWYG designer. It is an established concept and supports rapid application development.
- View State and Post Back forces behavior like programming a desktop application.
- There are many built in controls and a lot of third party control libraries

7.5.4 Disadvantages

According to Prajapati (2009), disadvantages of the Web Forms framework are:

- The JavaScript driven post back concept requires additional bandwidth.
- The complex view state (most of the time far more values than needed) decreases the performance.
- There are many situations where pages produced by Web Forms framework are not search engine friendly.

7.6 Asp.NET MVC

Inspired by Java and with much request from ASP.NET communities Microsoft has released a new web development framework that does not ship with the .NET Framework main distribution but can be downloaded separately.

MVC is the abbreviation for Model-View-Controller and describes a framework design pattern with three essential component roles:

- A model is responsible for representing the current state and the business logic of an application. It is intended to be persisted to permanent storage facilities such as databases.
- The view defines the user interface and is not responsible for manipulating the data model directly.
- The responsibility of the controller is to handle the end-user interactions with the model and selects which view has to be shown after an operation has finished.

Communication between the ASP.NET MVC components is completely interface driven. That means testing can be done without launching a web server process and model and controllers can completely get decoupled from the web scenario. It further supports a configurable URL mapping mechanism that allows defining clean request parameters which directly invoke specified methods. As a result, the method defines which view has to be displayed next. On the other side, the view can access result objects from the method to display the appropriate content. Because of the interface design, a view is always abstract to the controller. For legacy reasons it is possible to use classic ASPX pages for the view role (Guthrie, 2007).

7.6.1 Input Validation with Asp.Net MVC

Since the controller interacts directly with the web request, business logic validation has to be performed or at least triggered inside the handler method. That means also that it is more complex to pass appropriate and rich error feedback back to the view. The object which failed validation in the model should be passed back to the same view as before including a set of error information.

Microsoft has released an extension called Validator Toolkit for ASP.NET MVC which is capable of handling client-side and server-side validation logic. The framework manages each group of validation elements which belong to one form in a specific class that inherits from a basic validation set class. This class has to implement all built in and user defined validation rules. Afterwards the view and the controller get annotated by a custom attribute that associates the validation set with the application. Nevertheless, at this time it seems that the concept has some problems. Referring to JavaScript method names slightly breaks the concept of MVC. It is worth to use it while no alternative exists and to look forward to a better approach (Baeurle, 2009).

7.7 Conclusion

Today there is a huge market for web frameworks and different application layer approaches in both worlds Java and .NET Framework. Since most open source technologies have been ported between those platforms it is not that important to show differences between them but to show differences between several design approaches. Dealing with all open source and commercial frameworks that lay on top of ASP.NET would have gone far beyond the scope of this paper. As a result, the last chapters should have given an overview of the core components of the application environment and comparing the two central Microsoft concepts. It is not officially clear at this time which of these concepts would last for the next decade. According to Microsoft, both concepts will be shipped for the upcoming years. Sure, at this time dealing with that amount of different extensions and frameworks it is hard to keep compatibility between these systems, not only in domain of Microsoft technology.

8 Client Environment

8.1 General Technologies

Considering that a peer assessment system has to deal with a great variety of user groups HTTP and HTML seem to be the appropriate technologies to provide such an application. A short overview about those main technologies is given in the following chapter.

8.1.1 HTTP

HTTP means Hypertext Transfer Protocol and it is an application-level protocol which offers the speed and lightness that is necessary for a collaborative, hypermedia information system such as a modern peer assessment online system. It has been in use since 1990 and is based on a simple request/response paradigm. This means that the client connects to a server and sends a request message in a special form. The server answers with a response message that contains information which tells the client if the message was successful or not.

The World Wide Web Consortium (2009) also defines the most important terms in the HTTP context:

- Connection is the transport layer between two programs
- Message is the basic unit of a communication with a special syntax
- Request is a request message as described above
- Response is a response message as described above
- Client is a program which connects to a server to request services
- Server is an application program which offers services for clients
- Proxy is a program which acts as a server and a client
- Gateway is a server which acts as a middleman for some other server

8.1.2 HTML

HTML means Hypertext Markup Language and is used to create structured documents which represent web pages in an easy way. It was developed by Tim Berners-Lee in the year 1990 and is an improved version of the Standard Generalized Markup Language (SGML). Such documents are

divided into two main sections: First, the head section which contains the title and other information and second, the body which contains the content of the web page (World Wide Web Consortium, 2009).

8.2 Client Applications and Standards

An important fact of the Hypertext Markup Language is that it can load scripts in languages such as Java Script and appearance definitions in languages such as CSS which are processed by a web browser. These components will be described in the following chapter.

8.2.1 Browser

The web browser is a client application which connects to a web server to retrieve and show information resources on the World Wide Web. These resources can be web pages, videos, pictures or other sources of information. Accordingly, the main purpose of a web browser is to bring information that is mostly coded in HTML via HTTP to the user. The first web browser was developed by Tim Berners-Lee in the year 1990 and was called WorldWideWeb. 1993 the browser Mosaic was developed which was later refined to Netscape Navigator and Mozilla. The most used web browser, Internet Explorer, was introduced 1995 by Microsoft with their Windows 95 operating system (Steward, 1996).

A well known problem with web browsers that especially affects online systems is the problem of the compatibility between the different browsers. This is because actually a browser is only a translator and the result of this translation is like giving two humans a text in one language and let them translate it into another language. Probably they will produce the same meaning but expressed in different words. So it is really important to test web pages on different browsers to identify possible compatibility problems (Dahm, 2009).

8.2.2 Java Script

JavaScript is an object-oriented scripting language which is primary used to create programs or functions which run on the client side within an integrated component of the browser. Therefore, it is possible to include JavaScript code directly into HTML documents to create dynamic web pages. The big advantage is that JavaScript code runs directly on the user client side and not on the web server. So it is possible to validate the inputs of web forms before they are submitted to the server. Another good example for the use of JavaScript in such a peer assessment application is the idea of

the coloured annotations which was mentioned in the 5th chapter. The users can make their annotations and correct them without creating a post back (JavaScript, 2009).

8.2.3 CSS

CSS means Cascading Style Sheets and is a simple mechanism for adding style to documents, e.g. HTML or XML documents. This helps the developer to fulfil the principle of separating the document content from its style. The content is coded in one document, e.g. a HTML document, and the style is defined in a CSS file (World Wide Web Consortium, 2009).

8.3 Conclusion

This chapter has shown that the frameworks and ideas of the web are quite old; the main principles have even been developed nearly 30 years ago. Nonetheless, it is still possible to build modern and flexible web-based systems when combing them with client technologies and applications like JavaScript and browsers that support it.

9 Conclusion

9.1 Summary

To sum this up, peer assessment can be seen as an assessment type that fulfills both types of assessment, summative and formative assessment. This means that it can not only be used to determine grades but also to provide useful feedbacks created by students to help some other students. It is in the nature of things that the evaluation of many student evaluations needs time and so the peer assessment idea can be perfectly combined with methods of technology enhanced learning, especially e-learning and other forms of online peer learning. This combination defines the term of peer assessment in computer science and also clears the way for building a modern and flexible e-learning system around this approach. This can be implemented with classical three-tier software architecture with the use of special components in the fields of the data model, the application environment and of course the client environment.

9.2 Outlook into the Future

As an outlook into the future of peer assessment in computer science it can be assumed that the main idea and the main usage of peer assessment will not change. The research goals will focus on ways to improve the methods of creating and displaying the results of the formative assessment process. Some possible ideas for this, like the annotation markings, were already mentioned in the 5th chapter. Another goal of future developments will be a more flexible software design which supports different assessment and exam types and also more social aspects of the Web 2.0 approach. Lastly, the main challenge will be to make the formative assessment process as comfortable as it is in direct classroom assessments.

10 References

About-eLearning (2009). History of e-Learning. Retrived 2009, August, 26 from http://www.about-elearning.com/history-of-e-learning.html

Al-Smadi, M., & Guetl, C. (2008, September). *Past, Present and Future of e-Assessment: Towards a Flexible e-Assessment System*. Paper presented at the Interactive Computer Aided Learning Conference, Villach, CA.

Al-Smadi, M., Guetl, C., & Helic, D. (2009, April). *Towards a Standardized e-Assessment System: Motivations, Challenges and First Findings*. Paper presented at the International Conference on Interactive Mobile and Computer Aided Learning, Amman, JO.

Aranda, N. (2007, March 20). *A Brief History of E-learning and Distance Education*. Retrieved August 26, 2009, from http://ezinearticles.com/?A-Brief-History-of-E-learning-and-Distance-Education&id=496460

Baeurle, J. (2008, February 11). Creating Client and Server-Side Form Validation using the Validator Toolkit for ASP.NET MVC. Retrieved September 24, 2009, from http://www.codeproject.com/KB/aspnet/MvcValidatorToolkit.aspx

Barry, D. K. (2009). Object-relational mapping articles. Retrieved September 15, 2009, from http://www.service-architecture.com/object-relational-mapping/articles/index.html

Berardi, N. (2008, December 29). Introducing the ASP.NET MVC (Part 2) – ASP.NET MVC vs. ASP.NET Web Forms. Retrieved September 24, 2009, from http://www.coderjournal.com/2008/12/introducing-aspnet-mvc-part-2-aspnet-mvc-vs-webforms/

Bhalerao, A., & Ward, A. (2001, January 12). *Towards Electronically Assisted Peer Assessment: A Case Study*. Retrieved September 10, 2009, from http://www.dcs.warwick.ac.uk/~abhir/research/papers/new-alt-j.pdf

Bostock, S. (2000). Student peer assessment. *Learning Technology by Stephen Bostock*. Staffordshire, West Midlands: Keele University. Retrieved November 22, 2006, from http://www.keele.ac.uk/depts/aa/landt/lt/docs/bostock_peer_assessment.htm

Brown, J. S. (2008). Minds on fire: Open education, the long tail, and learning 2.0. *Educause Review*. Boulder, California. Retrieved February 2008, from http://www.educause.edu/EDUCAUSE+Review/EDUCAUSEReviewMagazineVolume43/MindsonFireOpenEducationtheLon/162420

CATME, (2009). Comprehensive Assessment of Team Member Effectivness. Retrieved September 27, 2009, from https://engineering.purdue.edu/CATME/index.htm

Christudason, A. (2003). Peer learning. *Successful Learning*. Singapore: University of Singapore. Retrieved September 16, 2009, from http://www.cdtl.nus.edu.sg/success/sl37.htm

Cockburn, A., & Williams, L. (2000). The Costs and Benefit of Pair Programming. *Proceedings of the First International Conference on Extreme Programming and Flexible Processes in Software Engineering.* Genova, Italy.

Dahm, T. (2009). *Browser compatibility tutorial*. Retrieved September 19, 2009, from http://www.netmechanic.com/products/Browser-Tutorial.shtml

DuPaul, G. J. (1998). Peer tutoring procedures in general education classrooms. In A.S. Canter & S.A. Carroll (Eds.), *Helping children at home and school: Handouts from your school psychologist*. National Association of School Psychologists, Bethesda: MD.

Earl, L. (2003). *Assessment as Learning. Using Classroom Assessment to Maximize Student Learning*. Thousand Oaks, California: Corwin Press.

Guetl, C. (2007). Moving towards a Fully Automatic Knowledge Assessment Tool. Paper presented at the International Conference on Interactive Mobile and Computer Aided Learning, Amman, JO.

Guetl, C. (2009). *Enhancements of peer assessment in computer based learning environments*. Manuscript submitted for publication.

Guthrie, S. (2007). ASP.NET MVC Framework. Retrieved September 24, 2009, from http://weblogs.asp.net/scottgu/archive/2007/10/14/asp-net-mvc-framework.aspx

JavaScript. (2009, September 21). The definitive JavaScript resource. Retrieved September 21, 2009, from http://www.javascript.com/

Kandlbinder, P. (2009). Peer assessment. *Involving Students in Assessment*. Sydney, Australia: University of Technology Sydney. Retrieved March 13, 2009, from http://www.iml.uts.edu.au/assessment/index.html

Kay, J. (2007). *Peer Assessment and Self Assessment in Student Work* [PDF document]. Retrieved from Teaching Quality Assurance Manual Online Web site: http://admin.exeter.ac.uk/academic/tls/tqa/

Kim, M. (2008). *Peer assessment as a learning method*. Saarbrücken, Saarland: VDM Verlag Dr. Müller.

