DI Josef Wachtler BSc

# Interaction-based Support of Selective Attention in Online Courses

**DOCTORAL THESIS**

to achieve the university degree of
Doktor der technischen Wissenschaften

submitted to
**Graz University of Technology**

Supervisor
Priv.-Doz. Dipl.-Ing. Dr. techn. Martin Ebner

Institute of Interactive Systems and Data Science

Graz, August 2019

This document is set in Palatino, compiled with pdfLaTeX2e and `Biber`.

The LaTeX template from Karl Voit is based on KOMA script and can be found online: https://github.com/novoid/LaTeX-KOMA-template

# Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present doctoral thesis.

Date

Signature

# Acknowledgments

At first, I'd like to thank the supervisor of this thesis, Dr. Martin Ebner. I say thank you for the opportunity to perform this work and for his extraordinary support.

Furthermore, I want to thank all the students, who contributed valuable parts of this work during their master theses. A huge thank you also goes to the teachers and students, who used the platform created with this work. The results of these usages provided instrumental insights in the ramifications of the usage of the platform.

Many thanks to my personal assistants, who supported me with the physical tasks of this work. Additionally, I'd like to thank all proofreaders for their professional work.

Finally, a special thank you is due to my parents, Ingrid and Josef Wachtler for their incredible support in all phases of this work and in the life beneath it.

# Abstract

Interaction and communication are considered to be very important influencing factors of human learning. They are able to support the attention of students in many positive ways. Because of that, it is often recommended to provide such technologies in many different forms at online courses. This should be done to increase both, the learning success and the satisfaction of the students.

To offer such technologies for learning videos, a web platform is developed which applies different methods of interactivity to videos or live-broadcastings. This platform is available for registered and authenticated users only. In addition, it substitutes between different groups of users, like students and teachers. The teachers are able to create events, which are providing a video or a live-broadcasting. While creating the events they are required to select the interaction methods which should be offered. The interaction methods are separated into four groups. The first one provides simple interactions, which are not related to the content of the event, in automatic and random ways. The second group consists of interactions which are triggered by the students manually. This includes a possibility to ask a question to the teacher. In contrast to that, the third group implements the opposite way. This means that the teacher is able to send questions to the students in real time. The final group provides the possibility to create questions which are planned before releasing the event. This means that such a question will occur at a specific position in the video according to the plan. To manage the interactions, an algorithm is implemented to handle the interactions during their life cycle. This includes the planning of the automatic interactions and displaying all interactions to the students.

All of the interaction methods offering questions are providing a detailed analysis of the students' answers. These analysis features are available to

the teacher for each student. In addition, a single student is able to view the analysis of her/his answers. To monitor the attendance of the students at the events, an algorithm is implemented. This algorithm provides two components. The first one consists of a detailed recording of the watched timespans in order to state for each student when she/he watched which part of the video or the live-broadcasting. In contrast to that, the second part of the algorithm calculates a so-called attention level, which states how attentive the students were during the watched timespans. This calculation is mainly based on the reaction times to the interactions. With this recording and the calculated attention level, it is possible to monitor the attendance of the students.

To evaluate the web platform as well as the concept of interactive videos or live-broadcastings, it was used in productive environments. These usages consisted of recordings of lectures, massive open online courses, flipped-classroom concepts and videos of lab experiments. It is pointed out that the platform has benefits for the students regarding their attention during the events and an increased long term learning success. A requirement for these benefits is the integration of the platform in the didactic concept of the course. In addition, it is shown that the workload of the teacher could be optimized by using the features of the platform.

# Contents

# List of Figures

# List of Tables

# 1 Introduction

> Communication - the human connection - is the key to personal and career success.
>
> *(Paul J. Meyer[1])*

Communication is considered to be a crucial factor for success. This not only includes personal matters or career related fields, but also the process of learning. This statement builds the base for this work. Because of the various benefits of interaction and communication (see Section 1.1), the idea of interactivity is applied to learning videos. This is done to evaluate several related issues as pointed out by Section 1.2.

## 1.1 Motivation

Selective Attention

It is widely known that today, students are confronted with a growing number of different pieces of information. This means that a lot of shapes, texts, colors, figures and sounds are presented to them. It is clear that they are only able to process a limited number at the same time (Shiffrin & Gardner, 1972). Because of that, most of these information is filtered out centrally (Moran & Desimone, 1985). Based on that, a mechanism called selective attention has been introduced. This mechanism is labeled as one of the most crucial resource for human learning (Heinze et al., 1994). If this attention is managed, it is possible to enhance both, behavioral and neuronal performance (Spitzer, Desimone, & Moran, 1988).

Interaction and Communication

According to Carr-Chellman and Duchastel (2000), such a management of attention could be achieved by providing interactivity because the addition

[1] https://www.brainyquote.com/quotes/paul_j_meyer_190945, last accessed October 4, 2018

of interaction and communication is seen as a key component for a successful online course. It is necessary to distinguish between synchronous and asynchronous communication. Furthermore, there are different groups of interacting participants. The typical tool for asynchronous communication is an online forum or a news group. As suggested by the term asynchronous, such tools are independent from the time aspect. This means that asking and answering people are not required to be online at the same time. In addition, a forum is a useful place to build learning communities.

The second way of communication is in a synchronous form. This means that the communication takes place in real time which is the case at online chats or something similar. It is clear that this form of communicating is more direct and because of that considered to be more casual. A further advantage is that because of this direct approach, there is no delay between a question and an answer.

As mentioned above, there are also different groups of communicating and interacting people. They should be enabled to perform the interaction in all possible directions. This means that on the one hand, a learning platform should provide the possibility for the students to communicate with other students and on the other hand, it is required to offer methods for the communication between the students and the teacher.

Overfeeding

The different methods of interaction and communication are useful techniques to support the attention of the students. This means that they could help to prohibit the overfeeding of the students with content (Helmerich & Scherer, 2007). The main reason for this overfeeding is the increased usage of multimedia content which is leading to an increased filtering of information. The teacher could use different forms of interactivity to avoid this effect. Mainly, the application of content related questions is considered to be most helpful.

From a teacher's point of view, interactive components are not only useful to manage students' attention, they are also a valuable resource for the teacher to answer the following questions for himself (Helmerich & Scherer, 2007).

- Are the students able to follow the explanations?
- Is the speed of teaching appropriate?

- Is the previous knowledge of the students sufficient?
- Was it possible for the participants to understand the content?
- Is the way of presenting suited for the target audience?
- Are the topics adequate for the target audience?

Massive Open Online Courses

All of the mentioned remarks regarding the attention of the students and the important management of it (e.g. with the help of interactivity) are becoming even more important with the increased usage of learning videos. This growing application of videos is mainly based on the lately evolved trend of so-called MOOCs[2] (Khalil & Ebner, 2013). However, videos were not considered very valuable for the purpose of learning due to the maxim "TV is easy and book is hard" (Salomon, 1984). The motivating factor for this maxim is based on the fact that the technical foundations of videos changed dramatically during the last decades, but the role of the watchers is still more or less unchanged. If the technical aspects are compared over time it can be seen that in the early days of videos, they were presented to the students by using projectors or something similar. Today, it is a common practice to use a mobile phone or other devices to search up a video on the internet. So it is possible to watch a video anytime and everywhere. In contrast to these changed watching possibilities, it is still true that a video is a passive medium and because of that, the watchers are inactive.

Most important digital Media

As stated by Lehner (2014), nowadays the most important digital media on the internet is the video. A reason for this importance might be the growing quality of the produced content. Due to that, creators of videos are forced to think about new and creative ways to stand out (Tembrink, Szoltysek, & Unger, 2013; dpa, 2015). As indicated above, a useful tool is the application of interactive components. This application seems to be the best choice because students are familiar to interactivity (Lehner, 2014). It is pointed out that students in general are favoring being challenged. Such a challenge could be achieved by providing interactive components. Because of the assumption that students are used to such interactive components, they are prone to like them.

[2]short for Massive Open Online Course

## 1.2 Research Questions

As indicated above, interactive components are likely to be of valuable use in the field of teaching. Because of that, this work tries to transfer these benefits to learning videos. This leads to the following research question:

*"Can students and teachers benefit from interactive videos used at institutions of higher education, at massive open online courses, or at school?"*

This general research question should be answered by assessing the following more detailed questions:

1. How to develop an interaction-supported web platform for live-broadcastings or on-demand videos?
2. Is it possible to monitor the attendance of students with this platform?
3. Can interactivity, provided by the developed platform, support the attention of attendees and therefore their learning success?
4. Is it possible to assess their understanding of the content using the interactive components of this platform?
5. Can the data generated by the platform used to perform learning analytics?

The first one covers the technical aspects of the development process of such a web platform. In contrast to that, question number two means that it is evaluated how genuine an attendance monitoring, which states which student watched which part of a video, could be. The following two questions are mainly related to the understanding of the students and a possible learning success achieved by a higher attention which should be established by the provided interactive components. The last question covers the usage of the obtained data for a better understanding of the students and the way of learning.

## 1.3 Overview

At first, some related work is presented by Chapter 2. This includes mainly different topics and studies about various applications of interactive components in different learning situations. This is followed by presenting the methods used to achieve the goals of this work (see Chapter 3). After that, Chapter 4 presents the implementation and the evaluation of an early prototype of the developed web application which worked only in a limited form at live-broadcastings. Based on this prototype, the new web platform is implemented (see Chapter 5). This developed web platform for learning videos and live-broadcastings is evaluated and discussed by the Chapters 7 and 8. Finally, an outlook to possible future work is given (see Chapter 9) and the main aspects of this work are summed up by Chapter 10.

# 2 Related Work

To gain a better understanding of this field of research, namely interactivity, this chapter shows some related work. This means that at first, some evaluations of audience response systems are shown by Section 2.1 because they are very similar to videos which are supported with interactivity. After that, an introduction of already existing interactive video platforms (see Section 2.2) is presented. This is followed by a presentation of some studies which are analyzing the usage of interactive videos (see Section 2.3). Finally, two possible fields of usage of the developed web platform are examined, namely online assessment and attendance monitoring (see Sections 2.4 and 2.5).

