Institute of Information Management

University of St.Gallen

Master of Arts in Business Innovation

Master's Thesis

Creation and evaluation of a computer-based formative assessment and feedback tool designed to support higher-level learning

Author

Kevin Duss

##########

######################

Supervisor

Prof. Dr. Matthias Söllner

matthias.soellner@unisg.ch

St.Gallen, 22. Mai 2018

Management Summary

Despite the immense potential of formative feedback, it is not or only insufficiently implemented in the teaching structure of today's universities. This circumstance can be attributed to high and constantly rising student numbers and limited financial resources of universities in combination with the time and costs required to provide feedback. However, the challenge of efficiently providing feedback to a high number of students can be overcome by exploiting current information technologies. The present thesis aims to reveal how an IT tool should be designed and implemented to provide automated formative feedback in large-scale lectures in order to increase student performance.

In the design of formative feedback, multiple variables must be considered that may influence the effect of feedback on student performance. Among these variables are feedback timing, frequency, specificity, complexity and goal orientation. It is crucial to understand that said variables interact with individual factors such as the complexity of a task, the achievement level of a student or the aim of feedback. Therefore, there is no ideal form of formative feedback for all students and performance goals – feedback needs to be designed with regard to the specific context. To evaluate the effects of automated formative feedback on student performance in the context of large-scale lectures in higher education, this thesis involves the creation of a computer-based formative feedback tool called LOOM. LOOM's development is based on the action design research method to ensure a contribution to the existing body of knowledge. Focusing on the second and third stage of this method, LOOM emerges in two cycles of building, intervention and evaluation while learnings are gathered. These learnings in combination with requirements derived from expert interviews (conducted prior to this thesis) and scientific literature result in the design elements discussed in the next paragraph that should be incorporated in a formative feedback tool.

A computer-based formative feedback tool should be implemented as a web-based application with a responsive user interface to be available from anywhere at any time. Since recurrent use is an important factor for the provision of formative feedback, the tool should also offer a high degree of usability. Feedback provided by the tool should be based on preceding self-assessment and computer-based assessment to allow an accurate evaluation of the students' skills and to reveal potential gaps between their perceived and actual level of knowledge. Bar charts visualizing the assessment results at different aggregation levels are a useful instrument to emphasize such gaps. These performance charts should also provide students with the possibility of comparing their results with the ones of fellow students who specified the same performance goal since social comparison can have a positive impact on performance. Lecturers should have access to similar charts with aggregated data, so they can monitor the assessment results in order to optimally design their lectures. When being created, computer-based assessments should be classified in a Taxonomy Table according to the learning objectives they cover to ensure that all specified objectives of a learning outcome are met. In this context, the tool should promote different assessment item types to enable feedback that targets both lower- and higher-level learning. During the completion of assessment items, the tool should retrieve a student's confidence level regarding each item and include it in the feedback since it may lead to an improvement in self-perception and performance. Finally, to minimize the cognitive load in students during the assessment process and to facilitate the creation of assessments for lecturers, the tool should include walkthrough wizards.

The final evaluation of LOOM and its design elements was based on an intervention in a large-scale master's course lasting one semester. Throughout the course, students repeatedly completed assessments and received immediate feedback from LOOM. LOOM's feedback was low in specificity and restricted to the two types knowledge of results and knowledge of correct response. The results of the evaluation prove that LOOM and the established design elements have a significant effect on both objective and perceived student performance. By documenting the knowledge generated through building and evaluating LOOM according to the action design research method, the present thesis makes a first step towards a design theory for formative feedback tools.

Keywords: Formative feedback, computer-based assessment, action design research

Table of Contents

List of Figures

List of Tables

1 Introduction

In today's fast-moving world, education is a determinant factor of success in the labor market. A skills mismatch consisting of a surplus of low-skilled workers and a shortage of medium- and high-skilled workers causes unemployment and underemployment around the world (Janta, Rathmann, Ghez, Khodyakov, & Yaqub, 2015). Until 2020, Dobbs et al. (2012) estimate a shortage of 16 to 18 million university-educated workers in advanced economies. The unmet need for skilled workers demonstrates the significance of education. To meet the high demands of the working world, more and more people enroll at universities (Nicol & Macfarlane-Dick, 2006). At the University of St.Gallen, for instance, the number of matriculated students has been rising by over 12 percent from 7'325 students in 2012 to 8'232 in 2015 (University of St.Gallen, 2017a). These high student numbers only allow little interaction between students and lecturers. Little interaction, consequently, limits the amount of feedback lecturers can provide students with (Cuseo, 2007). Feedback, however, plays a major role in the education process of students: If applied properly, it allows the enhancement of teaching and learning on various levels (Hattie & Timperley, 2007).

Despite the immense potential of feedback, it is not or only insufficiently implemented in the teaching structure of today's universities (Booth & Hyland, 2000). This circumstance can be attributed to the time and effort it takes to give feedback combined with the costs of specialists capable of conducting proper feedback (Price, Handley, Millar, & O'Donovan, 2010). Universities have limited financial resources to hire additional lecturers (Giroux, 2002). Therefore, the number of lecturers cannot keep up with the increasing number of students. While the student number at the University of St.Gallen increased by almost 1'000 between 2012 and 2015, solely eight new lecturers (full-time equivalent) were hired (University of St.Gallen, 2013, 2016, 2017a). The shortage of lecturers results in lectures with hundreds of students, characterized by a high degree of anonymity (MacGregor, Cooper, Smith, & Robinson, 2000). In such lectures, a student is bitterly aware that the lecturer does not even know that he or she exists (Cuseo, 2007). It is simply not feasible to establish the concept of regular, individual feedback in lectures of this magnitude (Miller, 2009).

The didactical power of feedback has been recognized already a long time ago from multiple researchers as various meta-analyses summarized by Hattie and Timperley (2007) show. Regarding the acknowledged value of feedback, there has not changed much during the last few years (London, 2015). However, what has changed are the possibilities to design feedback processes in a more efficient manner. During the ongoing digitization, web technologies have emerged and matured. The advanced research around these technologies has revealed new methods that are applicable to the feedback process (Chen & Chen, 2009). At the same time, the costs for (mobile) hardware declined making notebooks and handhelds affordable for the general public including students (Looi et al., 2010). These developments make feedback possible in teaching environments where it was unimaginable before because of the high student-lecturer ratio (Ng'ambi & Bozalek, 2013). Specifically, they can be exploited to enable automated assessments and feedback processes – accessible to any student from anywhere at any time (Bonk, 2009; Scalise & Gifford, 2006). To do so, it is vital to understand the elements and levels of valuable feedback on the one hand and to make the best use of existing technologies on the other hand. Accordingly, the main outcome of this work consists of a functional artifact that automates the feedback process based on the current state of technical and scientific knowledge. Said artifact allows for self-assessments (SA), computer-based assessments (CbA) as well as the comparison and tracking of the corresponding results. These assessments and feedbacks target particularly higher categories of the cognitive process defined by Krathwohl (2002): analyze, evaluate and create. To reach these higher categories, the artifact includes formative assessment consisting of different item types which allow for different forms of feedback. One possible item type within the assessment consists of the identification, classification and correction of errors in diagrams. This approach makes the students access their present knowledge to analyze a given case. Moreover, it enables them to develop the ability to evaluate visual and textual statements based on certain criteria and standards. Thus, this approach addresses two of the three highest categories of Krathwohl (2002): analyze and evaluate. After finishing an assessment, students receive a detailed feedback about their performance. Furthermore, the computer-assessed performance is compared to the foregoing SA which is a fixed component of the artifact. By providing these types of items, assessment and feedback, the application of the artifact is not limited to a specific subject. Rather, it represents a framework applicable to various disciplines such as business engineering or business administration.

As a result of gathering, processing and evaluating existing research about feedback and CbA, the theoretical part of this work establishes guidelines for the implementation of formative feedback. These guidelines lay the groundwork for the development of the artifact. During the practical implementation of the artifact, carried out as an iterative process of design and evaluation, learnings emerge. These learnings are not only continuously expanded but also challenged and revised when necessary. The design knowledge gained through this approach can then be generalized and used as a basis to achieve a general solution concept applicable to various field problems (Sein, Henfridsson, Purao, Rossi, & Lindgren, 2011). In this thesis, the artifact is not only being developed but also deployed and examined in terms of its impact on the student performance. This work contributes to the development of a general solution concept which itself is not within the scope of this thesis.

The structure of this research is based on the action design research (ADR) method. The application of ADR ensures that not only the technological view but also the organizational context is considered during the development of the artifact (Sein et al., 2011). Moreover, a thorough literature review assures that the current knowledge about assessment and feedback flows into the artifact. The development of the artifact is managed using the agile methodology Scrum. ADR and Scrum provide complementary as well as shared practices which make the simultaneous use of these methods possible. Among the shared practices is the active collaboration of different stakeholders within an iterative development process. Through this interaction, the artifact emerges step by step (Düchting, Zimmermann, & Nebe, 2007; Sein et al., 2011). Besides the ongoing evaluation during the development, the artifact is subject of two evaluations based on interventions in different courses. The latter of these evaluations validates the effect of the artifact on student performance.

The following research questions (RQ) are derived from the discussion above.

RQ1: How should an IT tool be designed and implemented to provide students in large-scale lectures with formative feedback?

RQ2: To what extent does the IT tool usage increase student performance?

2 Theoretical Background

The theoretical part of the present thesis is devoted to the gathering, processing and evaluation of information about feedback and assessment. The aim of this literature review is to catch up with the current state of research that is relevant for building a computer-based formative assessment and feedback tool and answering the research questions of this thesis. After defining the terms used within this thesis, this chapter discusses the provision of feedback in general and then specifically in a computer-based learning environment. Furthermore, the chapter covers the social comparison theory. Finally, it concludes by presenting 17 practical guidelines for the provision of formative feedback.

2.1 Terminology

This first chapter introduces essential terms in the field of feedback and assessment and helps to develop a basic understanding of the matter. This understanding is necessary to grasp subsequent theoretical explanations as well as explanations about the practical implementation of the artifact. In the present thesis, the terms “artifact” and “tool” are used interchangeably and refer to the software that is developed using the action design research (ADR) method.

2.1.1 Feedback

Hattie and Timperley (2007) define feedback as “information provided by an agent … regarding aspects of one’s performance or understanding” (p. 102) . Teachers, peers, parents as well as books can function as agents. However, feedback does not always come from the outside but can also be internally self-generated (Butler & Winne, 1995). The goal of feedback is to close the gap between the current understanding and the pursued understanding (Sadler, 1989). Ramaprasad (1983) explains that “information on the gap when used to alter the gap (most probably to decrease the gap) becomes feedback” (p. 5). Said gap can be decreased through affective and cognitive processes. Affective processes include raised effort, ambition or engagement. Cognitive processes involve validating results of students, providing additional information, giving directions or showing different ways to reach the aimed understanding. Moreover, feedback operates on different levels containing the task level, the process level, the self-regulation level and the self level. The impact of feedback on the gap varies between these levels. Feedback on task, process and self-regulation level has a positive effect on student performance. Self-oriented feedback, however, normally does not because it lacks task-related information (Hattie & Timperley, 2007).

The three most widely applied and studied types of feedback are knowledge of results (KR), knowledge of correct response (KCR) and elaborated feedback (EF). KR solely informs students whether an answer is correct or not while KCR provides the correct response or solution to it. Thus, both KR and KCR are restricted to the correction of the students’ answers (Attali & van der Kleij, 2017; Narciss & Huth, 2004). EF is often based on KR and KCR. However, the information involved in EF goes beyond corrective information (Jaehnig & Miller, 2007). With the increasing amount of incorporated information, the line between EF and instruction begins to blur until “the process itself takes on the form of new instruction, rather than informing the student solely about correctness” (Kulhavy, 1977, p. 212). Hints, extra study material or explanations are examples showing that EF can take various forms (van der Kleij, Feskens, & Eggen, 2015). Because of this variety of possible manifestations, some researchers subdivide EF into further feedback types according to their complexity (e.g. Kulhavy, White, Topp, Chan, & Adams, 1985; Mason & Bruning, 2001). Shute (2008) aggregated information about feedback types from various researchers and arrayed them along the dimension of complexity. The resulting compilation is displayed in appendix 1.

In this thesis, feedback of any type is based on a preceding self-assessment (SA) and a computer-based assessment (CbA) that objectively judges a student’s performance (Thelwall 2000). The feedback involves the presentation and direct comparison of the results of these two assessments. Thus, it points out a student’s divergence between the self-assessed and computer-based assessed knowledge regarding a learning outcome (LO).

2.1.2 Assessment

Assessment occurs in two different forms depending on its timing as well as purpose and effect. Formative assessment, on the one side, aims to improve the competences and skills of a student (Yorke, 2003). Feedback in connection with formative assessment is provided during a course of study and allows enough time for enhancement. Summative assessment, on the other side, is a final evaluation at the end of study and summarizes a student’s performance (Sadler, 1989). In its strongest form, summative assessment only involves feedback limited to marks, grades or scores. Even if offered earlier, this form of feedback has turned out to be less effective than descriptive feedback (Brown, Peterson, & Yao, 2016).

2.1.2.1 Self-Assessment

Formative assessment and feedback should be used to promote self-regulated learning since self-regulating students are the most effective (Butler & Winne, 1995). By doing so, it is important to provide students with external reference points such as LOs that help them to define their own goals. Students use these goals to assess their performance as well as monitor their progress and, thereby, produce internal feedback. To exploit the potential of self-monitoring, students need to have structured possibilities for SA (Nicol & Macfarlane-Dick, 2006). If organized properly, SA can result in significant improvement of student performance (McDonald & Boud, 2003). However, SA alone can result in students overestimating their skills. The tendency towards overestimation relates to metacognitive deficits of unskilled students (Kruger & Dunning, 1999). Therefore, SA should be combined with a form of objective assessment. This combined approach enables students to control their learning and consequently improve their metacognitive skills (Nicol & Macfarlane-Dick, 2006).

2.1.2.2 Computer-based Assessment

In contrast to SA, CbA judges student performance objectively. In higher education, diverse types of formative as well as summative assessment are realized digitally because CbA can lead to didactical benefits. Among these benefits are advanced possibilities of providing feedback. For instance, feedback based on CbA can be provided directly during assessment. This allows immediate interventions in the learning process with the objective of closing the gap between the current and the pursued understanding. Students are, therefore, able to timely adapt their learning to reach their performance goal (van der Kleij, Eggen, Timmers, & Veldkamp, 2012). To maximize the impact of feedback, it can be customized based on the answer retrieved from the student. Thus, it may be beneficial to deliver feedback in different variations. Feedback can provide the correct answer, give an explanation or suggest related literature for further studies (Lopez, 2009).

Besides the possibility of immediate interventions in the learning process, feedback based on CbA offers further advantages that differentiate it from traditional feedback. It remains unbiased and is accurate, provided it is carefully designed at the beginning. Moreover, this kind of feedback is nonjudgmental and ignores irrelevant characteristics of students (Mason & Bruning, 2001). If the learning impact does not differ between computer-based and traditional assessments, other factors are decisive for the most suitable assessment form. Is this the case, general advantages of computers promote the implementation of CbA. One main advantage is the efficiency of CbA which saves the lecturers time and the university costs. Once the initial programming is completed, computers are basically able to provide students with unlimited feedback. This circumstance makes CbA interesting for large-scale lectures. Other reasons for the use of CbA can be of strategic, policy-based or experimental nature. In some cases, the manual process connected to traditional assessment is laggard and impractical to an extend that leaves CbA as the only option (Thelwall, 2000).

To reach a high quality, Brown, Race, and Smith (1996) propose that assessment should be formative, valid, reliable, fair, equitable, timely, incremental, redeemable, demanding and efficient. Table 1 matches the characteristics of CbA with these quality criteria. The fact that in seven out of the ten criteria CbA is likely to be beneficial emphasizes its relevance.

In the context of CbA, Thelwall (2000) subdivides the established classification of formative and summative assessment in more sophisticated types of assessment. These subdivisions range from formative diagnostic tests over formative exercises to summative exams. The formative character of CbA is based on the feedback provided on performance or the skills gained by executing the task. Depending on the degree of these two factors, the diverse types of CbA can be distinguished and classified. Diagnostic tests, for instance, place the focus on feedback while exercises typically rely on the skills gained during their completion (Thelwall, 2000). A taxonomy including all types of CbA proposed by Thelwall is displayed in appendix 2.

Table 1: Quality Criteria of Assessment, Adapted from Higgins and Bligh (2006)

Criterion	**Realization through computer-based assessment**
Formative	CbA allows to perform assessment frequently with multiple submissions including feedback each time.
Valid	CbA measures specified aspects of the course (assuming good initial assessment design).
Reliable	Every submission underlies the same assessment procedure resulting in absolute consistency.
Fair	*Dependent on assessment design – CbA has no universal benefits.*
Equitable	Every submission underlies the same assessment procedure resulting in no discrimination.
Timely	CbA allows to perform assessment frequently including feedback during and at the end of assessment.
Incremental	*Dependent on assessment design – CbA has no universal benefits.*
Redeemable	CbA can allow multiple submissions.
Demanding	*Dependent on assessment design – CbA has no universal benefits.*
Efficient	CbA allows to save time and other resources.

2.1.3 Learning Objectives

Formative assessment and feedback can address learning objectives on various levels. Bloom's Taxonomy provides hierarchical models which allow the categorization of learning objectives according to their complexity. In the cognitive domain, the Taxonomy defines six major categories of learning. These categories are ordered from easy to difficult while the expertise of each subjacent category is essential for the understanding of the next higher one (Krathwohl, 2002). In a revision of Bloom's Taxonomy, Anderson and Krathwohl (2001) redefine these six categories of learning and show how they intersect with distinct levels of knowledge. Therefore, the revised Taxonomy is a two-dimensional framework. The two dimensions "Cognitive Process" and "Knowledge" and their intersections can be visualized within the Taxonomy Table. The Taxonomy Table allows the classification of assessments and the goal-oriented provision of feedback. Moreover, it reveals untapped potential in the way knowledge is transferred. Generally, the categories within the Cognitive Process dimension still vary in terms of complexity. However, since the revised Taxonomy involves a shift of focus towards teacher usage, "the requirement of a strict hierarchy has been relaxed to allow the categories to overlap one another" (Krathwohl, 2002, p. 215). In this relaxed hierarchy, the more complex categories are believed to be the crucial ones regarding sustainable student performance (Krathwohl, 2002).

2.1.4 Assessment Items

Methods inducing student responses during the process of assessment are usually called items and can appear in form of questions, tasks or activities. In the context of CbA, the term item is broadly defined. According to Scalise and Gifford (2006), an assessment item in relation with technology "is any interaction with a respondent from which data is collected with the intent of making an inference about the respondent” (p. 7). This definition leaves room for innovation regarding the design of computer-based items. The computer allows items to include innovative features such as audio, video as well as graphics and animations. However, the possibilities are not restricted to the structure of the items but can also innovate how they function. Rather than just providing multiple-choice formats, computer-based items can enable the student to highlight text, click on images, drag and drop objects across the screen or rearrange statements. Another opportunity is that the sequence of the items within a CbA is not rigid. Computer-based items can dynamically be chosen and added to meet the individual demand of a student. As a result, each student may take a unique path through the CbA based on his or her provided responses (Parshall, Davey, & Pashley, 2000).

Despite all these opportunities resulting from technology, multiple-choice items are still commonly used because of their practical benefits (Denton, Madden, Roberts, & Rowe, 2008; Lee, Lim, & Grabowski, 2010; Murphy, 2007). However, multiple-choice items are often restricted to lower-level learning objectives like “remember” (see chapter 2.1.5). They differ from constructed response items in terms of the cognitive requirements of the students resulting in different effects when used in combination with feedback (Attali & van der Kleij, 2017). Answering multiple-choice items does not require much effort since it only involves the selection of an option rather than producing a response. This circumstance may result in a lower engagement of students. Moreover, students pay more attention to constructed response than multiple-choice items (Attali, Laitusis, & Stone, 2016). According to Attali (2015), the increased effort and engagement resulting from constructed response items leads to an enhanced performance in terms of mathematical problem-solving (higher-level learning objectives). However, these findings are not in line with previous research in this field. Clariana and Lee (2001) found no significant differences between multiple-choice and constructed response items regarding their effect on recall (lower-level learning objectives). On this basis, Attali and van der Kleij (2017) speculate that item type gains relevance when learning complexity rises since the constructed response items indicate a deeper processing of feedback than multiple-choice items.

Introducing a more detailed categorization of items than just constructed response and multiple-choice, Scalise and Gifford (2006) present 28 item types which can be applied in CbA. These item types are categorized regarding the degree of constraint in their response format. The result of this approach are seven categories of item types. The response formats of these categories reach from fully selected to fully constructed. The categories between these two extremes contain “intermediate constraint item types” and have a declining level of constraint. With a declining level of constraint comes an increasing level of intricacy in the computer-based implementation of the items. For example, responses to a multiple-choice item (category 1) can be effortlessly checked via computer while the interpretation of essays (category 6) is significantly more challenging. Additionally, the item types within the seven categories of constraint are ordered regarding the complexity of innovation ranging from simple to complex (Scalise & Gifford, 2006). According to the discussed item characteristics, Scalise and Gifford organized the 28 identified item types into a taxonomy which is displayed in appendix 3.

2.1.5 Lower- and Higher-level Learning

In terms of outcome, learning is commonly divided in lower-level learning and higher-level learning. However, the existing literature provides differing definitions for this common categorization (e.g. Anderson & Krathwohl, 2001; van der Kleij, Timmers, & Eggen, 2011). The definition applied within this work is based on a revision of Bloom's Taxonomy. Lower-level learning, on the one hand, targets the objectives "remember", "understand" and "apply" in the cognitive process dimension of the revised Taxonomy involving the whole knowledge dimension. Higher-level learning, on the other hand, aims at the objectives "analyze", "evaluate" and "create" (Anderson & Krathwohl, 2001). Reasons for choosing this definition are that Bloom's Taxonomy is widely accepted (Krathwohl, 2002) and that elements of its revised version are integrated in the artifact developed in the context of this thesis (see chapter 4.2.2.3). Moreover, the research this thesis builds upon applies the same categorization (Rietsche, Söllner, & Seufert, 2017).

While lower-level learning contains memorization and comprehension, higher-level learning describes intellectual skills requiring students to apply their knowledge. The application of prior gained knowledge in new situations is called transfer (van der Kleij et al., 2015). Based on the type of these situations, Hattie, Biggs, and Purdie (1996) establish a further classification of higher-level learning into near and far transfer. Near transfer involves applying knowledge in a situation similar to the instructional task while far transfer, in contrast, describes knowledge application in a different situation. A far transfer task may appear in a different context or in form of a different item type resulting in a greater challenge for the student to evaluate the right strategy to solve the problem (Attali & van der Kleij, 2017).

A large part of the past research on feedback in the context of CbA, especially older studies, focused on lower-level learning (e.g. Jaehnig & Miller, 2007; Kulhavy et al., 1985; Kulik & Kulik, 1988). Attali and van der Kleij (2017) identify an overall pattern suggesting that "more complex learning requires more complex feedback" (p. 3). Chapter 2.2 is dedicated to the design of feedback and different variables that need to be considered in doing so.