Kuehnel, A. (2009). *Visual C# 2008. Das umfassende Handbuch*. Bonn: Galileo Computing

Matthews, R. (2009). The use of peer assessment to improve student journalists' court reporting skills. *Art / Design / Media Case Studies*. Birmingham, West Midlands: Coventry University. Retrieved January 2009 from http://www.adm.heacademy.ac.uk/resources/case-studies/the-use-of-peer-assessment-to-improve-student-journalists-court-reporting-skills

Mössenböck, H. (2003). Introduction to C#. Retrieved September 24, 2009 from http://www.ssw.uni-linz.ac.at/Teaching/Lectures/CSharp/Tutorial/Part1.pdf

MSDN Library (2009). Microsoft Developer Network. Retrieved September 22, 2009, from http://msdn.microsoft.com/en-us/library/default.aspx

MySQL. (2009, July 6). The world's most popular open source database. Retrieved July 6, 2009, from www.mysql.com

Nichols, M. (2008). E-Learning in context. *E-Primer Series*. New Zealand: Laidlaw College. Retrieved August, 2008 from http://akoaotearoa.ac.nz/sites/default/files/ng/group-661/n877-1---e-learning-in-context.pdf

Patterson, J. (2009). Simple Object Persistence with the db4o Object Database. Retrieved September 15, 2009 from http://onjava.com/pub/a/onjava/2004/12/01/db4o.html

Prajapati, C. (2008). ASP.NET MVC vs. Web Forms - Advantage Disadvantage. Retrieved September 24, 2009, from http://blog.zoomasp.net/?p=32

Phutela, D. (2006). Hibernate Vs JDBC. Retrieved September 15, 2009 from http://www.mindfiresolutions.com/mindfire/Java_Hibernate_JDBC.pdf

Rhem, J. (2003). *Classroom Assessment Techniques*. Retrieved from The National Teaching & Learning Forum Online Web site: http://www.ntlf.com/

Roberts, T. S. (2006). *Self, peer and group assessment in e-learning*. Bundaberg, Queensland: Information Science Pub.

Rowland, F. (2007). The peer review process. *A Report to the JISC Scholarly Communications Group*. Leicestershire, East Midlands: Loughborough University. Retrieved at September 18, 2009 from http://www.jisc.ac.uk/uploaded_documents/rowland.pdf

Steward, W. (1996). *Web browser history*. Retrieved September 19, 2009, from http://www.livinginternet.com/w/wi_browse.htm

TLRP (2009). Teaching and Learning Research Programme. Retrieved August 25, 2009, from http://www.tlrp.org/tel/current_projects.html

WebPA. (2009). An Online Peer Moderated Marking Tool for Group Work. Retrieved August 26, 2009, from http://www.webpaproject.com/

Winkels, Maarten (2009). Advanced Hibernate: Proxy Pitfalls. Retrieved September 14, 2009, from http://blog.xebia.com/2008/03/08/advanced-hibernate-proxy-pitfalls/

World Wide Web Consortium. (2009, September 19). Leading the web to its full potential. Retrieved at September 19, 2009, from http://www.w3.org/

Implementing a Flexible Peer Assessment System with Microsoft .NET:

Software Documentation and Manual

Practical Part of Bachelor's Thesis
at
Graz University of Technology
submitted by
Dominik Kowald and Joachim Maderer

Supervisor:
Dipl.-Ing. Dr. techn. Christian Gütl

Institute for Information Systems and Computer Media (IICM)
Graz University of Technology
A-8010 Graz, Austria

02.10.2009

Abstract

Actual E-Learning Systems usually only provide assessments of natural language answers by teachers or tutors and don't take notice about other assessment types like self assessment or peer assessment although this variants would have a great benefit, not only for teachers but also for students.

Students would have the chance to get into the role of the teacher and correct the answers of their colleagues. Furthermore they have to get deeper into the topic to make good corrections and so they have a higher learning success. Teachers and tutors would have the great advantage to save the time they would have needed to correct all the answers by themselves and to test a new assessment method in computer science.

To implement such an online e-learning system which supports the peer assessment idea, the Miscrosoft .NET framework seems ideal. It supports easy methods for creating web applications using asp.NET and runs fully on UNIX machines using the MONO framework.

This paper should give an overview about the implementation process of such an online peer assessment system. This also includes the used technologies, class hierarchies, the data model, the main programming ideas and a manual.

Content

1 Introduction

1.1 Motivation

The importance of the lifelong learning process is increasing from year to year. This means that there has to be a way of providing feedback for learning activities in an easy, significant and also correct way. The classic assessment process in which a teacher or tutor provides feedback and grades for a group of students satisfies these needs but there is also an aspect to think about: If students need more significant feedback from year to year to get the know-how that is necessary in economy, also more teacher and tutors are needed and this means higher cost.

This problem tends to the idea of peer assessment which is the assessment of students by other students to provide feedback and grades. With this assessment method it is possible to reach two goals: The workload of teachers and tutors decreases because students provide their feedbacks among each other and the second advantage is that students also increase their educational skills while doing this. On the contrary a high amount of student feedbacks is needed to get an evaluation which is very close to a teacher or tutor evaluation and so this means lot of administrative effort. To solve this problem peer assessment could be included in e-learning systems where the calculations of these evaluations can be made in microseconds.

To test the efficiency of peer assessment in computer science a prototype has been developed with the use of the .NET framework. This makes it possible to comfortably use it during an online test with a group of students to evaluate how good the peer assessed results match the teacher assessed results and also how many students and evaluations are needed to produce useful results.

1.2 Structure of the Paper

This paper consists of two main parts: First the documentation of the project and the software development process and second the manual of the completed system.

The first part consists of the chapters 2 and 3. Chapter 2 describes the project documentation including the user requirements document, the project schedule and also some early mockups of the system to show how the first design and usability ideas had been.

Chapter 4 implies the second part and contains the manual of the system including screenshots and descriptions of all functionalities. The 5th and last chapter is the conclusion and acts as a summary, an outlook into the future and some statements about the lessons that were learned while implementing this prototype.

1.3 Related Work

The practical use of e-assessment and peer assessment in computer science is an important component in modern e-learning systems. Al-Smadi and Guetl (2008) presented the past, present and future of e-assessment and have showed that technology is very important for the improvement of assessment. Guetl (2009) also described the use of peer assessment in computer science within an experiment that was performed at the Graz University of Technology. This experiment was based on a prototype that is very similar to the one that is described in this paper. Bhalerao and Ward (2001) defined some important design requirements for an online assessment system and also presented a self developed online assessment prototype.

Furthermore there also exist some modern peer assessment tools that are used in university courses including WebPA, CATME, Mni-PAT and OASYS (Bhalerao & Ward 2001, CATME 2009, ISCP 2009, WebPA 2009).

2 Project Documentation

2.1 User Requirements

This chapter contains the user requirements document which was created at the beginning of the project to determine all necessary functionalities and also points for future works.

2.1.1 Primary Components

- User Management and Login
- Question Catalogue
- Test Creation and Examination
- Reference Answers (online research)
- Peer Assessment Process
- Feedback (Examination Results)
- General GUI Issues

2.1.2 User Management and Login

- Authentication
    - Exchangeable Interface (external API Communication, e.g. TUG-Online)
    - Integrated simple user database (username and password)
    - User accounts can also be created by students on their own (teacher should be able to configure)
        - Required data for login: matriculum number, name and e-mail address
        - In the database the whole user data is saved of course (e.g. the attended courses or the assigned exams and so on)
- Assignment to roles (student, tutor [study assistant], teacher, administrator)
    - Possible per course, respectively question catalogue (e.g. Tutor for one course and normal student in another course)
    - Cumulative role system (role greater or equal to gain access – e.g..: tutor is allowed to do everything what a student can do + more)
        - Tutors for example can participate in an examination for system test purposes
- Administration area to manage users and courses
    - E.g. adding new accounts, assign examinations, courses and departments (special fields)
    - Administrator is allowed to use all features, teachers only within their areas (courses)

2.1.3 Question Catalogue (Record Creation and Management)

- Add new questions for later use in concrete examinations
- Assign them to different subjects (courses, respectively special fields)
    - Maybe chapters or sub-groups for fine tuning
- Different Question Types
    - Currently only short natural language answers
        - Possibly to enter a reference answer (sample solution)
    - Extendable, respectively exchangeable (e.g.: multiple choice, single choice, gap text, short essays, mathematic tasks)
        - Implemented as flexible, exchangeable module system

2.1.4 Test Creation and Examination Process

- Instantiation of a Test Object
    - Select previous entered questions
        - maybe through selection of special fields and chapters
    - Determine time slots and max. examination time
    - Determine student-groups, who are allowed to participate
        - a student group is a set of students that have something together (e.g. they attend the same course)
    - Additional settings, if logical possible
        - These settings are only available if their determination make sense (e.g. if you have a multiple-choice exam it makes no sense to enter settings for a peer assessment process
        - Determine Assessment-Process (peer, automatic, etc.)
- Examination phase for students
    - Learning phase should last longer according to prototype evaluation
    - Online test for each student
    - Status of each question can be saved through a button, if there is a fallout of internet connection or system
        - this means if the user answers a question and clicks the button to go to the next question, his answer will be stored and the status of his exam can be reloaded if the connection to the system is lost
    - The exam can be resumed immediately
    - Student also has to enter his self-confidence (1 – 10)

2.1.5 Sample Answers (Online Research)

- Begin date + duration configurable
- It should maybe be configurable how many questions of the test have to be researched (see assessment process) because this can be only a subset of the questions

2.1.6 Peer Assessment Process

- Can be performed by tutors and students
    - Tutors have to evaluate more answers per question (just a subset of the question or max. all)
- Special answers for calibration
    - A subset of the answers is already evaluated and selected for the peer assessment process
- Parameter
    - Separation of the test in real questions and questions for calibration
    - How many students have to evaluate one and the same answer
        - This determines the number of Assessment items per student
- Teacher can choose if it should be possible that student could evaluate their own answers
- Marking parts of the text with a mouse (3 states)
    - Irrelevant
    - Correct
    - False
- 2 additional text fields: 1 for remarks and comments and 1 to say which parts of the answer were missing
- Later it should be possible to type in HTML answers (e.g. formatted text) and formulas but marking the answers should also be possible without changing the format of the answer durable.

2.1.7 Feedback (Examination results)

- Show the whole examination again with the Assessment details (Test performance + performance)
    - All assessments and mean values + self confidence
- Quality of the own assessment – Performance of the Peer Assessment process
    - How did your assessments fit the calibrated answers
- Final Grade
    - Is calculated with a replaceable algorithm
    - Grade = k1 * Self-Assessment + k2 * Online-Test-Performance + k3 Peer-Assessment-Performance

- Extensible
    - e.g. show how your answers fit you reference answers
    - 2 Feedback versions for student and teacher
        - Detailed version for student
        - Compact version for teacher
        - But teacher should also be possible to view the assessments of the tutors
    - Feedback for tutors
        - so they can see how good their assessment was in comparison to the other tutors

2.1.8 General GUI issues

- Everything is in English
    - but the language information should be loaded from a so called language file to let the system in the future also work in other languages
- Flexible welcome screen which shows the available options depending on your role (teacher, student …)
- Status bar which shows how many questions or assessments you have left
    - Already answered questions will be specially marked (also already done assessments)
- Flexible navigation bar at the left side
- Appealing, user friendly design with tabs and other modern widgets

+ System should safe the times in which the students are logged in (or doing examinations or doing peer assessments) for statistical issues

+ the general 5 steps that will be performed with the system:

1.) Learning
2.) Online Test + Self Assessment
3.) Reference Answers (with access to the lecture notes and to net web)
4.) Peer Assessment
5.) Grading and Feedback

Students will perform all 5 tasks, Tutors only Tasks 3 - 5

2.2 Project Schedule

The whole project was developed in the year 2009 and the following listing should give a short overview about the specific project phases

Project Idea

The project idea was defined in January with the help of Professor Christian Guetl who has also been the supervisor of this project.

Project Start and Analysis

The project start was in March and contained the analysis phase and the creation of the user requirements document.

Design

The design phase was executed in April and included a refinement of the user requirements document, the creation of mock-ups to specify the graphical user interface and the use cases and definition of the database design.

Implementation

The implementation with Java started in May using Netbeans 6.5, the Hibernate framework and IceFaces but was stopped after two weeks because of several problems (see Lessons Learned chapter). In succession the .NET implementation was started in the mid of May and ended in August.

Test, Documentation and Approval

In September the system was tested and the documentation was written. Also a Linux test system was set up using Ubuntu 9.04 and Virtual Box to perform the approval.

2.3 Early Mock-Ups of the System

This mock-up shows the first idea of the assessment screen.