## 2.1 Audience Response Systems

An ARS[1] (Tobin, 2005) is a tool to increase the level of interactivity in standard classroom situations. It is a system which consists of both, software and hardware components. In general, it is used to perform the following actions: ARS Definition

- submitting questions or other similar little tasks to students
- these tasks or questions are answered by the students
- the results are shown to the teacher

For that, there are different possibilities. It could be that the questions are sent to the students using a special device or it is possible to simply use the existing projector of the classroom. Furthermore, the answering could be done using a special handset or an app could be used which runs on the

[1]short for Audience Response System

| Group | % Formal Collaboration | % Correct: Questions | % Correct: Quiz |
|---|---|---|---|
| standard lecture | - | - | 57 % |
| hand raising | 76 % | 98 % | 60 % |
| answering cards | 97 % | 92 % | 52 % |
| ARS | 100 % | 82 % | 60 % |

Table 2.1: The results of an ARS study of Stowell and Nelson (2007) are pointing out that interactive components are supportive to the attention.

mobile phones of the students. The same two possibilities (special device or existing hardware) are also available for the presenting of the results to the teacher.

The typical field of usage of an ARS is the classroom of a university. As indicated by Section 1.2, this work tries to transfer the possibilities of an ARS to online settings. This means that interactive components will be added to videos and live-broadcastings like an ARS does in a classroom or lecture theater. The main reason for presenting ARS related studies in the following is that it is tried to support the attention of the watchers of videos in the same way as an ARS does in a real classroom.

ARS Studies

A general study about ARS was done by Stowell and Nelson (2007). In this study it was tried to evaluate the usage of an ARS by looking at the ramifications to the collaboration, the learning success and the emotions. For that, a psychology lecture with a length of 30 minutes was held in four different forms:

1. conventional lecture in a standard classroom setting without interactivity
2. additional question answering by raising hands
3. using different cards to answer multiple choice questions
4. application of an ARS

Furthermore, it was necessary to measure the learning success. For that, the participants of the lecture were required to perform a quiz after the lecture.

The results of this study are shown by Table 2.1. It can be seen that the formal collaboration is significantly lower at the hand raising group in comparisons to the cards group or the ARS group. A possible reason for that might be that the level of anonymity is higher at the latter two groups. Furthermore, it could be observed that the correctness rate is higher at the hand raising group and at the group with the answering cards. Only the group with the ARS performed a little worse then the other two groups.

When comparing the results of the quiz which was performed after the lecture, one might notice that there are no significant differences between the four groups. However, it could be observed that the results to the quiz of the ARS group (60%) is the closest quiz result to the question result (82%). This suggests that an ARS delivers the best prediction of the learning success of the students.

To measure the emotions, an AEQ[2] was used (Pekrun, Goetz, Titz, & Perry, 2002). With this method, the level of pleasure, hope, anger and boredom was measured before, during and after the lecture. This measurement pointed out that there are no big differences of the emotional state of the participants of the four groups. The only group which showed an enhancement of the positive emotions however, was the group using the ARS.

A further study regarding ARS defined its goal in evaluating if the usage of an ARS could help to increase the learning experience. This study was done by Latessa and Mouw (2005) with a group of 46 participants of a lecture. For that, these attendees were confronted with questions provided by an ARS.

The questions presented by the ARS were asked after a brief introduction. A short while before the end of the lecture the same questions were asked again. After each round of questioning the results were shown to the students in a statistical and anonymous form.

To evaluate the usage of the ARS, the students were asked to answer four questions regarding this usage. With these questions the students were able to compare a lecture supported with an ARS with a lecture held in a common way. In Table 2.2 the questions and the results are printed. It can be seen

[2]short for Academic Emotions Questionaire

| | A Lot | Some | Little | None |
|---|---|---|---|---|
| To what degree did the ARS make this lecture more fun that traditional lecture formats? | 84 % | 16 % | 0 % | 0 % |
| To what degree did the ARS make you more attentive than traditional lecture formats? | 67 % | 33 % | 0 % | 0 % |
| To what degree did the ARS help you learn more than traditional lecture formats? | 22 % | 63 % | 15 % | 0 % |
| Financial considerations aside, how likely are you to consider using the ARS in your work? | 44 % | 33 % | 15 % | 8 % |

Table 2.2: Exactly these questions have been asked to the students of a lecture supported by an ARS by Latessa and Mouw (2005). It was pointed out that the attention and the learning success was increased by using an audience response system.

that the usage of an ARS led to an increased level of attention as well as an elevated entertainment value. Furthermore, the students got the feeling that they are learning more with the help of the ARS than without it.

A more detailed study consists of the evaluation of the usage of a system similar to an ARS in conjunction with an electronic whiteboard. The data for this study were gathered at a preparation course for foreign students at a university located in Great Britain. In the course of the study, Cutrim (2008, 1) used different methods for the evaluation. For example, the classroom was observed and the students were required to fill in a questionnaire before and after the lecture. Furthermore, the students were interviewed and the teachers were asked for feedback.

The study pointed out that the collaboration of the students is increased by the usage of an ARS. As mentioned by the study presented above, the existence of an entertainment value was confirmed. In addition, such an ARS is a valuable tool to gather feedback from the students. A reason for these benefits might be that the level of anonymity is quite high when using an

ARS. However, the level of interactivity is still low. To address this issue, it is recommended to provide the possibility for discussion after each question presented by an ARS.

Silliman, Abbott, Clark, and McWilliams (2004) performed a very detailed study which evaluated the strengths and the weaknesses of an ARS. Furthermore, the students were required to report their experiences while using such a system. To cover a broad range of different usage scenarios, the study was done with different types of audience. This means that the ARS was used at two different courses where the first was held for beginners in the field of engineering and the second, for older and experienced technicians. In addition, an ARS was used at a class in school and in contrast to that, such a system was the tool of choice for obtaining and evaluating the learning goals of a large group of students. The data for the evaluation of the ARS were gathered using different methods:

- examining the feedback of the students given through the course evaluation
- a survey
- the learning success of the students

With these different types of data, it is possible to state some advantages and disadvantages of using an ARS. At first the main advantages are presented:

- Long and therefore taunting lectures are becoming more agile.
- The collaboration and the attention is increased.
- Students as well as teachers are receiving valuable feedback.
- There are many possibilities for analysis.

In contrast to these advantages, there are also some disadvantages:

- The lecture is unnecessary slowed down.
- Students are forced to give answers to the questions presented by the ARS very fast. This is reported to be a little bit disturbing.
- The risk of technical problems is quite high.
- The distribution of the handsets to answer the questions could be time consuming.

Finally, it is pointed out that the usage of an ARS is able to improve the setting for the learners considerably. Furthermore, it is also able to create a positive effect on the learning success of the students. However, it is important not to overuse such a system.

ARS Systems

The usage of an ARS is documented by Helmerich and Scherer (2007). They used a self-developed ARS in a productive environment at a lecture with 60 to 70 participants. The developed ARS is called Ping-Pong and in contrast to a typical ARS, it is using the devices of the students instead of special hardware handsets to answer the questions. From a technical point of view, this system is based on a LAMP-server[3]. This server provides the complete logic of the program which means that it implements all functionalities of the ARS and provides an API[4]. The provided API is used by different front-ends which are offered for various devices and platforms. Furthermore, there exists a web application.

In general, the usage of Ping-Pong could be divided in three steps. The first step is the creation of the questions. For that, it is possible to create pools of questions which could be used at different lectures. From these pools, a so-called collection of questions has to be created for each individual lecture. This indicates that the questions, created in the pools, are reusable. It seems to be obvious that the second phase of using Ping-Pong is the actual usage at the lecture. Here the teacher can select a question from the previous created collection and send it to the students. Now, they are asked to answer it. The results of the questions are presented to the teacher and based on that, it is recommended that the teacher discusses the results with the students and adapts the lecture accordingly. The final step consists of the evaluation of the questions and the results by the teacher. For that, there are different forms of statistical methods of analysis. This should also be done to gain feedback which should help to improve the lecture.

The usage of Ping-Pong points out some observations and implications. It is observed that the majority of the students were expecting an increased quality of the lecture because of the ARS. In addition, the evaluation shows that the number of answering students is constantly over 50%. This is a higher value in comparison to answering the questions by hand raising.

[3] short for Linux, Apache, MySQL and PHP
[4] short for Application Programming Interface

Based on that, it is possible to state that the level of interactivity is increased by the usage of an ARS. This elevated degree of interactivity should help to avoid negative effects of listening to lectures in a passive way only. From a teacher's point of view, the offered possibilities of analysis during the lecture are a valuable feedback to instantly react to the understanding (or misunderstanding) of the students.

A very similar tool was developed at Graz University of Technology by Haintz, Pichler, and Ebner (2014). This tool was originally called realfeedback and is now commercially available under the name feedbackr[5]. In contrast to the ARS presented above, this system is completely web-based. This means that the devices of the students are used to answer the questions through the web browser. In addition to this commercial tool which has its roots in an academic environment, there are a lot of other more or less similar systems. The following list presents some of them:

- Turning Technologies: Assessment Delivery and Data Collection Solutions [6]
- PowerComARS: Audience Response System [7]
- IML: Harnessing Audience Insight [8]
- and many others

In comparison to the presented solutions which are based on software only, these listed systems are in need of a special handset to answer the questions (Helmerich & Scherer, 2007). This leads to some problems. At first, there is the requirement to distribute the handsets to the students at the beginning of the lecture. Furthermore, it seems to be obvious that the lecture theaters have to be equipped with the handsets and some piece of hardware to receive the answers from the handsets. Such a system is typically much more expensive than software based solutions as well as the level of maintenance is increased (e.g. changing batteries in the handsets). However, such commercial systems are often well integrated in the most commonly used presentation tools like LibreOffice Impress[9].

[5] https://www.feedbackr.io, last accessed October 8, 2018
[6] http://www.turningtechnologies.com/, last accessed October 8, 2018
[7] http://www.powercomars.com/, last accessed October 8, 2018
[8] http://imlworldwide.com/, last accessed October 8, 2018
[9] http://www.libreoffice.org/, last accessed October 8, 2018

## 2.2 Interactive Video Platforms

Overview Interactive features have increased lately in different applications. This also includes the field of videos. Because of that, this section explains the functionalities and the advantages as well as disadvantages of the following platforms and technologies for enriching videos with different methods of interactivity:

- YouTube[10]
- TEDed[11]
- Adways[12]
- Wirewax[13]
- QuizCram
- ILVP

YouTube When examining video platforms, it is necessary to take a look at YouTube. This popular platform provides some interactive features too. This means that after uploading a video, interactive content could be added. Mainly, these interactive components are overlays over the video which are displayed at a given position in the video. The mentioned overlays could be used to display different things. This ranges from simple text and links over images to interactive polls. The creation of such interactive components is very simple however there are some drawbacks. This means that the features for analyzing the watchers reaction to the interactive parts is fairly basic. Furthermore, the time of occurrence is marked in the timeline of the video. The latter problem is quite important because it limits the usefulness of such interactive parts for learning purposes. This means that the watchers are enabled to jump from an interactive part to another to simply get the interactive exercises done.