2.1.6 Student Performance

While a variety of definitions of the term student performance have been suggested, this thesis uses the definition of Gupta, Bostrom, and Huber (2010) who break it down into multiple learning goals. This definition is the most suitable for the present research since it involves findings of both IT and education. The learning goals consist of skill-based, cognitive, affective and meta-cognitive goals. Skill-based goals describe the ability of students to apply learned methods. Cognitive goals refer to the mental awareness and judgement of students. They involve the knowledge necessary to transfer learning to new situations. Affective goals focus on the emotional aspects of the students' behavior. Among these affective goals are motivational knowledge referring to the usefulness of an artifact and satisfaction with the learning process which is crucial for the continuous use of an artifact. Meta-cognitive or self-regulated learning knowledge describes the students' understanding about their own learning processes. In the context of meta-cognitive learning, a commonly examined variable is self-efficacy which also affects other learning goals (Gupta et al., 2010).

The goals defining student performance can be considered from two different perspectives. The final exam of a course provides insight into the objective performance of students while surveys allow to gather information about their perceived performance. This thesis analyzes skill and cognitive goals in terms of both objective and perceived student performance. Affective and metacognitive goals, however, are analyzed only regarding perceived student performance because they cannot be measured by evaluating exam results (Chemers, Hu, & Garcia, 2001).

2.2 Design of Feedback

Feedback aims to signal and close gaps in the understanding of a student. Different variables may influence the success of feedback in doing so. This chapter discusses said variables and their effects on student performance by focusing on formative feedback.

2.2.1 Feedback Timing

For decades, researchers have analyzed the timing of feedback and its effects on student performance (Kulik & Kulik, 1988). In the resulting studies, feedback is commonly divided in immediate and delayed feedback (Shute, 2008). However, the meta-analysis of Kulik and Kulik (1988) shows that these two terms are defined relatively and, therefore, bear inconsistent, sometimes overlapping meanings. In one study, for instance, immediate feedback may be considered as right after the completion of the assessment and delayed feedback as 24 hours afterwards. In another study, immediate feedback may be considered as feedback provided right after the response to an item and delayed feedback as after the completion of the assessment (Attali & van der Kleij, 2017). These divergent definitions are partly responsible for findings pointing in different directions (Mory, 2004). In this thesis, feedback is considered immediate if provided directly after the completion of an assessment and delayed if provided a certain period after the completion of an assessment. Some researchers believe delayed feedback to be more effective while others endorse immediate feedback (Shute, 2008).

2.2.1.1 Support for Delayed Feedback

Researchers supporting delayed feedback commonly build on the interference-perseveration hypothesis presented by Kulhavy and Anderson (1972). Kulhavy and Anderson argue that during the delay interval students forget their errors. This prevents initial errors from interfering with the correct responses when the feedback is provided. Thus, it is simpler for students to absorb the right information. Moreover, after a delay an increased attention to the feedback was detected. In their research, Kulhavy and Anderson acknowledge the delay-retention effect originally defined by Brackbill and her collegues (e.g. Brackbill, Bravos, & Starr, 1962; Brackbill & Kappy, 1962). This effect indicates that students receiving immediate feedback retain less information than students receiving feedback after a certain delay. However, many researches refute the superiority of delayed feedback implied by the delay-retention effect (e.g. Kippel, 1974; Phye & Andre, 1989; Phye & Baller, 1970).

An experiment of Schroth (1992) shows that a delay of feedback decreases the speed of the initial learning process. At the same time, however, it makes the transfer easier for students. Schroth considers hypothesis testing as a possible explanation for the results of his experiment. Following this model, students sequentially test different hypotheses until they find the right one. In this case, the delay period during practice needs to be at least 20 seconds. Another possible explanation is the interference-perseveration hypothesis discussed above (Kulhavy & Anderson, 1972).

2.2.1.2 Support for Immediate Feedback

Researchers arguing for immediate feedback regard it as a way of preventing students from memorizing errors. Accordingly, the probability of efficient retention rises the sooner corrective information is provided (Phye & Andre, 1989). In a meta-study involving 53 studies about feedback timing Kulik and Kulik (1988) conclude that in terms of conventional educational purposes “to delay feedback is to hinder learning” (p. 94). Moreover, the studies of Dihoff, Brosvic, Epstein, and Cook (2004) present evidence that immediate feedback enhances performance of students on examinations. The results of the studies involve that the performance increases particularly when the items of the examination have the same phrasing as the ones of the preceding practice tests (near transfer of knowledge).

Corbett and Anderson (2001) analyzed the timing of feedback in the context of CbA. They examined three CbA versions with differing levels of control over error feedback and correction. One group of students received feedback and error correction immediately. In another group, errors were immediately flagged while the students were responsible for their correction. Students within yet another group only received feedback on demand and were also in charge of error correction. In addition, a fourth group of students received no feedback at all. The results of this experiment revealed that with increasing control the learning efficiency rises. Therefore, students receiving immediate feedback experienced the most efficient learning, followed by students receiving error flagging, on-demand feedback and eventually the students without any feedback. On exams, the three groups that received feedback performed without statistically significant differences. However, students in the group without feedback yielded worse results (Corbett & Anderson, 2001).

2.2.1.3 Guidance Hypothesis

Connecting the discussed findings of Corbett and Anderson (2001) with the ones of Schroth (1992), immediate feedback may increase the efficiency of the learning process while delayed feedback may facilitate the transfer of learning. Schmidt, Young, Swinnen, and Shapiro (1989) came to a similar conclusion and explain these findings with the guidance hypothesis. The guidance hypothesis indicates that immediate feedback guides students through the first learning stages resulting in an increased initial performance. However, this guidance can cause students to rely on feedback and neglect secondary skills needed to perform the task independently. Such secondary skills include, for example, the ability to detect and self-correct errors (Schmidt et al., 1989). The findings of Schooler and Anderson (2008) are consistent with the benefits and drawbacks of immediate feedback identified by Schmidt et al. (1989). To get the most out of feedback, Schooler and Anderson propose to evaluate a form of feedback that optimally balances these benefits and drawbacks.

2.2.1.4 The Right Timing

Hattie and Timperley (2007) criticize most of the research about feedback timing for not considering the levels at which feedback operates. They found evidence that immediate feedback is probably more suitable at the task level while delayed feedback is more suitable at the process level. Immediate error correction, for instance, may accelerate task acquisition. However, during fluency building, immediate error correction can jeopardize the learning of automaticity. Clariana, Wagner, and Roher Murphy (2000) confirm these findings in their study about the impact of feedback on memory. Observing the retention of initial error responses, the study presents that difficult items presumably require a higher extend of processing regarding the task. Delayed feedback offers the time needed for this processing. In the case of simple items, which do not involve extensive processing, delayed feedback brings no benefits and is therefore not desirable. Another factor that should be considered when defining the timing of feedback are the students and their level of achievement. Immediate feedback is rather suitable for low-achieving students while delayed feedback may benefit high-achieving students. A framework for the design of effective feedback that distinguishes low- and high-achieving students will be discussed in chapter 2.3.1 (Mason & Bruning, 2001). To summarize, there is no consensus in research whether immediate or delayed feedback is preferable in general. However, depending on the situation and the aim of feedback, one of both approaches tends to be more beneficial than the other as Table 2 shows.

Table 2: When to Use Immediate or Delayed Feedback

	Immediate feedback	**Delayed feedback**
Desired outcome	More efficient initial learning	Facilitates transfer of learning
Level at which feedback operates	Task level	Process level
Student achievement	Low-achieving students	High-achieving students

2.2.2 Feedback Frequency

Most of the theories in the domain of psychomotor learning recognized that a higher number of feedback would result in increased student performance (Adams, 1987; Schmidt, 1975). Therefore, frequent feedback was generally considered beneficial. Past research suggests that these benefits appear during the acquaintance of skills as well as during assessment involving the retrieval of these skills (Salmoni, Schmidt, & Walter, 1984).

However, recent studies show that more feedback is not always better. Feedback is only beneficial for student performance up to a certain extend. Once feedback exceeds this point and enters high levels of frequency, its benefits diminish and negative effects on the learning process occur. At these high frequency levels, students are forced to put their efforts towards processing feedback and responding to it. Thus, the cognitive resources of students are taken away from the actual task resulting in a lower task performance. Eventually, the student performance suffers because of frequent feedback (Lam, DeRue, Karam, & Hollenbeck, 2011).

Schroth (1997) shows that feedback can have diverging effects on the learning process. He investigates the percentage of trials in which students receive feedback and its impact on immediate- and delayed-transfer tasks. The investigation reveals that "while lowering the percentage of trials in which feedback was given slowed down concept acquisition on the training task, it facilitated transfer on both the immediate and delayed task" (Schroth, 1997, p. 76). Therefore, circumstances that complicate the initial learning of a task may simplify the transfer to other tasks. Formulated the other way around, frequent feedback may have a positive impact on the practice performance, however, it makes the transfer more difficult (Winstein & Schmidt, 1990).

In conclusion, recurrent feedback is vital for the learning process. However, if provided too frequently, feedback stands in the way of long-term individual learning and performance enhancement (Lam et al., 2011).

2.2.3 Feedback Specificity

The specificity of feedback defines the information value included in feedback messages (Goldstein, Emanuel, & Howell, 1968). Multiple researchers report that specific feedback suggesting ways to improve an answer is more beneficial than feedback merely verifying the answer (e.g. Bangert-Drowns, Kulik, Kulik, & Morgan, 1991). This has also been found true for feedback in the context of CbA (Pridemore & Klein, 1995).

However, Goodman and Wood (2004) point out that the assumption that increasing feedback specificity results in enhanced performance and learning "has become an accepted generalization, despite a lack of evidence to support its positive impact on learning" (p. 809). Only the positive effect on immediate or short-term performance is proven and documented in research analyzing feedback specificity (Ilgen, Fisher, & Taylor, 1979; Kluger & DeNisi, 1996). This effect can be attributed to the fact that more specific feedback promotes the capability of giving detailed information about errors. The information is not necessarily limited to the location of the errors but can also introduce corrective actions and show appropriate approaches to improve performance (Payne & Hauty, 1955). Feedback with low specificity, for instance, may solely inform the student that errors occurred or provide general statements about the performance level. High specificity feedback may additionally inform the student about the actions that caused errors as well as how they can be prevented (Goodman, Wood, & Chen, 2011). “Thus, more specific feedback will provide information not only on the outcome of a performance episode, but also on the process, or behaviors, that led to the outcome” (Davis, Carson, Ammeter, & Treadway, 2005, p. 412).

Students may consider feedback lacking in specificity as useless or frustrating since it requires more effort to interpret and understand the message behind unspecific feedback. Furthermore, students may not know how to respond to this kind of feedback. This uncertainty can result in lower levels of learning and infiltrate the motivation of the students to respond to feedback (Shute, 2008). However, with increasing specificity of feedback unfavorable effects may arise as well. While specific feedback may immediately improve performance, it discourages the exploration process which positively affects learning. The undermined learning would be required for the transfer of training and subsequent independent performance. Thus, the transfer to other situations with modified or more challenging tasks is more difficult for the students. Moreover, the discussed benefits of high specificity for performance are restricted in time and do not remain in the long run (Goodman & Wood, 2004; Goodman, Wood, & Hendrickx, 2004). Too specific feedback can even interfere with student performance as it directs the resources of a student away from the task (Kluger & DeNisi, 1996).

The findings of this chapter are consistent with the findings about feedback frequency. Both specificity and frequency improve short-term performance but can negatively affect long-term learning. Therefore, specificity and frequency of feedback are only beneficial to a certain point which should not be crossed (Goodman et al., 2011; Lam et al., 2011). Similarly, immediate feedback tends to lead to a more efficient initial learning process while delayed feedback facilitates the subsequent transfer of learnings (Schmidt et al., 1989; Schroth, 1992). The following Table 3 simplifies how the so far discussed feedback variables timing, frequency and specificity should be designed to achieve either short-term or long-term student performance.

Table 3: Design of the Feedback Variables Timing, Frequency and Specificity

	Short-term performance Efficiency of learning	**Long-term performance** Transfer of learning
Feedback timing	Immediate	Delayed
Feedback frequency	High	Low
Feedback specificity	High	Low

2.2.4 Feedback Complexity

A variable connected to the specificity of feedback is the complexity and length of the provided information. Morrison, Ross, Gopalakrishnan, and Casey (1995) consider feedback types to be complex if they "require additional engagement of the learner through either reading or responding" (p. 32). If feedback is too difficult to understand or simply too long, the addressed student may ignore it. In this scenario, feedback fails to take effect and remains pointless. Moreover, excessively long feedback can negatively affect the strength of the message (Shute, 2008). Thus, what information and how much of it is included in feedback may be crucial to the learning behavior and performance (Kulhavy et al., 1985).

In case that a corrective function is attributed to feedback, Shute (2008) specifies two basic tasks it must fulfill. First, feedback needs to verify if an answer is correct or not (KR). Second, feedback should provide information about the right answer (KCR). Similarly, Jacoby, Mazursky, Troutman, and Kuss (1984) differentiate two different types of feedback: Outcome feedback that verifies the correctness of the answer and cognitive feedback that describes why the answer is correct or not. A more detailed categorization of feedback types based on their complexity is presented in appendix 1. Studies investigating the effect of feedback complexity on learning in terms of the type and amount of involved information are not compatible (Kulhavy, 1977).

2.2.4.1 No or Negative Effect of Complex Feedback

Studies about the complexity of feedback often show no post-test differences. Thus, complex feedback seems to do no more than cause work for its provider and receiver (Kulhavy, 1977). For example, Merrill (1965) concludes that receiving further correction is progressively more time consuming while the student performance remains the same.

To measure the effects of feedback complexity, Kulhavy et al. (1985) constructed four feedback types varying in level of complexity (similar to the feedback types introduced in chapter 2.1.1). These types build upon each other, meaning that every level involves new corrective information while the information of the lower levels is retained. The least complex level is restricted to the verification of the answer (KR). The most complex level includes additional information explaining, for example, why error choices are incorrect or where the correct answer can be found (EF). These different types of feedback were then assigned to experimental student groups after they responded to a set of questions. Comparing the time students needed to read the feedback to the associated post-test corrections, the study's main finding is that feedback complexity is inversely related to the capability of error correction and learning efficiency. Therefore, Kulhavy et al. (1985) conclude that "the most efficient procedure may be that of simply telling learners whether they are right or wrong" (p. 291).

2.2.4.2 Positive Effect of Complex Feedback

According to Merrill and Stolurow (1966), a positive effect on performance can be achieved by inflating feedback to an extend that it functions as review or summary of the lecture. Moreover, Gilman (1969) achieved positive results by providing multiple types of feedback in the context of CbA. In strong contrast to Kulhavy et al. (1985), Gilman (1969) describes that feedback including the reason why an answer is correct “may be of much more value than merely telling him ‘correct’ or ‘wrong’” (p. 6).

The inconsistent results of the past studies discussed above allow no systematic statement about the impact of feedback complexity on student performance. Therefore, Mory (2004) urges researchers “to go back and study further the complexities of feedback” (p. 777).

2.2.5 Goal Orientation

Instead of providing feedback on single responses, goal-directed feedback provides students with information about their development towards certain goals. The expectations that the desired goals can be reached influence the motivation and engagement of students (Fisher & Ford, 1998; Ford, Smith, Weissbein, Gully, & Salas, 1998). Goals that are too ambitious probably result in failure and discourage students. At the same time, goals so low that their achievement is definite lose the ability to motivate students (Shute, 2008). These findings are notable because motivation has a significant impact on the performance of students (Ames, 1992; Covington & Omelich, 1984).

In this context, goal orientation of individual students should be considered during the design of instruction since it can be a “powerful predictor of performance” (Davis et al., 2005, p. 410). Goal orientation describes that individuals are driven to work towards goals either through learning or performance orientation. Learning orientation specifies the extent to which a student desires to develop new competencies in a task believing that intelligence is malleable. Performance orientation is characterized by the desire to demonstrate task competencies and receive favorable evaluation from others believing that intelligence is innate (Button, Mathieu, & Zajac, 1996; Heyman & Dweck, 1992).

The two dimensions of goal orientation induce different reactions to task difficulty and failure. Students with learning orientation are likely to be persistent while facing failure and seek challenging tasks. Students with performance orientation are more likely to withdraw and seek less challenging tasks involving higher chances of success. In accordance with these characteristics, research presents that learning orientation is generally linked to favorable outcomes while performance orientation is associated with negative or ambiguous outcomes (Button et al., 1996; Elliott & Dweck, 1988; Phillips & Gully, 1997; VandeWalle, Brown, Cron, & Slocum, 1999).

Formative feedback can be used as an instrument to influence the goal orientation of a student, preferably from performance to learning orientation. One approach to achieve this is to alter the student’s view of intelligence. Goal-orientation feedback may show a student that skills can be acquired and increased through practice and by investing effort. Moreover, feedback should help to understand that mistakes are not failures but a necessary part of the skill-acquisition process (Hoska, 1993).

Davis et al. (2005) conducted a study analyzing the relationship between goal orientation and feedback specificity (see chapter 2.2.3) and their interactive effects on student performance. The study was built around a computer simulation of management decision-making including three different degrees of feedback specificity. The results showed that if students had no previous experience with the task, feedback specificity achieved the highest impact on performance for students low in learning orientation. If students already gained task experience, feedback specificity had the highest impact on performance for students high in performance orientation. In short, feedback with high specificity is more likely to be beneficial for students who have either low learning or high performance orientation (Davis et al., 2005).

2.3 Design of Computer-based Feedback

After discussing the design of feedback in general, this chapter focuses explicitly on the provision of feedback in a computer-based learning environment. The two following frameworks demonstrate how feedback should be implemented in a software application to maximize student perfomance.

2.3.1 A Framework Proposed by Mason and Bruning

CbA typically implies an independent learning environment with little human interaction. This kind of environment probably requires an enhanced focus on feedback compared to a conventional environment in form of a classroom. Therefore, designers of CbA need to be careful which types of feedback they include in their programming (Mason & Bruning, 2001).

Mason and Bruning reviewed and summarized available research about computer-based feedback. Moreover, they categorized feedback in different groups depending on the degree of verification and specificity. On this basis, they propose a framework for the design of effective feedback and its inclusion in educational software and programs. Said framework addresses not only lecturers but also programmers and instructional design specialists. Accounting for various feedback conditions, the framework presents different variables to define the appropriate type of feedback. These variables include student achievement, task level (complexity of a task), feedback timing, prior knowledge and learner control. Latter variable describes to which extend students can choose the type and elaboration of feedback. The framework considers the level of student achievement in the beginning because the students' expertise may influence their capability to make use of different types of feedback (Mason & Bruning, 2001). Figure 1 shows Mason and Bruning's framework to facilitate the decision-making about computer-based feedback.

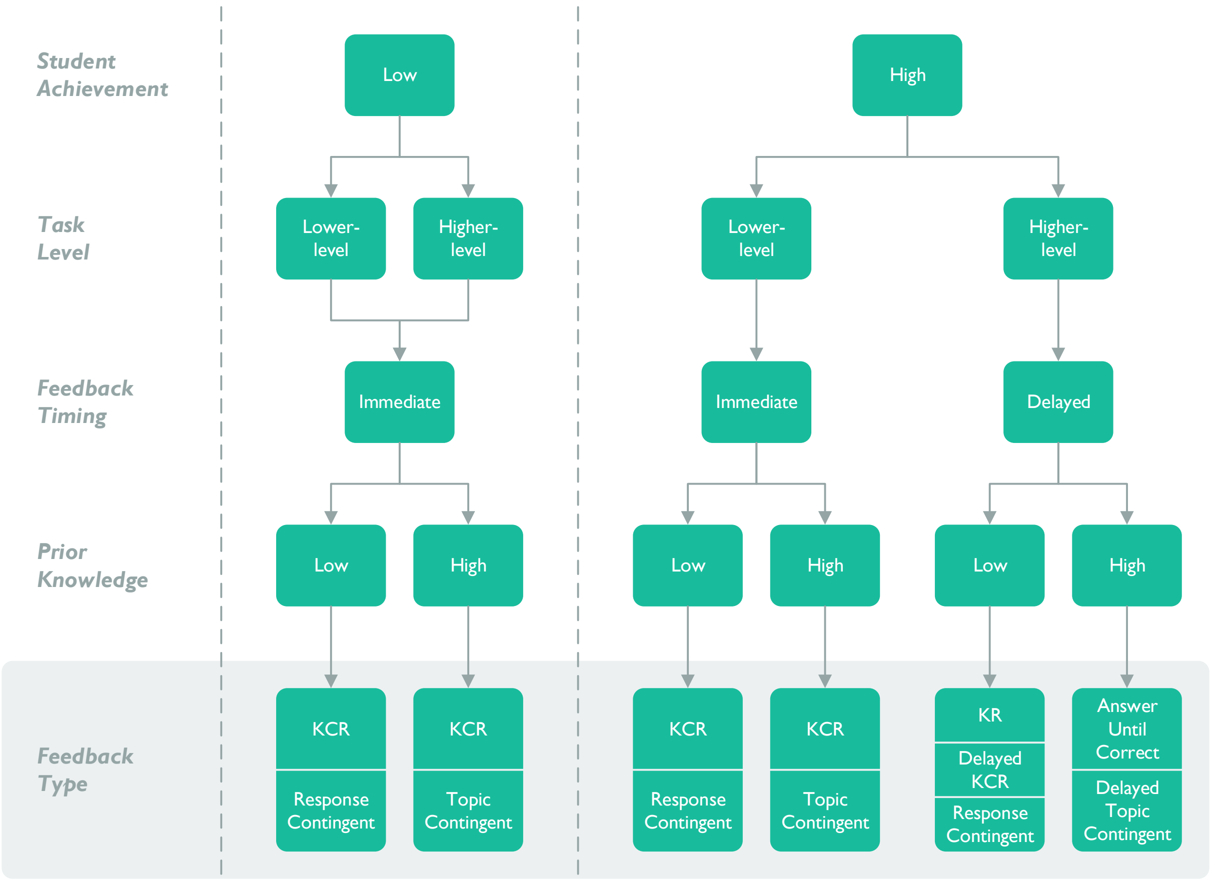

Figure 1: Framework to Determine Feedback Types, Adapted from Mason and Bruning (2001)

The resulting feedback types of the framework correspond with the ones defined in the beginning of this thesis (see chapter 2.1.1). However, Mason and Bruning further subdivide EF based on the level of verification and elaboration into response- and topic-contingent feedback. Response-contingent feedback explains why a specific student response is wrong or correct while topic-contingent feedback provides the student with general information to the topic being covered without revealing the correct response. Moreover, with answer-until-correct Mason and Bruning apply a modification of KR in their framework. In the context of answer-until-correct, students have multiple attempts to answer an item while the feedback is restricted to KR, meaning that students have to find the correct response on their own by trying again (Mason & Bruning, 2001). These feedback types are among the ones defined by Shute (2008) that are presented in appendix 1.

Regarding the discussion about timing of feedback (see chapter 2.2.1), Mason and Bruning emphasize that immediate and delayed feedback can and should be implemented simultaneously in a single program. "In this manner, students can have immediate knowledge of the correctness of their response but still have time to think about the error before informational feedback is given" (Mason & Bruning, 2001, p. 14). Moreover, they suggest including multiple types of EF which can be applied according to the achievement level of the individual student. This suggestion is supported by previous research that approves the potential of applying various feedback types (e.g. Morrison et al., 1995; Pridemore & Klein, 1995).

2.3.2 A Framework Proposed by Narciss and Huth

Narciss and Huth (2004) recognized as well that informative feedback plays a major role in computer-based learning environments. Therefore, they propose a framework for the design of "psychologically well-founded informative tutoring feedback forms" (Narciss & Huth, 2004, p. 2). This framework is based on past studies as well as cognitive task and error analyses. However, in their research, Narciss and Huth not only define design principles but also show ways to implement them. Moreover, the research involves computer-based learning experiments measuring the effect of the proposed feedback forms.