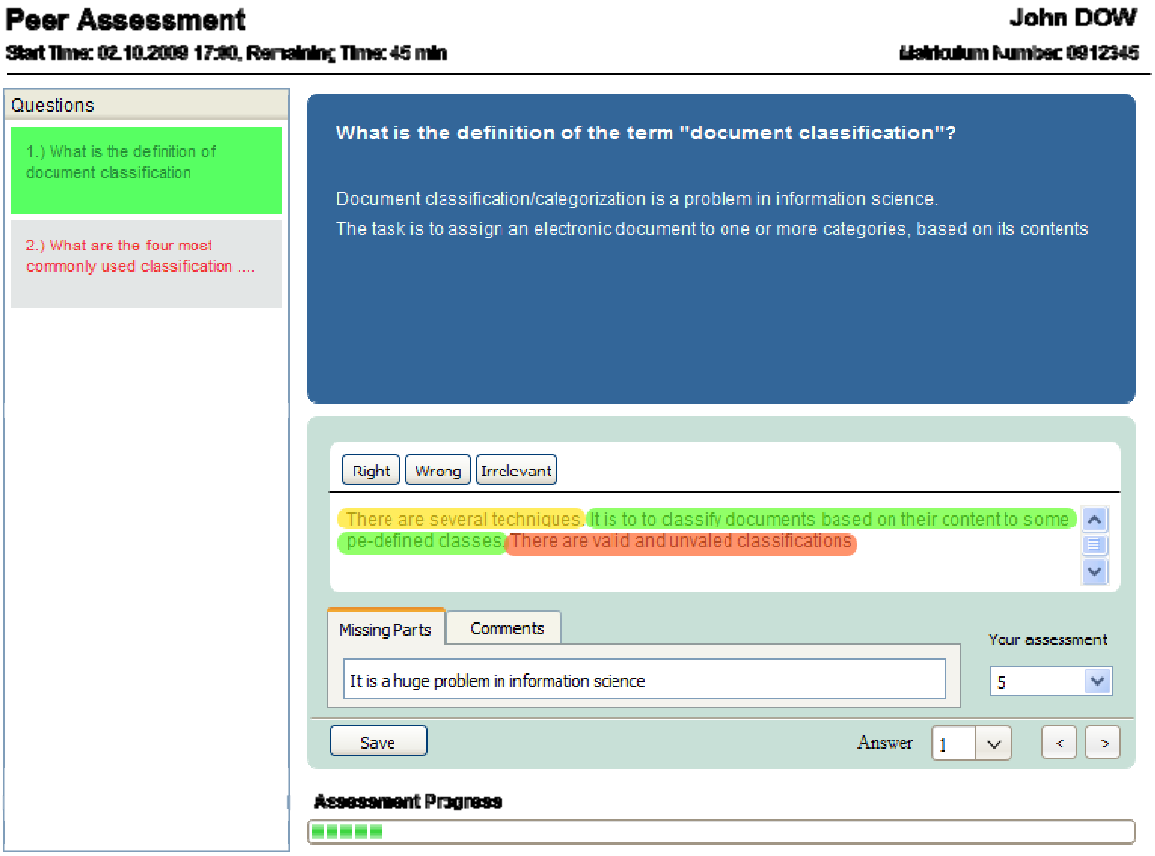

Figure 1: Early mockup of the peer assessment screen

When comparing this early mock-up in Figure 1 with the final version of the program, it can be seen that the style remained the same. This means that the .NET framework provides many possibilities to create a user-friendly and modern graphical user interface design.

3 Software Documentation

3.1 System Architecture

This architecture diagram in figure 2 shows the main components of the system which are described in the next chapter.

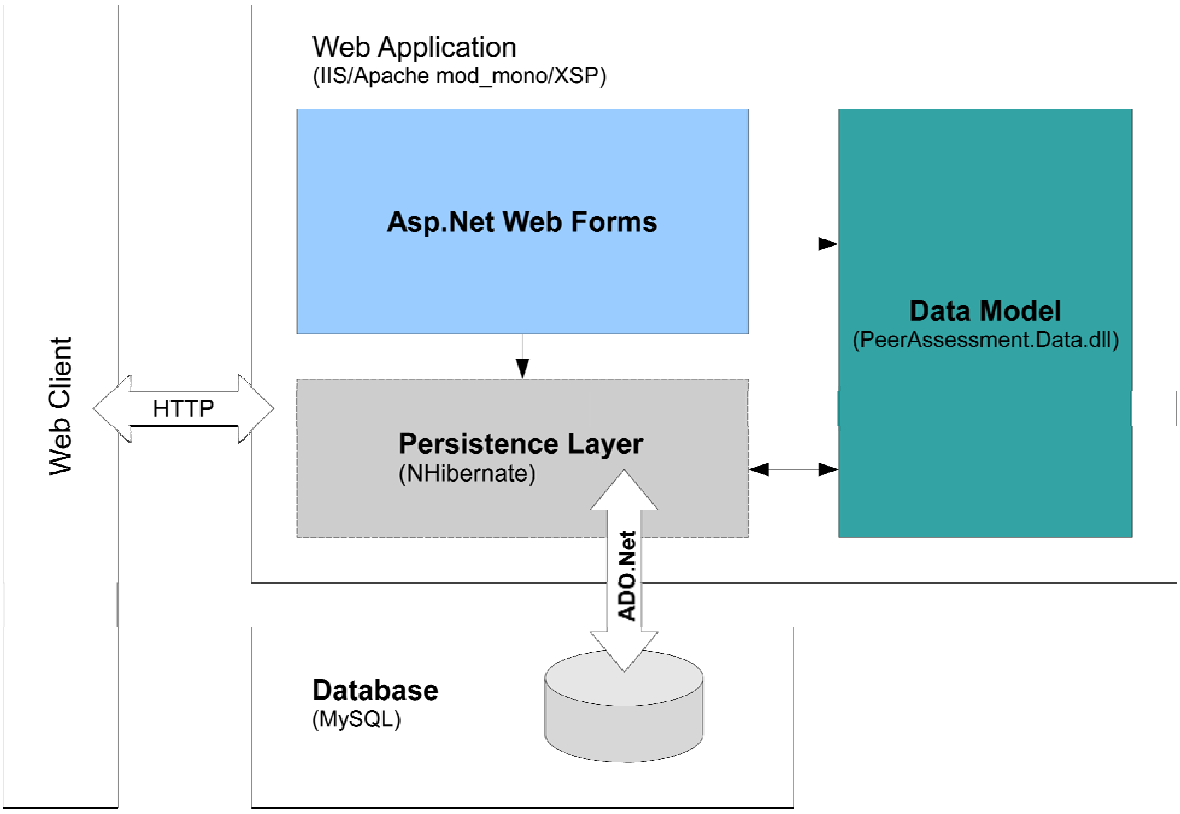

Figure 2: System architecture

3.2 Used Technologies and Frameworks

This chapter acts as a listing of all used technologies, frameworks and tools

MySQL

MySQL is a relational database management system. This project uses the community edition which is available as free open source software.

NHibernate

A .NET Framework port of the Java persistence framework Hibernate. This framework maps the relational data model that is stored in the MySQL database to plain .NET objects. Further it provides independent query definitions which are decoupled from SQL.

There are several dependency libraries, i.e. for creating dynamic proxies, which ship together with the NHibernate download package.

MySQL Connector.Net

The MySQL database connector for .NET is used to establish a connection between NHibernate and the MySQL database server.

ASP.NET

As a web application environment, ASP.NET handles all incoming requests and delegates them to the appropriate implementation of the attached web framework.

Web Forms

The Web Forms framework is a central part of the official .NET Framework and ships together with ASP.NET. It provides a WYSIWYG compatible development interface that follows the same standards and guidelines as the Window Forms Framework that is used for desktop applications.

JavaScript

JavaScript is a client side scripting language that could be embedded in HTML. The client implementation which is responsible for annotating natural language answers makes extensive use of JavaScript and the Document Object Model (DOM). Also the Web Forms framework uses JavaScript for some of its integrated functionality.

Mono

Mono as an open source implementation of the Common Language Runtime (CLR) is used to deploy the prototype application on the future production system. It was tested on Mono during the development phase. Due to compatibility issues the application is laid-out for .NET Framework version 2.x.

Microsoft Visual Web Developer 2008 Express Edition

The express edition is a freely available integrated development environment (IDE) that offers a visual Web Forms designer and full code completion features. Beside some limitations like integrated testing system and cooperation services, it is a fully operational environment that could even be used for commercial projects.

The web developer was used to manage the whole solution including the separate data library that defines the business logic for NHibernate.

Firebug

Firebug is an add-on for Mozilla Firefox. It offers functionality to debug and edit JavaScript, Cascading Style Sheets and HTML while browsing any website.

3.3 Data Model

The data model is divided into two main components, the assessment side and the question pool side. The two components are connected with cross-tables between the question class and the user test, user research and user peer assessment classes. For the database work the MySQL GUI Tools (Administrator and Query Browser) and the MySQL Workbench were used.

3.3.1 Assessment Side of the Database

This diagram shows all tables/classes that are needed to maintain the course, exam, test, research and assessment processes.

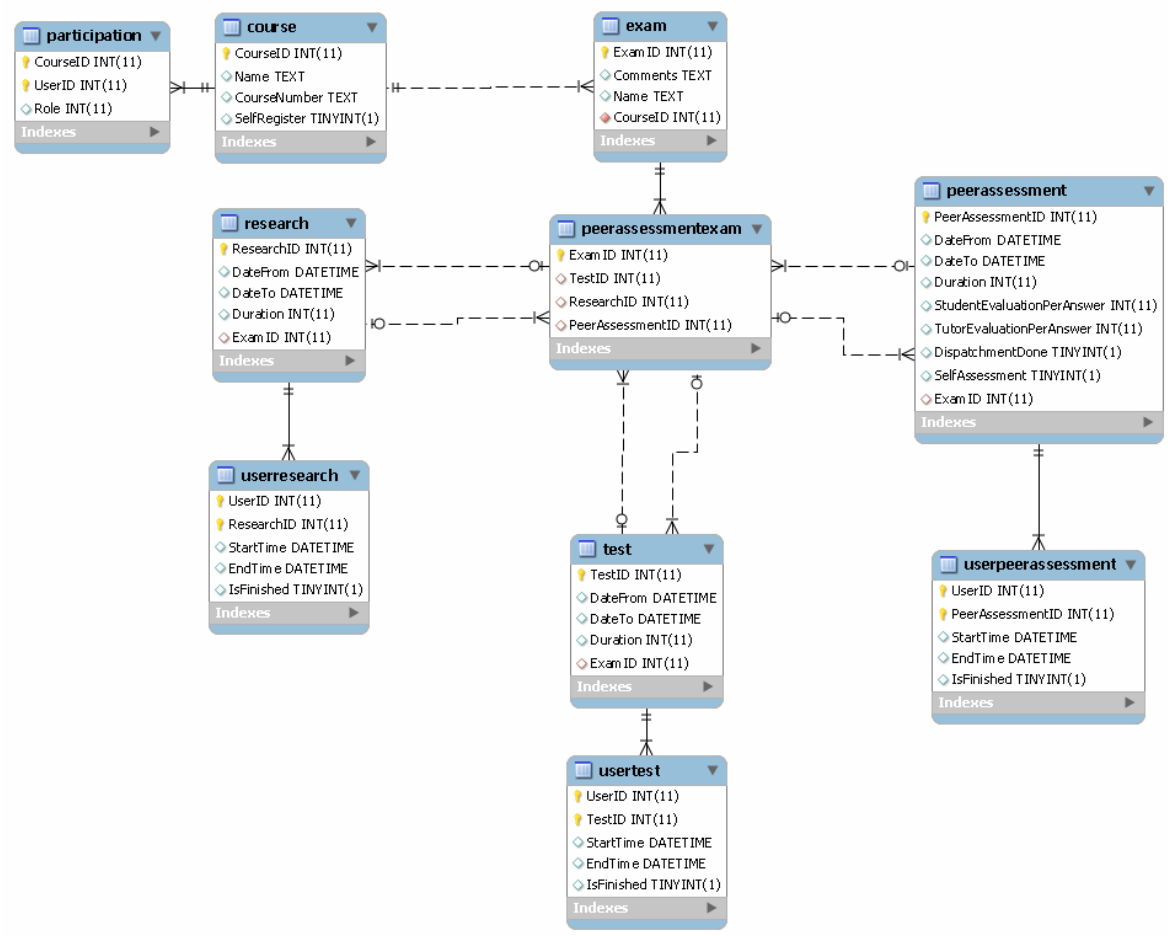

Figure 3: ER model of the assessment side

3.3.2 Question Pool Side of the Database

This entity relationship diagram in figure 4 shows all tables / classes that are needed to maintain the question, answer and evaluation pools.