TEDed In comparison to that, the interactive components offered by TEDed are using a completely different approach. This platform simply embeds a video

[10] https://creatoracademy.youtube.com/page/lesson/community-tab, last accessed October 8, 2018
[11] https://ed.ted.com/, last accessed October 8, 2018
[12] http://www.adways.com/, last accessed October 8, 2018
[13] https://www.wirewax.com/, last accessed October 8, 2018

(e.g. from YouTube) and displays its interactive parts beneath the video. This indicates that the video and the interactive components are not connected and therefore displayed during the whole runtime of the video. It is possible to display different types of questions and polls or also simple text marker to support the video. As mentioned, this is a problematic scenario for the purpose of learning because all of the interactive components are available at all time.

A further platform for interactive videos is called Adways. As suggested by the name, the foundations of this platform are in the field of creating interactive videos for advertisement. However, it also provides a solution for the creation of learning videos with interactive content. The possibilities of interactive components are ranging from simple text overlays and point-and-click interactions to more complex parts like questions and polls. Unfortunately, the position of the interactive parts is given away in the timeline of the video like it is the case with the built-in possibilities of YouTube. `Adways`

Very similar to Adways is the platform named Wirewax. It has its focus on different point-and-click components, which are called hotspots. These hotspots are used to bind interactions to elements shown in the video. This means that for example in a video promoting different goods, it is possible to place a hotspot on it and if the watcher clicks on the good, she/he may be linked to a shop, where the good is on sale. For the purpose of learning this feature could be used to place explanations of the shown elements in the video. Furthermore, it is possible to let the watcher decide how the video should proceed based on some interactive components, which are asking what should happen next. However, there are no possibilities to place assessment questions or polls in the video. Again, this platform marks the position of the interactive components in the timeline of the video. `Wirewax`

In summary, it can be seen that there is one common problem affecting the most platforms for interactive videos. Namely, they are pointing out the position of the interactive parts in the timeline of the video. This is unfortunate because in the course of learning, the interactive components are typically used to assess the understanding of the students of the contents of the video. For that, the students are required to watch the video but with the position of the interactive components given away, students are tending to jump from interaction to interaction without really watching the video.

QuizCram To address this common behavior of the watching students, an application was introduced by Kovacs (2015). The application named QuizCram tries to use this behavior of the students. For that, it divides the video or even a combination of multiple videos in smaller segments. Each segment is associated with a question summarizing the content of the segment. This segmentation allows the students to progress through the video from segment to segment by answering the questions. For instance, it is possible for the students to jump to the next segment as soon as they are able to answer the question of the segment correctly. This approach is called "question directed video viewing". It is assumed that there are two benefits for the students:

1. For those students who know the answer to the question of a segment, it is possible for them to skip the segment and to continue with the next one.
2. In contrast to that, the students who do not know the answer to the question are getting a summary of the content of the segment. This should help them to focus on the relevant parts.

Based on the watched segments, a timeline is computed for each student. This means that during watching a segment, the student also sees a list of previous segments. In this list the relevance to the current segment is expressed by using different colors. This highlighting of associated parts should encourage the students to review these segments.

A further feature of QuizCram is that it helps the students to identify the segments, which they should review for a better understanding. For that, it assigns a score to each mastered segment which indicates how well they performed at it. This score is a weighted sum of the following values:

- performance on the question of the segment
- watched part of the segment
- number of views of the segment

This feature helps the students to identify whole segments, which they should review. In contrast to that, QuizCram additionally marks the watched part of the video. With this marking, it is assumed that the students focus on the parts they haven't watched of the video segment when they are reviewing it. An evaluation of this application points out that the students are more likely to answer the questions and to review parts of the video (see Section 2.3).

A further web platform for interactive learning videos is introduced by Cummins, Beresford, and Rice (2016). This platform is named ILVP[14] and is mainly based on a javascript library which records different events during playback to a server. This includes actions like play, stop, pause, resume and seek. In addition, it reports the current position in the video to the server every 20 seconds. Because the students are required to authenticate at the platform, it is possible to state for each student when she/he watched which part of the video. Further control elements allow the students to jump from certain points in the video to other points (e.g. from slide to slide in the case of a screencast). Additionally, there is the possibility to jump in steps of +/- 5 seconds. ILVP

Teachers are able to place two types of questions at given positions in the video. This includes multiple-choice questions or text-based ones. The playback of the video stops at the position of a question and the students are required to provide an answer. In the case of a wrong answer, students are allowed to skip the question or to try it again. In addition, some feedback to the questions is provided after they are answered. As seen by all of the presented platforms for interactive videos, this platform allows to jump from question to question too. For that, it displays appropriate control elements listing the questions.

## 2.3 Interactive Video Studies

When talking about interactive videos, it often depends on the period to which it is referred. This means that in the first decade of the 21st century, the interactive features in videos were limited to functionalities to control the playback of the video. These features included actions like play, stop, pause, or control of the speed of the playback. In comparison to that, nowadays, interactivity in videos provides additionally in-video quizzes

[14] short for Interactive Lecture Video Platform

or other possibilities to interact with the content or the teacher. Because of that, this section presents at first some earlier studies evaluating interactive features to control the playback and after that, some evaluations of quiz-based methods of interactivity are shown.

Interactivity refers to playback control

A study comparing the performance of students using interactive videos with students watching traditional videos was done by Schwan and Riempp (2004). In this case, interactive features were limited to possibilities to control the video playback. To compare the two methods of watching videos, two learning environments were created:

1. *Non-interactive*: In this learning environment the videos were presented to the participants without the possibility to control the playback of the videos. This means that they were forced to watch the complete video in its normal speed. They only had the possibility to watch the videos as often as wanted.
2. *Interactive*: This learning environment additionally enabled the students to fully control the playback of the video. Based on that, they were allowed to interrupt the video, to play in slow motion and in timelapse, and to change the direction from forward to backward and vice versa.

The 36 participants of the study were recruited at the University of Offenburg. They were required to learn to tie four different nautical knots. For each knot, there was a video where the process of tying is shown. The length of the videos ranged from 14 seconds to 35 seconds. In both of the mentioned learning environments the participants were equipped with a rope and sat in front of a monitor showing the videos. It was allowed to alternate between watching the videos and trying to tie the knots. However, watching and tying at the same time was forbidden.

To evaluate the two learning environments, the overall time required to learn to tie the knots was measured. This included the time spent watching the videos and the time invested in the practical knot tying phase. At the interactive videos it was additionally recorded how often the interactive control features had been used.

The results of the evaluation of the times required to learn to tie the knots is shown by Table 2.3. On examining the overall time it can be seen that it

| Time | | | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|---|
| Overall | Non-interactive | M | 13.5 | 31.6 | 25.7 | 34.3 |
| | | SD | 5.7 | 13.9 | 11.9 | 13.6 |
| | Interactive | M | 7.6 | 19.0 | 13.0 | 20.5 |
| | | SD | 4.4 | 12.0 | 6.8 | 9.1 |
| Viewing | Non-Interactive | M | 6.6 | 10.5 | 10.1 | 12.3 |
| | | SD | 2.5 | 4.2 | 4.0 | 4.2 |
| | Interactive | M | 4.1 | 9.7 | 7.7 | 9.9 |
| | | SD | 1.5 | 4.7 | 3.3 | 3.9 |
| Practicing | Non-interactive | M | 6.9 | 21.1 | 15.6 | 22.0 |
| | | SD | 3.4 | 11.0 | 8.9 | 10.9 |
| | Interactive | M | 3.6 | 9.2 | 5.2 | 10.6 |
| | | SD | 3.3 | 8.2 | 4.1 | 5.6 |

Table 2.3: Learning times to tie a nautical knot with interactive and non-interactive videos (Schwan & Riempp, 2004).

took the participants significantly longer to learn to tie the knots at the non-interactive learning environment. Varying from knot to knot the differences are ranging from 66% to 95%. It is clear that this effect is also visible at the viewing time and the practicing time. However, it has to be pointed out that the participants spent considerably more time at practicing the knots at the non-interactive videos.

On evaluating the usage of the interactive playback controls, it was noted that the students used these features quite heavily. Most popular were the possibilities to control the speed and the direction of the videos. The study additionally pointed out that the usage of the interactive features increased by the difficulty of the knots to tie.

A further study of Zhang, Zhou, Briggs, and Nunamaker Jr (2006) focuses on evaluating the learning success and the satisfaction of the students in different learning environments. For that, four different environments were evaluated and compared:

1. with interactive videos
2. with non-interactive videos

3. without videos
4. standard classroom environment

These four environments were staffed with 138 undergraduate students, where the first two environments consisted of 35 students and the last two of 34. In the course of this study, the topic of internet search engines was taught to the students. After a brief introduction about the procedure of the course, the students were required to participate in a written pretest. This pretest did not show any significant differences between the four groups. The students of the first three groups additionally received a short introduction about the learning environment system. Now the lecture took place as a 50 minute unit. In the cases of the video groups, students were required to watch a video with a length of 29 minutes. The interactive video group had the possibility to control the video in multiple ways. In contrast to that, the normal video group was only able to rewatch the video. Group number three learned by using the slides and some other text-based material. The group which was taught in the standard classroom environment got the content explained in more or less the same manner as at the video and after that, they had the possibility to ask questions. Finally, the students of all groups went through a post-test and were required to fill in a questionnaire to assess their satisfaction.

The evaluation points out that, there are positive effects of using interactive videos in the course of e-learning. It is shown that the learning success is not significantly different between the group without videos and the group with non-interactive videos. This suggests that simply adding a video to an online course is not enough. However, the difference in the learning success is significant for the group with interactive videos in comparison to all other groups. In addition, the analysis of the questionnaire to measure the satisfaction of the students indicated that the approach is liked more than the other methods.