The framework looks at feedback from different perspectives instead of just analyzing its information value. The resulting multi-dimensional view consists of three major factors of relevance. The first factor accounts for the nature and quality of the feedback message itself. The other two factors consider characteristics of the instructional context and characteristics of the student. This view of feedback is consistent with researchers urging to design feedback in consideration of the student's individual needs (Smith & Ragan, 1993). The feedback message is analyzed regarding the three facets function, content and presentation. Said facets define the quality of feedback. Even though in past research these facets were normally not distinguished, they are vital for feedback design since, depending on the function of feedback, different contents and presentation forms are required. Regarding the specification of feedback content, Narciss and Huth (2004) present design principles in form of five successive steps shown in Figure 2.

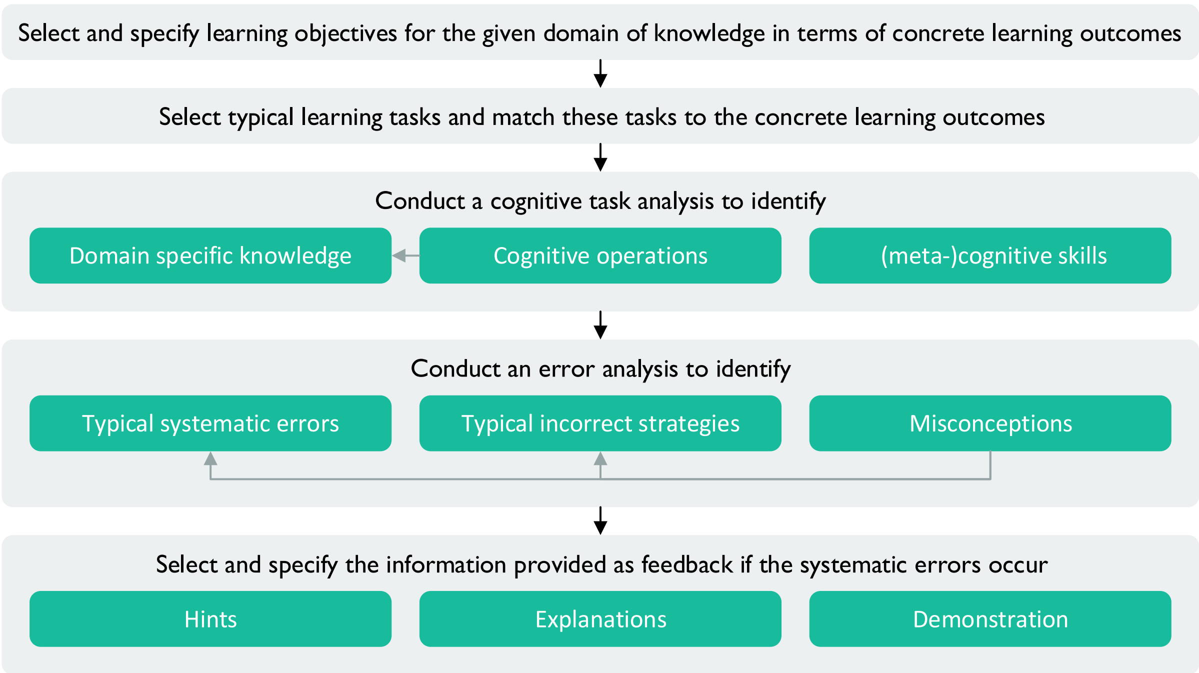

Figure 2: Design Principles for the Content of Feedback, Adapted from Narciss and Huth (2004)

Regarding the presentation of feedback, Narciss and Huth (2004) introduce the following guidelines.

1. Do not provide feedback especially KCR before learners have tried to solve the learning tasks on their own.
2. Do not immediately combine EF components with the correct response or solution (KCR).
3. Provide the EF information stepwise in manageable pieces and offer the opportunity to apply the information provided on a second or multiple try.
4. Implement a mastery level in order to check if a specific learning criterion (e.g. the correct application of a procedural rule) is achieved.
5. Use the potential of multi-media systems in order to avoid interferences or perceptual and cognitive overload due to modality effects.

Exemplarily, Narciss and Huth (2004) applied the defined design principles in a computer-based environment in the form of bug-related feedback. Bug-related feedback directly provides the student with information in case procedural errors occur (Schimmel, 1988). The intention is to offer students just as much information needed to correct the errors by themselves. This type of feedback is only applicable if the computer can identify specific errors caused by students. However, in comparison to yes/no, correct answer and explanatory feedback, it is considered the most useful type of feedback for learning enhancement (Overbaugh, 1994).

Following their own guidelines for the presentation of feedback, Narciss and Huth (2004) incorporated feedback components in an adaptive feedback algorithm. This incremental algorithm involves three additive levels of feedback presented as part of a multiple try strategy. The first time students answer incorrectly, they receive feedback in form of KR. If the reconsidered answer to the same task is incorrect again, bug-related feedback is presented in addition to KR. In case students are unsuccessful even at the third attempt, the feedback not only involves KR and bug-related information but also KCR (Narciss & Huth, 2004).

To verify the impact of the bug-related feedback algorithm, Narciss and Huth (2004) executed two studies. The studies aimed to reveal differences in efficiency between the bug-related feedback algorithm and a standard feedback algorithm. The results of the studies present that bug-related feedback facilitates both error correction and task completion. Moreover, it enhanced the motivation and the performance in delayed assessments. These findings contradict previous studies arguing that more complex feedback does not lead to increased student performance and, thus, is merely a waste of time (see chapter 2.2.4.1). One reason behind these differences may be that previous studies focused on achievement and neglected the effect on motivation (Narciss & Huth, 2004).

2.4 Social Comparison Theory

Originally introduced by Festinger (1954), the social comparison theory analyzes the habit of humans, or other species (Gilbert, Price, & Allan, 1995), to compare themselves to others in order to allow an accurate self-evaluation. In the social comparison process, individuals try to resolve uncertainties about their own opinions and abilities (Festinger, 1954).

Social comparison manifests itself in two different directions. Downward comparison of the self involves other people that are perceived to be worse off while upward comparison involves people that are thought to be better off (Buunk, Collins, Taylor, VanYperen, & Dakof, 1990). In the context of downward comparison, people can enhance their subjective well-being by comparing themselves to others who are less fortunate (Wills, 1981). As a result, downward comparison can be a way of dealing with problems simply by realizing that someone else's problems are more severe (Taylor, Wood, & Lichtman, 1983). In contrast, upward comparison can be painful (Garcia, Tor, & Gonzalez, 2006). However, past research in the academic sector shows that upward comparison leads to better results than downward comparison. Blanton, Buunk, Gibbons, and Kuyper (1999) observed in their experiment that students who compared themselves with well performing students increased their own academic performance the most during semester.

Both downward and especially upward comparison can lead to competitive behavior. Downward comparison triggers competitive behavior to retain a superior position while upward comparison generates competitive behavior to achieve a superior position. Social comparison fosters competition not only in school or at work but also in the social life. The reason being that humans generally strive for a superior position than their counterpart – regardless of the context (Garcia, Tor, & Schiff, 2013). Festinger (1954) describes this urge of being better and better in terms of abilities as "unidirectional drive upward" (p. 124).

2.5 Feedback Guidelines

Table 4 on the next page summarizes the insights gathered through the literature review by proposing guidelines for the design of formative feedback. These guidelines are structured according to the theoretical part of this thesis and include references to the chapters they are based upon. As a result, the guidelines involve suggestions for the design of formative feedback in general as well as in the context of a computer-based learning environment. Within the table, equivocal findings are not considered, and the literature references are not exhaustive, but representative.

Table 4: Feedback Guidelines

Guideline		Chapter
G01	CbA as well as feedback need to be equitable and valid (Brown et al., 1996; Higgins & Bligh, 2006).	2.1.2.2 Computer-based Assessment
G02	Do not restrict feedback to lower-level learning but also provide feedback that supports higher-level learning (Krathwohl, 2002).	2.1.5 Lower- and Higher-level Learning
G03	Choose immediate feedback if you aim for an efficient initial learning process and delayed feedback in order to facilitate the transfer of learning (Corbett & Anderson, 2001; Schmidt et al., 1989; Schroth, 1992).	2.2.1 Feedback Timing
G04	Use immediate feedback at the task level and delayed feedback at the process level (Hattie & Timperley, 2007).	
G05	Apply immediate feedback for low-achieving students and delayed feedback for high-achieving students (Mason & Bruning, 2001).	
G06	Provide feedback more frequently if you strive for efficiency in the short run and less frequently if you focus on facilitating the transfer to other tasks (Schroth, 1997).	2.2.2 Feedback Frequency
G07	Provide feedback with high specificity to increase short-term performance and feedback with low specificity to enhance independent long-term performance (Goodman et al., 2011).	2.2.3 Feedback Specificity
G08	Design feedback as simple as possible but no simpler (Shute, 2008).	2.2.4 Feedback Complexity
G09	Make use of formative feedback to alter a student's goal orientation from performance to learning orientation (Hoska, 1993).	2.2.5 Goal Orientation
G10	Offer more specific feedback to students who have low learning orientation or high performance orientation (Davis et al., 2005).	
G11	Implement immediate and delayed feedback simultaneously to exploit the full potential of feedback (Mason & Bruning, 2001).	2.3.1 A Framework Proposed by Mason and Bruning
G12	Apply different feedback types according to the achievement level of a student (Morrison et al., 1995; Pridemore & Klein, 1995).	
G13	Consider characteristics of the instructional context and characteristics of the student when providing feedback (Narciss & Huth, 2004).	2.3.2 A Framework Proposed by Narciss and Huth
G14	Do not restrict the mode of feedback presentation to text but use the potential of multi-media systems (Narciss & Huth, 2004).	
G15	Only provide feedback after the student has attempted a solution (Narciss & Huth, 2004).	
G16	Present elaborated feedback in manageable portions (Narciss & Huth, 2004).	
G17	When allowing for social comparison let students compare themselves with better performing students (upward comparison) to increase their own performance (Blanton et al., 1999).	2.4 Social Comparison Theory

3 Methodology

The present research is based on the action design research (ADR) method proposed by Sein et al. (2011). The ADR method is suitable for this research because it provides a scientific approach that allows to solve current issues of practitioners by developing an innovative IT artifact. Moreover, this method ensures that learnings and insights gained through interventions during the development process are gathered to eventually generate knowledge applicable to a group of problems (Sein et al., 2011).

The first of four stages proposed by the ADR method consists of the problem formulation which can be inspired by practice and/or theory. This initial stage defines the problems which the artifact should solve and, thus, declares the reason behind the research (Sein et al., 2011).

In the second stage of ADR, the problem formulation is used as a basis for the initial draft of the artifact. Afterwards, the artifact is further formed through organizational intervention and evaluation in an iterative process. Consequently, the second stage is called building, intervention, and evaluation (BIE). This stage reveals the main source of innovation which can be found within a continuum between IT-dominant BIE and organization-dominant BIE. The position of an artifact along this continuum defines which of both BIE forms is more suitable for a project. In case of an IT-dominant BIE, the focus of the project lies on innovative artifact design. If the BIE is organization-dominant, innovation originates from organizational intervention (Sein et al., 2011).

The third stage concentrates on making a solution built for a specific case applicable to a wider category of problems. This stage is realized in parallel to stages one and two since it involves continuous and conscious reflection of the problem and the emerging solution. That way, it is assured that contributions to knowledge occurring in the course of the ADR project are recognized and captured (Sein et al., 2011).

The fourth stage of ADR emphasizes the formalization of learning. In this stage, the specific knowledge gained during the research is further elaborated and defined more broadly to solve a group of field-problems. In other words, both the solution and the problem are generalized to a more generic form. This final stage summarizes the findings and achievements of the research based on the IT artifact and discusses the organizational outcomes. From those outcomes researchers can derive design principles which provide guidance in the process of creating solutions for a predefined class of problems (Sein et al., 2011).

Fellow researchers of the Institute of Information Management of the University of St.Gallen already started the ADR project by defining the problem (stage 1) and deriving requirements for the artifact (Rietsche et al., 2017). At this point of the research, the artifact is being built, deployed and evaluated (stage 2) while learnings are gathered (stage 3). The formalizing of learning (stage 4) is beyond the scope of this thesis because the artifact in its final version will not be finished by the time the paper is submitted. Therefore, the learning process is still in progress and the definition of design principles would be too early. As a result, the focus of this thesis lies on the second and third stage of ADR.

4 Design and Evaluation

Following the ADR method, the design of the artifact starts with the problem formulation before the implementation begins. However, the implementation can still affect the initial problem formulation. During the process of problem formulation and implementation, learnings are gathered and used to further improve the artifact. This chapter is structured according to this approach. It starts by discussing the problem and goes over to documenting the BIE stage while defining the learnings and how they affect the further development of the artifact.

4.1 Problem Formulation

Nowadays, higher education is characterized by a high ratio between students and lecturers (Fortes & Tchantchane, 2010). The University of St.Gallen, for example, counted 8’337 students by the end of 2016 while 127 lecturers were employed. In other words, there were about 65 students for every lecturer (University of St.Gallen, 2017b, 2017a). At the same time, the number of matriculated students rises from year to year making it hard for universities to keep pace (Nicol & Macfarlane-Dick, 2006). While being cost-effective, the high and rising student-lecturer ratio turns the provision of feedback to individual students into a yet unresolved challenge (Miller, 2009). As a result, higher education typically does not involve any formative feedback but only summative feedback at the end of a course when it is too late for students to improve their performance (Brown et al., 2016).

Prior to this thesis, researchers of the Institute of Information Management of the University of St.Gallen gathered information about field problems in the context of formative feedback through six semi-structured interviews with practitioners and end-users. The interview partners consisted of lecturers as well as students. The insights gained through the interviews confirm the circumstances discussed in the previous paragraph. Therefore, not only researchers but also practitioners recognize the problem of insufficient formative feedback in the learning process of students. The fact that the emerging artifact generates knowledge based on field problems results in a research that follows a practice-inspired approach (Rietsche et al., 2017).

4.2 IT-Dominant Building, Intervention, and Evaluation

This chapter focuses on building, intervention and evaluation (BIE) regarding the software artifact LOOM (standing for “learning objective and outcome manager”). In this project, an innovative technological design of LOOM has higher priority than innovation through organizational intervention. Therefore, IT-dominant BIE cycles are applied. The IT-dominant approach involves an alpha version and a more mature beta version of LOOM used for recurring interventions and evaluations. As a result, the second ADR stage of this project consists of two BIE cycles in which LOOM is developed (Sein et al., 2011). Figure 3 visualizes the IT-dominant BIE and shows at which point of the project the two BIE cycles take place. The first cycle is partly based on the findings of Rietsche et al. (2017). The learnings gathered in the first cycle then serve as basis for the second cycle which results in LOOM's beta version.

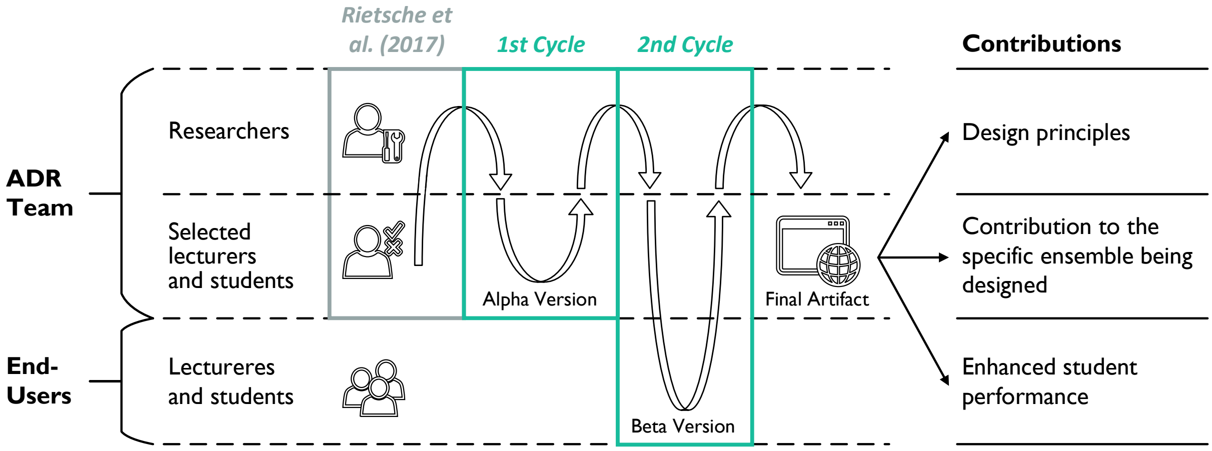

Figure 3: IT-Dominant BIE for the Development of LOOM, Adapted from Sein et al. (2011)

To demonstrate the functionalities of LOOM, screenshots of its front-end are integrated in this chapter. These screenshots are based on the interventions of LOOM in the context of real university courses. Since the courses are held in German while the initial versions of LOOM are released in English, the screenshots may involve content in both these languages (e.g. Figure 8). Nevertheless, the screenshots fulfill their purpose of providing a visual impression of LOOM.

4.2.1 Fundamentals of LOOM

Before the BIE cycles are discussed, this chapter introduces the emerging artifact LOOM by briefly discussing its fundamental software architecture and describing its data structure.

4.2.1.1 Software Architecture

LOOM is deployed by GlassFish Server implementing Java EE 7. On the server-side, no frameworks of third-party providers are applied. On the client-side, jQuery and Bootstrap are in use (see chapter 4.2.2.1). The development of LOOM is based on the design pattern model-view-controller (MVC). All user requests are accepted by a main controller on the server. The primary task of the main controller is to analyze the incoming requests and distribute them among specialized modules which are responsible for different functions. Every of these modules consists of a controller, a data model and one or multiple views.

The controllers of the modules receive the user requests from the main controller. The user requests are forwarded until they eventually reach the function in charge for their processing. Thereby, the object-oriented data model represents how the data is structured and enables the communication with the underlying SQL database (DB). The responsible controller then forwards the response to the view where the data is formatted and transferred to the user. Figure 4 summarizes how LOOM manages user requests and reveals the composition of its modules.

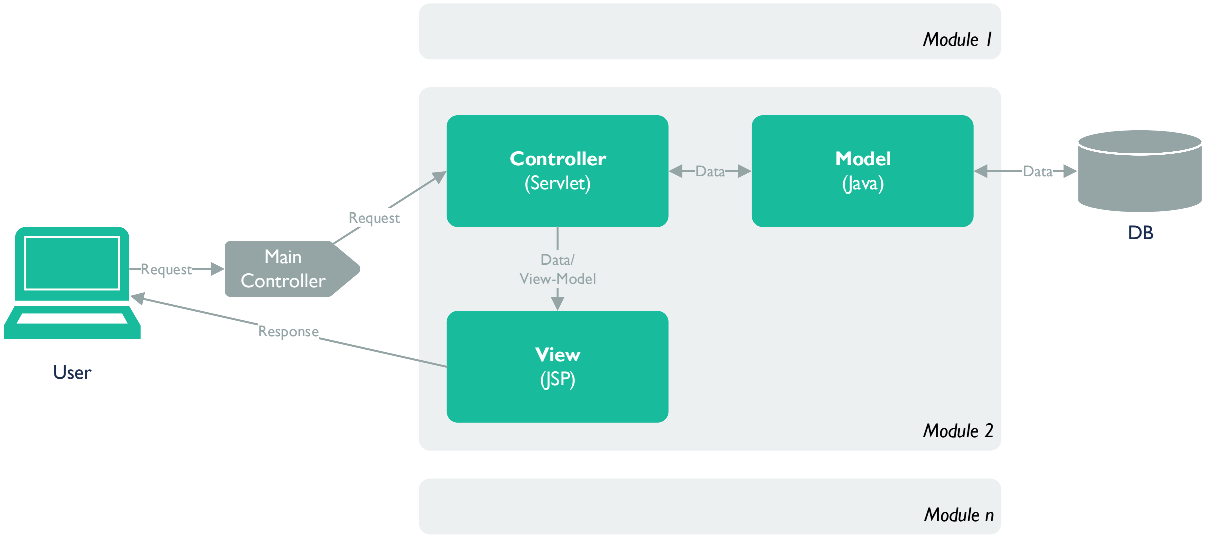

Figure 4: MVC Pattern Applied in LOOM

4.2.1.2 Structure

The basic levels in LOOM are courses, learning units (LU), learning outcomes (LO) and computer-based assessments (CbA) including items of type multiple-choice question (MCQ) or flaw diagram (FD). FDs may involve MCQs and/or free text questions (FTQ). Using crow's foot notation, the entity-relationship model in Figure 5 shows the section of the DB that represents the relationship between these elements.

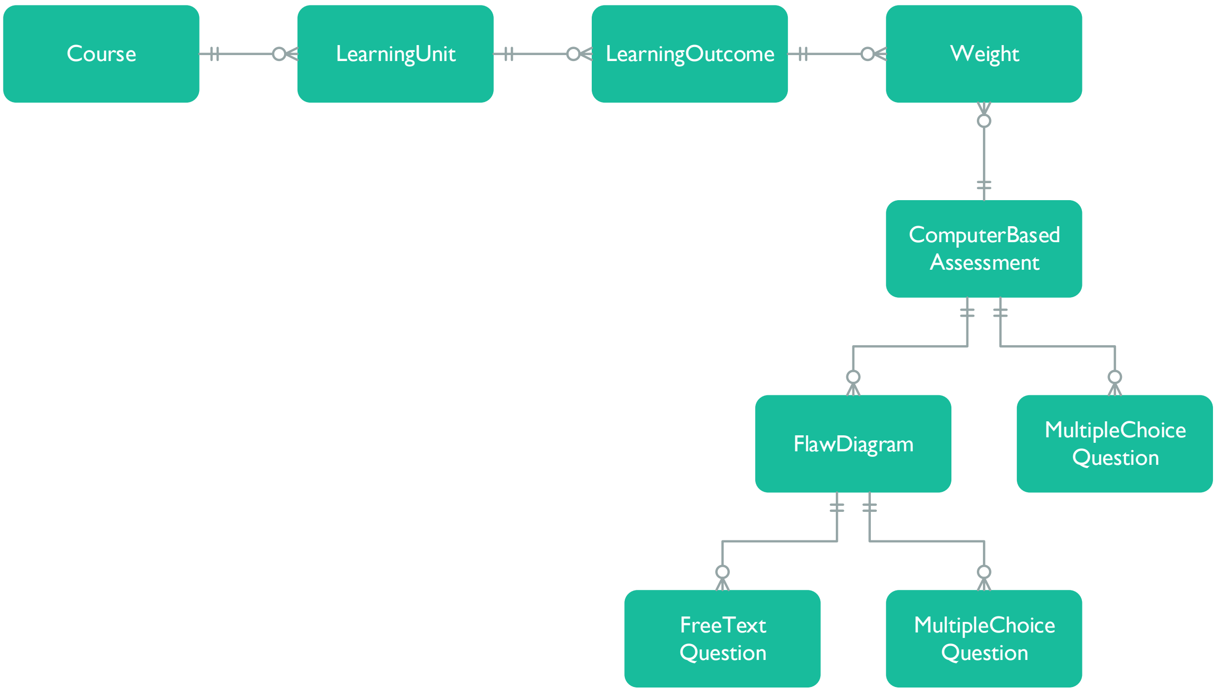

Figure 5: Entity-Relationship Model Representing a Part of LOOM's Structure

Every course in LOOM can have multiple LUs. An LU normally represents one lecture and the topics covered in it. In turn, every LU can have multiple LOs. An LO defines a goal pursued by the students within an LU and is usually formulated in one sentence. Every LO can involve multiple CbAs checking the knowledge of students. However, a CbA is not necessarily assigned exclusively to one LO. It can also be integrated in other LOs which may belong to different LUs or courses. The lecturers weight CbAs when assigning them to an LO by allocating a number between 10 and 100 percent. Depending on a CbA's weight, the points of its questions have a higher or lower impact on the overall result of the LO and, consequently, the LU.