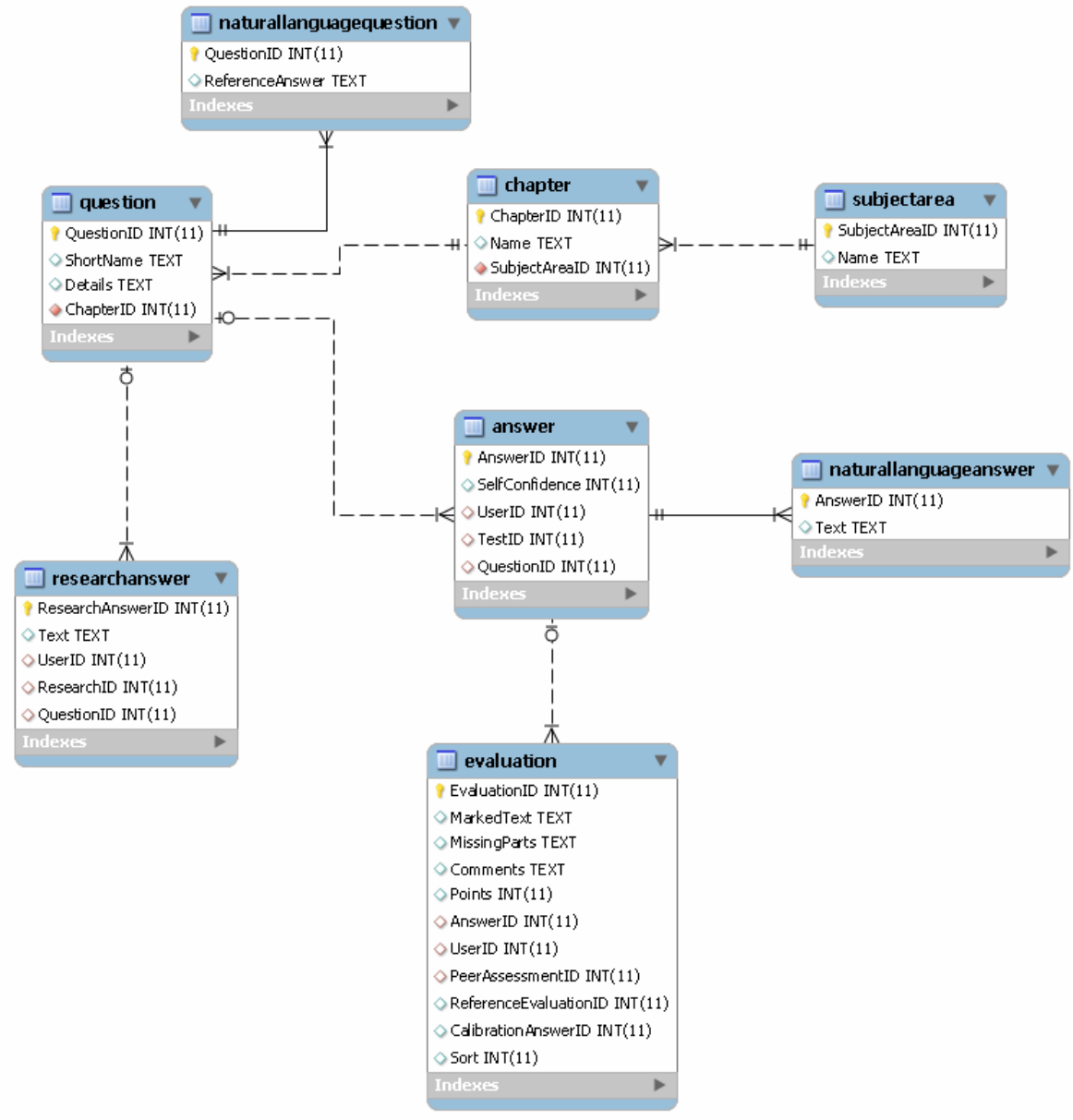

Figure 4: ER model of the question side

3.4 Implemented Algorithms

The following sections describe the two logical core features of the application. All other parts are based on common Web Forms scenarios that could easily be understood by an ASP.NET programmer without further instruction. Finally it describes the used performance classes for the peer assessment performance.

3.4.1 Client Annotation

Based on the Document Object Model (DOM) and selection features of the browser (at this time only Mozilla based clients are supported) a collection of script files was written to implement the ability to select a specific text part and apply a colored annotation.

While the selection object already features a function to wrap text with a specified document node, it was necessary to keep the integrity of the annotated structure and provide an easy way to overwrite parts that were selected before.

Following steps describe the annotation process after selecting a piece of text with the cursor, inside of the declared container element:

- Take the selected range form the browser and walk down the selected sub tree from the first node to the last node
- Create logical sub-selections for each involved block element (i.e. paragraphs)
  Hint: this step is not implemented yet but it is intended for future improvements that allow basic html based answers including paragraphs.
- Surround the sub-selections with a newly created node
  - Use the current block element (or at this time just the clear text container) and break the first node of the selection apart as well as the last node. That means creating a deep copy sub-tree of the node and separating the text content of the deepest child node.
  - Wrap all nodes between the resulting gap with a newly created reference node which holds the annotation or style information.
  - Neutralize all deprecated annotation nodes within the new annotation node.
  - Optimize the structure of the whole block element or container by joining adjacent nodes that share the same name and set of style properties (also including original html tags within the answer).
- Collapse possible remaining selection ranges to improve visual appearance.

Finally, an XML writer class takes the annotated structure and transforms it to a simple XML based annotation definition, which features "true", "wrong" and "irrelevant" as annotation types. In inverse direction an XML reader class creates a DOM representation again with a hidden text area as data source.

3.4.2 Evaluation Distribution Algorithm

One of the most problematic features is to distribute random evaluation tasks to participating students. Such an algorithm claims for following properties:

- Provide acceptable time of execution (desirable would be nearly just-in-time)
- Ensure that no student ever gets his own answers to evaluate
  Remark: this claim should not be confused with the additional and decoupled option to evaluate own answers.
- Each student should get the same number of evaluations per question

The previously described considerations have led to this algorithm (per question):

- Select all participants (users) of the current exam and all answers for the given question
- Create a hashed lookup-table ("account") that maps a user object to an integer. The integer value indicates the number of already assigned evaluation objects per user.
- For each user and answer create as many raw evaluation objects as determined by "student evaluations per answer"
- Shuffle the list of created evaluation objects
- Do the following n times ("student evaluations per answer"), and for each user
  - Iterate through all raw evaluations until an appropriate combination for the current user is found (remember index)
  - If the index is within the valid range, connect the current user with the evaluation object and move it to "assigned evaluations"
  - Otherwise try to find a suitable swap candidate from "assigned evaluations" and exchange the user association with the current evaluation object. Move the current evaluation object also to "assigned evaluations". It this is not possible either (due to wrong configuration), keep the invalid setting (fail safe to avoid a stuck system).

While this is the complex part, the generator system also incorporates self-assessment creation (for each answer), entire tutor evaluation and evaluations which are associated with calibration answers to calculate the peer assessment performance. Since the creation order of this decoupled components appear linearly, the system generates a random sort number that is used to display the evaluations

during the peer assessment step. As a consequence the participant is not aware of which answer is the calibration answer or self-assessed answer (if the latter option was selected).

3.4.3 Performance Classes

For the calculation of the summative self assessment and peer assessment performance some performance classes were defined by Guetl 2009 based on the results of an experiment.

Class 5: e < 1.1
Class 4: 1.1 <= e < 1.9
Class 3: 1.9 <= e < 2.6
Class 2: 2.6 <= e < 3.1
Class 1: 3.1 <= e < 3.5
Class 0: 3.5 <= e

Theses classes determine how many points a student will get for a specific answer and were defined between 0 and 5. When calculating the self assessment the variable e is the difference between the self confidence value and the online test performance value that was determined with the arithmetic mean value of all received points through the peer assessment process. In terms of calculating the peer assessment performance the variable e is the difference between the points of the calibration answers and again the online test performance value.

4 Manual

4.1 General Functionalities

This chapter shows the general functionalities like the login or registration process.

4.1.1 Login

Figure 5: Login screen

The login screen in figure 5 is the first and only screen that every user will see if he is not logged in already. To log in the user uses his chosen username and password or uses the registration link on the left side to create a new user account. An error message will be raised if the login information were wrong.

4.1.2 Registration

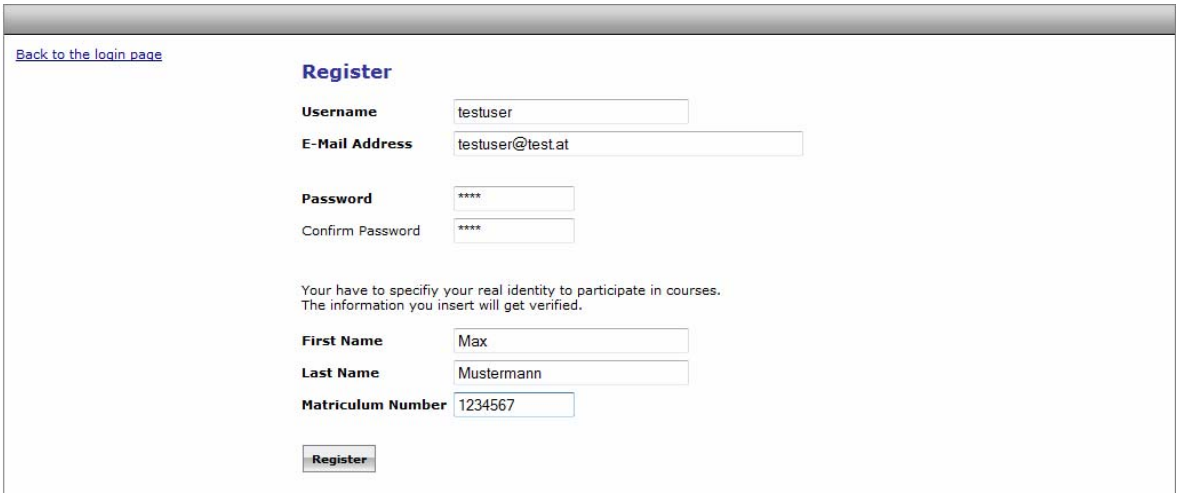

Figure 6: Registration screen

The registration screen in figure 6 is used by a new student to create a user account. Accounts that need access to the question pool or other teacher or tutor privileges have to be created by the administrator.

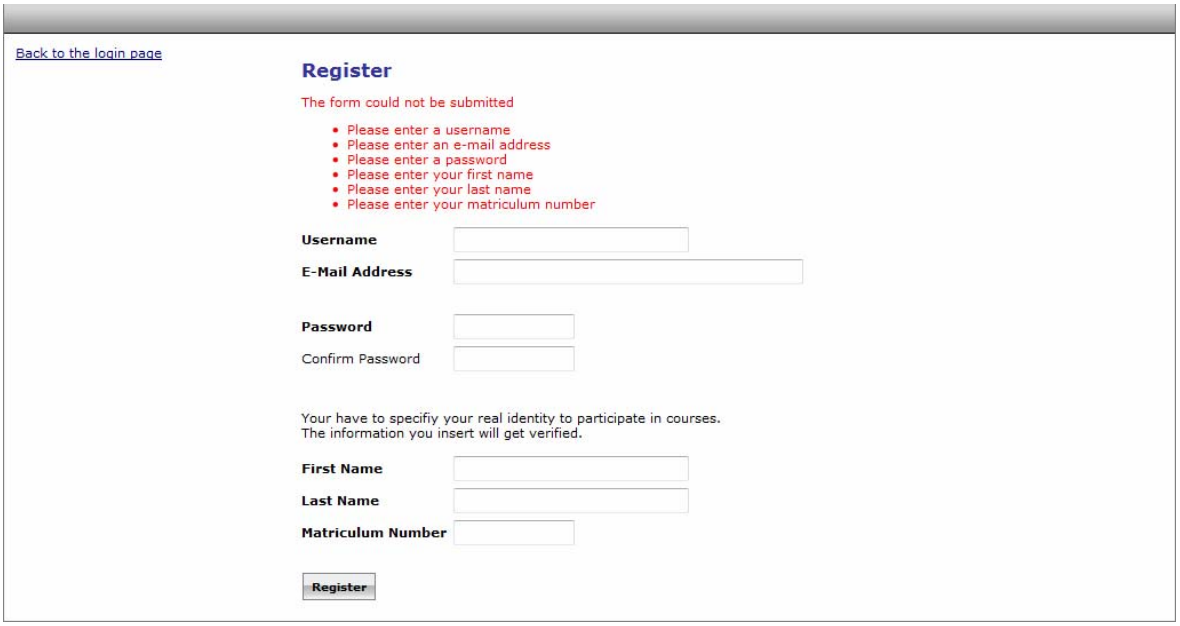

Figure 7: Client validation screen

This screen in figure 7 shows the used client validation if the typed in data does not match the awaited values. This assures that no wrong data will be send to the server. Some sort of client validation is used in every screen where the user has to type in data.

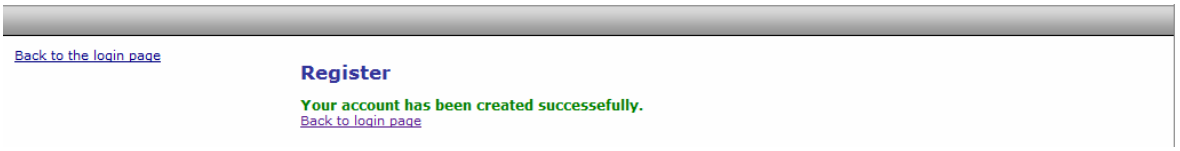

Figure 8: Success message

A success message (see figure 8) will be raised after a successful registration. This shows the user that no errors occurred. Some type of success message is used in every form that executes a database operation.

4.2 Administrator Area

The administrator or super user has the ability to use all functionalities that the system offers. This chapter describes some special administrator screens that differ from the screens of the other user types.

4.2.1 Courses

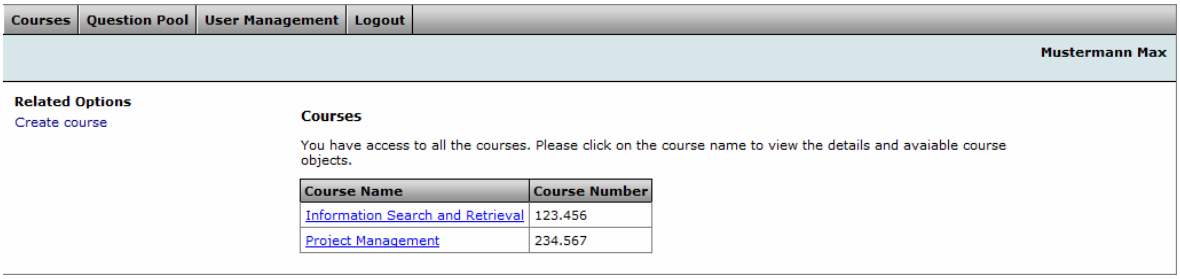

Figure 9: Courses main page for administrator

The administrator has access to all the courses (see figure 9) to manage the exams within them. This screen also shows the menu bar on the top of the screen. These buttons can be used to navigate through the different areas.