`Interactivity means quizzes`

In contrast to these two studies evaluating the usage of videos with very simple means of interactivity, a study of Kovacs (2015) examined the performance of a system called QuizCram (see above). This system provides advanced interactive components to videos like quizzes which are related to segments of a video. The study compared the system with normal videos with simple in-video quizzes. For that, the study prepared two sections

| Exam | QuizCram | In-Video | Significant |
|---|---|---|---|
| in-video questions | 85.4% | 81.3% | yes |
| unit exam | 65.1% | 63.4% | no |
| muliple-choice questions | 85.5% | 76.0% | yes |
| text-based questions | 67.6% | 49.0% | yes |

Table 2.4: Exam results at QuizCram and standard in-video systems (Kovacs, 2015).

of an online course for the QuizCram system and for a normal in-video question system. 18 participants were recruited and they were required to learn for 40 minutes with QuizCram and after that, 40 minutes with the normal system using in-video questions. On the following day, the students were required to perform some exams and a survey. The exams consisted of the built-in questions, the questions of the exam of the online course, some additional multiple-choice questions and extra text-based questions.

The results of the exams are shown by Table 2.4. It can be seen that the performance of the students is significantly better at the videos presented by QuizCram in three out of four exams.

The survey regarding the satisfaction of the students points out that on a scale of 1 to 7 the average for QuizCram is located at 5.28. In comparison to that, the average for in-video quizzes lies at 5.17. The students reported that they prefer QuizCram in 61% of the cases when they want to remember the content for the long term. It is clear that non of these values is significant.

On comparing the way of interacting of the students with the two systems, it was shown by the study that the built-in questions were answered more often if QuizCram was used. Additionally, students jumped back to already answered questions more often in the case of QuizCram. This observation might be an explanation for the elevated scores at the exams.

The behavior of students at in-video quizzes is influenced by the way of presenting them to the students. Because of that, the mentioned behavior is evaluated by Kovacs (2016). The evaluation is based on data gathered at a course for machine learning on Coursera. This course consisted of 113 lecture videos with a total length of 19.5 hours of video material. This material was supported with 109 in-video quizzes. From the 96,195 registered users, 42,437 of them answered at least one in-video quiz.

The study reported several observations about the behavior of the students. First, it was pointed out that the participation at the in-video quizzes was quite high. This means that 74% of the watchers tried to answer a quiz. A typical way of watching the videos was that students sought forward to the in-video quizzes. After seeing the question, most of the backward seeks happened. An explanation for this behavior might be that students sought backward to review the video in order to find an answer to the question. As a consequence, students were not skipping the questions in most of the cases. An extreme form of the "seek forward and jump backward" strategy was employed by some students. This means that they jumped only from question to question and tried to answer it. Independent from the way of watching, it was observed that the answers to the questions were correct at the first try with a rate of 76%. By comparing the length of the watched parts of the videos with in-video quizzes with normal videos it could be seen that the watched part was longer at the first group. This suggests that the in-video dropout was lower at interactive videos. The importance of this observation is underlined by the fact that videos with a length of more than six minutes were more often left earlier by the watchers (Kim et al., 2014).

A further study (Cummins et al., 2016) evaluated the engagement of the participants at a flipped-classroom course with the topic of programming with prolog[15]. The evaluation was done by analyzing the performance of the students of two different years. The course was attended by 81 students in 2012/13 and by 84 students in 2013/14. In summary the complete course consisted of 18 videos, where 16 were supported by in-video quizzes. There were 30 such questions distributed across the videos. To show the videos to the students and to enrich them with questions, an own platform, named ILVP, was developed (see above). The analysis of the participation of the students to the questions showed that in the first year, 71.5% of the watching students tried to answer the questions. In the following year this value increased to 86.4%. On average, each question was answered correct at 30 attempts and wrong at 14 attempts.

[15] a programming language for logic

In addition to this evaluation of the answers to the questions, the study tried to identify the motivation of the students for answering the questions. For that, students were required to fill out a questionnaire. Based on this questionnaire, four different types of motivation were identified.

1. *Completionism*: The single motivation for students of this group is the pure existence of the questions. Because of that, a video is considered incomplete if not all of the questions are answered.
2. *Challenge Seeking*: The questions are only answered by the students of this group if they consider the questions challenging. This means that they ignore questions which are trivial based on their opinion.
3. *Feedback*: Students of this group are answering the questions in order to receive feedback about their understanding of the content of the video.
4. *Revision*: In this group students view the questions multiple times to learn from viewing them. This is done typically short before a scheduled exam.

Based on the findings of this study, the application of in-video quizzes is recommended by Cummins et al. (2016). This is done because the increased interactivity is supportive for the formative assessment in a learning environment using videos.

## 2.4 Online Assessment

With interactive components embedded in videos, it is possible to use them for assessment. Online assessment means that the learning success of the students is measured with the help of a computer or more accurate the internet. This includes several different methods of assessment: (Gaytan & McEwen, 2007) Definition

- evaluation of the messages of the students posted in forums, chats or other similar online media
- online tests or quizzes
- weekly questions regarding the understanding
- self-tests

- and other similar means of testing

Studies In addition to these different means of online assessment, it has been shown by Ricketts and Wilks (2002) that the way of presenting the questions of the assessment is important. It is pointed out that the correctness rate is increasing if only a single question is presented per page. This means that the answers of the students are more often wrong if all of the questions are displayed on a single page and they are required to scroll down to reach all of the questions.

A common problem of most of the available methods of online assessment is the fact that it is difficult to identify the student who performs the assessment (Hernandez, Ortiz, Andaverde, & Burlak, 2008). In most of the cases only simple means of authentication are used. This means that a combination of a username and a password is the primary method. However, such a method does not really identify a person because it is easy to share the user credentials with somebody different.

It seems to be obvious that more strict methods of identifying a person are required. This is motivated by the fact that without such methods, the online assessment has to take place in a controlled environment where the process of identifying is done in an analog form (Sim, Holifield, & Brown, 2004). It is clear that this makes online assessment obsolete. The range of more strict methods includes, for instance, the possibility to use the webcam of the person for identification (Bailie & Jortberg, 2009). This means that face recognition software is used to validate the identity of the students participating in the online assessment. Furthermore, it is possible to use biometrical data like fingerprints (Hernandez et al., 2008).

## 2.5 Attendance Monitoring

As mentioned above, the requirement of identifying students in a valid form is also a challenge when it comes to monitoring the attendance. Because of that, this section points out how the attendance is monitored in standard classroom situations, so that it is possible to compare the accuracy of these methods with an approach for online videos. Before that, the assumed benefits (Rodgers, 2002) of employing compulsory attendance are presented by showing the results of some studies.

Studies

One of these studies was done by Devadoss and Foltz (1996). It is pointed out that the students, who attended all of the units of a course are more likely to receive a better grade. The mentioned grade is a full degree higher in comparison to students, who are only attending 50% of the units. A second study which points out very similar results was performed by Romer (1993). This study compares the grades of students with a full attendance, with other students who are only employing an attendance of one quarter. On average, the students of the first group scored the second best grade, where in contrast the members of the second group received the third best. These arguments for the employment of compulsory attendance are further encouraged by Bai and Chang (2016) by stating that students like occasionally performed checks of attendance because they are feeling more supported by the teacher. Furthermore, it is reported by Park and Kerr (1990) that a key factor in avoiding a negative grade is the attendance in most of the units of a course.

Signed Lists

As mentioned above, it seems to be obvious that there is the need to monitor the attendance of the students. In standard classroom situations there are different possibilities to achieve this goal. First, there is the solution to use lists which have to be signed by the attending students. This is one of the most popular methods but it has some drawbacks. One of these drawbacks is that students could leave the classroom after signing the list. A further possibility to cheat is that one student signs the list on behalf of many others. These drawbacks are especially exploitable in larger classes with a huge number of students. A countermeasure is the placement of some personnel at the doors of the lecture theater to oversee both, the signing process and the (early) leaving of the students. Furthermore, it is possible to check the ID-cards of the students before signing the list.

Audience Response Systems

In comparison to the approach using the signed lists, a completely different method is the usage of an audience response system (see Section 2.1) (Haintz et al., 2014). In this scenario for attendance monitoring, the students are required to authenticate at the ARS. During the lecture, questions are presented to the students which must be answered by using the ARS. Now, it is possible to use a list of answering students as a proof of attendance. It is clear that this only works accurate if such questions are presented quite often.

`GPS Tracking with Mobile Phone`

A third way of monitoring the attendance is presented by the Aoyama Gakuin University (apa, 2009). For this approach it is planned that each student is equipped with a smartphone given to them by the university. The smartphone should be used to track the movements of the students at the campus using the built-in GPS-module. In the case of a session in a class with compulsory attendance, it has to be checked if the smartphones of the correct students are in the correct classroom. One might notice that to falsify the attendance monitoring, it is simply enough to give the smartphones to a single student, who carries more of them. However, it is assumed that this will not be the case because a smartphone holds usually sensitive and personal data which should not be shared. Unfortunately, currently there is no research available to prove the usefulness of this method.

# 3 Methodology

When examining the research questions investigated by this work (see Section 1.2), it can be concluded that there are two main areas of goals. The first one consists of the actual development of a web platform and the second one covers different applications of the developed platform. This means that the second area is mostly evaluating the developed platform in its different possibilities of usage. To develop the web platform two methods of software development are used. On the one hand, a strategy known as Test Driven Development (see Section 3.1) is used. On the other hand, the means of Rapid Prototyping are applied to develop and to evaluate the platform with a real audience (see Section 3.2). For developing the web platform, established concepts of programming are used and presented by Section 3.3. Furthermore, the concept of a state machine (see Section 3.4) is used to implement some interaction related parts. As mentioned, the developed web platform is evaluated at its different usages. For that, the methods of qualitative and quantitative research are used (see Section 3.5).

## 3.1 Test Driven Development

Three parts of TDD

Generally speaking, TDD[1] is a process where the software to develop is broken down in small units and before any code is written, tests are defined for each unit (Beck, 2003). This indicates that the unit tests are defining the requirements of the software. When the tests are all finishing without any failing, it is clear that the requirements are fulfilled. Based on that, one might assume that TDD is simply a testing strategy. However, if the three

[1]short for Test Driven Development

different parts are examined more deeply, it can be seen that there is more (Janzen & Saiedian, 2005):

1. Test — The aspect of testing consists of two tasks, namely writing the tests and executing it. As mentioned, a test is written for each smallest possible part of the software which is called unit. For that, it is required to define what a unit is. There are different interpretations of this term. In the field of object oriented programming both, the class and the method were listed as the ideal candidates for a unit. In the most cases it is sufficient to declare a method as a unit. To write the tests, an interface or at least method-stubs have to be declared. This is necessary because the tests are meant to be executed and for that, they at least need the interface of the method which shall be tested. This execution of the tests could be done by the developers themselves, by separated testing personnel or by completely automated testing facilities.