A CbA can either consist of MCQs or FDs. When creating an MCQ, lecturers enter a question and offer multiple response options for the students to choose from. Depending on the form of the question selected by the lecturers, either one or multiple response options can be correct. An FD basically consists of an image with incorporated errors. Students need to find these errors within a limited number of attempts and answer associated questions. In order to enable this, lecturers upload and crop flawed images and then mark the faulty areas. To each of those incorporated errors, lecturers can either enter an MCQ or an FTQ.

4.2.2 First BIE Cycle

The researchers of the Institute of Information Management mentioned in chapter 4.1 transformed the information about field problems into requirements for LOOM. Based on those requirements, they defined design elements (DE) for the initial version of the artifact. This groundwork is not part of the present thesis. Nevertheless, its results are used as a starting point for the development of the artifact. Five out of the six originally defined DEs are pursued in the development of LOOM. These predefined DEs are complemented by additional DEs that emerged in the context of this thesis. Table 5 summarizes all DEs covered in the first BIE cycle. Within this cycle, two primary user roles must be considered: on the one side, lecturers who create CbAs and, on the other side, students who complete them. Besides describing the DEs, the upcoming table categorizes them regarding the user role(s) they affect. Furthermore, the last column of the table indicates which DEs are implemented in the context of this thesis. Note that the first DE "Web-based application with responsive UI" is overarching and must be considered in the implementation of every other DE.

DE03 as well as DE05 to DE07 (displayed in *italics* in Table 5) were newly added to the list while the remaining DEs were defined from the beginning. However, some of the original DEs changed. First, DE02 "Wizard for Lecturers", formerly called "Walkthrough Wizards", involved an additional wizard for the import of LOs from lecture slides. However, this wizard has low priority for lecturers and is, therefore, currently not considered in the implementation of LOOM. Second, DE04 "Course Enrollment", formerly called "Assessment Questions", was limited to the assessment of a student's competitiveness during the course enrollment process. This DE was extended and now describes the whole enrollment process including a question regarding the students' competitiveness. Moreover, the sequence of the DEs was altered so it resembles the process implemented in LOOM from creating a

CbA as lecturer over participating in it as student up to reviewing the performance both as lecturer and student. All DEs are discussed separately and in more detail in the upcoming chapters.

Table 5: Design Elements of the First BIE Cycle, Adapted from Rietsche et al. (2017)

Design element (DE)		User role(s)	Description	Implementation within this thesis
DE01	Web-based Application with Responsive User Interface	Lecturer, Student	LOOM is based on the latest web technologies and offers a responsive user interface for the different screen sizes of smartphones, tablets or notebooks.	(✓)
DE02	Wizard for Lecturers	Lecturer	LOOM provides a walkthrough wizard to the lecturer during the creation of LOs and CbAs.	✓
DE03	*Taxonomy Table*	*Lecturer*	*The lecturer uses a Taxonomy Table to define objectives for an LO, to classify CbAs and to check if the objectives are achieved.*	✓
DE04	Course Enrollment	Lecturer, Student	Students are enrolled by themselves or by lecturers. During this process students answer a question concerning their competitiveness.	
DE05	*Wizard for Students*	*Student*	*LOOM provides a walkthrough wizard to the student during the assessment process.*	✓
DE06	*Self-Assessment*	*Student*	*The student assesses his or her current knowledge regarding an LO before taking CbAs.*	✓
DE07	*Computer-based Assessment*	*Student*	*The student participates in a CbA created by a lecturer.*	✓
DE08	Feedback	Student	LOOM provides feedback based on preceding SA and CbA.	✓
DE09	Student Performance Charts	Lecturer, Student	The lecturer gets a chart of the aggregated student performance per LO. The student gets a graphical view of his or her performance.	

4.2.2.1 Web-based application with responsive UI (DE01)

LOOM should be available for lecturers and students from anywhere at any time independent of the operating system in use. To meet this requirement, LOOM is implemented as a web-based application accessible via web browser. Moreover, students should be able to efficiently use LOOM from any web-enabled device regardless of its screen size. To support various screen sizes, LOOM incorporates a responsive user interface (UI). As a result, users can access LOOM with different devices such as notebooks, tablets or smartphones running on different operating systems such as Windows, macOS, iOS or Android.

The front-end of LOOM is developed with Hypertext Markup Language version 5 (HTML5), Cascading Style Sheets (CSS) and JavaScript (JS). To achieve a fully responsive UI, LOOM makes use of the front-end web framework Bootstrap 3. The plugins of Bootstrap are based on the JS library jQuery which is also integrated in LOOM. However, jQuery is not only used to run Bootstrap but also to facilitate the programming of LOOM's front-end with JS.

The consistent use of the Bootstrap framework affects the appearance and structure of the whole front-end. Among the components influenced by Bootstrap are (Bootstrap terms in brackets) the navigation header (navbar), dialog and pop-up boxes (modals and popovers) and alert messages (alerts). Moreover, Bootstrap offers design templates for various front-end elements including typography, forms, tables and buttons. At the same time, Bootstrap's grid system ensures that the content of LOOM's webpages is optimally presented on each screen size by applying the concept of liquid layout.

Additionally, every plugin or library integrated into LOOM supports responsive UI design. The ability to flawlessly adapt to different screen sizes represents a main criterion in the selection of these software components. Figure 6 shows the front-end of LOOM on a smartphone, tablet and notebook. Students face these views during the assessment process of an LU.

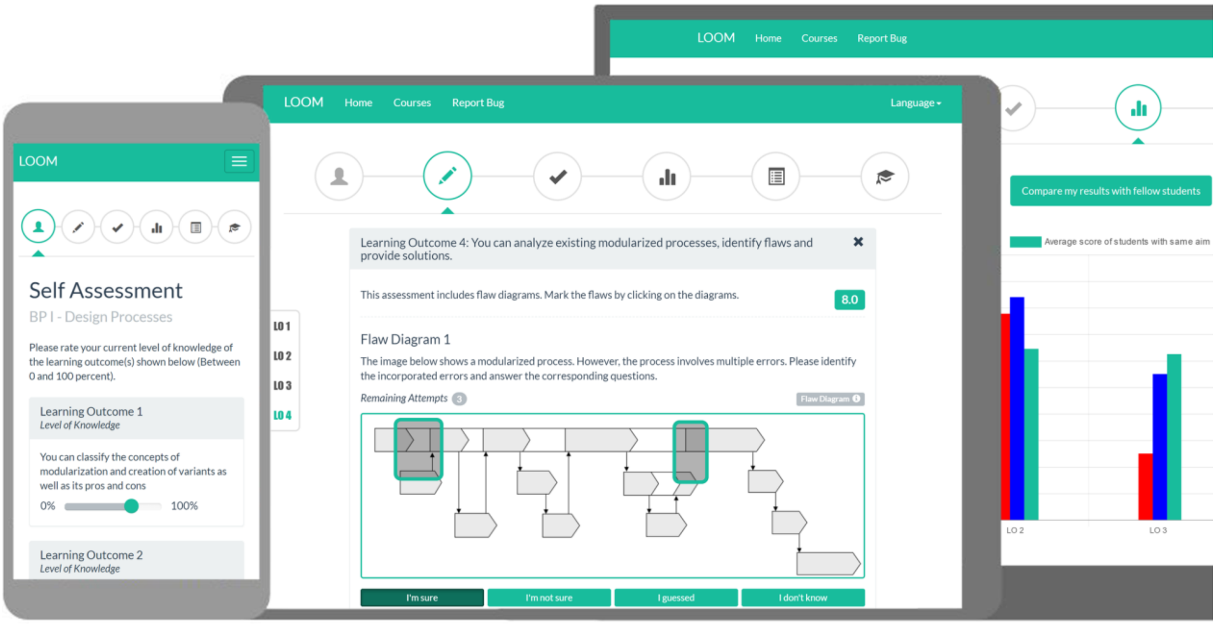

Figure 6: LOOM's Front-End for Students on a Smartphone, Tablet and Notebook

4.2.2.2 Wizard for Lecturers (DE02)

After defining a course with at least one LU, a lecturer can begin to create LOs and CbAs. When creating a new LO, LOOM automatically starts a walkthrough wizard. This wizard guides lecturers from the creation of the LO through the assignment of CbAs up to the creation of new or the modification of existing CbAs. These three visualized steps building on one another provide a simple way to use LOOM and help lecturers to better understand and manage the relationship between LOs and CbAs. Figure 7 shows the navigation of the wizard followed by an explanation of the individual steps.

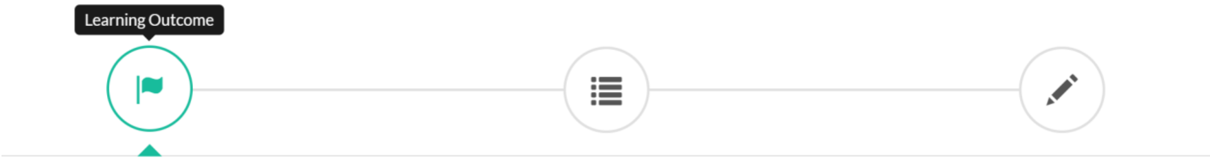

Figure 7: Wizard Navigation for the Creation of Learning Outcomes

Step 1: Creation of Learning Outcome

The creation of an LO involves its assignment to an LU. LOOM already preselects an LU according to the context in which the wizard is called. However, lecturers have the possibility to change this selection. Moreover, lecturers categorize the specific objectives of an LO in a Taxonomy Table. Afterwards, they assign new or existing CbAs to the LO. While assigning CbAs to the LO, the lecturers recognize through the Taxonomy Table if the CbAs achieve the defined objectives. The Taxonomy Table is discussed in detail in chapter 4.2.2.3. To allow an efficient assignment of CbAs, LOOM offers a search function which dynamically alters the list of available CbAs according to the user input. As soon as lecturers select a CbA, it is removed from said list and transferred to a table containing all CbAs assigned to the LO (see section 2 in Figure 10).

Step 2: Management of Assigned Computer-based Assessments

In a next step, the wizard provides the lecturer with an overview of the assigned CbAs. At this point, the lecturer has multiple options to choose from. The main options involve creating, modifying or deleting CbAs. If the lecturer decides to create or modify a CbA the last process step is invoked.

Step 3: Creation or Modification of Individual Computer-based Assessment

Within this last step, CbAs are managed on an individual basis. A switch defining if a CbA is active or not gives lecturers the possibility to exclude certain CbAs from the assessment process of the students. Furthermore, lecturers can select the item type which is included in the CbA. Depending on the selection, LOOM automatically prefills the CbA's Taxonomy Table indicating which objectives are being covered. However, lecturers can alter this suggestion of LOOM. After choosing the item type, lecturers can either add items of the type MCQ or FD to the CbA. If they choose to add MCQs, they are asked to weight each question individually with points. In case of FDs, LOOM distinguishes between the number points achievable for finding errors and for correctly answering the corresponding questions. Lecturers determine both of those numbers for every FD item. Additionally, they can define the maximum number of attempts students have to find errors within a FD (see Figure 8).

Including images is optional for MCQs while it is mandatory for FDs. Lecturers can either upload new images from their computer or select from the existing ones on the webserver. The component responsible for the upload of images is discussed in the following chapter. After selecting an image for a FD, LOOM asks the lecturer to mark the area of the image that is required. LOOM crops the image accordingly, embeds the cropped image in the FD item and stores it on the webserver. Subsequently, lecturers mark the faulty areas in the embedded image with their cursor as illustrated in Figure 8. To every area they can define a question of type multiple-choice or free text.

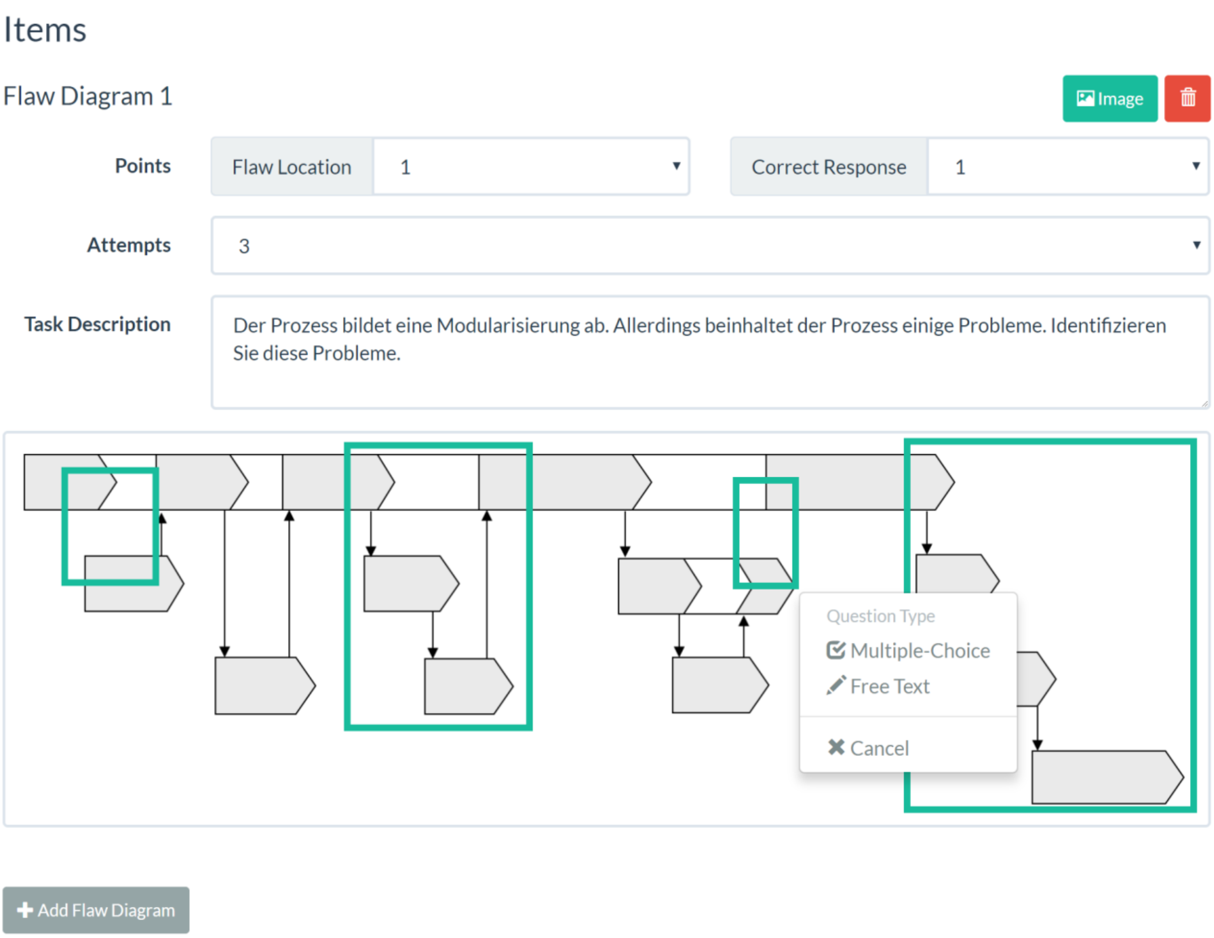

Figure 8: Creation of a Flaw Diagram Item in LOOM

The wizard allows lecturers to navigate back to one of the preceding steps with a single click. However, moving forward in the process is only possible when LOOM successfully validated the user entries. Lecturers can also directly create CbAs without going through all the steps of the wizard by accessing a page designed especially for this purpose. This alternative way of creating CbAs is intended for lecturers who are already experienced with LOOM and do not require the wizard anymore. In this context, the form for defining a CbA includes an additional drop-down list to assign the CbA to an LO. If a lecturer tries to save a CbA that is assigned to an LO involving other CbAs, a dialog box is called. This dialog box lists all the assigned CbAs including the one being created and asks the lecturer to reallocate the weights, so they do not exceed 100 percent. Only when this requirement is met, LOOM grants the creation of a CbA.

Image Uploader

As discussed in the previous chapter, images are primarily necessary for FDs but can also be used to enrich MCQs. The component responsible for enabling lecturers to save images on the webserver is called image uploader. This uploader can either be accessed directly via navigation header or within the process of creating an FD or MCQ. Lecturers can choose one or multiple images for the upload by selecting them in the file explorer or by simply dragging and dropping them into LOOM. The explorer only shows image files for selection while the drag and drop functionality verifies the data format of the selected files when they are dropped into LOOM. This is to ensure that users only upload files in form of images to the webserver. Figure 9 shows the dialog box allowing the upload of images.

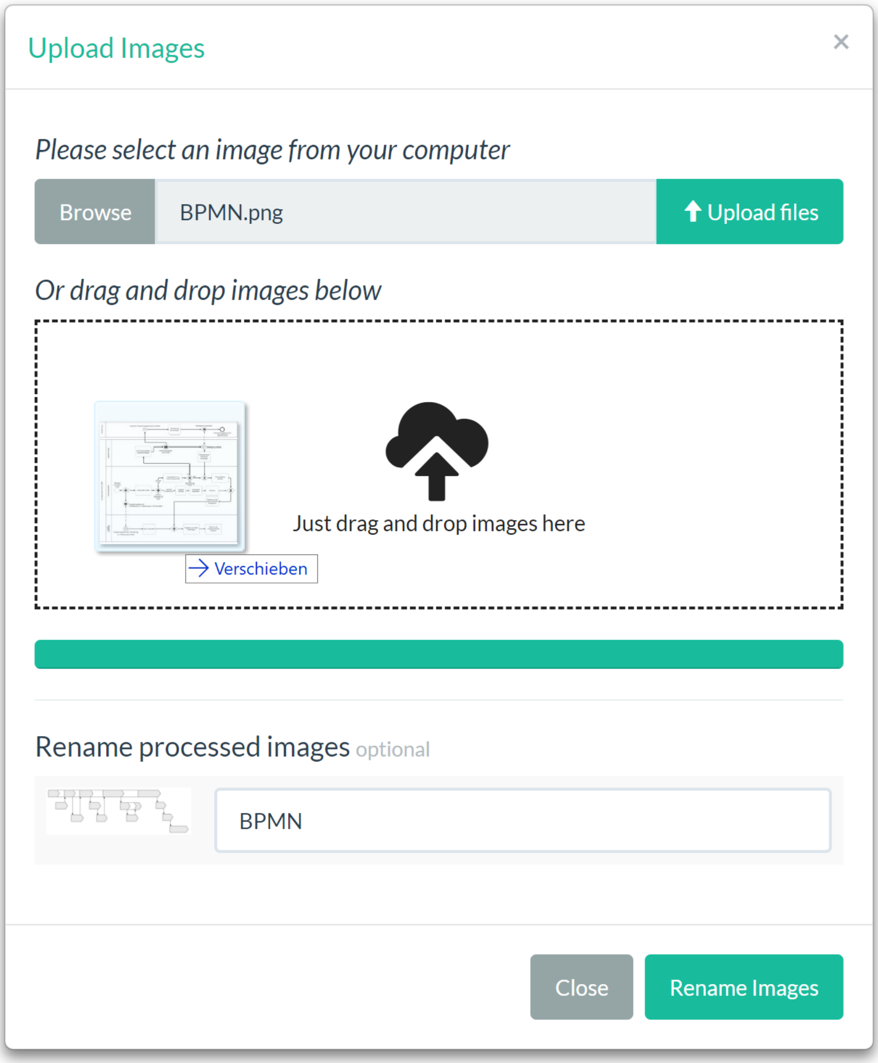

Figure 9: Image Uploader of LOOM

LOOM stores uploaded images in the file system of the server. Simultaneously, it creates corresponding DB entries with references to the new image files. The default name of the images in LOOM is based on the name of the original files but users can change it directly after the upload. The file name in the file system of the server, however, matches the ID of the DB entry and cannot be changed. That way, unique file names are guaranteed at all time. When an image is cropped during the creation of an FD, LOOM generates a new image file and stores it in the server file system so that the original image is reusable. At the same time, a reference from the FD to the new file is stored in the DB. Similarly, if an image is added to an MCQ, LOOM writes a reference to the image into the DB entry of the MCQ.

A team member implemented the functionality responsible for creating MCQs. All remaining functionalities of the DE discussed in this chapter were implemented by the author of this thesis.

4.2.2.3 Taxonomy Table (DE03)

Since the feedback generated by LOOM is to a large extent based on CbA, the quality of CbA is crucial for the quality of feedback. Regarding CbA, it is important to bear in mind that the structure and content of each CbA depends on the lecturer designing it. Therefore, LOOM provides lecturers with an instrument that supports them in the process of designing CbA and that ensures that they meet the targeted learning objectives. Lecturers use the Taxonomy Table to define which objectives they want to achieve when creating an LO. For those unfamiliar with the revised Taxonomy, LOOM offers an assistant that determines the right placements in the Table by posing a few cumulative questions. In the process of assigning CbAs to an LO, the lecturers then continuously see in the Taxonomy Table if the selected CbAs correspond with the defined objectives (see Figure 10). If the user tries to create an LO without meeting all the objectives, a warning is being displayed.

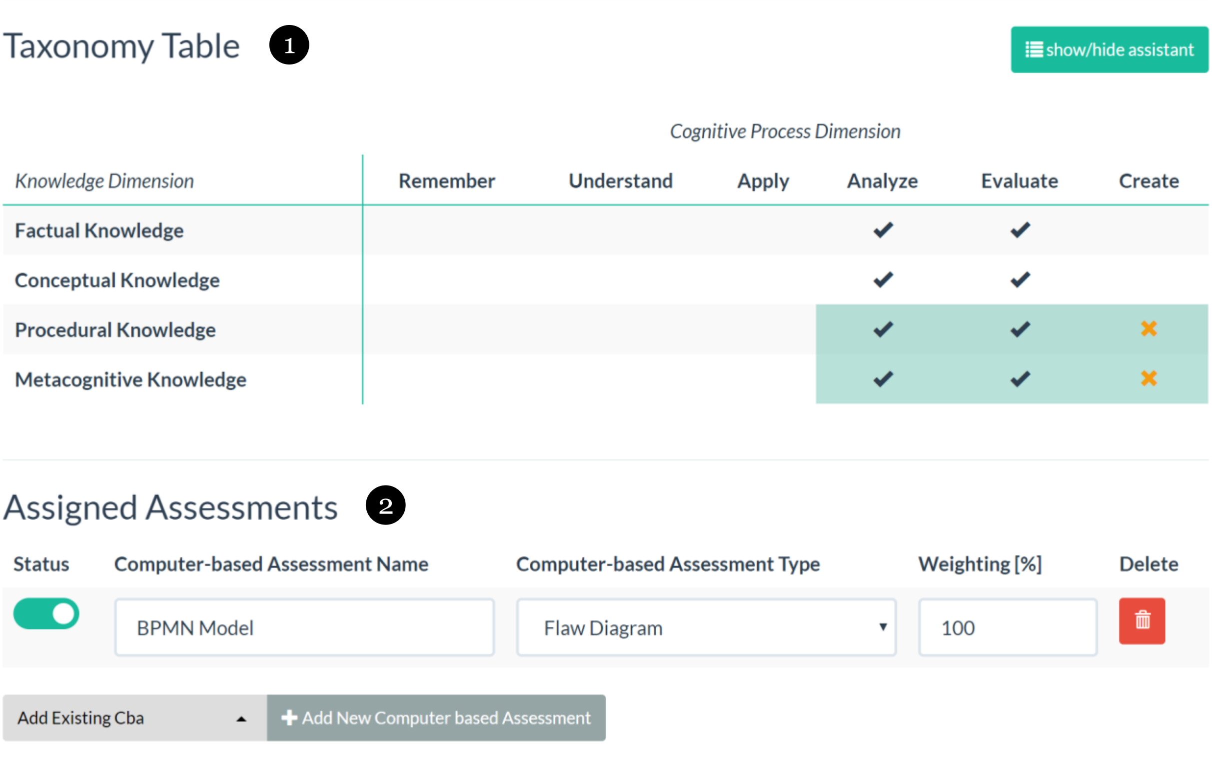

Figure 10: Taxonomy Table During the Assignment of Computer-based Assessments

The highlighted area in the Taxonomy Table (see section 1 in Figure 10) represents the learning objectives the lecturer wants to achieve with the LO that is being created. The black check marks in the same table indicate which learning objectives are actually achieved with the assigned CbAs (see section 2 in Figure 10). If there are pursued objectives that remain uncovered by the assigned CbAs, LOOM warns the lecturer by filling the affected cells with orange crosses.