4.2.2 User Management

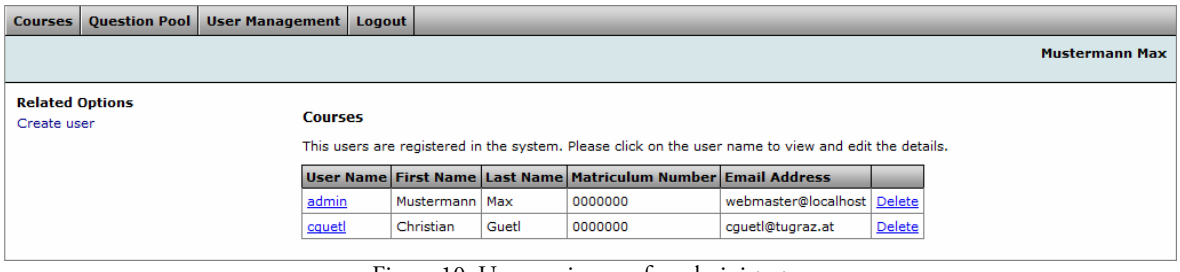

Figure 10: Users main page for administrator

The administrator group is the only user group that has access to the user management area (see figure 10). When clicking on the user name he has the ability to edit the user data. The create user link on the left opens the form to create a new user and the delete link will delete the user from the database.

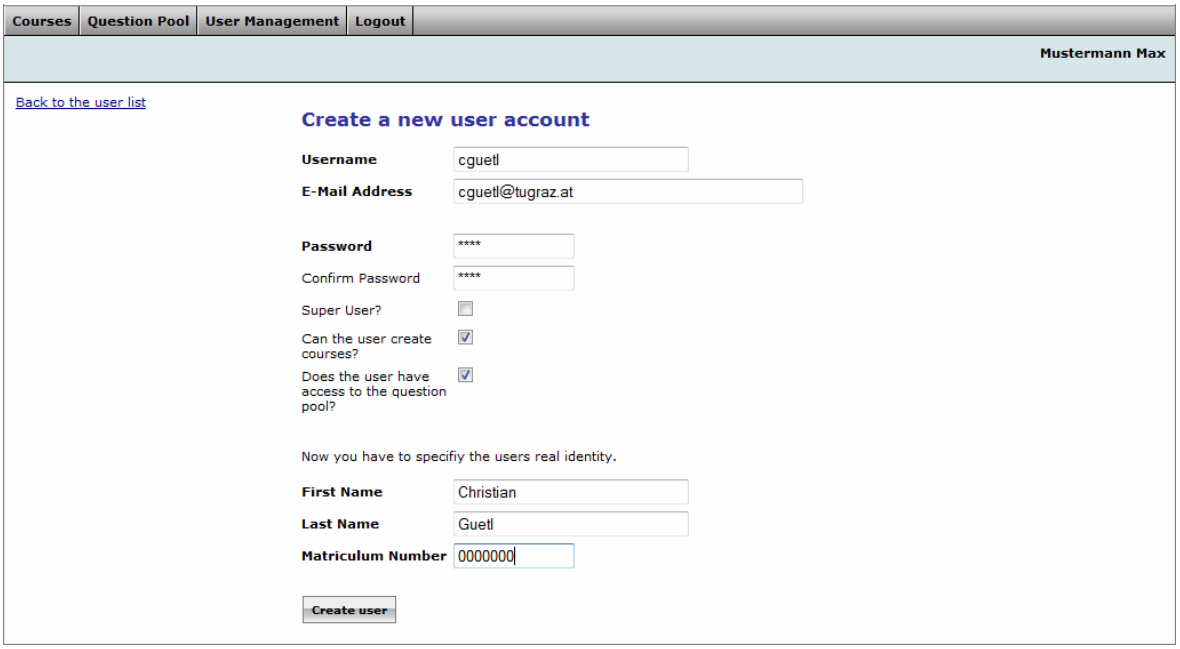

Figure 11: User creation screen

This screen in figure 11 creates a new user account. It works the same as the registration form has worked but it is also possible to set some flags: the super user flag to create another administrator, the create courses flag to create a user with teacher privileges and the access question pool flag to create a user with teacher or tutor privileges.

4.3 Teacher Area

This chapter explains the teacher area which includes the course and exam administration.

4.3.1 Courses

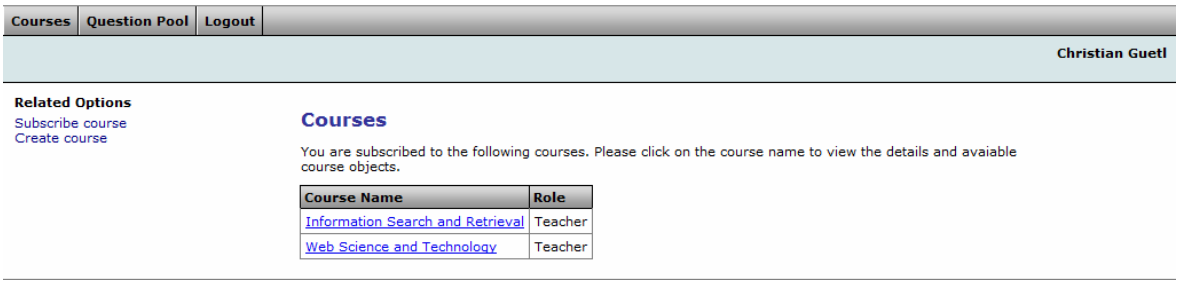

Figure 12: Courses main page

Unlike the administrator all other users only have access to their subscribed courses. The links on the left side offer the functionalities to subscribe to a new course with student privileges or to create a new course if the current user has the privileges for this (see figure 12).

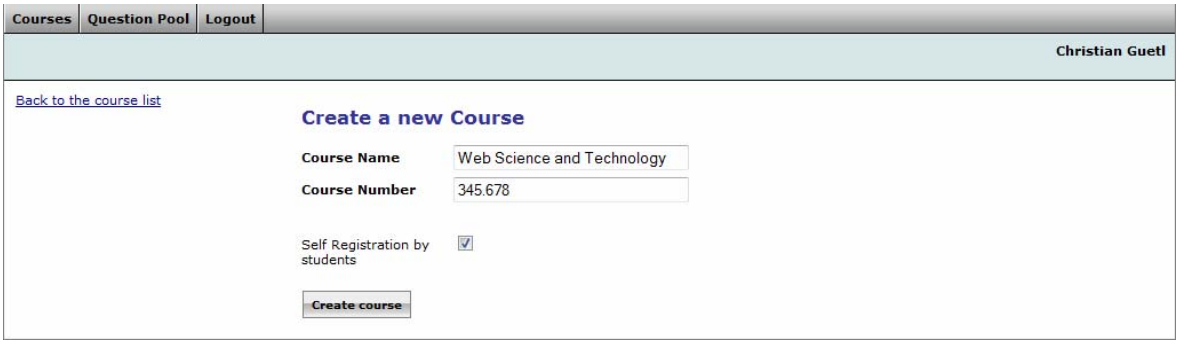

Figure 13: Course creation screen

When creating a new course also a checkbox is available which determines if students have the ability to subscribe to the new course by them self (see figure 13).

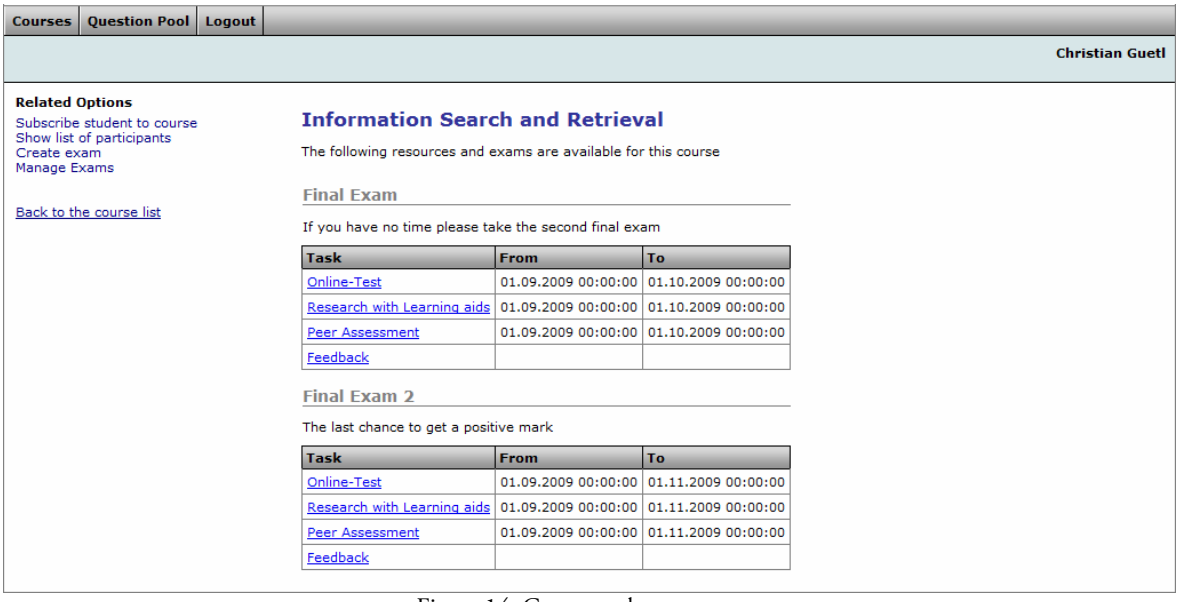

Figure 14: Course and exams screen

This screen in figure 14 shows the course main page with all the available exams and their exam items (test, research, peer assessment and feedback). It also shows the timeslots that constitute when the exam items are available for the students. The link list on the left side presents the other functionalities within the current course.

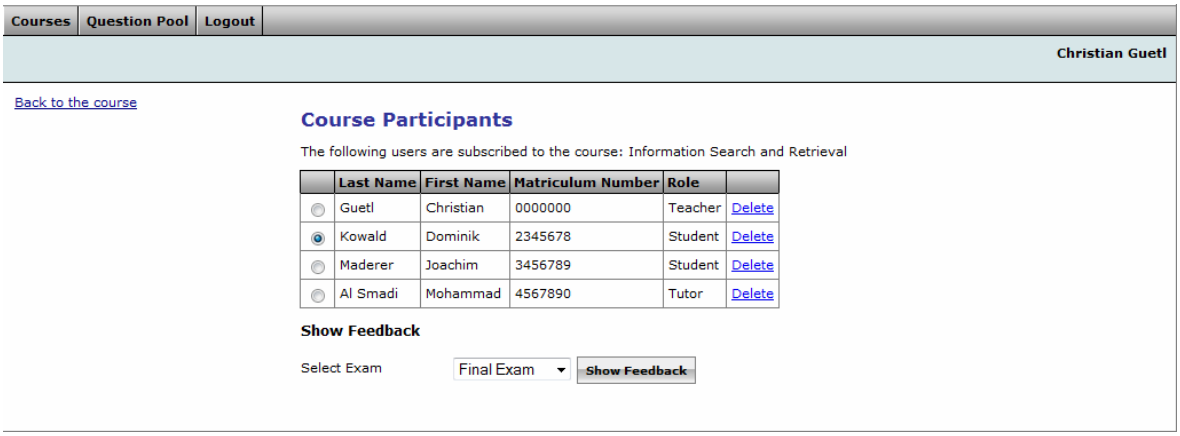

Figure 15: Course participants screen

The user will receive this page in figure 15 when he chooses the “Show list of participants” option. Beside the standard functionalities like deleting the participation of a user, he also has the ability to get the feedback of a specific user for a chosen exam.

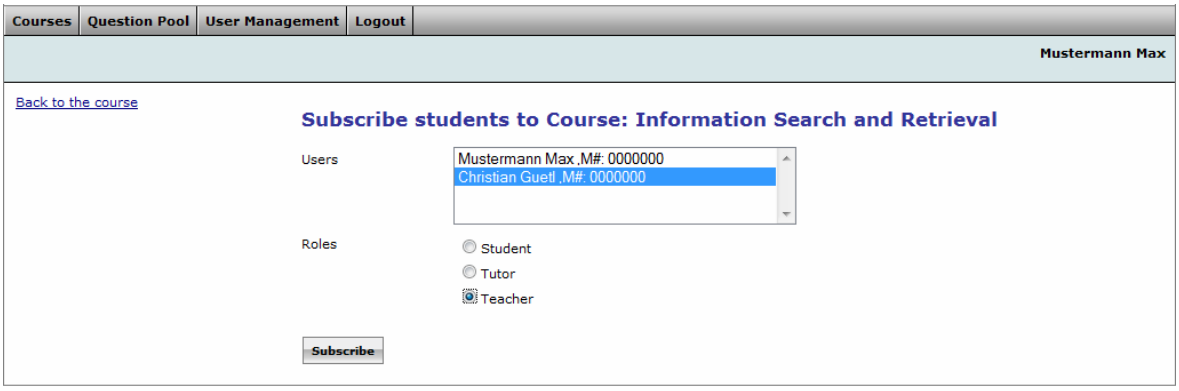

Figure 16: Course subscription screen

The teacher also has the chance to subscribe a user to one of his courses. This screen in figure 16 shows the admin mode of this functionality because the logged in user can also create teacher participations. The teacher is only able to create student and tutor roles.