2. Driven — The second part of TDD has its focus on the aspect driven. This means that it is incomplete to think of TDD as a testing concept which takes place after writing the code. The driven aspect centers on the phases analysis and design of the standard software development process which consists of four phases, namely analysis, design, implementation and testing. This suggests that with TDD, also the first two phases are covered. At the beginning, the defining of the requirements of the software take place which leads to the creation of the unit tests. For that, the second phase consists of the defining of an interface of the methods to be implemented (see above).

3. Development — Finally, the aspect of TDD named development states that TDD is not a complete software development process. It could be seen as a strategy to find the requirements of the software under development. In addition, there has to be a mechanism to ensure its correct implementation. This indicates that TDD has to be a part of a larger process. In the case of this work TDD is used with Rapid Prototyping (see below) to develop and evaluate the software.

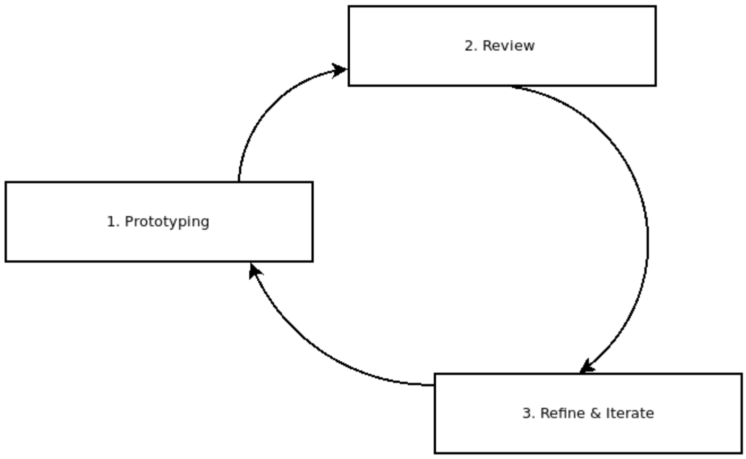



Figure 3.1: The three phases of Rapid Prototyping are forming a loop.

## 3.2 Rapid Prototyping

As mentioned above, the process used to develop the web platform is called Rapid Prototyping which relies on the principle that a prototype is brought to productive usage very fast and the gathered feedback is used to refine the prototype (Yan & Gu, 1996; Jacobs, 1992). This indicates that this approach consists of three phases which are forming a circle (see Figure 3.1): Three phases

1. prototyping
2. review
3. refine and iterate

The first phase is responsible for developing a working prototype. This has to happen according to a development strategy. In the case of this work, the means of TDD are used. This suggests that the prototype is developed according the requirements defined by unit tests. A prototype is ready for the next phase if all of the unit tests are passing successfully. At the second phase, the prototype is under review by test users or even in use in a productive environment. The result of this phase of review is some feedback of the users and other data usable for evaluation (e.g. log files). With these outputs of the review process the requirements are refined in the third phase. With these new or adapted requirements it is possible to update the unit tests to match the mentioned requirements. Now, the loop can start again by adapting the prototype.

## 3.3 Design Pattern

In order to develop the web platform, many parts are implemented following certain design patterns. Such patterns are general and reusable plans for given problems in the various phases of software design (Vlissides, Helm, Johnson, & Gamma, 1995). It is valid to think about design patterns as templates or best practice advices which could be used to implement common tasks in software engineering. This means that such patterns are not directly executable.

The following sections are explaining the used design patterns while developing the web platform.

### 3.3.1 Model View Controller

General MVC

When developing a web application, a common design pattern is known as MVC[2]. This architectural pattern is typically used for the development of user interfaces and web applications. The web platform developed in the course of this work uses the Django-Web-Framework[3]. This framework requires the usage of a variant of the MVC pattern. Because of that, this section explains the common principles of MVC and after that, the Django's way of implementing it, is shown.

As suggested by its name, the MVC pattern consists of three individual parts, namely the model, the view and the controller. This separation is done to encapsulate the internal storage and logic from the interactions with the user. In this case this means that the way of accepting input from the user and sending output to her/him is independent of the implementation of the methods used for saving the data and the associated programming logic.

[2]short for Model View Controller
[3] https://docs.djangoproject.com/, last accessed October 8, 2018

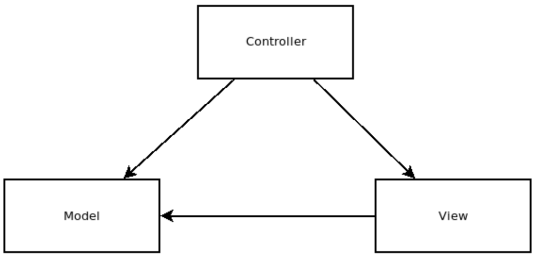



Figure 3.2: The three parts of the common MVC pattern.

The three parts of MVC as shown by Figure 3.2 are typically defined in the following way:

- *Model*: This is the most prominent part of MVC. It is responsible for saving the data and providing methods to access them. Furthermore, the core logic of the application is provided by the model.
- *View*: Any representation of the data for the output is handled by a view. This means that the data provided by the models is presented to the user.
- *Controller*: It acts as the delegator not only between the models and the views, it also connects them to the user by accepting her/his input.

`Django's MTV`

The general description of the MVC pattern leaves some leeway for the actual implementation. In the case of the Django-Web-Framework this means that it is more accurate to declare the variant of the MVC as a MTV[4] pattern. In detail, the three components are responsible for the following tasks (Holovaty & Kaplan-Moss, 2012; Django Team, 2018a):

- *Model*: This part is in the most cases the same as at the common MVC pattern. Django defines the models with an ORM[5]. This means that the models are defined as Phyton classes which are translated to tables in the database. Furthermore, there is an API to query the database.
- *Template*: The complete instructions which are required to define the way of displaying are provided by the templates.

[4]short for Model Template View
[5]short for Object Relational Mapper

- *View*: The connection between the models and the templates is the view. This means that it acquires the requested data from the models and uses the templates to actual display them.

The missing part in this definition is the controller. For Django projects this means that the responsibilities of the controller are mostly handled automatically. The controller is implemented as the definition of the URLs[6] which redirects from a requested URL to the corresponding view.

Flow of Events

For a better understanding of the Django's way of implementing the MVC pattern, the flow of events of a request is illustrated by Figure 3.3. The numbered steps are responsible for the following tasks:

1. At first there is the request of a client to the web server.
2. The server routes the request to Django where it is run through the URL configuration.
3. Based on that, the request is delivered to a view.
4. If required, a request to the database is initiated with the provided API.
5. The results of this request are returned to the view.
6. The obtained data is rendered by the template. For that different tags and filters are used.
7. The template returns the rendered data in a display ready way (e.g. HTML[7]).
8. This response is passed back to the web server.
9. Finally, the response is returned to the client.

### 3.3.2 Plugin

As mentioned earlier, the developed web platform provides different forms of interactivity. Each of these forms is implemented as an independent plugin. Because of that, this section explains the basic principles of an infrastructure with plugins.

Offered Interface

The design pattern called plugin, is a member of the group of behavioral

[6]short for Unique Resource Locator
[7]short for Hyper Text Markup Language

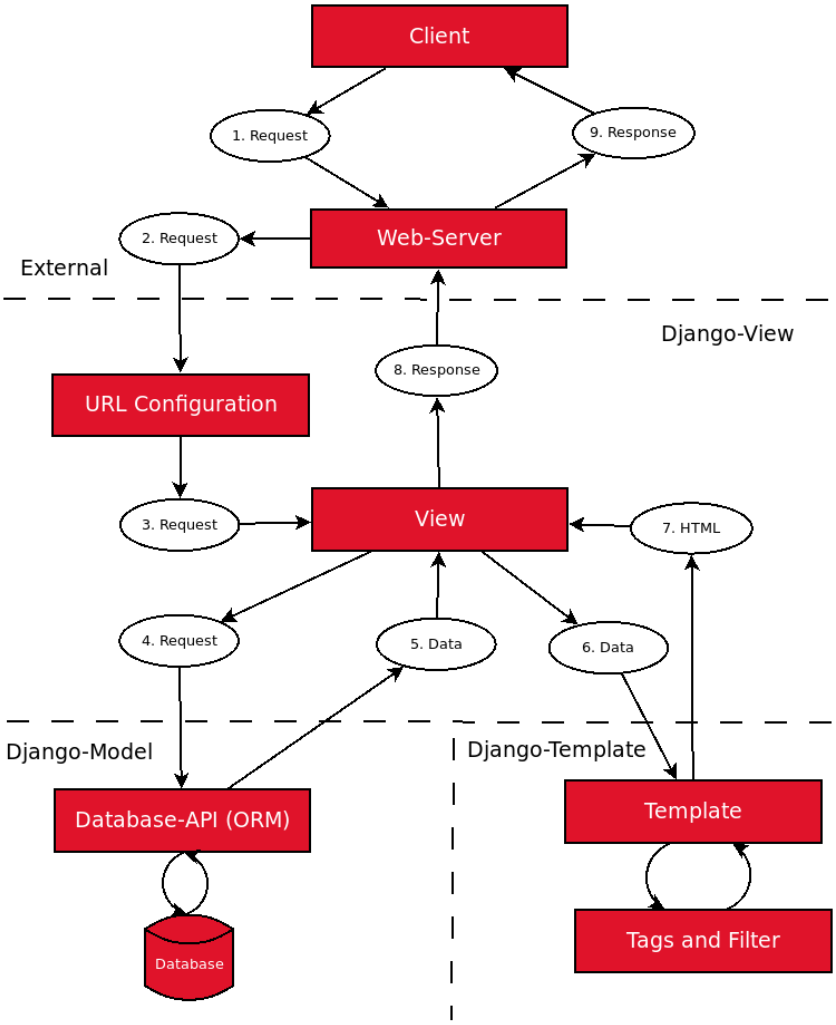



Figure 3.3: The handling of a request with Django illustrates the implementation of the MVC pattern as MTV (Probst, 2012).

patterns. It works on the assumption that there is an application which provides an interface for the addition of further functionalities. This interface clearly defines the possibilities of the plugins implementing it. Based on that, it is possible that the plugins are implemented by external personnel which means that they are able to improve the basic application with functionalities of their choice. From the point of view of the developers of the basic application providing the interface, it is required to define the rules for the plugins. This means that the interface only allows access to parts of the basic application which are meant to be extended.