The classification of objectives a CbA is covering depends to a certain degree on the item type included in the CbA. In the context of LOOM, MCQs are primarily used to address the cognitive process dimensions "remember" and "understand" and, thus, to target lower-level learning (Attali & van der Kleij, 2017). In FDs, items of type "Substitution/Correction" (Scalise & Gifford, 2006), students are asked to find and revise errors. According to Krathwohl (2002) these cognitive processes mainly take place within the dimensions "analyze" and "evaluate" and target higher-level learning. These classifications are set as default when adding a new assessment. However, the specific classification within the Taxonomy Table can vary from CbA to CbA even if they are of the same type. Therefore, lecturers can adjust the default classification of individual CbAs based on the characteristics of the involved questions.

LOOM aggregates the classifications of CbAs applied in LOs at LU level and at course level giving a visual overview over the dimensions of the revised Taxonomy that are covered or not. This makes it possible to "examine relative emphasis, curriculum alignment, and missed educational opportunities" (Krathwohl, 2002). These insights enable lecturers to optimize the CbAs and improve the learning process for the students.

4.2.2.4 Course Enrollment (DE04)

After registering themselves, students see a list with all available courses in LOOM. This list enables them to send enrollment requests for the courses they want to attend. LOOM forwards these enrollment requests to the lecturers of the selected courses who can either accept or decline them. Only if a lecturer accepts a request, the requesting student is allowed to enter the corresponding course in LOOM. Lecturers can also create and enroll students by themselves. They can either do so for every student separately or they can use the bulk import functionality provided by LOOM which reads the data of multiple students from a CSV file. Latter is especially useful if all the students of a course are expected to work with LOOM.

Regardless of who is responsible for the enrollment, students face an enrollment question before entering the course for the first time. This question asks about the goal they pursue by attending the course in order to determine their competitiveness. Students may select one of the following response options:

- To learn something new
- Achieve a 1.0
- Just pass
- Do better than my fellow students
- Avoid being bad

At this point, the chosen response does not affect the subsequent student interaction with LOOM. However, this DE forms the basis for an upcoming component allowing students to compare themselves with others that chose the same performance goal. Said component enables research in the field of social comparison and is implemented within the second cycle (see chapter 4.2.3.3).

4.2.2.5 Wizard for Students (DE05)

In LOOM, CbA and feedback are based on LUs. These LUs are covered throughout the course and leave the students enough time to improve themselves until the final exam takes place. Therefore, LOOM offers formative CbA and feedback. Students can autonomously initiate the assessment process for an LU during a time frame predefined by the lecturer of the course. Additionally, the lecturer has the possibility to manually activate or deactivate the assessment of an LU. The assessment process involves multiple steps. To maintain a clear structure and allow for an intuitive use of LOOM, a walkthrough wizard guides the students through the steps of the process. Figure 11 shows how the process is visually represented on top of the page to offer students an overview of all the steps and indicating their progress.

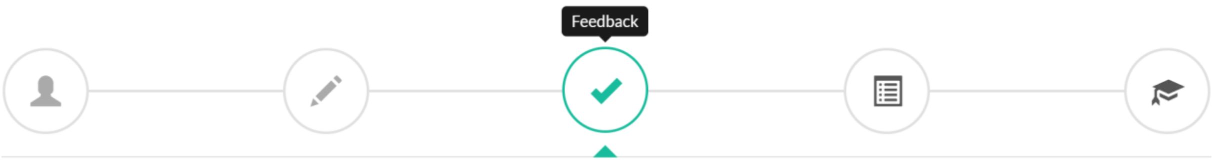

Figure 11: Wizard Navigation Offering an Overview of the Assessment Process

The following list outlines the five process steps (one additional step will be added in the second BIE cycle). Each of the first three steps represents an individual DE and will be covered in a separate chapter.

1. Self-assessment (see chapter 4.2.2.6)
2. Computer-based assessments (see chapter 4.2.2.7)
3. Feedback (see chapter 4.2.2.8)
4. Review
5. Completion

Once the students completed the SA and the CbAs, LOOM provides them with automated feedback to their performance. After receiving the feedback to the CbAs, the students are asked to evaluate it within the fourth step of the assessment process. They express their opinion by declaring the level of agreement for a series of 14 statements. A seven-point Likert scale allows the students to choose how much they agree or disagree with a statement. An eighth option also allows the students to abstain from voting. In addition, the students are asked to comment the learning process of the LU and how they perceived the role of LOOM in it. Only when a student evaluated every statement and left a comment, the completion of the assessment process is possible.

The fifth and final step confirms the completion of the assessment process and offers two choices. The students can either review their individual feedback or continue to the overview of the LU. Regardless of the choice, students can always return to this last step. However, the other steps become inaccessible right after their completion.

4.2.2.6 Self-Assessment (DE06)

Self-assessment (SA) is the first step of the assessment process. It forces students to assess themselves in terms of their level of knowledge. One slider for each LO allows the students to define their current understanding of the corresponding topic within a range from 0 to 100 percent (see Figure 12).

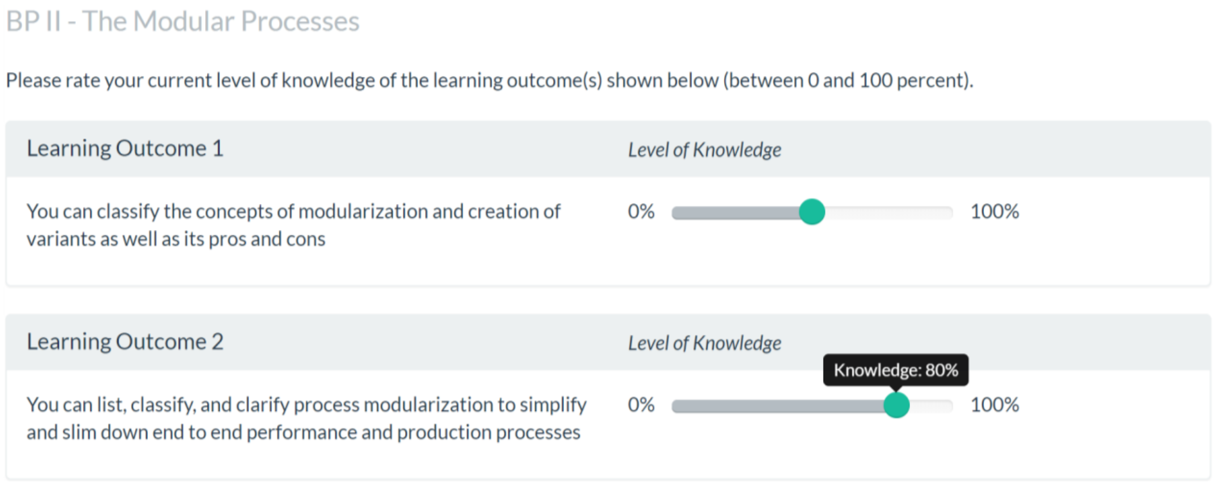

Figure 12: Self-Assessment for All Learning Outcomes of a Learning Unit in LOOM

This step is mandatory, meaning that without completing the SA for each LO, students cannot proceed to the CbAs. When saving the SA, all the data is written into LOOM's DB. Moreover, LOOM inserts entries that prepare the DB for the next step. By doing so, LOOM establishes the basic structure for the storage of the students' CbA results.

4.2.2.7 Computer-based Assessments (DE07)

After the students assessed themselves, the CbA created by the lecturer begin. In the student view, all CbAs assigned to the LOs of an LU are presented to the user collectively. In other words, LOOM shows all the separately created CbAs within an LU in a single page to the students. The students then can choose their own path through the CbAs. Each CbA either consists of FDs or MCQs.

Flaw Diagrams

Students reveal errors in FDs by clicking on the faulty area of the image. In doing so, they have a limited number of attempts determined by the lecturer. Every click triggers an Ajax request to the webserver containing the coordinates of the click (x and y). The webserver then checks if the coordinates lie within a faulty area. If not, LOOM saves the failed attempt including the coordinates in its DB and informs the student about the remaining number of attempts via dialog box. In case there are no attempts left, the server deactivates the FD. However, if the student successfully identified an error, the webserver responds with the coordinates of the discovered faulty area (x, y, width and height). Moreover, the server puts together a dialog box with a question and returns it in HTML format to the client which presents it to the user (see Figure 13). It is no coincidence that the code for the verification of student inputs is implemented entirely server-sided. LOOM is structured that way to prevent students from getting solutions or hints out of the client-sided code before they finish the CbAs.

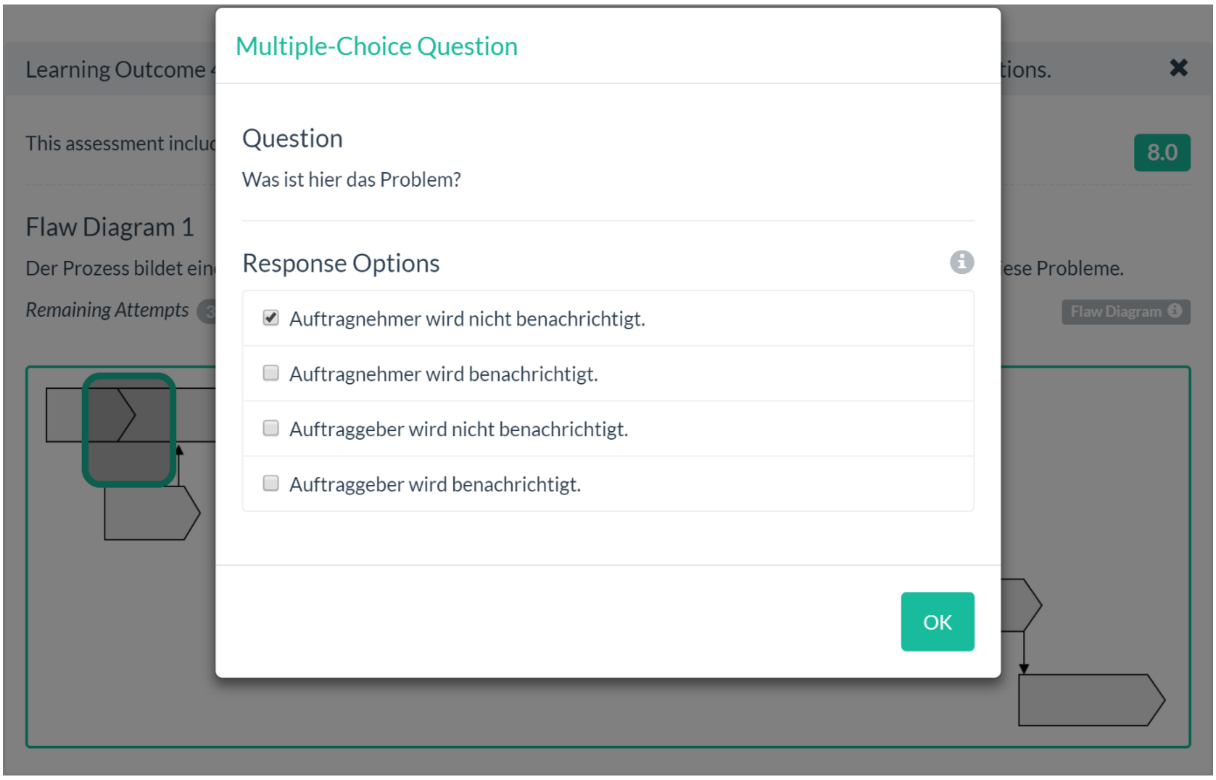

Figure 13: Dialog Box with a Multiple-Choice Question to an Identified Error in a Flaw Diagram

After receiving the response from the server, the client uses the coordinates to visually mark the area and disable left clicks on it (since the error was already discovered). If it was the last error to be found, the whole FD is being deactivated. At the same time, the dialog box assembled by the server pops up allowing the student to answer the incorporated question. The question within the dialog box can either require an answer in form of multiple-choice or free text. In either case, as soon as the student confirms the dialog box, the answer is transferred via Ajax to LOOM's webserver where it is stored into the DB.

Note that even if an FD is deactivated due to the student having used up all attempts or having found all errors, the answers to the questions regarding the discovered errors can still be edited by selecting "Edit" after right clicking the corresponding faulty area.

Multiple-Choice Questions

Students answer MCQs by choosing the right response option. Every time a student selects or changes a response option, an Ajax request is triggered that leads to an update of LOOM's DB. This means that, similar to FDs, each answer to MCQs is instantly saved by LOOM. Through the continuous saving of student answers to FDs and MCQs, LOOM can prevent the loss of data when the browser window is closed or the browser crashes. Also, students cannot gain any advantage by reloading the page with the intention of resetting the counter of FD attempts.

As the students select “continue” to proceed to the next step in the assessment process, a dialog box pops up summarizing the number of FDs and MCQs that remained unanswered. Then, the students can either go back to the CbAs to complete those questions or go ahead knowing that they cannot complete or modify their answers afterwards.

4.2.2.8 Feedback (DE08)

A benefit of CbAs is the possibility to directly and efficiently generate feedback. In the third step of the assessment process, LOOM makes use of this benefit and provides students with immediate feedback to their performance in the foregoing step. In doing so, LOOM applies the feedback types knowledge of results (KR) and knowledge of correct response (KCR) introduced in chapter 2.1.1. This means that LOOM notifies students whether their answers were correct and/or provides the right answers or sample solutions to the questions.

In the context of FDs, LOOM gives feedback at different times. Students already receive feedback in the form of KR, more specifically answer-until-correct (see chapter 2.3.1), directly in the process of solving FDs. When clicking on the image of an FD, they instantly find out if they found an error or not. Only if a student successfully identifies an error, LOOM displays the corresponding faulty area. After the completion of a CbA, LOOM additionally shows the faulty areas of an FD the student was unable to find in a different color (KCR). Moreover, LOOM visually marks the failed attempts indicating the incorrect answers of the student (KR). Figure 14 exemplarily shows LOOM’s feedback regarding an FD. By depicting different symbols in the identified areas of FDs, LOOM indicates whether the answer to an MCQ was correct or not. Yet another symbol is shown for FTQs. When selecting an area, the corresponding question and answer are opened within a pop-up box (see Figure 14). In comparison to the dialog boxes used for the CbAs, it is possible for students to keep multiple pop-up boxes open at once making it easier to check and compare answers.

The questions to the errors incorporated in FDs differ in terms of the connected feedback type. In case of MCQs, both KR and KCR are involved. The same goes for autonomous MCQs that are not attached to an FD. LOOM evaluates and displays if the provided answers were correct and marks the correct response option(s). However, FTQs only involve KCR since LOOM provides sample answers but does not inspect the student answers for correctness. The reason being that the item type FTQ involves a low level of constraint and a high level of complexity which makes an automated evaluation of student responses difficult. Scalise and Gifford (2006) confirm this circumstance by categorizing this item type as *6D* within their taxonomy (see appendix 3).

Students also receive feedback in form of points. LOOM calculates and displays the achieved points for each LO. This calculation takes the individual weights of the CbAs assigned to the respective LO into account while adding up the points. In case the weights defined by the lecturer do not add up to 100 percent, LOOM generates a conversion factor and incorporates it into the calculation. The points of the

LOs are then summed up for the whole LU showing students their total score in comparison to the maximum possible score.

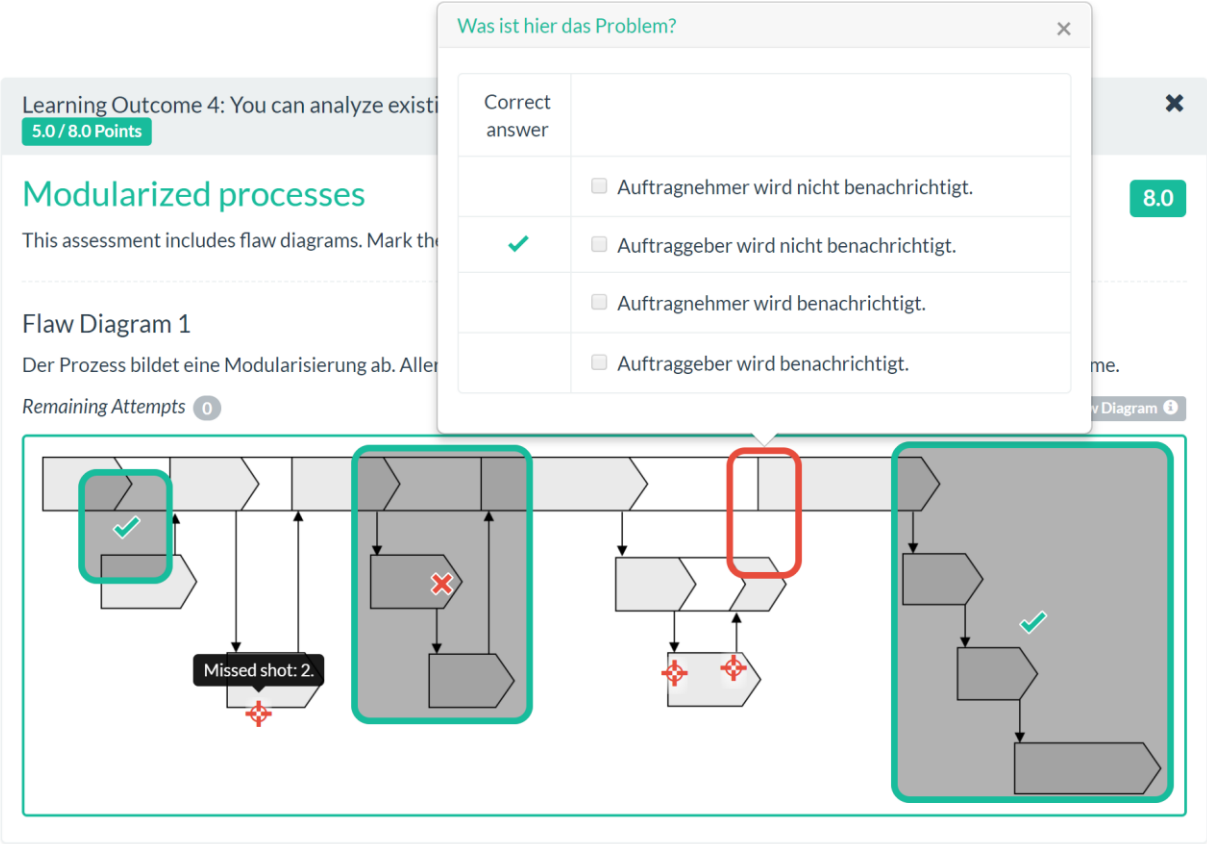

Figure 14: LOOM's Feedback to a Flaw Diagram

4.2.2.9 Student Performance Charts (DE09)

Line charts present the results of the SAs and CbAs to students on an individual basis and to lecturers in aggregated form (the chart type will change in the second BIE cycle).

Chart presented to Students

LOOM uses individual student results of the SAs and CbAs aggregated either at LO or LU level to generate a line chart in the students' overview. One line of the chart represents the SA results and the other represents the aggregated CbA results expressed as a percentage of the maximum points. Consequently, the y-axis of the chart has a range from 0 to 100 percent and the y-axis displays, depending on the aggregation level, all LOs of an LU or all LUs of a course.

Chart presented to Lecturers

While LOOM visualizes the individual results for students, it summarizes the results of all students of a course for lecturers. Lecturers may observe a chart showing the average SA and CbA results of their class aggregated at LO or LU level. However, lecturers do not have access to the results of individual students.

4.2.2.10 Intervention of Alpha Version

The intervention of the alpha version of LOOM took place in the course "Info 1" at the University of Kassel. 63 bachelor students participating in the course took assessments to two LUs treating the basics of business engineering. During class, the lecturer reserved time frames dedicated to the completion of LOOM's assessments. Thus, students used LOOM simultaneously and in a controlled environment.

4.2.2.11 Evaluation of Alpha Version

LOOM was evaluated at two different points of time by people with various backgrounds. Before undertaking the first intervention, an internal evaluation took place involving employees from two universities. Besides finding and fixing bugs, the main goal of this first evaluation was to verify that LOOM is mature enough to be used with students. After the first intervention, a second evaluation was conducted with students who worked with LOOM. The results served as source for learnings that lay the foundation for the development of the beta version of LOOM.

Internal Evaluation during the Cycle

During the first BIE cycle, an internal evaluation of LOOM's alpha version in the form of a focus group workshop was held at the Institute of Information Management of the University of St.Gallen. Additionally, an audiovisual connection to the University of Kassel was established. The 17 participants of the evaluation were divided into two groups. The majority was instructed to register as students and took a predefined assessment. Four selected participants took the role of lecturers. They created courses, LUs, LOs and CbAs by themselves. Among the participants of the evaluation were software developers, doctoral candidates as well as professors. This concentrated charge of expertise led to a reasonable amount of bug reports with valuable feedback and suggestions for improvement which flowed into the learnings of this first BIE cycle. At the same time, the evaluation proofed LOOM to be ready for the first intervention.

Evaluation at the End of the Cycle

The final evaluation of the alpha version of LOOM is based on the intervention in the course "Info 1" at the University of Kassel. Since this intervention was restricted to one lecture of the course, LOOM could only provide the participating students with feedback once and the students' LOOM usage could not be monitored over time. Therefore, the measurement of LOOM's effect on objective student performance is not part of this evaluation. Being the first interaction with real users, the intervention was rather evaluated in terms of user experience as well as LOOM's functionality.

After using LOOM, the students completed a survey involving more than 40 statements regarding their impressions of LOOM. For each statement, the students were asked to choose their agreement level on a seven-point Likert scale. However, they could also abstain from voting. The statements included in the survey are summarized in eight constructs. Table 6 presents these constructs and shows the corresponding mean value of the student responses. Furthermore, the table involves literature references indicating where the statements originate from. Note that this final evaluation of the first BIE cycle is limited to the analysis of mean values while the evaluation of the next cycle is based upon the more sophisticated approach of structural equation modeling.