4.3.2 Exams

These screens in figures 17 and 18 show the main functionalities of the exam objects: Creating and editing them and having an overview of all available exams.

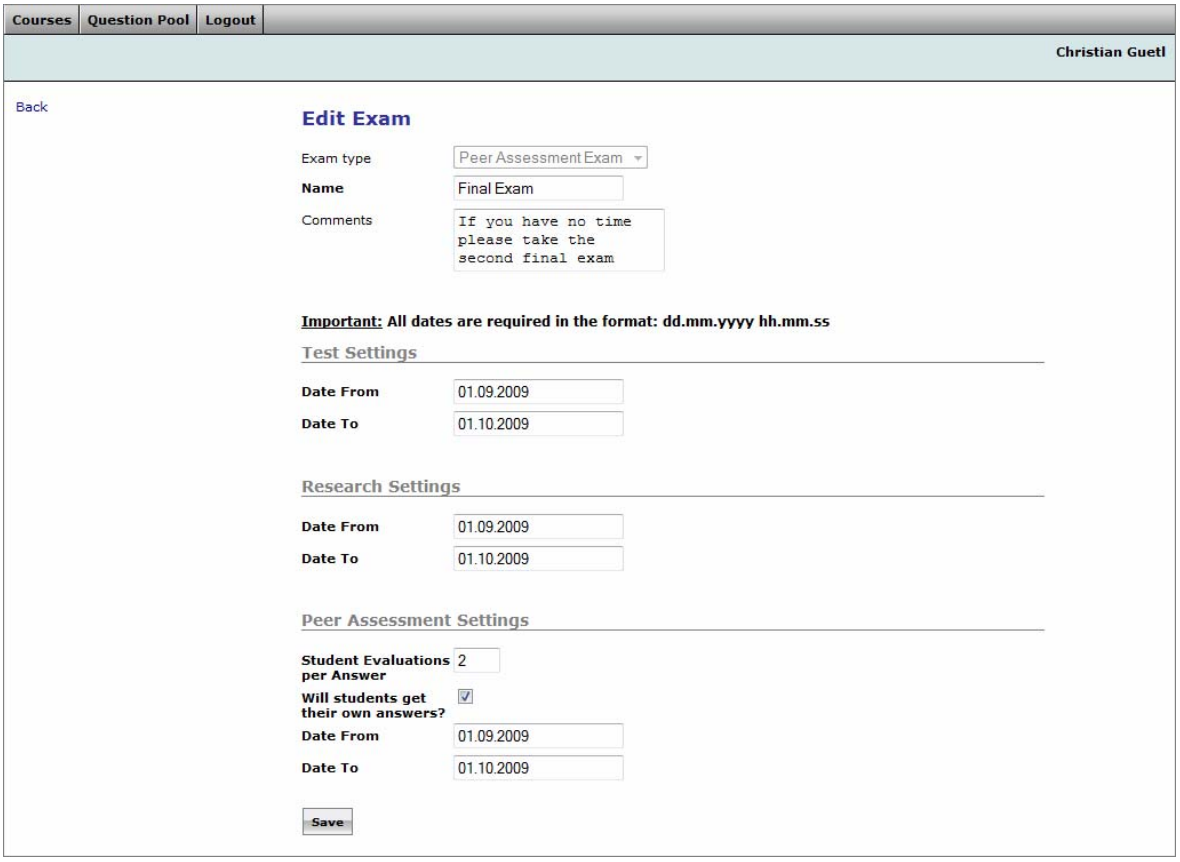

Figure 17: Create / Edit exam screen

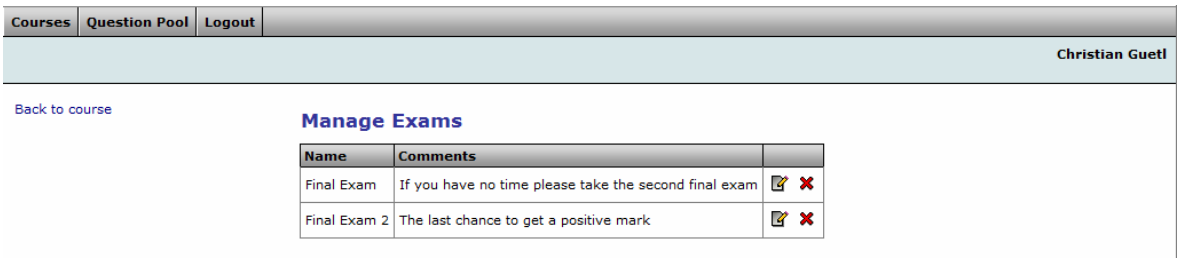

Figure 18: Manage exams screen

The next three screens in figures 19, 20 and 21 show the main peer assessment adjustments: In the test configuration the teacher chooses the questions he wants to add to the online test and in the research configurations he can choose a subset of the chosen test questions.

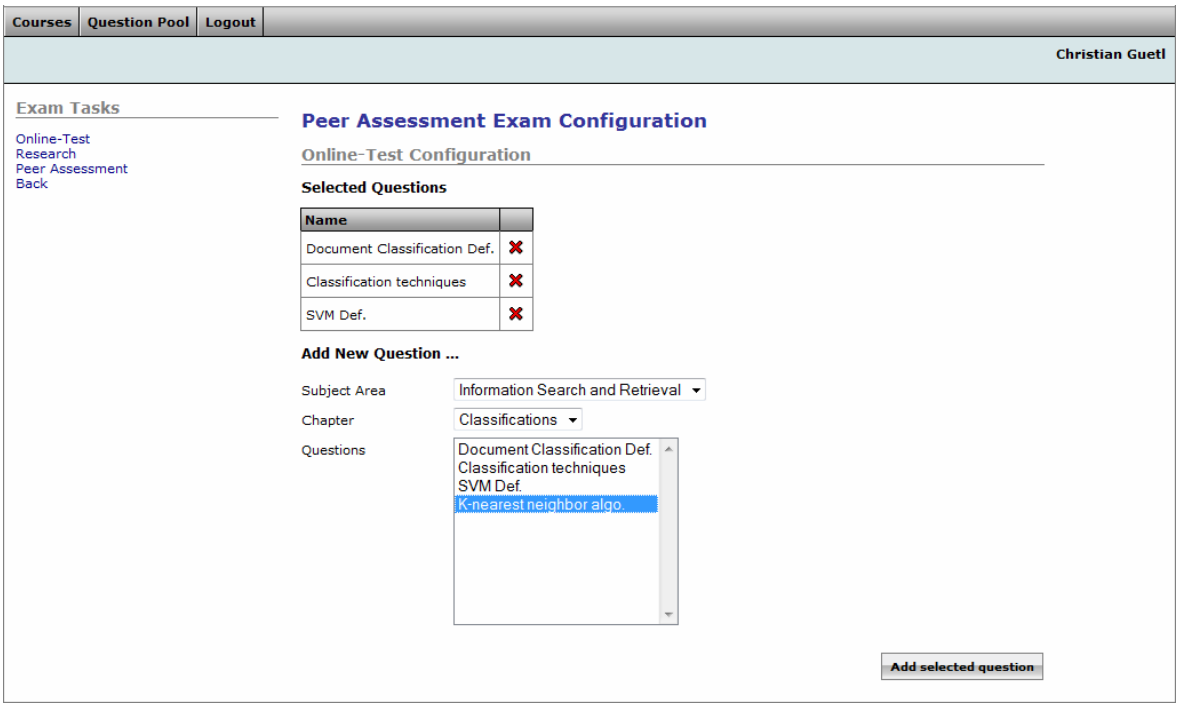

Figure 19: Test configuration screen

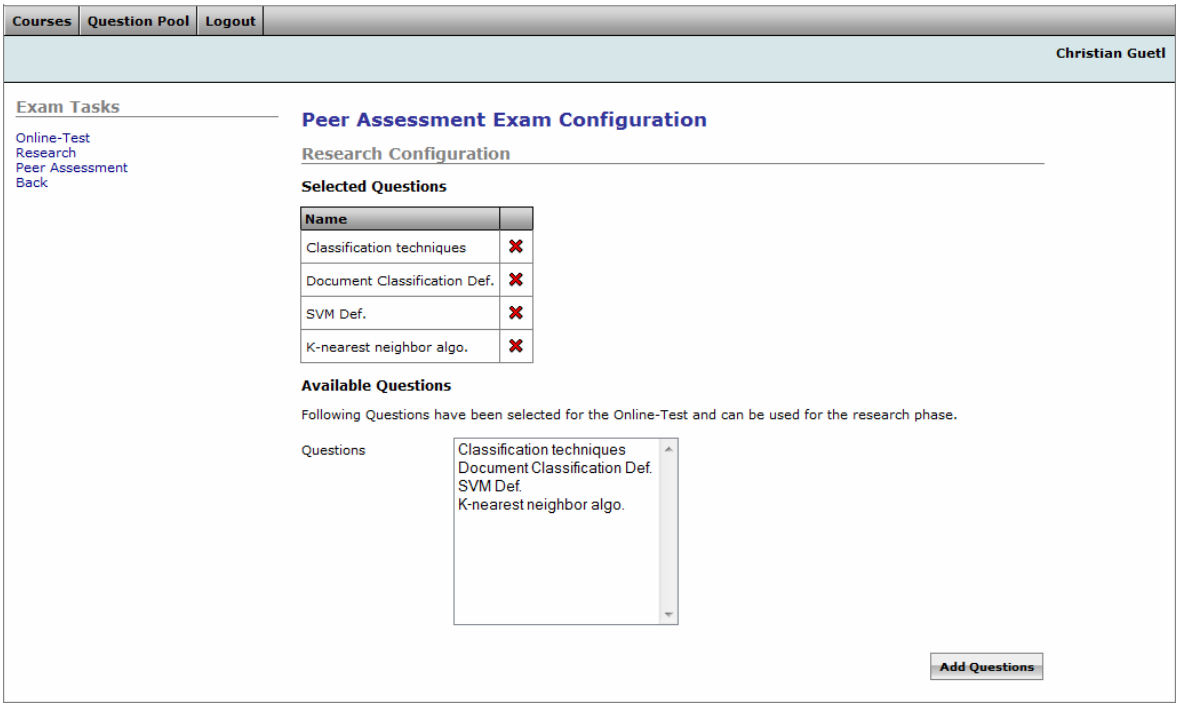

Figure 20: Research configuration screen

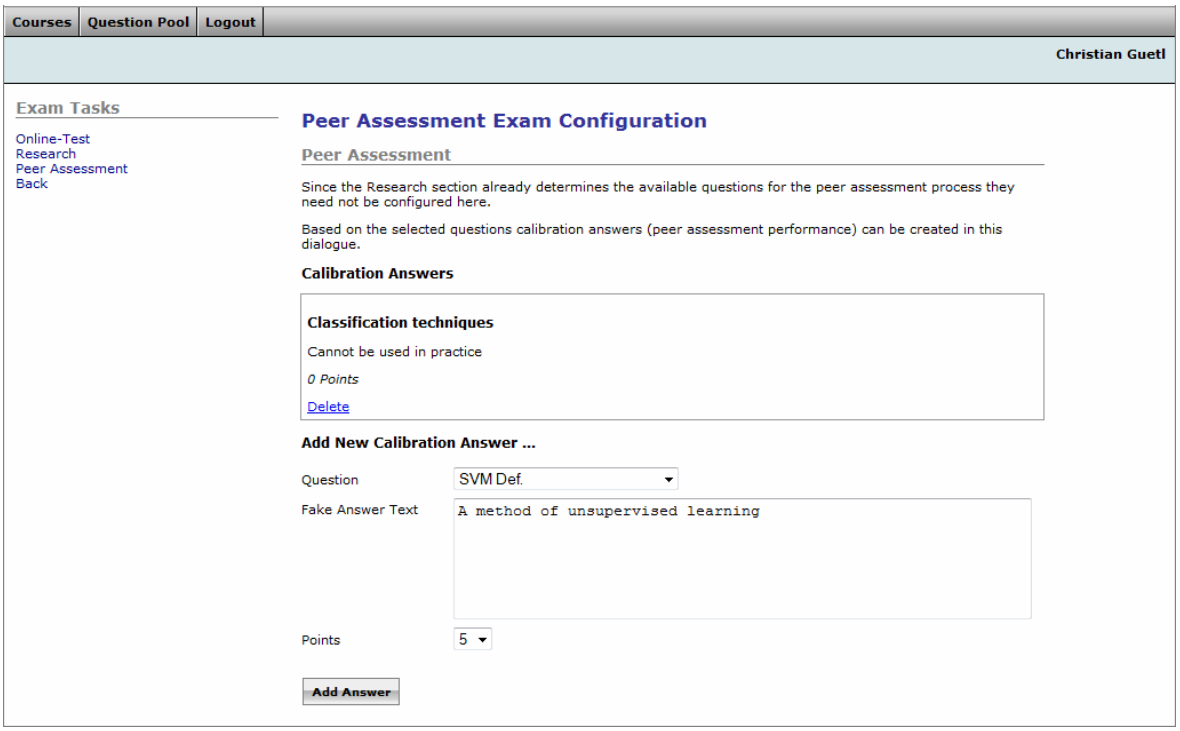

Figure 21: Peer assessment configuration screen

The peer assessment configuration in figure 21 is a bit more complicated. Here the teacher has to create calibration answers for all the chosen research questions. These calibration answers are needed to calculate the peer assessment performance.