`Extend Functionalities`

The major advantage of using the plugin approach is that it is possible to develop the basic application without thinking about further functionalities provided by the plugins as long as the interface is unchanged. This advantage could also be seen as a disadvantage because the need to keep the interface unchanged may be a restriction for further developments. Furthermore, there may be performance issues because the means to handle the plugins are probably leading to an overhead.

### 3.3.3 Command

The command pattern is a behavioral design pattern introduced by Vlissides et al. (1995). It encapsulates an action from the caller. This pattern is used if the following problems are required to address:

`Decouple Request from Invoker`

Direct coupling of a request to an invoker is problematic because it leads to the effect that it is not possible to specify a particular request at run time. For instance, this scenario happens when adding concrete actions to menus provided by libraries for implementing user interfaces. In this case a button or an item of a menu is provided by the library and the associated action is added like a parameter at run time. A further problem addressed by the command pattern is the need to implement chains of commands. Such chains are most useful when implementing functionalities like undo and redo.

A solution for these problems is the usage of the command pattern as illustrated by Figure 3.4. It can be seen that a base class (*Command*) defines an interface for the commands. This interface is implemented by the different

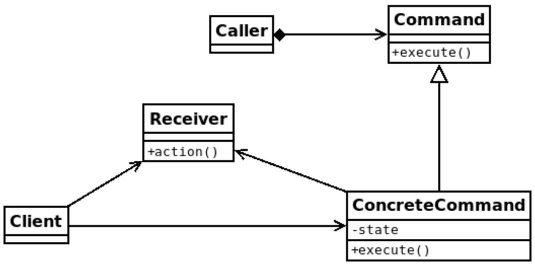



Figure 3.4: The general structure of the command pattern (Vlissides, Helm, Johnson, & Gamma, 1995).

concrete commands. Such an implementation of a command holds the state which is required to execute the command. This typically includes a reference to the receiver of the command. The client is responsible for creating the concrete commands and to equip them with all required information. In most of the cases this means that the concrete command is linked to the receiver. A further duty of the client is the linking of the concrete commands to one or more caller. In the example of the user interface library, the caller is a button or a menu item which accepts commands which are defined by the base class. It executes the concrete command by calling the method defined by the interface. On execution the concrete command performs its action by manipulating the receiver.

In this work, the command pattern is used to let the different methods of interactivity, which are implemented as plugins, provide their result of the calculation of the attention level to the web application. For more details see Section 5.4.

## 3.4 State Machine

As mentioned earlier, the concept of a state machine is used for the implementation of the interactive components. Because of that, this section explains the basic way of working of such a concept.

A mechanism known as a state machine is a mathematical model of computation. This means that it is no physical machine just an abstract concept

`Mathematical Model`

for describing an automata. To define such a state machine three parts are required:

1. a list of possible states
2. an initial state which is one of the possible states
3. a description of the state changes

The first item is a simple list of names for each state. A state is typically a description of the current status of the modeled system. The state machine could reach each of these listed states during its execution. Because the execution has to start in a particular state, item number 2 defines such a starting state. To switch from state to state, there are transitions. Such a transition typically happens on some specific condition like an external input. Based on that, item 3 is a list of such conditions which are stating the corresponding change of state.

When executing the state machine it starts in its initial state and waits for a condition which could happen in this state. If this condition occurs, the associated transition is performed which leads again to a state. This state may be the same or a different one. Now the state machine is waiting again for a specific condition and the related transition. The mentioned relations between conditions and transitions are defining the logic of the state machine and are part of the definition (see above).

## 3.5 Qualitative and Quantitative Research

When evaluating the performance of the web platform it is required to analyze a lot of data. For that, the following strategies are used:

- quantitative research
- qualitative research

Quantitative Research

The first strategy, namely quantitative research, has its roots in the quantification of data (Neuman, 2013). This means that with this method, it is possible to generalize results from a sample to a complete population. This means that statistical methods are used to measure different values provided by a given sample. With that, it is tried to state rules which are valid for the whole population.

In contrast to that, qualitative research focuses on gathering a deeper understanding of the founding motivations and reasons (Neuman, 2013). It can be seen that the tools of choice for this strategy are consisting of different methods of surveys or other methods of evaluating the opinion of test persons. Qualitative Research

These definitions of the two methods of research are suggesting that it is possible to use them in some kind of a loop (Mason, 2002; Sandelowski, 2000). This means that the results of quantitative research are often required to be evaluated in more detail. For that, the means of qualitative research could be applied. Based on that, it is possible that qualitative research generates new hypotheses. Such newly generated hypotheses are required to be analyzed. For that, the methods of quantitative research could be used. This suggests that the loop is started again.

# 4 Interactions for Live-Broadcastings

A very first prototype for interactivity at live-broadcastings of lectures was implemented and evaluated by the authors' master thesis (Wachtler, 2012). Because of that, this section presents the most important parts of this thesis. To show the differences to the current work, the goals of the master thesis are summarized to the following points:

- Creating a web platform which provides interactive components for other web content.
- The mentioned web content is typically a live-broadcasting of a lecture served by an independent streaming platform.
- The content with the interactive parts is available only at a given date which is defined by the teacher. Such a session is called an event. It has to be started and stopped manually by the teacher.
- Users should be able to join the events.
- The different methods of interactivity should be implemented as plugins.
- To analyze the performance of the attendees of an event, they should be listed and an attention level should be calculated for each of them to indicate how seriously they watched the live-broadcasting.
- Evaluating the web platform at different usage scenarios.

At first, this chapter gives an overview of the developed web platform and its components (see Section 4.1). After that, the main parts of the implementation are explained by Section 4.2. Finally, the important findings of the evaluation and the discussion are recapped by Section 4.3.

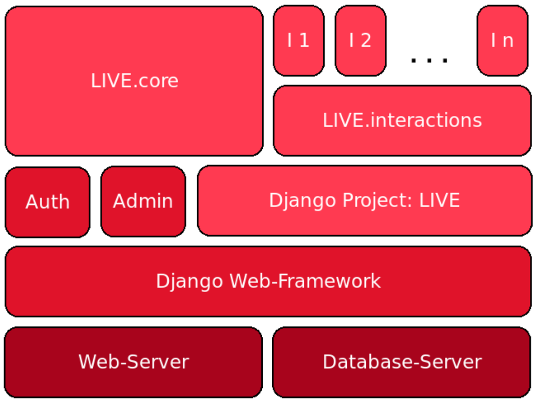



Figure 4.1: The architecture of the first prototype of LIVE.

## 4.1 Overview

Architecture

To address the defined goals of the master thesis, a web platform was developed to provide a mechanism for showing interactive components at given dates for any other kind of web content. This web platform was built on top of the Django-web-framework. With this framework, the basic architecture of the web platform was defined as shown by Figure 4.1.

It can be seen that the base of the framework is a database server and a web server. On top of these servers operates Django by providing the general mechanisms required for developing web applications. The actual web platform was implemented as a Django project named LIVE. Furthermore, the packages Admin and Auth of Django were used. The base system of the web platform was implemented in the package called *LIVE.core*. It was responsible for providing the basic functionalities like the joining of users to events. Because of the requirement that interactive components were implemented as plugins the package, *LIVE.interactions* provided an interface which had to be implemented by the interaction plugins. Furthermore, it was responsible for scheduling and showing the interactions to the joined users. The plugins providing the interactions were operating on top of this interface.

## 4.2 Implementation

As indicated above, the implementation of the web platform consisted of three major parts. The first one was the base system which is introduced by Section 4.2.1. After that, the interface for the interactions and the interactions themselves are presented (see Section 4.2.2). Finally, there was the analysis of the attendees which consisted of a mechanism to calculate a level of attention (see Section 4.2.3).

### 4.2.1 Base System

The base system of the web platform was implemented in the package *LIVE.core*. Its functionalities could be summarized as followed:

- Handling the registration and authentication of the users.
- Managing the events.
- Joining the users to the events.
- Creating the user interface for the students and for the teacher.

`User Management`

The web platform was usable for registered and authenticated users only. This was done because of the analysis features. These features required an association to a user because the teacher wanted to see the names of the watching students. As mentioned, the *Auth* package of Django was used to implement the handling of the tasks of the user management. To manage the privileges of the user accounts there were three different roles:

- Ordinary users were only allowed to join the offered events and to participate at the interactions of these events. Furthermore, they were able to take a look at the analysis of their own performance.
- Users with teacher privileges were additionally allowed to create events and decide the offered methods of interactivity.
- Finally, the admins were able to access the admin panel which was mainly used to distribute the user privileges.

`Event Management`

As mentioned, the creation of events was limited to the teachers. In contrast to that, students were only allowed to attend the events. In both cases the

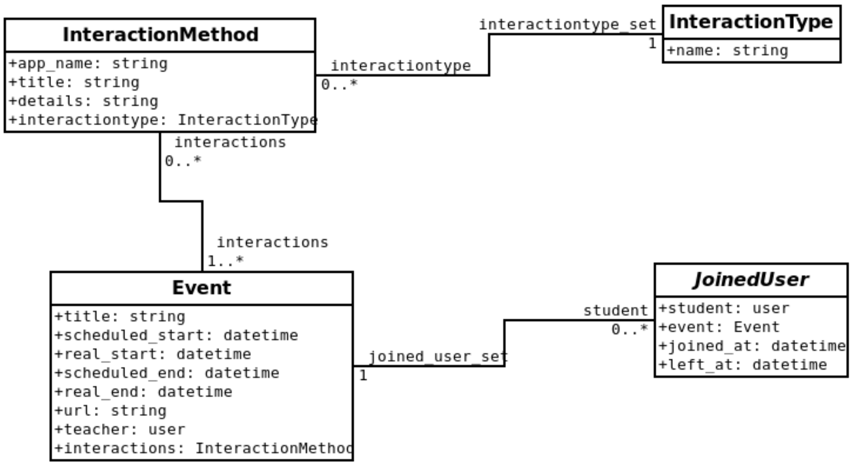



Figure 4.2: The models of the event management of the first prototype of LIVE.

event management of the web platform was involved. This management offered the following functionalities:

- Providing the models to save all the data required for the events.
- Offering all the dialogues and parts of the user interface to create, change, delete, start, join, stop and analyze the events.