Table 6: Constructs Measured in the Evaluation at the End of the First BIE Cycle

Construct		Mean value
C01	Learning process quality (Söllner, Bitzer, Janson, & Leimeister, 2017)	4.01
C02	Task technology fit (McGill & Klobas, 2009)	4.31
C03	Technology acceptance model (Venkatesh, Morris, Davis, & Davis, 2003)	4.30
C04	Information systems success model (DeLone & McLean, 1992)	4.27
C05	Learning outcome quality (Söllner et al., 2017)	4.03
C06	Satisfaction (Arbaugh, 2001)	3.88
C07	Security beliefs (Bart, Shankar, Sultan, & Urban, 2005)	3.14
C08	Perceived student performance (Alavi, 1994)	3.83

In the survey, the construct *security beliefs* achieved the lowest result. This can be attributed to the fact that no privacy statement was presented to the students in this version of LOOM. However, a privacy statement will be added in the beta version. The other construct's mean values are all close to four, the center of the applied Likert scale. The mean values of the constructs s*atisfaction* and *perceived student performance* are below four. This could indicate that students are not yet fully convinced of the didactical benefits achievable by using LOOM. However, the values of these constructs are not considerably lower than four. The mean values of the constructs *learning process quality* and *learning outcome quality* are almost exactly at the center of the Likert scale while the values of the remaining three constructs go beyond that. The best rated constructs *task technology fit*, *technology acceptance model* and *information systems success model* involved questions about the usability, availability and correctness of LOOM as well as the intention of students to use LOOM again. The insights gained through the survey and the focus group workshop are documented in the form of learnings in the following chapter.

4.2.2.12 Learnings

The learnings listed in this chapter are primarily based on the evaluations discussed in the previous chapter. These learnings summarize the findings of the first BIE cycle and, at the same time, form the basis for the second BIE cycle. The following Table 7 presents the learnings and matches them with the affected user roles. Note that during the first cycle it became apparent that an additional user role "researcher" is necessary. This user role will be introduced in the second cycle.

Table 7: Learnings of the First BIE Cycle

Learning		Affected user role(s)
L01	Students feel unfairly treated if they accidently click on an FD and consequently lose one attempt.	Student
L02	Students may overlook CbAs because the content is initially hidden in collapsible panels.	Student
L03	Dialog boxes may prevent students from seeing important parts of an FD.	Student
L04	Depending on the size of an FD, its content may be difficult to read for students.	Student
L05	Retrieving the confidence level of students when they respond to a CbA item may increase their performance and lead to relevant scientific findings.	Student, Researcher
L06	A functionality allowing students to compare their results is missing.	Student, Researcher
L07	Students need to be divided into a treatment and control group to allow a valid social comparison experiment.	Researcher
L08	A line chart is not the best form of visualization to compare SA and CbA.	Student, Lecturer
L09	Lecturers need a possibility to review student responses to FTQs.	Lecturer
L10	Researchers must be able to extract the gathered data out of LOOM in a form that allows for an effective analysis.	Researcher

Some of the learnings require the adjustment of LOOM's existing components while others cause the implementation of new ones. The learnings L01 to L04, for example, require the enhancement of LOOM's component responsible for the completion of CbAs in terms of usability. The consideration of learning L09, however, leads to the implementation of a new component. All the learnings of the first BIE cycle flow into the second BIE cycle which is dedicated to the refinement of LOOM.

4.2.3 Second BIE Cycle

The second BIE cycle has the same structure as the first one. However, it puts increased emphasis on the evaluation since it is the final evaluation of LOOM in the context of this thesis. Based on the intervention, evaluation and the resulting learnings of the first BIE cycle, DEs for the second BIE cycle are established. Table 8 presents these DEs in the same form as in the first cycle. However, the table includes an additional column indicating which learning(s) of the previous cycle the DEs address. Moreover, in this cycle an additional user role is in use for components that serve scientific purposes and must not be accessible by regular lecturers or students. This user role, called "researcher", builds upon the privileges of a lecturer and extends them. Consequently, researchers look at the front-end of lecturers with some additional functions. Besides managing social comparison experiments, a user of type researcher enjoys the right to export student results within a course.

Table 8: Design Elements of the Second BIE Cycle

Design element (DE)		User role(s)	Description	Addressed learning(s)	Implementation within this thesis
DE10	Usability of Computer-based Assessments	Student	LOOM is easy to use and offers students a great user experience during the assessment process.	L01, L02, L03, L04	(✓)
DE11	Confidence Level	Student	Students state their confidence level when responding to an item.	L05	✓
DE12	Social Comparison	Student, Researcher	Students can compare their results with the ones of fellow students.	L06, L07, L08	
DE13	Presentation of Student Responses	Lecturer	Lecturers can inspect student responses to FTQs.	L09	✓
DE14	Data Export	Lecturer, Researcher	LOOM allows authorized users to extract student data.	L10	✓

4.2.3.1 Usability of Computer-based Assessments (DE10)

LOOM was positively evaluated regarding its usability in the first BIE cycle. Nevertheless, the evaluation revealed shortcomings in LOOM's front-end. More specifically, the component allowing students to complete CbAs is concerned. The following paragraphs discuss how the usability is being improved within the second BIE cycle to offer students a better user experience.

Enabling of Flaw Diagrams

When revealing errors in FDs, students have a restricted number of attempts. While scrolling through the CbAs, students could accidently click on an FD and instantly lose one attempt. Affected students perceive this scenario as unfair. To avoid counting unintended clicks as attempts, FDs are reconfigured so that the first click is designated to enable the underlying FD. On the initial click everything darkens in LOOM but the enabled FD. Every following click then counts as attempt to reveal an error. As soon as the user scrolls or clicks outside the image, LOOM disables the FD again allowing the student to proceed with the remaining CbA. However, FDs can always be reenabled as long as there are still unrevealed errors and remaining attempts.

Collapsible Panels and Navigation

LOOM structures CbAs in collapsible panels, each panel presenting all CbAs of one LO. For the CbAs to remain compact, LOOM was initially configured that only one panel is expanded at a time. This means that every time a student opened one panel another one was automatically closed. However, because by default only the first LO panel was open at the beginning, some students overlooked the CbAs of further LOs.

The obvious solution to this problem consists of expanding all panels from the beginning. This, however, happens at the expense of the compact and transparent structure of the CbAs. In order to still provide the students with an overview over all LOs, the CbA page now involves an additional navigation bar. This navigation bar lists all LOs and shows the students which LO they are currently working on. Moreover, it indicates which LOs are already completed and lets the students directly jump to any desired LO.

Draggable Dialog Boxes

If students find an error within an FD, a corresponding question pops up in a dialog box. Since this dialog box appears in the middle of the screen, it often hides the discovered error of the FD. However, it is helpful for students to see the error when answering a question that refers to it. Therefore, a function making the dialog boxes draggable was introduced. That way, students are enabled to freely drag the dialog boxes across the screen to uncover certain areas of the FD if necessary. The possibility to relocate the dialog boxes results in an improved user experience.

Flaw Diagram Zoom

Depending on the amount of detail included in a cropped image representing an FD, certain aspects of the image may appear too small on the screen. For example, the text of activities in an extensive BPMN model can quickly become difficult to decipher. Therefore, LOOM now magnifies the area over which a student hovers with the curser and shows it right on top of the enabled FD. While all other functions described in this chapter were implemented by the author of this thesis, the zoom function was realized by a team member.

4.2.3.2 Confidence Level (DE11)

When responding to the items of a CbA, LOOM asks students to select their confidence level. The levels consist of “I’m sure”, “I’m not sure”, “I guessed” and “I don’t know”. This information allows to monitor changes in the students’ confidence levels over time and serves a scientific purpose. Monitoring confidence may help students to improve their performance and the assessment of their own knowledge (Nicol & Macfarlane-Dick, 2006). Neither responding to the items within a CbA nor stating the confidence level is mandatory for students. However, if students respond to an item, a confidence level needs to be selected as well to continue the assessment process.

4.2.3.3 Social Comparison (DE12)

In the context of the DE regarding social comparison, two components available to two different user roles must be considered. First, researchers have the possibility to start an experiment within a course which automatically divides participating students into a control and treatment group. Second, students of the treatment group can use charts to compare their results with fellow students who resemble themselves in terms of competitiveness.

Experiments

Researchers have the possibility to start an experiment within the scope of a course. The moment they do so, LOOM randomly assigns each of the participating students to either a treatment or control group. Students assigned to the treatment group encounter the enrollment questions discussed in chapter 4.2.2.4. Afterwards, LOOM enables the students of the treatment group to compare their CbA results with the results of fellow students who have chosen the same goal. To facilitate this comparison, LOOM visualizes the results within a chart that is discussed in the following chapter. Students assigned to the control group neither answer the enrollment question nor have the possibility to compare themselves with others. The control group is necessary to ensure that conclusions drawn from social comparison experiments are valid. However, all other aspects of LOOM and the assessment process are the same for both groups. The functionality to manage experiments, as described in this chapter, was implemented by a team member.

Bar Chart

As discussed in chapter 4.2.2.9, LOOM uses the results of the SAs and CbAs expressed in percentage to generate a chart that provides an overview of the performance. Originally displayed as a line chart, LOOM now visualizes the student results through a bar chart. The reason being that line charts are intended to depict trends over a period of time while all the CbAs of an LU are usually completed at once. Moreover, bar charts allow a better comparison of the results. After the transformation to a bar chart, LOOM clearly reveals possible gaps between SA and CbA by representing those values as bars standing next to each other.

If an experiment regarding social comparison is in progress, the data visualized by the chart is not restricted to the results of the individual student. Depending on the experiment group (see preceding chapter), a student can also activate a graph showing the average score of fellow students with the same performance goal. Therefore, these students are enabled to compare their results with those of others who are similar to them. The following Figure 15 shows the discussed bar chart for an LU with two LOs. LOOM only displays the bars for social comparison if the student clicks on the button “Compare my results with fellow students”.

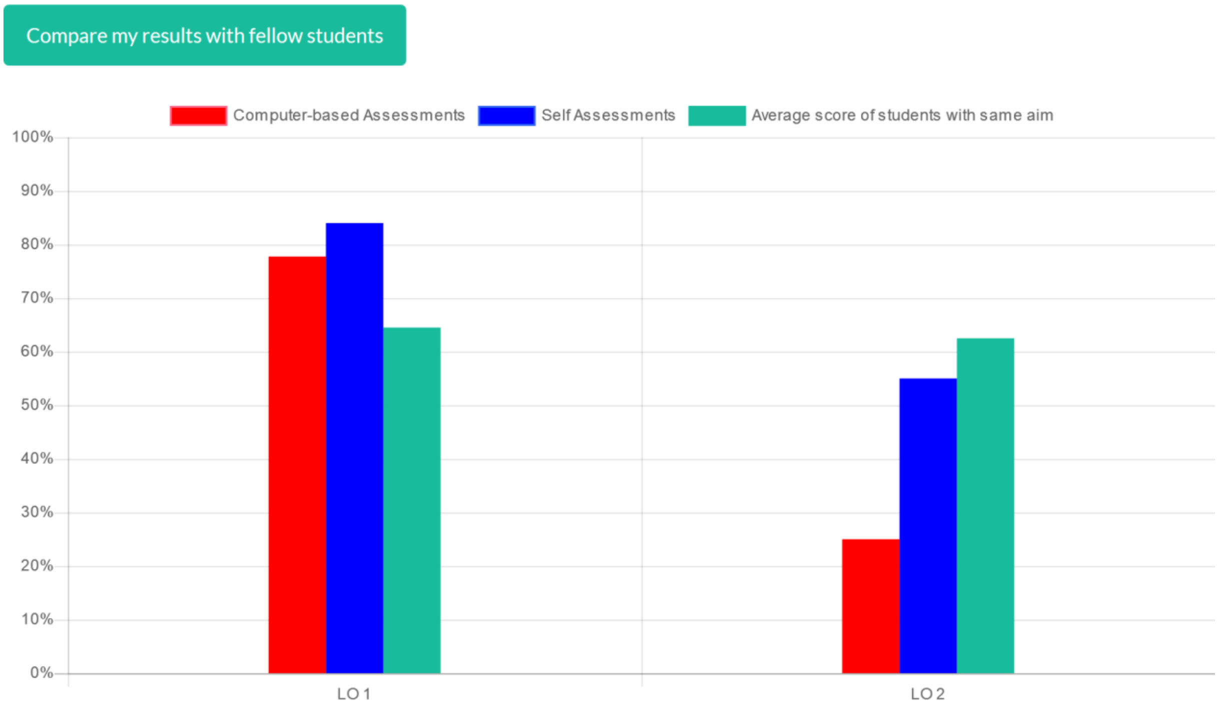

Figure 15: Chart Visualizing Student Results and Allowing for Social Comparison

This bar chart is displayed in the overview of each LU and the course. However, the first time students encounter it, is directly in the assessment process. The chart is considered as part of the feedback and, therefore, appears right after the already implemented feedback mechanism as an individual process step. Being shown before the review step, the comparison chart is considered by the students when they give their opinion to the feedback received from LOOM. The author of this thesis incorporated the chart into the assessment process. However, the social comparison mechanism and the chart in the overview were implemented by a team member.

4.2.3.4 Presentation of Student Responses (DE13)

Regarding FDs, lecturers can either define MCQs or FTQs to incorporated errors. LOOM is capable of correcting MCQs automatically while FTQs need the intervention of a lecturer. Therefore, LOOM enables lecturers to analyze student responses to FTQs. For each CbA with FDs that involve FTQs, lecturers can call a page presenting all individual student responses. This page offers lecturers to choose an FD and then simply click on an error to list all submitted responses. Note that the student data is treated anonymously, so lecturers do not know the names of the students behind the responses.

4.2.3.5 Data Export (DE14)

In two scenarios, LOOM allows authorized users to extract data from its DB. The data export in these two scenarios is discussed in the following subchapters.

Responses to Free Text Questions

The previous chapter about DE13 discusses an online page offering lecturers to review student responses to FTQs. To facilitate this review process, an export component allows lecturers to extract all student responses within a CbA at once in form of a CSV file. This file is generated on the webserver and then transferred to the client. Within the CSV file, all student responses are clearly structured by FD and FTQ. CSV is the chosen format for the export file because it is commonly used to exchange data and is supported by nearly all spreadsheets or database management systems. Lecturers are expected to primarily use Microsoft's spreadsheet Excel to inspect the data. Like in the online presentation of the responses, student information is anonymized in the exported data.

Course Data

In the context of this project, the programming language R is used to prepare the data gathered throughout the semester for its analysis. Therefore, the data of LOOM's DB must be transferrable to the software environment R. This transfer is achieved by an export component implemented in LOOM. Said component collects all the student data of a selected course from the DB and writes it into a file. More specifically, it transforms the data originating from multiple relational DB tables into a single two-dimensional table stored as a CSV file. In the new format, the data for each student is merged into a single row. The resulting export file forms the basis for the data preparation and can directly be imported into R. The export file includes the following data of each enrolled student.

- Personal information
  - First and last name
  - Email address
- Data regarding the usage of LOOM
  - Time spent to complete all CbAs of an LU
  - Number of LUs for which the CbAs were completed
- Data regarding social comparison
  - Performance goal
  - Experiment group
  - Number of clicks on the social comparison function
- Further data collected during the assessment process
  - SA values for LOs
  - CbA results for MCQs and FDs (including number of failed attempts) on different aggregation levels
  - Selected confidence levels aggregated for each CbA
  - Feedback evaluation

Since the sensitive data listed above is intended for scientific purposes only, regular lecturers are not entitled to access the export component. Only users of type researcher can initiate the export of course data.

4.2.3.6 Intervention of Beta Version

The intervention of the beta version of LOOM took place in the course "Business Engineering – Digital Business & Transformation" at the University of St.Gallen. Around 200 master students participating in the course took assessments to eight LUs treating different aspects of business engineering. Four out of the eight times, the students completed the assessments in class. The remaining four times, students used LOOM independently outside the teaching hours.

4.2.3.7 Evaluation of Beta Version

The evaluation of LOOM's beta version is based on the intervention within the second BIE cycle. It consists of a field study involving students who participated in the course "Business Engineering – Digital Business & Transformation". The evaluation is discussed in three steps. This chapter first describes the data preparation process, then discusses the measurement of constructs and finally provides an interpretation of the data.

Data Preparation

The evaluation is based on data from various sources including LOOM's DB, two surveys and the final exam of the course. Throughout the course, LOOM saved all the student inputs in its DB. This DB can be accessed via export component that gathers data to every student's performance, competitiveness and individual extent of usage (see chapter 4.2.3.5 for a detailed list of the involved data). Moreover, data is available from two online surveys conducted at different points of time. In the first survey, conducted prior to the students' first use of LOOM, participating students specified how much they agree or disagree with statements regarding mastery approach and self-efficacy. In the second survey, conducted after the students' last use of LOOM, participants faced statements addressing their experiences and satisfaction with LOOM as well as the perceived student performance. To gather the opinions of the students in a way that allows for a scientific analysis, both surveys applied a seven-point Likert scale with an eighth option to abstain from voting. Additionally, the results of the course's final exam in form of points provide information about the objective student performance.

The data from the above described sources is extracted as CSV files. Afterwards, the software environment R is used to transform the data of those files into a form that enables a data analysis using structural equation modeling (SEM). For the analysis, the transformed data is loaded into the software SmartPLS. The following Figure 16 visualizes this extract, transform, load (ETL) process before it is discussed in detail.

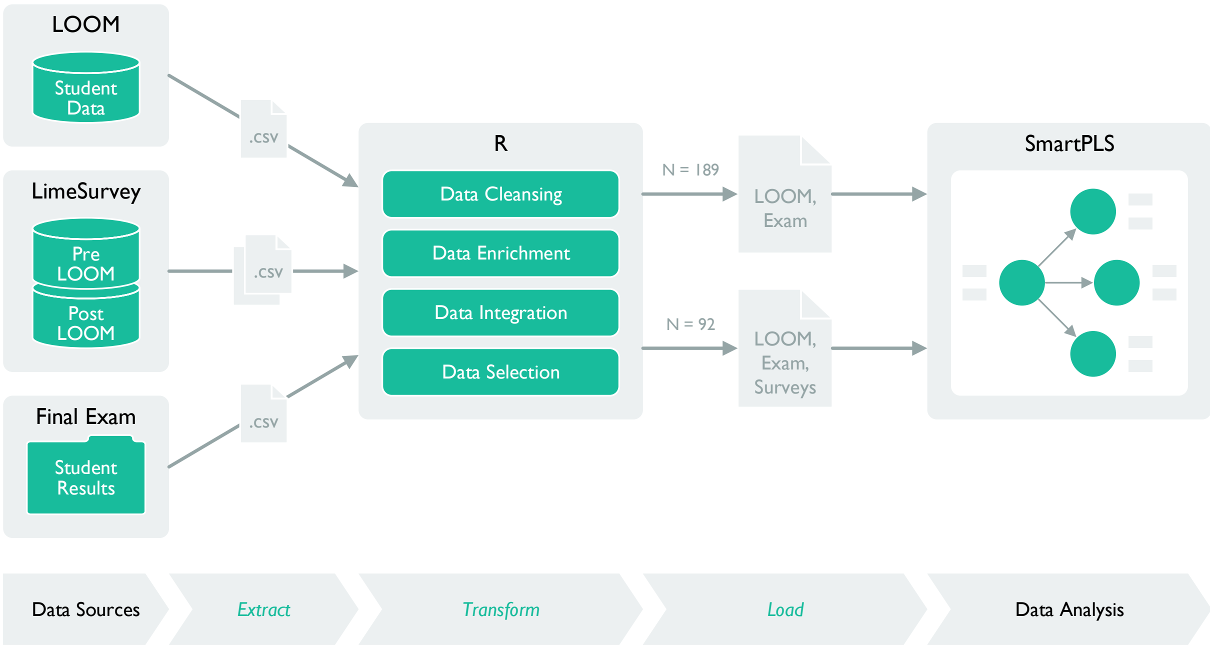

Figure 16: Data Preparation Process for the Evaluation of LOOM's Beta Version

After retrieving the CSV files from the data sources, entries of test users that do not represent real students are removed from the LOOM data. Moreover, to prevent straightlining from affecting the survey results, students are excluded who selected the same response to different statements more than 20 times in a row. Afterwards, all datasets are anonymized by replacing the personal Information of each student with a unique identifier. Only then, the data is imported into the environment of R.

In R, the data originating from the surveys is cleaned by removing entries of students who did not complete the questionnaire. Also, columns involving no data at all are excluded from the datasets. Additionally, the column naming of the survey datasets is standardized by matching it with the naming convention of the other datasets. Regarding the data originating from LOOM, the boxplot method is applied to identify and exclude outliers in terms of the usage duration at LU level. Thereby, every duration that exceeds 1.5 times the interquartile range (the middle 50% of the data) above the upper and below the lower quartile is considered an outlier. After removing the outliers, LOOM's data is enriched with the total usage duration and the average usage duration of each student.

After cleaning and enriching the individual datasets, R is used to integrate the data. To begin with, the data of LOOM is merged with the students' results of the final exam (dataset 1). In the resulting dataset, only students who are registered in LOOM and attended the final exam are considered. This dataset is merged with the results of the surveys conducted prior and after the usage of LOOM leading to another new dataset (dataset 2). Both derived datasets are then reduced to the columns that are relevant for the data analysis and exported as CSV files. These CSV files serve as input for the software SmartPLS used for the analysis of the data. The following Table 9 presents descriptive statistics for the datasets 1 and 2. The measurement of constructs and the data analysis are discussed in the upcoming chapters.

Table 9: Descriptive Statistics of the Students in Dataset 1 and Dataset 2

	Dataset 1 Objective student performance		**Dataset 2** Objective and perceived student performance	
Sample size	N = 189		N = 92	
Male	69.31% (n = 131)		73.91% (n = 68)	
Female	30.69% (n = 58)		26.09% (n = 24)	
	Mean	**SD**	**Mean**	**SD**
Age	25.90 years	1.92 years	25.92 years	1.65 years

Measurement of Constructs

The aim of this chapter is to describe the operationalization of the measurement constructs used in the research models of this evaluation (see Figure 17). Straub, Limayem, and Karahanna-Evaristo (1995) summarize various approaches to measure the usage of an IT tool. Usage measures can either be subjective or objective in nature. However, research indicates that subjective measures are not accurate enough to describe the actual usage of a tool (e.g. Rice & Borgman, 1983; Trice & Treacy, 1988). Therefore, the independent construct *LOOM usage* is determined using solely objective measures. Frequency of use, the first applied measure, describes the number of assessments a student has carried out (Ginzberg, 1981). Duration, the second measure, calculates the timespan of a student between completing the first and the last item of an assessment (Srinivasan, 1985). The decision to use these two measures is based on the fact that both can be precisely calculated using database entries of LOOM.

The dependent construct *objective student performance* is measured by the number of points students achieved in the final exam ranging from 0 to 40 points. This construct is determined by points instead of grades to achieve a higher measurement precision. Student performance, as defined in this thesis, is not restricted to objective student performance but also involves perceived student performance. The measurement of perceived student performance is based on commonly used constructs suggested by (Alavi, 1994) consisting of *perceived skill development*, *self-reported learning* and *learning interest*. Moreover, *satisfaction* can be a determinant for student performance and is, therefore, also considered in this evaluation (Arbaugh, 2001). Additionally, the evaluation involves the constructs *mastery approach* (Baranik, Barron, & Finney, 2007) and *self-efficacy* (Hardin, Looney, & Fuller, 2014) in order to check for their effect on LOOM usage. The measurement of constructs other than LOOM usage and objective student performance is based on the surveys described in the previous chapter.