4.4 Question Pool

This chapter explains the usage of the question pool which is divided into subject areas, chapters and questions.

4.4.1 Subject Areas

The subject area in figures 22 and 23 is the highest object in the question pool hierarchy and contains a number of chapters. The user has access to all the subject areas and their chapters and can create new ones.

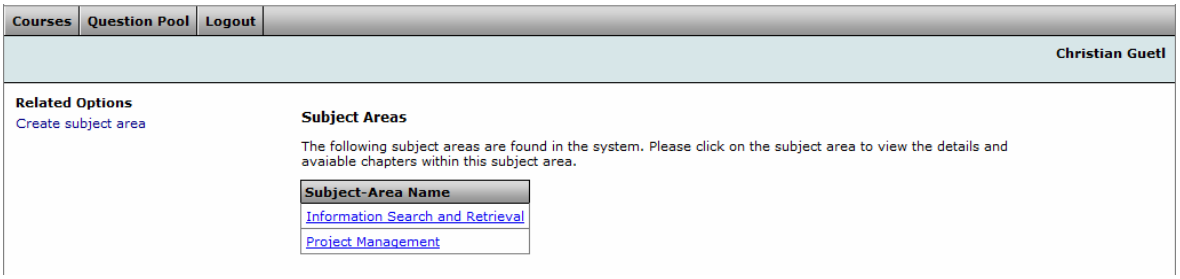

Figure 22: Subject areas main screen

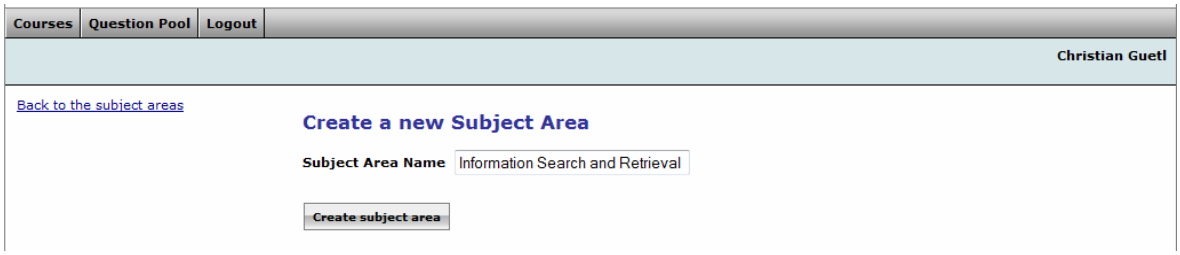

Figure 23: Subject area creation screen

4.4.2 Chapters

The chapter in figures 24 and 25 is the next step in the question pool hierarchy and contains a set of questions. It offers the same functionalities as the subject area main form: creating a new chapter and click onto a chapter to see all of its questions.

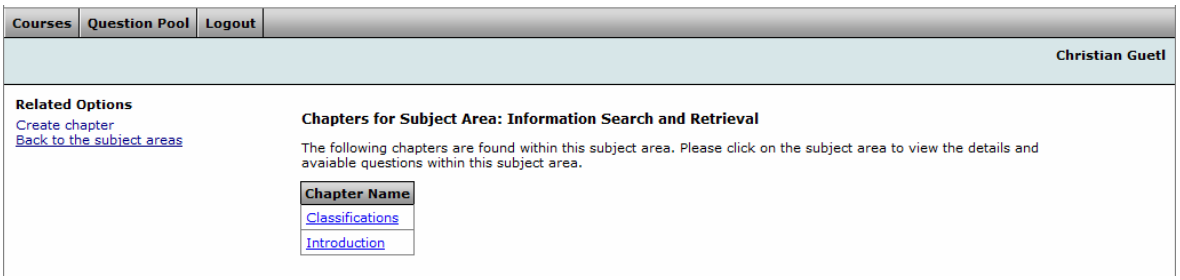

Figure 24: Chapters main screen

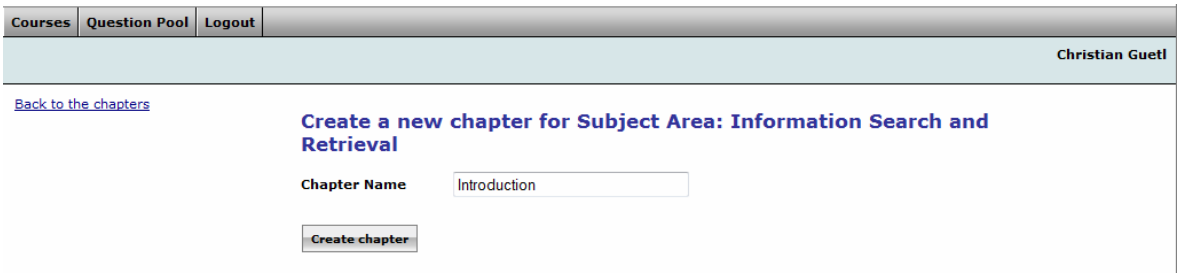

Figure 25: Chapter creation

4.4.3 Questions

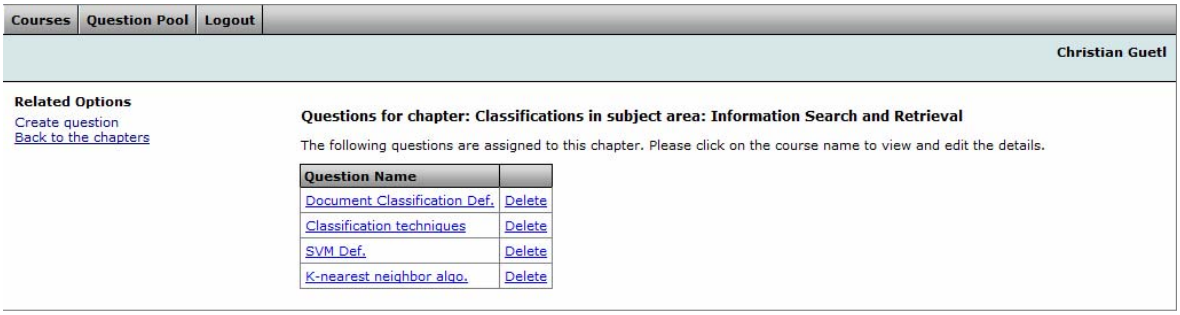

Figure 26: Questions main screen

This screen in figure 26 shows all the questions within the current chapter with the ability to click on a question for editing it.

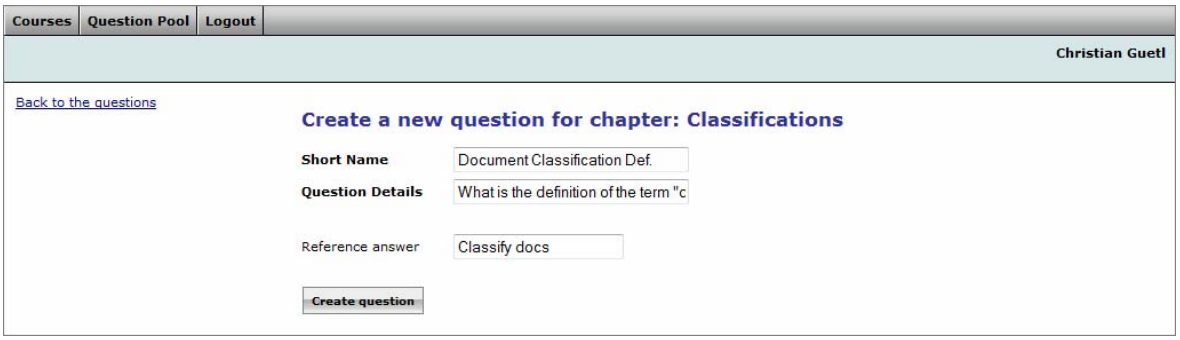

Figure 27: Create / edit question screen

The user will see the same screen in figure 27 when creating or editing a question. The short name of a question is only for displaying and the question details field holds the whole question. The reference answer field is only for internal purposes and has not to be entered.

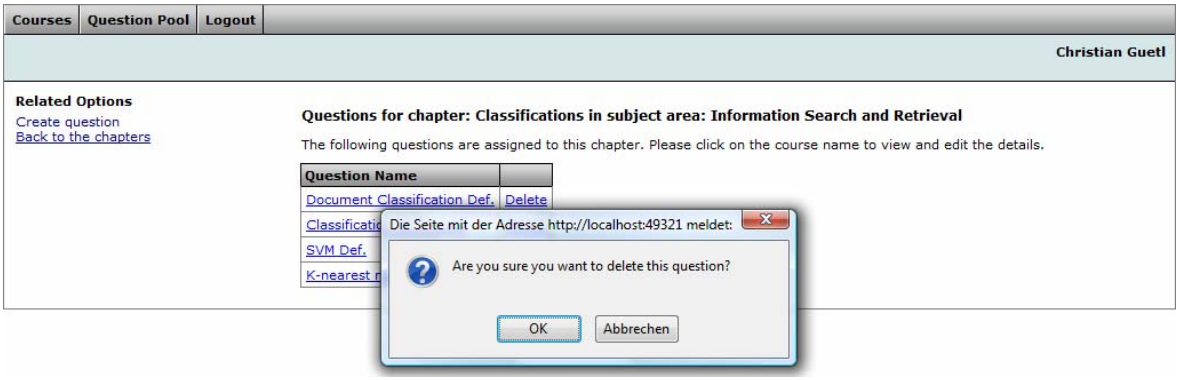

Figure 28: Delete confirmation screen

After clicking on a delete button the user will see a message box as in figure 28 where he has to confirm the delete operation.

4.5 Student Area

The main activity of a student in this application is the performing of a peer assessment exam which is described in the following chapter.

4.5.1 Courses

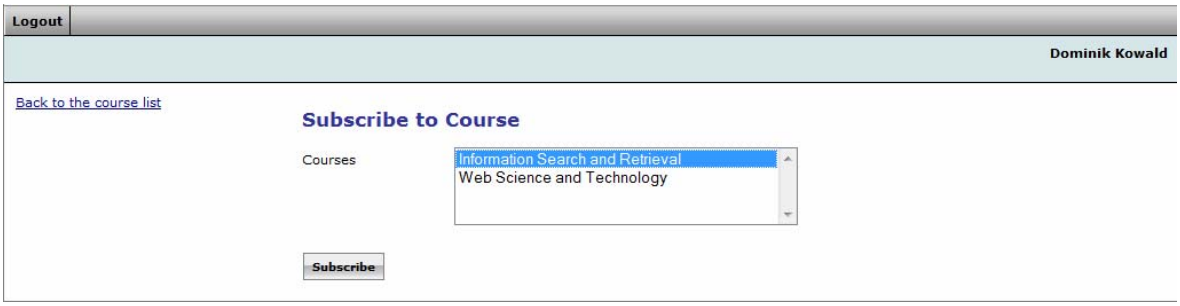

Figure 29: Course subscription

Besides the performing of an exam the student also has the ability to subscribe himself to courses (see figure 29) which is shown in the screenshot above.

4.5.2 Exam Items

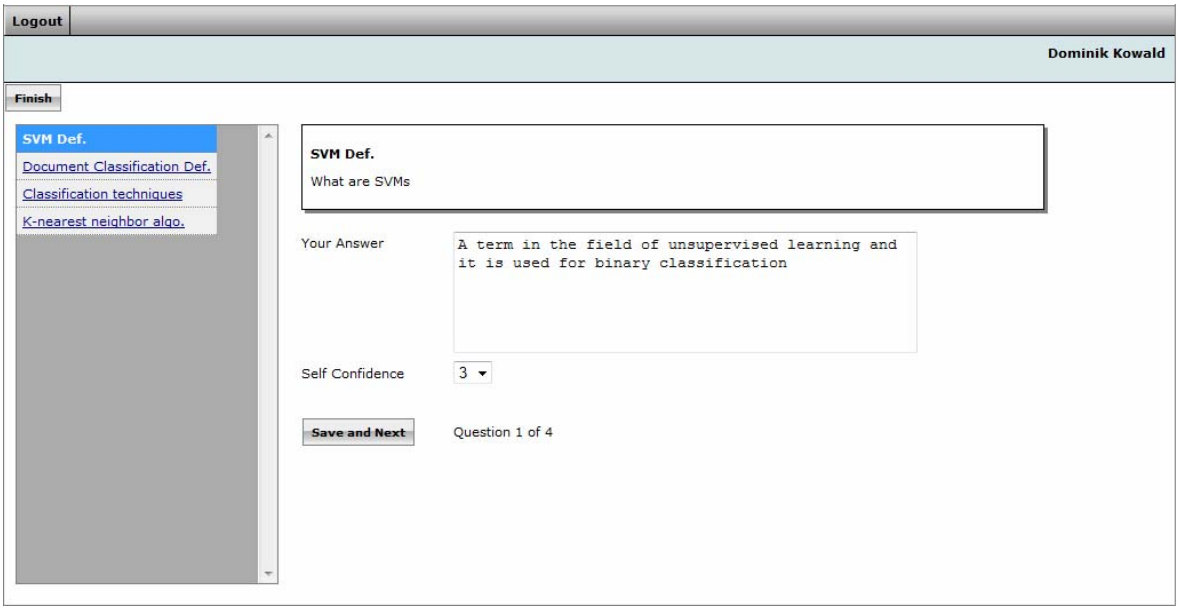

Figure 30: Test screen

This first exam item is the test in figure 30. The student can navigate through the questions on the left side and can finish his test with the finish button. The self confidence is for calculating the self assessment.