To save all the data, some models were used by the event management as presented by Figure 4.2. At first, it can be seen that there were two models related to interactions. The first one was called *InteractionType* and represented the type of the interaction method. At this first prototype there were three types available. The first type represented the group of automatic and random interaction methods. This means that the interactions of such a method of this type were shown to the students in a completely automatic way. Furthermore, they were occurring at random positions during the event. In contrast to that, the types number 2 and 3 were defining groups of interaction methods where the interactions are triggered by the students or by the teacher. For that, the interaction method had to provide control elements in the user interface to invoke the interactions.

The different interaction methods were represented by the model *InteractionMethod*. For that, it saved the app name of the plugin providing the interaction method as well as a title and some details of it. Furthermore, the type was defined by a relation to the model mentioned above. The available interaction methods are presented in more detail by the Section 4.2.2.

The key model of the event management was named *Event*. It held all the information defining a single event. This included a title and a relation to a user identifying the teacher of the event. In addition, it saved the scheduled and real date and time of the start and the end of the event. This means that an event was scheduled to begin and to end at a given date and time but the real date and time values for the beginning and the ending was set when the teacher started and ended the event actually. The web content which is shown at the event was also saved by the model as a URL. The typical usage scenario was a link pointing to a streaming platform showing a live-broadcasting. Finally, the model saved the interaction methods which were used at the event.

To link a user to an event there was the model *JoinedUser*. It held a relation to the joining user and to the event to which she/he had joined. In addition, it saved the date and time of joining and leaving the event.

With the models forming the backend of the event management, it was possible to create all the dialogues and other parts of the user interface required to handle the tasks of the event management. These tasks could be arranged according to the following workflow:

1. *Creating the event*: The teacher was allowed to use a dialog to create an event. During the creation he had to offer a title and a link to the web content. Furthermore, she/he had to specify the scheduled start and end of the event as well as the interaction methods offered at the event.
2. *Joining of the students*: After the teacher had created the event, students were able to join to the event until the teacher stops the event. If the event was not started by the teacher, the joining students were seeing a message, indicating that they will be redirected to the event when it starts.
3. *Starting the event*: If the event actually took place, the teacher had to start it manually. This starting of the event led the teacher to a user interface (see below) which provided all the actions required during the event.

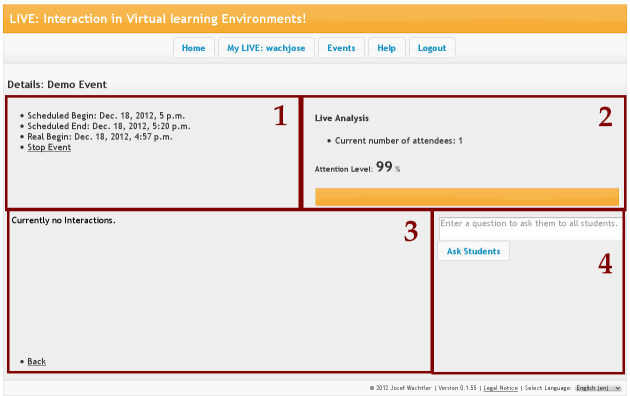



Figure 4.3: The view for the lecturer during a live-broadcasting of the first prototype of LIVE.

4. *During the event*: The students were offered a user interface which mainly showed the web content and a sidebar for handling the interactions (see below). Now was the time for the interactions to take place.
5. *Leaving the event*: During the event, students were able to leave the event on their own accord.
6. *Stopping the event*: If the event was finally over, the teacher had to stop it. This automatically forced all of the students to leave the event.
7. *Analyzing the event*: The teacher was able to view the analysis of the joined students. The students were also allowed to see the analysis of their own performance.

User Interface

As mentioned above, there were different parts of the user interface created by the event management. This included a dialog for creating the event. It was a simple dialog presenting text boxes to enter the relevant information for the event. Furthermore, a list of available methods of interaction was shown and the teacher had to select the methods which should be offered at the event.

After creating the event, the teacher saw a site summarizing the details of the event and had the opportunity to start it by clicking a button. The starting of the event redirects the teacher to a page as shown by Figure 4.3. It can be seen that this user interface consisted of four parts. At first, there

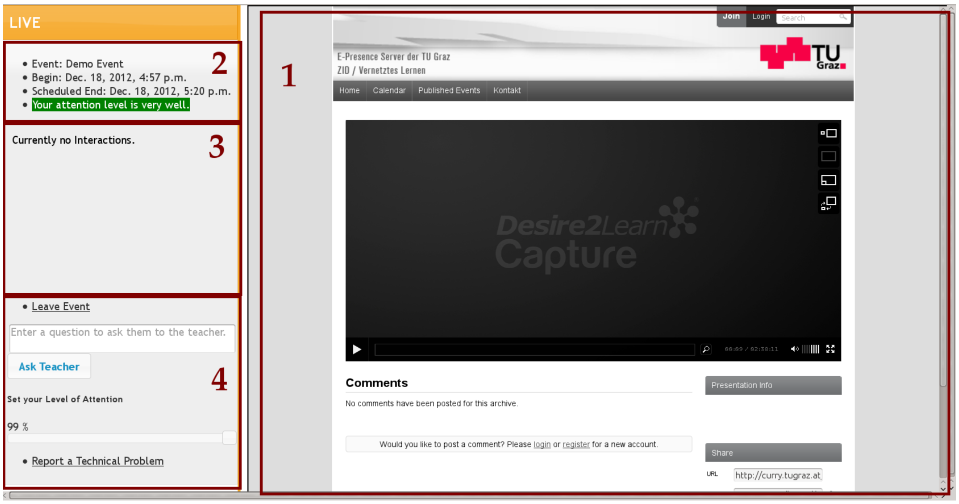



Figure 4.4: The view of a live-broadcasting for students of the first prototype of LIVE.

was an area (1) showing some metadata of the event and a link to stop it. Area number (2) presented a live analysis of the students. This included the number of joined students and their calculated level of attention. The biggest area (3) was reserved for interactions to occur. At the right hand side of this interactions area was the place (4) for control elements to invoke an interaction. The actual visible control elements were depending on the selected methods of interaction at the creation of the event. For more details of the interactions and control elements of the areas number (3) and (4) see Section 4.2.2.

In contrast to the user interface for the teacher during the event, the equivalent for the students looked very different. As shown by Figure 4.4 the main area (1) was reserved for the web content which was supported by the interactions of the event. The primary usage scenario was embedding a streaming platform. In this example, a live-broadcasting presented by a platform used at Graz University of Technology was embedded. At the left hand side of the web content, a sidebar showed some metadata and a statement about the attention level of the watching student at the top (2). Below that, area number (3) provided some space for occurring interactions. Finally, area number (4) included the possibility to leave the event and it held control elements to invoke interactions (see Section 4.2.2).

### 4.2.2 Interactions

As indicated above, there were three types of interaction methods. These types were defining the way of triggering the related interactions. Each of the following interaction methods was implemented as an independent plugin and was part of a single interaction type:

1. **Automatic**
   - Simple Questions
   - Solve CAPTCHA[1]
2. **Student-Triggered**
   - Ask Teacher
   - Set Attention
   - Report Problem
3. **Teacher-Triggered**
   - Ask Students

`Type 1` The two methods of interaction of type number 1 were very similar. The first one simply showed questions of general nature to the students. In contrast to that, the second one presented a CAPTCHA to the students. This happened completely automatic and in a random way. The process of scheduling was done by the interface which had to be implemented by the plugins. It was only required for the plugin to provide two things. First, there had to be a view which actually showed the question. Second, a value had to be provided, which stated how often the interactions should be shown per hour. With that, the scheduler of the interface showed the interactions of the methods of type 1.

`Type 2` Of the second type, there were three different methods of interaction. The first one provided the possibility to the students to ask a text-based question to the teacher. For that, the plugin had to provide a template with the necessary control elements. In this case it showed a field to enter the question and a button to send it. A click on this button invoked two interactions. On the one hand, the question was shown to the teacher with the possibility to

[1]short for Completely Automated Public Turing test to tell Computers and Humans Apart

answer it. On the other hand, the question was also distributed to all of the watching students, but the field to answer the question was missing. When the teacher eventually answered the question, the answer was shown to all students in combination with the question.

The next method of interaction which belonged to the second type provided a control element. This was a slider which enabled the students to specify their current level of attention. The slider could be moved from 0% (completely absent) to 100% meaning that the student thought she/he was fully attentive. Each change of position of the slider saved the new value to the database.

The final method of interaction of type 2 was a simple dialog offering the possibility to report a (technical) problem to the teacher. In most of the cases this feature was used by the students to report issues with the offered web content (e.g. live stream not working). If a student used the dialog to report a problem, the teacher received a pop up message stating the problem. This message was placed quite prominently at the top of the control panel of the teacher. A dialog was associated to the problem report in order to provide the possibility to the teacher to sent an answer to the reporting student. This answer was shown to the student as a typical interaction in the reserved area of the panel at the left hand side.

Of type 3, there was only one method of interaction. It implemented the opposite way of asking questions as the first method of interaction of type 2. This means that it provided some control elements to the teacher to ask a question to all of the students. For that, the teacher had to enter the question in a text box and after sending it to the students a list was shown to the teacher in the reserved area for the interactions. This list instantly showed the answers of the students. The students received the questions through interactions and a dialog asked them to send an answer to the teacher. After finishing the round of questioning, the teacher was able to send the correct answer to all of the students. Type 3

### 4.2.3 Analysis of Students Attention

As mentioned above, the web platform offered different features of analysis. This included a recording of the time of joining and leaving the event for each student. In addition, a level of attention was calculated for each student.

`Simple Recording` For the first part of the analysis, namely the recording of joining and leaving the values saved by the model *JoinedUser* were used. With these values it was possible to state how much of the event a student had attended. It is clear that this showed additionally how much from the beginning and the end of the event had been missed. The teacher was able to view this analysis for each student who had attended an event.

`Attention Level` The second part of the evaluation of the students' performance consisted of the calculation of a so-called level of attention. This level of attention was calculated under two conditions. The first one covered the analysis during the event runs. As shown above, the attention level was shown to the teacher as a value in percent indicating how attentive the joined students were at the current moment. The second condition took place at the evaluation after the event. In this case, a list of all joined students was shown to the teacher (see Figure 4.5). It can be seen that a level of attention was assigned to each student.