Data Analysis

Structural equation modeling (SEM) is applied to the gathered data to analyze if the usage of LOOM increases student performance. This approach suits the present research for three reasons (Backhaus, Erichson, & Weiber, 2015). First, the nature of this research is rather confirmatory than exploratory. Second, the research model involves independent, dependent and control variables and, thus, is rather complex. Third, the constructs of the model are latent, meaning they are not tangible or directly measurable. According to Mertens, Pugliese, and Recker (2017), these conditions endorse the use of multivariate instead of univariate regression models.

For the model identification, partial least squares (PLS) modeling is used instead of covariance-based modeling because the objective of this research is not to evaluate a theoretical model but to assess the effects of LOOM usage on the objective and perceived student performance (Mertens et al., 2017). The statistical software applied for SEM in the context of this research is SmartPLS version 3.2.7 developed by Ringle, Wende, and Becker (2015). SmartPLS applies SEM by estimating unknown parameters like loadings, path weights or shared variances and presenting them graphically. By doing so, the software is configured to replace not available (NA) values with the mean value of the remaining data of the affected measurement item (mean replacement) (Mertens et al., 2017).

In SmartPLS, the PLS algorithm is run with the path weighting scheme and 300 iterations. The settings for the bootstrapping algorithm involve 5000 subsamples and individual sign changes. These algorithms are run for both prepared datasets resulting in two research models. All common quality criteria for the research models A and B are measured (Backhaus et al., 2015). The factor loadings of all measurement items included in the constructs are higher than 0.6. For all constructs, Cronbach's alpha is above 0.8 and the average variance extracted (AVE) is above 0.62. Moreover, all constructs contain a composite reliability that exceeds 0.83 while their variance inflation factor (VIF) values remain below 4.2. As a result, the research models fulfill the quality criteria proposed by Hair, Hult, Ringle, and Sarstedt (2014).

Figure 17 shows the models identified by SmartPLS that fit the data. Research model A is based on dataset 1 and focuses on showing the impact of the LOOM usage on the objective student performance. It proves that the extent to which students use LOOM has a significant effect on their performance in the final exam. Students who made use of LOOM generally performed better than those who did not.

To analyze student performance, however, not only the objective but also the perceived student performance needs to be considered. Research model B reveals that LOOM usage has a significant positive effect on objective as well as perceived student performance consisting of learning interest, self-reported learning and perceived skill development. Moreover, it positively affects the students' satisfaction. Model B is also used to rule out effects originating from the variables mastery approach and self-efficacy. It shows that neither of these two control variables has a significant effect on LOOM usage.

Note that the R^2 values of the dependent variables in both models are not high which means that the LOOM usage alone is not responsible for a large portion of the variables' variation. However, these low values are not problematic for this evaluation since the focus lies on assessing how significant the effect of the LOOM usage is on student performance and satisfaction.

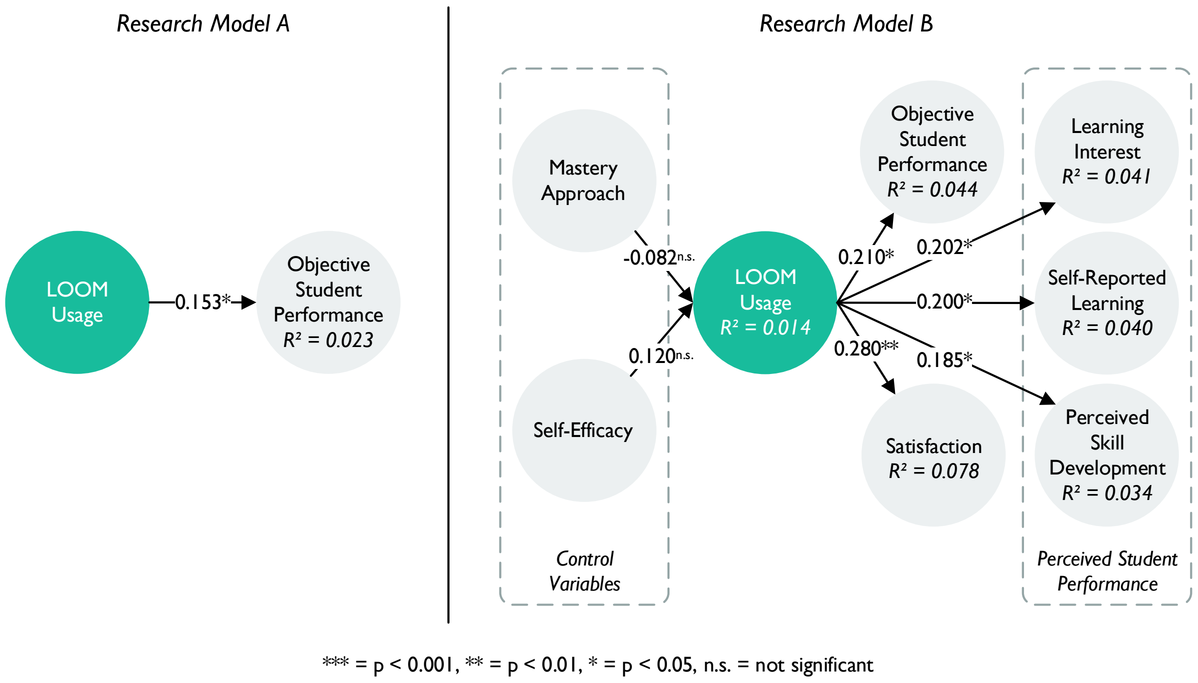

Figure 17: Results of Research Model A and Research Model B

4.2.3.8 Learnings

The evaluation of this cycle revealed that the usage of LOOM has a significant effect on the objective student performance. Based on a subgroup of students, who participated in a survey before and after the usage of LOOM, the evaluation additionally confirmed a significant increase in the perceived student performance and satisfaction. Therefore, students who used LOOM more extensively not only achieved more points in the final exam but also experienced higher satisfaction and perceived performance consisting of learning interest, self-reported learning and perceived skill development. To further enhance the already significant effect of LOOM usage on student performance, Table 10 presents specific learnings gathered within the second BIE cycle applicable for the future development of LOOM.

Table 10: Learnings of the Second BIE Cycle

Learning		Affected user role(s)
L11	Some students want to take the same CbAs multiple times, especially for exam preparation.	Student
L12	It is more convenient for students to log into LOOM with their existing university credentials instead of creating a new account (single sign-on).	Student
L13	The feedback regarding MCQs needs a clearer distinction between a student's answer and the correct solution.	Student
L14	LOOM needs to be available in English and German.	Student, Lecturer
L15	Lecturers may want to import presentation slides into LOOM.	Lecturer

The learnings of this cycle will be applied for the further development of LOOM. However, the resulting implementation steps are not part of this thesis anymore.

5 Discussion

Current literature agrees on the importance of formative feedback in the context of learning. Several studies and meta-studies confirm that feedback generally enhances student performance (e.g. Hattie & Timperley, 2007; Kluger & DeNisi, 1996). However, there is no consensus among researchers about the design of formative feedback. The design of feedback involves variables such as feedback timing, frequency, specificity, complexity and goal orientation which influence the effect of feedback on student performance. Different studies report disparate findings about the impact of feedback variables. These inconsistent findings can partly be explained with individual factors that interact with the feedback variables (Shute, 2008). The aim of feedback, the achievement level of a student and the level at which feedback operates have, among other factors, an influence on the form of feedback that is the most suitable in a specific context. In short, there is no ideal form of formative feedback for all students and performance goals. A framework for the design of computer-based feedback, proposed by Mason & Bruning (2001), shows that this conclusion also applies for computer-based learning environments. Therefore, Table 4 summarizes the theory about feedback by providing 17 guidelines for the design of formative feedback in general as well as in computer-based environments that account for different situational factors.

Feedback does not necessarily come from the outside but can also be internally self-generated as result of self-assessment (SA) (Butler & Winne, 1995). However, to prevent students from overestimating their skills, SA should be paired with a form of objective assessment such as computer-based assessment (CbA) (Nicol & Macfarlane-Dick, 2006). Another approach to support students in accurately evaluating themselves is social comparison. The social comparison theory states that individuals try to resolve uncertainties about their own opinions and abilities by comparing themselves to others (Festinger, 1954). Studies show that this habit can be used to increase student performance (Blanton et al., 1999). However, since feedback, as defined in this thesis, is built upon SA as well as CbA and may also involve CbA results of fellow students for social comparison, it is crucial to find a way to keep the cognitive load in students within reasonable limits (Janson, Söllner, & Leimeister, 2015). Moreover, CbA must be carefully designed because its content is decisive for the quality of the feedback (Mason & Bruning, 2001). A large part of the past research on feedback in the context of CbA, especially older studies, focused on lower-level learning (e.g. Kulhavy et al., 1985). However, CbA should also allow feedback that supports higher-level learning to maximize its effect on student performance (Attali & van der Kleij, 2017). To facilitate learning on both levels, CbA needs to involve item types that can target different learning objectives within the cognitive process dimension of Bloom’s revised Taxonomy (Krathwohl, 2002).

The literature review in combination with the expert interviews conducted prior to this thesis served as foundation for the realization of the computer-based formative assessment and feedback tool LOOM. The development of LOOM was structured according to the ADR method of Sein et al. (2011). This systematic approach ensured that the learnings gathered during the development process were continuously captured to be eventually communicated according to scientific standards (Sein et al., 2011). These learnings formed the basis for the DEs of the second BIE cycle of this ADR project. The DEs of the first cycle, however, are primarily based on requirements derived from the literature review and expert interviews. In total, LOOM involves 14 DEs which are summarized in the next paragraph.

LOOM is implemented as a web-based application with a responsive user interface (DE01) to be available from anywhere at any time. Moreover, LOOM offers a high degree of usability, especially during the assessment process (DE10). When creating a CbA, lecturers can either apply items of type MCQ or FD. MCQs primarily target lower-level learning while FDs are intended to target higher-level learning. However, a Taxonomy Table (DE03) allows lecturers to classify CbAs by themselves according to the specific characteristics of the involved items. LOOM uses this classification to check if the targeted objectives of an LO are achieved with the assigned CbAs. To simplify the creation of LOs and CbAs for lecturers and reduce the cognitive load in students during the assessment process, LOOM involves walkthrough wizards for both these user roles (DE02 & DE05). In accordance with the reviewed literature, LOOM provides feedback (DE08) based on preceding SAs (DE06) and CbAs (DE07) that judge a student's performance. When completing a CbA, students state their confidence level for each of their responses (DE11). LOOM provides feedback of type KR and/or KCR for every item involved in a CbA. Simultaneously, it enables lecturers to retrieve and review student responses to FTQs (DE13). Moreover, LOOM aggregates the CbA results at LO and LU level to visualize them within student performance charts (DE09). These charts additionally offer students the possibility to compare their CbA results with the average score of fellow students who have the same performance goal (DE12). Students specify their performance goal during course enrollment (DE04). In the end of a course, authorized users have the possibility to extract the accumulated student data from LOOM for research purposes (DE14).

During the intervention in the master's course, LOOM applied the feedback variables of the theory as follows: LOOM's feedback was immediate and occurred frequently throughout the semester. Moreover, it was low in specificity and restricted to the types knowledge of results (KR) and knowledge of correct response (KCR) making it rather simple. The final evaluation of LOOM based on this intervention confirmed previous research regarding the importance of feedback. The usage of LOOM had a significant positive effect on student performance.

6 Conclusion

The aim of the present thesis has been to examine how an IT tool should be designed and implemented to provide students in large-scale lectures with formative feedback and to investigate the tool's impact on student performance. To reach these research goals, the computer-based formative feedback tool LOOM was developed and evaluated according to the action design research (ADR) method. Guidelines and requirements derived from literature and practical insights gained through expert interviews (not part of this thesis) formed the starting point for the development of the tool. This knowledge in combination with further learnings gathered during the development process of LOOM led to the specification of 14 design elements for formative feedback tools. The final evaluation of LOOM and the incorporated design elements consisted of a field study in the context of a large-scale lecture. Examining the student's results in the final exam, it revealed that the usage of LOOM has a significant effect on the objective student performance. Based on a subgroup of students, who participated in a survey before and after the usage of LOOM, the evaluation additionally confirmed a significant increase in the perceived student performance. In other words, students who used LOOM more extensively not only tended to achieve more points in the final exam but also experienced increased learning interest as well as higher perceived learning and skill development. By generating design knowledge through building and evaluating LOOM, this thesis laid the foundation for a design theory regarding formative feedback tools.

6.1 Limitations

Some limitations to this thesis need to be acknowledged. One limitation is that the emerging artifact LOOM is not finalized by the time this thesis is finished. Therefore, the learning process is still in progress and the formalizing of learning (stage 4 of the ADR method) is too early. As a result, the outcomes cannot be generalized and transformed into design principles yet. Moreover, the interventions in the first and the second BIE cycle took place in the same context: a large-scale university course about business engineering. This limitation means that to achieve a more diversified insight into LOOM's effect on student performance, it should be evaluated in other contexts. One other possible field of application are massive open online courses (MOOC) allowing for distance education. Finally, for the evaluation of LOOM, students were not separated into a treatment and control group. The lack of a control group presents a limitation in form of the absence of a real experiment setting.

6.2 Future Research

This chapter presents recommendations for future research as well as the further development of the artifact LOOM. The evaluation within the second BIE cycle proved that LOOM as a whole has a significant positive effect on student performance. However, to reveal which of the DEs have the biggest impact on performance, they need to be analyzed on an individual basis. Moreover, the evaluation did not involve an analysis of the students' self-assessment (SA) and computer-based assessment (CbA) results during the usage of LOOM. Future research should not only investigate how SA and CbA results change over time but also how the gap in between these two values develops. LOOM already offers the functionality to extract the data necessary for such an investigation.

The learning objectives that lecturers can target with CbAs are to a certain extent dependent on the item types supported by LOOM. At this point, items of type multiple-choice question (MCQ) are primarily used to target lower-level learning while items of type flaw diagram (FD) primarily target higher-level learning. However, to enable lecturers to easily cover any desired learning objective in the Taxonomy Table, further item types are required. For example, a new item type could ask students to reorder or assign a series of statements or images using drag and drop. This specific item type would support lecturers mainly in targeting the cognitive process dimension "analyze".

In the context of FDs, lecturers can either define MCQs or free text questions (FTQ) to incorporated errors. Currently, LOOM is capable of automatically correcting student responses to MCQs. However, while LOOM provides students with solutions to FTQs, it is not yet able to autonomously correct their individual responses. To correct student responses, lecturers can retrieve individual responses from LOOM and export them as CSV file if desired. This procedure could be automated by applying text mining to inspect student responses for correctness. Given the complexity of text mining methods, it is advisable to source this component from an external software supplier that holds expertise in this field. The integration of text mining in LOOM would simplify the use of FTQs for lecturers and enhance the feedback for students. After integrating text mining, the satisfaction and approval of students regarding the automated correction should be measured and analyzed.

The feedback provided by LOOM is static and limited to the types knowledge or results (KR) and knowledge of correct response (KCR). Not all the guidelines proposed in the theoretical background of this thesis were considered in the initial implementation of LOOM because of time restrictions. Nevertheless, these guidelines should be applied in the further development of LOOM to exploit the full potential of formative feedback. LOOM's feedback should not only consist of KR and KCR, but also involve elaborated feedback (EF). For example, if students are not equal to a CbA, LOOM could provide feedback in the form of explanations or references to corresponding study material. LOOM would, therefore, empower students to independently gain the knowledge necessary to complete the CbA. Furthermore, LOOM should be able to alter the type of feedback or single feedback variables to account for different feedback conditions. LOOM should provide feedback, for instance, according to the goals a lecturer pursues and in consideration of the characteristics and preferences of an individual student. In doing so, LOOM must not be restricted to one form of feedback but combine different variations of feedback. For example, LOOM could apply immediate as well as delayed feedback, so students can benefit from the advantages of both approaches. It is advisable to measure LOOM's effect on student performance before and after modifying the automated feedback in order to analyze the impacts of individual changes.

7 References

Adams, J. A. (1987). Historical review and appraisal of research on the learning, retention, and transfer of human motor skills. *Psychological Bulletin*, *101*(1), 41–74.

Alavi, M. (1994). Computer-Mediated Collaborative Learning: An Empirical Evaluation. *MIS Quarterly*, *18*(2), 159.

Ames, C. (1992). Classrooms: Goals, structures, and student motivation. *Journal of Educational Psychology*, *84*(3), 261–271.

Anderson, L. W., & Krathwohl, D. R. (Eds.). (2001). *A taxonomy for learning, teaching, and assessing: a revision of Bloom's taxonomy of educational objectives*. New York: Longman.

Arbaugh, J. B. (2001). How Instructor Immediacy Behaviors Affect Student Satisfaction and Learning in Web-Based Courses. *Business Communication Quarterly*, *64*(4), 42–54.

Attali, Y., Laitusis, C., & Stone, E. (2016). Differences in Reaction to Immediate Feedback and Opportunity to Revise Answers for Multiple-Choice and Open-Ended Questions. *Educational and Psychological Measurement*, *76*(5), 787–802.

Attali, Y. (2015). Effects of multiple-try feedback and question type during mathematics problem solving on performance in similar problems. *Computers & Education*, *86*, 260–267.

Attali, Y., & van der Kleij, F. (2017). Effects of feedback elaboration and feedback timing during computer-based practice in mathematics problem solving. *Computers & Education*.

Backhaus, K., Erichson, B., & Weiber, R. (2015). *Fortgeschrittene Multivariate Analysemethoden: Eine anwendungsorientierte Einführung* (3., überarbeitete und aktualisierte Auflage). Berlin Heidelberg: Springer Gabler.

Bangert-Drowns, R. L., Kulik, C.-L. C., Kulik, J. A., & Morgan, M. (1991). The Instructional Effect of Feedback in Test-like Events. *Review of Educational Research*, *61*(2), 213.

Baranik, L. E., Barron, K. E., & Finney, S. J. (2007). Measuring Goal Orientation in a Work Domain: Construct Validity Evidence for the 2 × 2 Framework. *Educational and Psychological Measurement*, *67*(4), 697–718.

Bart, Y., Shankar, V., Sultan, F., & Urban, G. L. (2005). Are the Drivers and Role of Online Trust the Same for All Web Sites and Consumers? A Large-Scale Exploratory Empirical Study. *Journal of Marketing*, *69*(4), 133–152.

Blanton, H., Buunk, B. P., Gibbons, F. X., & Kuyper, H. (1999). When Better-Than-Others Compare Upward: Choice of Comparison and Comparative Evaluation as Independent Predictors of Academic Performance. *Journal of Personality and Social Psychology*, *76*(3), 420–430.

Bonk, C. J. (2009). *The World is Open: How Web Technology is Revolutionizing Education*. San Francisco, CA, USA: Jossey-Bass.

Booth, A., & Hyland, P. (Eds.). (2000). *The practice of university history teaching*. Manchester: Manchester Univ. Press.

Brackbill, Y., Bravos, A., & Starr, R. H. (1962). Delay-improved retention of a difficult task. *Journal of Comparative and Physiological Psychology*, *55*(6), 947–952.

Brackbill, Y., & Kappy, M. S. (1962). Delay of reinforcement and retention. *Journal of Comparative and Physiological Psychology*, *55*(1), 14–18.

Brown, G. T. L., Peterson, E. R., & Yao, E. S. (2016). Student conceptions of feedback: Impact on self-regulation, self-efficacy, and academic achievement. *British Journal of Educational Psychology*, *86*(4), 606–629.

Brown, S. A., Race, P., & Smith, B. (1996). *500 tips on assessment* (Vol. 1). London: Kogan Page.

Butler, D. L., & Winne, P. H. (1995). Feedback and Self-Regulated Learning: A Theoretical Synthesis. *Review of Educational Research*, *65*(3), 245–281.

Button, S. B., Mathieu, J. E., & Zajac, D. M. (1996). Goal Orientation in Organizational Research: A Conceptual and Empirical Foundation. *Organizational Behavior and Human Decision Processes*, *67*(1), 26–48.

Buunk, B. P., Collins, R. L., Taylor, S. E., VanYperen, N. W., & Dakof, G. A. (1990). The affective consequences of social comparison: Either direction has its ups and downs. *Journal of Personality and Social Psychology*, *59*(6), 1238–1249.

Chemers, M. M., Hu, L., & Garcia, B. F. (2001). Academic self-efficacy and first year college student performance and adjustment. *Journal of Educational Psychology*, *93*(1), 55–64.

Chen, C.-M., & Chen, M.-C. (2009). Mobile formative assessment tool based on data mining techniques for supporting web-based learning. *Computers & Education*, *52*(1), 256–273.

Clariana, R. B., & Lee, D. (2001). The effects of recognition and recall study tasks with feedback in a computer-based vocabulary lesson. *Educational Technology Research and Development*, *49*(3), 23–36.

Clariana, R. B., Wagner, D., & Roher Murphy, L. C. (2000). Applying a connectionist description of feedback timing. *Educational Technology Research and Development*, *48*(3), 5–22.

Corbett, A. T., & Anderson, J. R. (2001). Locus of feedback control in computer-based tutoring: impact on learning rate, achievement and attitudes (pp. 245–252). ACM Press.

Covington, M. V., & Omelich, C. L. (1984). Task-oriented versus competitive learning structures: Motivational and performance consequences. *Journal of Educational Psychology*, *76*(6), 1038–1050.

Cuseo, J. (2007). The Empirical Case against Large Class Size: Adverse Effects on the Teaching, Learning, and Retention of First-Year Students. *Journal of Faculty Development*, *21*(1), 5–21.

Davis, W. D., Carson, C. M., Ammeter, A. P., & Treadway, D. C. (2005). The Interactive Effects of Goal Orientation and Feedback Specificity on Task Performance. *Human Performance*, *18*(4), 409–426.

DeLone, W. H., & McLean, E. R. (1992). Information Systems Success: The Quest for the Dependent Variable. *Information Systems Research*, *3*(1), 60–95.

Denton, P., Madden, J., Roberts, M., & Rowe, P. (2008). Students' response to traditional and computer-assisted formative feedback: A comparative case study. *British Journal of Educational Technology*, *39*(3), 486–500.

Dihoff, R. E., Brosvic, G. M., Epstein, M. L., & Cook, M. J. (2004). Provision of feedback during preparation for academic testing: Learning is enhanced by immediate but not delayed feedback. *Psychological Record*, *54*(2), 207–231.

Dobbs, R., Madgavkar, A., Barton, D., Labaye, E., Manyika, J., Roxburgh, C., … Madhav, S. (2012). *The world at work: Jobs, pay, and skills for 3.5 billion people* (No. 28). Greater Los Angeles: McKinsey Global Institute.

Düchting, M., Zimmermann, D., & Nebe, K. (2007). Incorporating User Centered Requirement Engineering into Agile Software Development. In J. A. Jacko (Ed.), *Human-Computer Interaction. Interaction Design and Usability* (Vol. 4550, pp. 58–67). Berlin, Heidelberg: Springer Berlin Heidelberg.

Elliott, E. S., & Dweck, C. S. (1988). Goals: An approach to motivation and achievement. *Journal of Personality and Social Psychology*, *54*(1), 5–12.

Festinger, L. (1954). A Theory of Social Comparison Processes. *Human Relations*, *7*(2), 117–140.

Fisher, S. L., & Ford, J. K. (1998). Differential Effects of Learner Effect and Goal Orientation in Two Learning Outcomes. *Personnel Psychology*, *51*(2), 397–420.