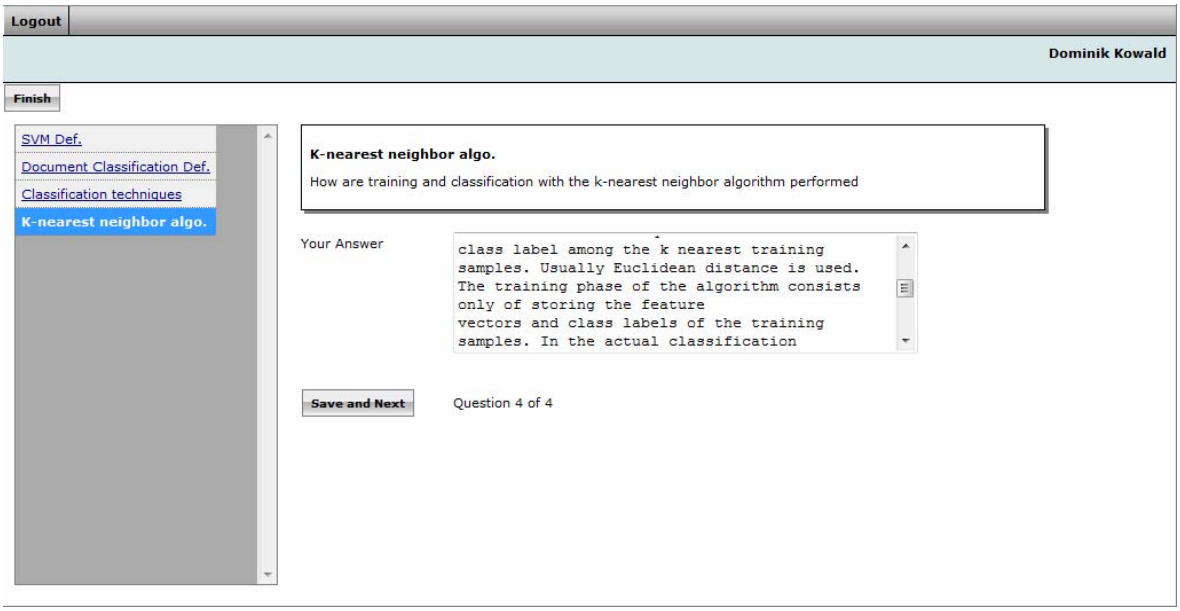

Figure 31: Research screen

The research item in figure 31is very similar to the test item but without the entering of the self confidence.

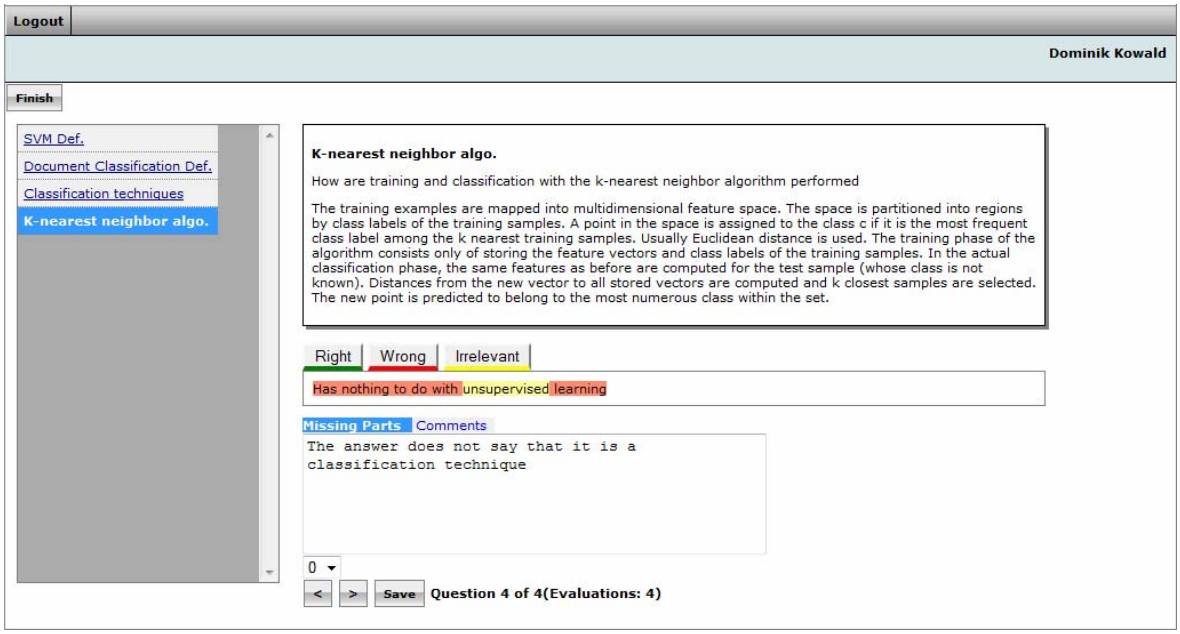

Figure 32: Peer assessment screen

The peer assessment seen in figure 32 is a bit more complicated. The student navigates through the evaluations per question with the "< / >" buttons and uses the save button to persist his given evaluation. To use the implemented evaluation tool the assessor has first to click on the right, wrong of irrelevant button and then mark the relevant answer sections.

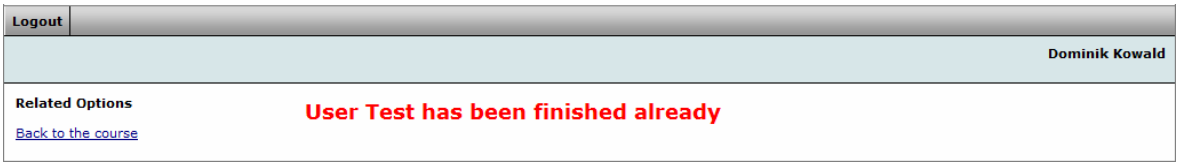

Figure 33: Manage exams screen

This error message is an example of the error page seen in figure 33 that appears when the student doesn't use the system in the correct order or if he wants to enter an exam item a second time.

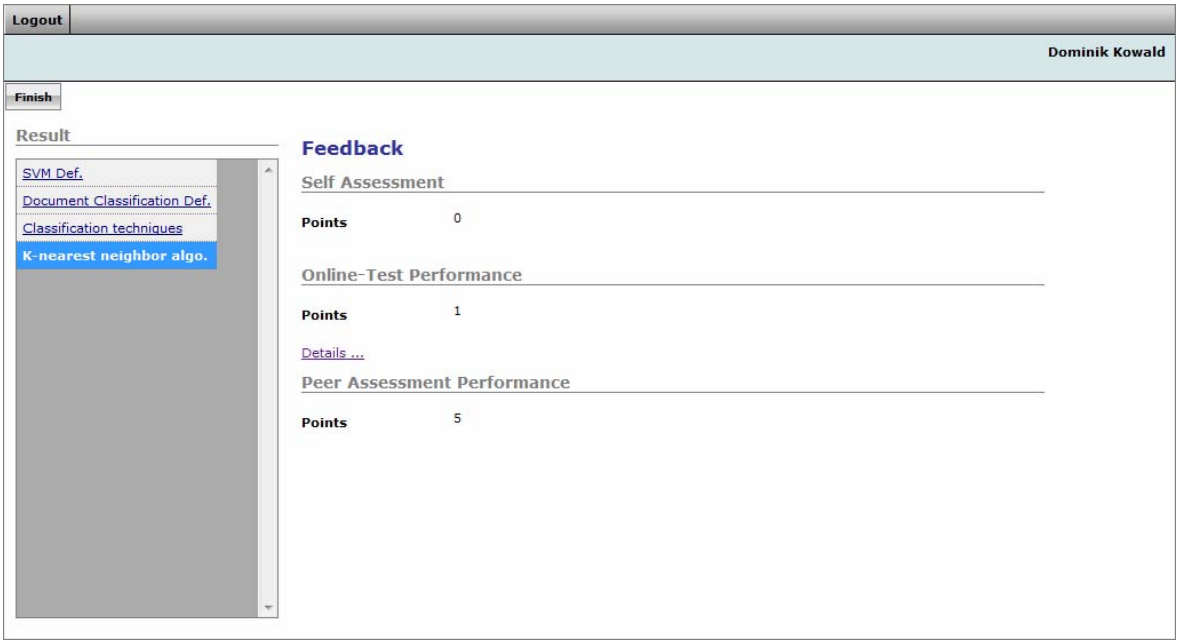

Figure 34: Feedback screen

The feedback screen in figure 34 shows the points for self assessment, the online test performance and the peer assessment performance per question. If the user clicks the details button he will see all evaluations for the given answer for the selected question.

5 Conclusion

5.1 Summary

Peer assessment can be seen as an assessment type that fulfills both types of assessment, summative and formative assessment. This means that it not only can be used to determine grades but also to provide useful feedbacks created by students to help some other students. It is in the nature of things that the evaluation of many student evaluations needs time and so the peer assessment idea can be perfectly combined with methods of technology enhanced learning, especially e-learning and other forms of online peer learning. This combination defines the term of peer assessment in computer science and also clears the way for building a modern and flexible e-learning system around this approach. This can be implemented with classical three-tier software architecture with the use of special components in the fields of the data model, the application environment and of course the client environment. To implement these components the .NET framework seems ideal because it offers an easy way to create a powerful web application with asp.NET and the MONO framework makes it possible to use this application also with UNIX based systems like Linux.

5.2 Outlook into the Future

Since this system is a prototype, a large set of ideas raised during the development phase, some of them are:

- Messaging system to report senseless or wrong assessments. This could also include a back-feedback for the exam results.
- An automatic variance reporting system that informs tutors about strange results, inspired by Online Assessment System (OASYS).
- Improved interface for answering messages, including basic HTML tags.
- Advanced features for the teacher to observer and manipulate results, as well as more flexible configuration screens.

For accomplishing a large and more consistent system, several reconsiderations with focus on architecture and reliable high-level frameworks would be necessary. There is a great chance to implement first approaches of computer based peer assessment for the next generation of e-learning systems, which can be used at schools but also at university. Beside limited experiments, time seems right for first larger monitoring in natural learning environments to collect practical experiences.

5.3 Lessons Learned

After first tests and a try to begin with the implementation of the system on Java based technology (including original Hibernate version for Java), several problems rose. Not even that the startup times of testing environment and reactions of the IDE was incredibly slow, also integrated debugging features did not work reliable. When evaluating the feasibility of implementation on .NET Framework based technology, it was possible to recreate the work and drafts of about two weeks of work within two afternoons.

Since the Web Forms framework enables a rapid application development it was ideal for implementing a prototype on top of .NET Framework technology. However, the practical work has shown that on some point the implementations become sloppy due to time pressure and on the other hand the view state concept of ASP.NET Web Forms produce a lot of overhead regarding request and response data. For more advanced long-term solutions an MVC based framework like ASP.NET MVC Framework will result in cleaner and better maintainable code. Furthermore the concept forces to strictly separate code and design. In return MVC requires more initial skill adaptation training.

Regarding the businesslike aspects of this prototype it has shown that the administrative part is the primary problem for such an application. To fulfill all the flexible requirements of the peer assessment system described before, expansive and complex input scenarios are needed. This would lead to enormous dimensions in the sense of human computer interaction which are hard to implement in the context of web based applications.

But also technical aspects complicate the implementation. Beside the possibility of creating complex data and process abstractions with object oriented data models very easily, it turned out that mapping these structures to user interfaces results in various problems. Creating generic interfaces is heavily possible with existing technology. One fact that encumbers binding the data in interface direction is the weakness of NHibernate (and also Hibernate for Java) to cast generic objects to their concrete representation. Although the lazy loading features can be disabled this will also lead to dynamic loading issues regarding the associated lists of the class hierarchy. Sure there are some workarounds for this problem on the internet, but none is really convenient. Maybe the best solution is to avoid inheritance as far as possible.

6 References

Al-Smadi, M., & Guetl, C. (2008, September). *Past, Present and Future of e-Assessment: Towards a Flexible e-Assessment System*. Paper presented at the Interactive Computer Aided Learning Conference, Villach, CA.

Bhalerao, A., & Ward, A. (2001, January 12). *Towards Electronically Assisted Peer Assessment: A Case Study*. Retrieved September 10, 2009, from http://www.dcs.warwick.ac.uk/~abhir/research/papers/new-alt-j.pdf

CATME, (2009). Comprehensive Assessment of Team Member Effectivness. Retrieved September 27, 2009, from https://engineering.purdue.edu/CATME/index.htm

Guetl, C. (2009). *Enhancements of peer assessment in computer based learning environments*. Manuscript submitted for publication.

ISCP (2009). Intercollegiate Surgical Curriculum Programme. Retrieved September 25, 2009, from https://www.iscp.ac.uk/Home.aspx

WebPA. (2009). An Online Peer Moderated Marking Tool for Group Work. Retrieved August 26, 2009, from http://www.webpaproject.com/