Ford, J. K., Smith, E. M., Weissbein, D. A., Gully, S. M., & Salas, E. (1998). Relationships of goal orientation, metacognitive activity, and practice strategies with learning outcomes and transfer. *Journal of Applied Psychology*, *83*(2), 218–233.

Fortes, P. C., & Tchantchane, A. (2010). Dealing with Large Classes: A Real Challenge. *Procedia - Social and Behavioral Sciences*, *8*, 272–280.

Garcia, S. M., Tor, A., & Gonzalez, R. (2006). Ranks and Rivals: A Theory of Competition. *Personality and Social Psychology Bulletin*, *32*(7), 970–982.

Garcia, S. M., Tor, A., & Schiff, T. M. (2013). The Psychology of Competition: A Social Comparison Perspective. *Perspectives on Psychological Science*, *8*(6), 634–650.

Gilbert, P., Price, J., & Allan, S. (1995). Social comparison, social attractiveness and evolution: How might they be related? *New Ideas in Psychology*, *13*(2), 149–165.

Gilman, D. A. (1969). Comparison of several feedback methods for correcting errors by computer-assisted instruction. *Journal of Educational Psychology*, *60*(6, Pt.1), 503–508.

Ginzberg, M. J. (1981). Early Diagnosis of MIS Implementation Failure: Promising Results and Unanswered Questions. *Management Science*, *27*(4), 459–478.

Giroux, H. (2002). The Corporate War Against Higher Education. *Workplace*, *9*, 103–117.

Goldstein, I. L., Emanuel, J. T., & Howell, W. C. (1968). Effect of percentage and specificity of feedback on choice behavior in a probabilistic information-processing task. *Journal of Applied Psychology*, *52*(2), 163–168.

Goodman, J. S., & Wood, R. E. (2004). Feedback Specificity, Learning Opportunities, and Learning. *Journal of Applied Psychology*, *89*(5), 809–821.

Goodman, J. S., Wood, R. E., & Chen, Z. (2011). Feedback specificity, information processing, and transfer of training. *Organizational Behavior and Human Decision Processes*, *115*(2), 253–267.

Goodman, J. S., Wood, R. E., & Hendrickx, M. (2004). Feedback Specificity, Exploration, and Learning. *Journal of Applied Psychology*, *89*(2), 248–262.

Gupta, S., Bostrom, R. P., & Huber, M. (2010). End-user training methods: what we know, need to know. *ACM SIGMIS Database*, *41*(4), 9.

Hair, J. F., Hult, G. T. M., Ringle, C., & Sarstedt, M. (2014). *A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM)*. Thousand Oaks, CA: Sage.

Hardin, A. M., Looney, C. A., & Fuller, M. A. (2014). Self-efficacy, learning method appropriation and software skills acquisition in learner-controlled CSSTS environments: Learning method appropriation. *Information Systems Journal*, *24*(1), 3–27.

Hattie, J., Biggs, J., & Purdie, N. (1996). Effects of Learning Skills Interventions on Student Learning: A Meta-Analysis. *Review of Educational Research*, *66*(2), 99–136.

Hattie, J., & Timperley, H. (2007). The Power of Feedback. *Review of Educational Research*, *77*(1), 81–112.

Heyman, G. D., & Dweck, C. S. (1992). Achievement goals and intrinsic motivation: Their relation and their role in adaptive motivation. *Motivation and Emotion*, *16*(3), 231–247.

Higgins, C. A., & Bligh, B. (2006). Formative computer based assessment in diagram based domains (p. 98). ACM Press.

Hoska, D. M. (1993). Motivating learners through CBI feedback: Developing a positive learner perspective. In J. V. Dempsey & G. C. Sales (Eds.), *Interactive instruction and feedback* (pp. 105–132). Englewood Cliffs, N.J: Educational Technology Publications.

Ilgen, D. R., Fisher, C. D., & Taylor, M. S. (1979). Consequences of individual feedback on behavior in organizations. *Journal of Applied Psychology*, *64*(4), 349–371.

Jacoby, J., Mazursky, D., Troutman, T., & Kuss, A. (1984). When feedback is ignored: Disutility of outcome feedback. *Journal of Applied Psychology*, *69*(3), 531–545.

Jaehnig, W., & Miller, M. L. (2007). Feedback Types in Programmed Instruction: A Systematic Review. *The Psychological Record*, *57*(2), 219–232.

Janson, A., Söllner, M., & Leimeister, J. M. (2015). Towards a Holistic Understanding of Technology-mediated Learning Appropriation. In *Proceedings of the International Conference on Information Systems - Exploring the Information Frontier, ICIS 2015*.

Janta, B., Rathmann, N., Ghez, J., Khodyakov, D., & Yaqub, O. (2015). *Employment and the changing labour market* (Global societal trends to 2030 No. 5).

Kippel, G. M. (1974). Information Feedback Schedules, Interpolated Activities, and Retention. *The Journal of Psychology*, *87*(2), 245–251.

Kluger, A. N., & DeNisi, A. (1996). The effects of feedback interventions on performance: A historical review, a meta-analysis, and a preliminary feedback intervention theory. *Psychological Bulletin*, *119*(2), 254–284.

Krathwohl, D. R. (2002). A Revision of Bloom's Taxonomy: An Overview. *Theory Into Practice*, *41*(4), 212–218.

Kruger, J., & Dunning, D. (1999). Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments. *Journal of Personality and Social Psychology*, *77*(6), 1121–1134.

Kulhavy, R. W. (1977). Feedback in Written Instruction. *Review of Educational Research*, *47*(1), 211–232.

Kulhavy, R. W., & Anderson, R. C. (1972). Delay-retention effect with multiple-choice tests. *Journal of Educational Psychology*, *63*(5), 505–512.

Kulhavy, R. W., White, M. T., Topp, B. W., Chan, A. L., & Adams, J. (1985). Feedback complexity and corrective efficiency. *Contemporary Educational Psychology*, *10*(3), 285–291.

Kulik, J. A., & Kulik, C.-L. C. (1988). Timing of Feedback and Verbal Learning. *Review of Educational Research*, *58*(1), 79.

Lam, C. F., DeRue, D. S., Karam, E. P., & Hollenbeck, J. R. (2011). The impact of feedback frequency on learning and task performance: Challenging the "more is better" assumption. *Organizational Behavior and Human Decision Processes*, *116*(2), 217–228.

Lee, H. W., Lim, K. Y., & Grabowski, B. L. (2010). Improving self-regulation, learning strategy use, and achievement with metacognitive feedback. *Educational Technology Research and Development*, *58*(6), 629–648.

London, M. (2015). *The power of feedback: giving, seeking, and using feedback for performance improvement*. New York, NY: Routledge, Taylor & Francis Group.

Looi, C.-K., Seow, P., Zhang, B., So, H.-J., Chen, W., & Wong, L.-H. (2010). Leveraging mobile technology for sustainable seamless learning: a research agenda. *British Journal of Educational Technology*, *41*(2), 154–169.

Lopez, L. (2009). *Effects of delayed and immediate feedback in the computer-based testing environment*. Indiana State University.

MacGregor, J., Cooper, J. L., Smith, K. A., & Robinson, P. (2000). *Strategies for energizing large classes: from small groups to learning communities*. San Francisco: Jossey-Bass.

Mason, B. J., & Bruning, R. H. (2001). *Providing Feedback in Computer-based Instruction: What the Research Tells Us* (CLASS Project Research Report No. 9). University of Nebraska-Lincoln.

McDonald, B., & Boud, D. (2003). The Impact of Self-assessment on Achievement: The effects of self-assessment training on performance in external examinations. *Assessment in Education: Principles, Policy & Practice*, *10*(2), 209–220.

McGill, T. J., & Klobas, J. E. (2009). A task–technology fit view of learning management system impact. *Computers & Education*, *52*(2), 496–508.

Merrill, D. M. (1965). Correction and review on successive parts in learning a hierarchical task. *Journal of Educational Psychology*, *56*(5), 225–234.

Merrill, M. D., & Stolurow, L. M. (1966). Hierarchical Preview VS Problem Oriented Review in Learning an Imaginary Science. *American Educational Research Journal*, *3*(4), 251–261.

Mertens, W., Pugliese, A., & Recker, J. (2017). Models with Latent Concepts and Multiple Relationships: Structural Equation Modeling. In W. Mertens, A. Pugliese, & J. Recker, *Quantitative Data Analysis* (pp. 37–59). Cham: Springer International Publishing.

Miller, T. (2009). Formative computer-based assessment in higher education: the effectiveness of feedback in supporting student learning. *Assessment & Evaluation in Higher Education*, *34*(2), 181–192.

Morrison, G. R., Ross, S. M., Gopalakrishnan, M., & Casey, J. (1995). The Effects of Feedback and Incentives on Achievement in Computer-Based Instruction. *Contemporary Educational Psychology*, *20*(1), 32–50.

Mory, E. H. (2004). Feedback research revisited. In David H. Jonassen (Ed.), *Handbook of research on educational communications and technology* (2nd ed, pp. 745–783). Mahwah, N.J: Lawrence Erlbaum Associates.

Murphy, P. (2007). Reading comprehension exercises online: The effects of feedback, proficiency and interaction. *Language Learning & Technology*, *11*(3), 107–129.

Narciss, S., & Huth, K. (2004). How to design informative tutoring feedback for multi-media learning. In H. M. Niegemann, D. Leutner, & R. Brünken (Eds.), *Instructional design for multimedia learning* (pp. 181–195). Münster: Waxmann.

Ng'ambi, D., & Bozalek, V. (2013). Editorial: Emerging technologies and changing learning/teaching practices: Editorial. *British Journal of Educational Technology*, *44*(4), 531–535.

Nicol, D. J., & Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: a model and seven principles of good feedback practice. *Studies in Higher Education*, *31*(2), 199–218.

Overbaugh, R. C. (1994). Research-based guidelines for computer-based instruction development. *Journal of Research on Computing in Education*, *27*(1), 29–47.

Parshall, C. G., Davey, T., & Pashley, P. J. (2000). Innovative Item Types for Computerized Testing. In W. J. van der Linden & G. A. W. Glas (Eds.), *Computerized Adaptive Testing: Theory and Practice* (pp. 129–148). Dordrecht: Springer Netherlands.

Payne, R. B., & Hauty, G. T. (1955). Effect of psychological feedback upon work decrement. *Journal of Experimental Psychology*, *50*(6), 343–351.

Phillips, J. M., & Gully, S. M. (1997). Role of goal orientation, ability, need for achievement, and locus of control in the self-efficacy and goal-setting process. *Journal of Applied Psychology*, *82*(5), 792–802.

Phye, G., & Baller, W. (1970). Verbal retention as a function of the informativeness and delay of informative feedback: A replication. *Journal of Educational Psychology*, *61*(5), 380–381.

Phye, G. D., & Andre, T. (1989). Delayed retention effect: Attention, perseveration, or both? *Contemporary Educational Psychology*, *14*(2), 173–185.

Price, M., Handley, K., Millar, J., & O'Donovan, B. (2010). Feedback: all that effort, but what is the effect? *Assessment & Evaluation in Higher Education*, *35*(3), 277–289.

Pridemore, D. R., & Klein, J. D. (1995). Control of Practice and Level of Feedback in Computer-Based Instruction. *Contemporary Educational Psychology*, *20*(4), 444–450.

Ramaprasad, A. (1983). On the definition of feedback. *Behavioral Science*, *28*(1), 4–13.

Rice, R. E., & Borgman, C. L. (1983). The use of computer-monitored data in information science and communication research. *Journal of the American Society for Information Science*, *34*(4), 247–256.

Rietsche, R., Söllner, M., & Seufert, S. (2017). Digital formative learning assessment tool - Torwards helping students to take ownership of their learning. In *Proceedings of the 25th European Conference on Information Systems (ECIS), Guimarães, Portugal, June 5-10, 2017* (pp. 2960–2969).

Ringle, C. M., Wende, S., & Becker, J.-M. (2015). SmartPLS (Version 3.2.7). Bönningstedt: SmartPLS GmbH.

Sadler, D. R. (1989). Formative assessment and the design of instructional systems. *Instructional Science*, *18*(2), 119–144.

Salmoni, A. W., Schmidt, R. A., & Walter, C. B. (1984). Knowledge of results and motor learning: A review and critical reappraisal. *Psychological Bulletin*, *95*(3), 355–386.

Scalise, K., & Gifford, B. (2006). Computer-Based Assessment in E-Learning: A Framework for Constructing "Intermediate Constraint" Questions and Tasks for Technology Platforms. *The Journal of Technology, Learning, and Assessment*, *4*(6).

Schimmel, B. J. (1988). Providing meaningful feedback in courseware. In D. H. Jonassen (Ed.), *Instructional Desing for Microcomputer Courseware* (pp. 183–195). Hillsdale, NJ: Lawrence Erlbaum Associates.

Schmidt, R. A. (1975). A schema theory of discrete motor skill learning. *Psychological Review*, *82*(4), 225–260.

Schmidt, R. A., Young, D. E., Swinnen, S., & Shapiro, D. C. (1989). Summary knowledge of results for skill acquisition: Support for the guidance hypothesis. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *15*(2), 352–359.

Schooler, L. J., & Anderson, J. R. (2008). The Disruptive Potential of Immediate Feedback (pp. 702–708). Presented at the Research Showcase @ CMU.

Schroth, M. L. (1992). The effects of delay of feedback on a delayed concept formation transfer task. *Contemporary Educational Psychology*, *17*(1), 78–82.

Schroth, M. L. (1997). Effects of Frequency of Feedback on Transfer in Concept Identification. *The American Journal of Psychology*, *110*(1), 71.

Sein, M. K., Henfridsson, O., Purao, S., Rossi, M., & Lindgren, R. (2011). Action Design Research. *MIS Quarterly*, *35*(1), 37–56.

Shute, V. J. (2008). Focus on Formative Feedback. *Review of Educational Research*, *78*(1), 153–189.

Smith, P. L., & Ragan, T. J. (1993). Designing Instructional Feedback for Different Learning Outcomes. In J. V. Dempsey & G. C. Sales (Eds.), *Interactive instruction and feedback*. Englewood Cliffs, N.J: Educational Technology Publications.

Söllner, M., Bitzer, P., Janson, A., & Leimeister, J. M. (2017). Process is king: Evaluating the performance of technology-mediated learning in vocational software training. *Journal of Information Technology*.

Srinivasan, A. (1985). Alternative Measures of System Effectiveness: Associations and Implications. *MIS Quarterly*, *9*(3), 243.

Straub, D., Limayem, M., & Karahanna-Evaristo, E. (1995). Measuring System Usage: Implications for IS Theory Testing. *Management Science*, *41*(8), 1328–1342.

Taylor, S. E., Wood, J. V., & Lichtman, R. R. (1983). It Could Be Worse: Selective Evaluation as a Response to Victimization. *Journal of Social Issues*, *39*(2), 19–40.

Thelwall, M. (2000). Computer-based assessment: a versatile educational tool. *Computers & Education*, *34*(1), 37–49.

Trice, A. W., & Treacy, M. E. (1988). Utilization as a dependent variable in MIS research. *ACM SIGMIS Database*, *19*(3–4), 33–41.

University of St.Gallen. (2013). *The HSG: a portrait*. St.Gallen: University of St.Gallen. Retrieved from http://www.unisg.ch/~/media/internet/content/dateien/unisg/universitaet/portrait/zahlenfakten/2013_zahlenfakten_e.pdf

University of St.Gallen. (2016). *The HSG: a portrait*. St.Gallen: University of St.Gallen. Retrieved from https://www.unisg.ch/~/media/internet/content/dateien/unisg/universitaet/portrait/zahlenfakten/2016_factsfigures_e.pdf

University of St.Gallen. (2017a). Teaching and Learning. Retrieved March 25, 2017, from https://www.unisg.ch/en/universitaet/ueber-uns/portraet/hsginzahlen/lehren

University of St.Gallen. (2017b). *The HSG: a portrait*. St.Gallen: University of St.Gallen. Retrieved from https://issuu.com/universityofstgallen/docs/unisg_zahlen_fakten_e_2017_web van der Kleij, F. M., Feskens, R. C. W., & Eggen, T. J. H. M. (2015). Effects of Feedback in a Computer-Based Learning Environment on Students' Learning Outcomes: A Meta-Analysis. *Review of Educational Research*, *85*(4), 475–511.

van der Kleij, Fabienne M., Eggen, T. J. H. M., Timmers, C. F., & Veldkamp, B. P. (2012). Effects of feedback in a computer-based assessment for learning. *Computers & Education*, *58*(1), 263–272.

van der Kleij, Fabienne M., Timmers, C. F., & Eggen, T. J. H. M. (2011). The effectiveness of methods for providing written feedback through a computer-based assessment for learning: a systematic review.

VandeWalle, D., Brown, S. P., Cron, W. L., & Slocum, J. W., Jr. (1999). The influence of goal orientation and self-regulation tactics on sales performance: A longitudinal field test. *Journal of Applied Psychology*, *84*(2), 249–259.

Venkatesh, Morris, Davis, & Davis. (2003). User Acceptance of Information Technology: Toward a Unified View. *MIS Quarterly*, *27*(3), 425.

Wills, T. A. (1981). Downward Comparison Principles in Social Psychology. *Psychological Bulletin*, *90*(2), 245–271.

Winstein, C. J., & Schmidt, R. A. (1990). Reduced frequency of knowledge of results enhances motor skill learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *16*(4), 677–691.

Yorke, M. (2003). Formative Assessment in Higher Education: Moves Towards Theory and the Enhancement of Pedagogic Practice. *Higher Education*, *45*(4), 477–501.

8 Appendix

The appendix involves detailed categorizations of feedback types, computer-based assessment types and assessment item types.

8.1 Appendix 1: Feedback Types

The following Table 11 presents feedback types categorized by Shute (2008) including knowledge of results, knowledge of correct response and different forms of elaborated feedback. The feedback types are ordered from the least complex to the most complex one.

Table 11: Feedback Types Arrayed by Complexity, Adapted from Shute (2008)

Feedback type	Description
No feedback	Refers to conditions where the student is presented a question and is required to respond, but there is no indication as to the correctness of the student's response.
Knowledge of results (KR)	Also called *verification*, or *knowledge of outcome*, it informs the student about the correctness of the response(s), such as right/wrong or overall percentage correct.
Knowledge of correct response (KCR)	Also known as *correct response*, it informs the student of the correct answer to a specific problem with no additional information.
Try-again	Also known as *multiple try, answer-until-correct*, or *repeat-until-correct* feedback, it informs the student about an incorrect response and allows the student one or more attempts to answer the question.
Error-flagging	Also known as *location of mistakes*, error-flagging highlights errors in a solution, without giving correct answer.
Elaborated feedback (EF)	A general term, it refers to providing an explanation about why a specific response was correct, and it might allow the student to review part of the instruction (see below for six types of elaborated feedback).
Attribute isolation	Elaborated feedback that presents information addressing central attributes of the target concept or skill being studied.
Topic-contingent	Elaborated feedback that provides the student with information relating to the target topic currently being studied. This might entail simply re-teaching material.
Response-contingent	Elaborated feedback that focuses on the student's specific response. It may describe why the answer is wrong and why the correct answer is correct. This does not use formal error analysis.
Hints/cues/prompts	Elaborated feedback that guides the student in the right direction (e.g., strategic hint on what to do next or a worked example or demonstration). It avoids explicitly presenting the correct answer.
Bugs/misconceptions	Elaborated feedback that requires error analysis and diagnosis. It provides information about the student's specific errors or misconceptions.
Informative tutoring	The most elaborated feedback, this presents verification feedback, error-flagging, and strategic hints on how to proceed. The correct answer is not usually provided.

8.2 Appendix 2: Computer-based Assessment Types

The following Table 12 presents different types of summative and formative computer-based assessment defined by Thelwall (2000).

Table 12: Taxonomy of Applications of Assessment, Adapted from Thelwall (2000)

Area	Assessment type	Description
Summative	Exam	An assessment solely for grading purposes such as an exam at the end of a unit of study.
Formative/summative	Grading test	An assessment for grading but which also provides feedback intended to direct future studies. Includes a mid-unit small test, or weekly problem sets.
Formative	Open access test	A grading test that doubles as a set of exercises because students are allowed to practice before sitting the test.
Formative	Self-test	An assessment designed to give feedback to a student on their progress with a section of a unit of study.
Formative	Exercises	A problem set designed to consolidate learning on a section of a unit of study.
Formative	Programmed learning tool	A linear computer assisted learning package – based upon a question and answer session.
Formative	Computer assisted learning quiz	A marked exercise integrated into a computer assisted learning package, for example a multiple-choice question presented after a slide containing new information.
Formative	Adaptive computer assisted learning quiz	A marked exercise integrated into a computer assisted learning package used to test the student but also used to adapt the teaching of the package to student weaknesses.
Formative	Diagnostic test	An assessment of prior learning taken before a unit of study.

8.3 Appendix 3: Assessment Item Types

In the following Table 13, Scalise and Gifford (2006) categorize assessment items along two dimensions. Item types are organized from left to right according to their level of constraint. Within these types, an additional ordering indicates the complexity level of an item.

Table 13: Taxonomy for Assessment Items, Adapted from Scalise and Gifford (2006)

		Most Constrained →						***Least Constrained***
		Fully Selected	*Intermediate Constraint Item Types*					*Fully Constructed*
		1	**2**	**3**	**4**	**5**	**6**	**7**
		Multiple-Choice	**Selection/ Identification**	**Reordering/ Rearrangement**	**Substitution/ Correction**	**Completion**	**Construction**	**Presentation/ Portfolio**
Less Complex	**A**	True/False	Multiple True/False	Matching	Interlinear	Single Numerical Constructed	Open-Ended Multiple-Choice	Project
↓	**B**	Alternate Choice	Yes/No with Explanation	Categorizing	Sore-Finger	Short-Answer & Sentence Completion	Figural Constructed Response	Demonstration, Experiment, Performance
↓	**C**	Conventional or Standard Multiple-Choice	Multiple Answer	Ranking & Sequencing	Limited Figural Drawing	Cloze-Procedure	Concept Map	Discussion, Interview
More Complex	**D**	Multiple-Choice with New Media Distractors	Complex Multiple-Choice	Assembling Proof	Bug/Fault Correction	Matrix Completion	Essay & Automated Editing	Diagnosis, Teaching

9 Declaration of Authorship

“I hereby declare

- that I have written this thesis without any help from others and without the use of documents and aids other than those stated above;
- that I have mentioned all the sources used and that I have cited them correctly according to established academic citation rules;
- that I have acquired any immaterial rights to materials I may have used such as images or graphs, or that I have produced such materials myself;
- that the topic or parts of it are not already the object of any work or examination of another course unless this has been explicitly agreed on with the faculty member in advance and is referred to in the thesis;
- that I will not pass on copies of this work to third parties or publish them without the University’s written consent if a direct connection can be established with the University of St.Gallen or its faculty members;
- that I am aware that my work can be electronically checked for plagiarism and that I hereby grant the University of St.Gallen copyright in accordance with the Examination Regulations in so far as this is required for administrative action;
- that I am aware that the University will prosecute any infringement of this declaration of authorship and, in particular, the employment of a ghostwriter, and that any such infringement may result in disciplinary and criminal consequences which may result in my expulsion from the University or my being stripped of my degree.”

Date and signature

....

By submitting this academic term paper, I confirm through my conclusive action that I am submitting the Declaration of Authorship, that I have read and understood it, and that it is true.