From a technical point of view, the level of attention was computed by each method of interaction. These results were grouped together to form the overall level of attention for each student. At the calculation of the level of attention during the event, the levels of attention of the students were combined to a single value. This definition suggests that the different methods of interaction were performing the calculation of the level of attention on their own. Strictly speaking, this was true however, all but one of the methods of interaction were using a standard function to compute the level of attention. This was done because in these cases the level of attention was based on the reaction times of the students to the interactions. In general, this means that if the students were reacting slower to an interaction, the level of attention was decreasing. The way of calculating the level of attention is shown in all details by Section 5.5. Only the method of interaction which provided a slider to set the current level of attention did not use the time-

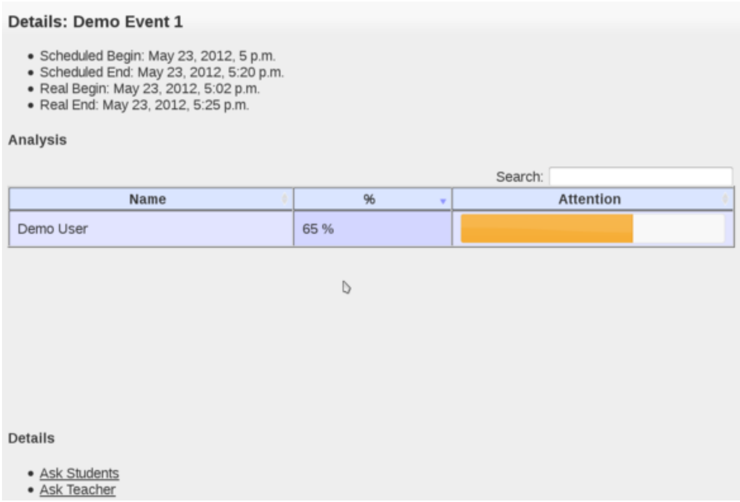



Figure 4.5: The analysis of the attendees of the first prototype of LIVE.

based calculation of the level of attention. It simply used the values set by the students.

Question Analysis

In addition to these overall possibilities of analysis, there were features of evaluation offered by the methods of interaction. Such features were provided by the methods of interaction which were responsible for asking questions to the students and vice versa. In both cases this consisted of a list of the questions and the answers. Additionally, it was stated who asked the question and who answered it at which time.

## 4.3 Evaluation and Discussion

The web platform was evaluated at three different scenarios with different goals. At first, a proof of concept test was performed. After that, it was used to support the live-broadcastings of two lectures with interactions.

Proof of Concept

The goal of the first test was defined to prove the correct way of working of the basic functionalities of the web platform and to find realistic parameters to state how often interactions should be displayed. For that, multiple test runs with testing personnel were performed.

Productive Usage

In comparison to this first test done in the lab, the web platform was used in productive environments at the two proceeding evaluations. In summary, 25 students used the platform to watch a live-broadcasting of six lectures. To evaluate the platform, the behavior of the students was observed and they were required to submit feedback.

Based on that, it is possible to state the following remarks about the usage of interactive components provided by the web platform.

- Benefits for the students under specific conditions
- Communication between the students and the teacher
- Analysis of the attention of the students
- Not suitable for on demand videos
- Technical limitations

The feedback of the students regarding the interactive components pointed out that in most of the cases they felt that the interactions had been helpful for a better understanding of the content and for staying attentive during longer broadcastings. However, it was stated that the number of interactions should not be too high or too low. This suggests that it is important to use the interactions with care. To state recommendations on the perfect number of interactions, a lot of further research is required (see Section 7). In addition to the feedback about the number of interactions, it was reported that interactions presenting content related questions had been more favored than general ones. Because of that, the usage of general interactions should be used only to overcome phases of no other interactivity. Furthermore, it was helpful for the students if the usage of the different types of interactions had been explained at the beginning of an event.

As indicated, the possibility for the teacher to ask questions to the students was liked by the students. Additionally to that, the feature to ask questions to the teacher was important for the satisfaction of the students. Because of this feature, students got the feeling that they were more integrated in the course.

From the teacher's point of view it has to be noted that additional workload was required because the students were more favoring interactions which had been related to the content of the broadcasting. This means that the teacher had to enter the question manually during the lecture which could slow it down. However, the benefit for the teacher was that she/he received an analysis of the attention of the students and an insight in their understanding by viewing the answers to the questions. It was also possible to use this analysis as a very basic form of attendance monitoring.

A major disadvantage of the web platform was that it was bound to given dates and times. This means that an event happened at a specific timestamp where a teacher had to control the flow of the interactions in real time. It can be seen that because of that, the usage of the web platform was limited to live-broadcastings only. It was observed that the usage of live-broadcastings of lectures decreased because the recordings were offered as on-demand videos after the lecture. Due to that, the possibility to use the web platform at videos would be from high importance.

From a technical point of view, it was observed that it was problematic that the web platform was completely independent from the presented content. This means that on the one hand, it was not possible for the web platform to detect if the students changed the URL of the content shown to them at the event. On the other hand it could happen that the presented content used a fullscreen feature. If the fullscreen was enabled it completely overlayed the interaction sidebar. This problem is illustrated by Figure 4.6. To point out the problem, the overlay effect has been rendered semi-transparent. It can be seen that the sidebar presenting the interactions is now in the background. Because of that the students were not able to make use of the interactive features. It was simply the case that they completely missed the occurring interactions. It is clear that this problem led to a decreasing level of attention because the students were not able respond to the interactions.

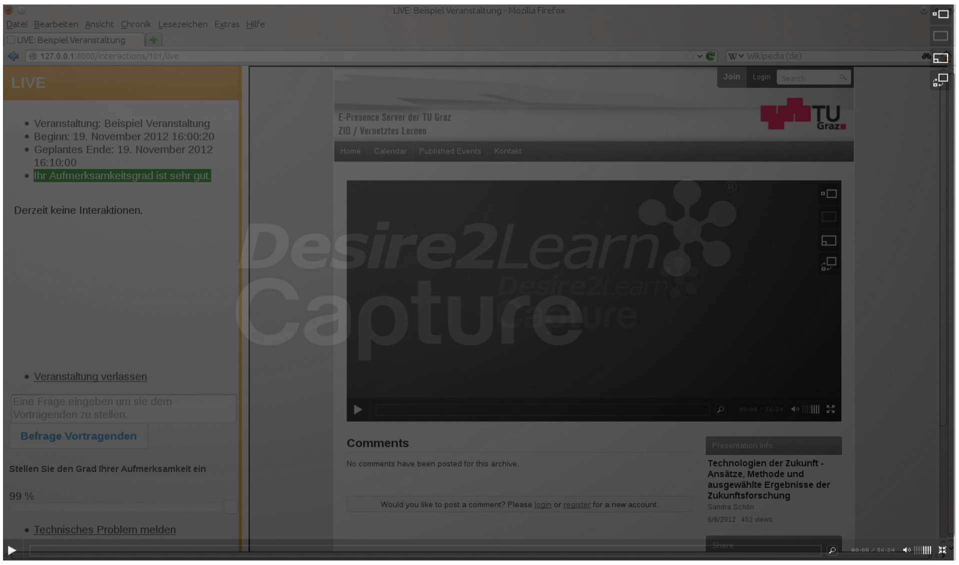



Figure 4.6: The fullscreen problem of the first prototype of LIVE.

# 5 Implementation

As indicated by the previous chapter, the very first prototype was limited in several ways. Because of that, the complete web platform for interactive components in learning videos has been re-implemented. Only the very basic concepts of the prototype were re-used. Due to that, the complete development process of the web platform is documented by this chapter. This includes a description of the functionalities of the web platform as well as a detailed explanation of the way of working of these features.

At first, an overview of the different components of the web platform is given (see Section 5.1). After that, Section 5.2 presents the used technologies for the implementation. Finally, the rest of the chapter presents the different parts in all their details.

## 5.1 Overview

The implemented web platform consists of several components. For a better understanding of the purposes of these components, this section states their main functionalities. In addition, it is explained how the components are working together.

Architecture

The schema shown by Figure 5.1 illustrates the parts of the web platform. It can be seen that a web-server and a database-server are forming the base. On top of these servers the Django-Web-Framework builds a basic abstraction for the implementation of the web platform. For that, additional third party libraries are used. The actual Django project, called LIVE, holds the complete implementation. It consists of a package providing all the core functionalities and a package offering the possibility to download all the data acquired by the web platform for research purposes. Furthermore,

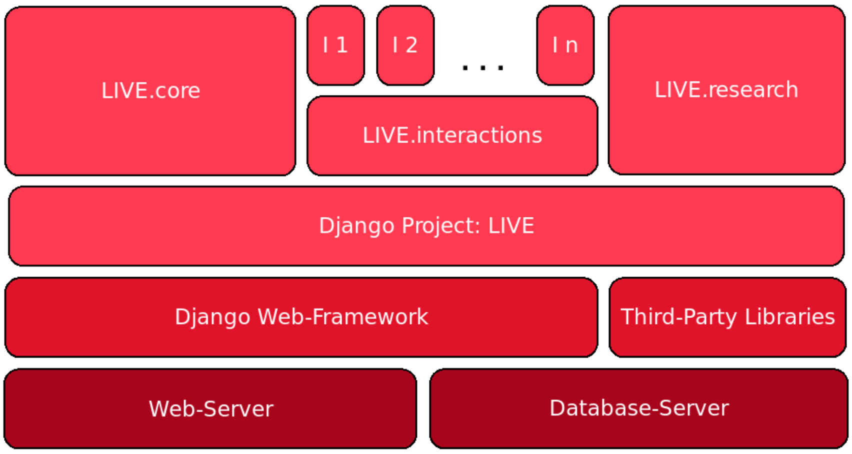



Figure 5.1: The architecture of the web platform.

there is a package providing an interface for the implementation of the different methods of interaction on top of it.

`Web-Server` It seems to be obvious that the web-server is responsible for providing the access to the web platform. It is required that such a server is compatible with Django. Because of that, one of the following servers has to be used:

- Apache HTTP Server[1]
- Gunicorn Web-Server[2]
- NGINX[3]

`Database-Server` In contrast to that, the database-server is used to save all of the data produced by the web platform. Again it is important that a supported server is used. This requirement limits the choices to the following ones:

- MySQL[4]
- PostgreSQL[5]
- ORACLE[6]

[1] http://httpd.apache.org/, last accessed October 30, 2018
[2] http://gunicorn.org/, last accessed October 30, 2018
[3] https://www.nginx.com/, last accessed October 30, 2018
[4] https://www.mysql.com/, last accessed October 30, 2018
[5] https://www.postgresql.org/, last accessed October 30, 2018
[6] https://www.oracle.com/database/index.html, last accessed October 30, 2018

- SQLite[7] (development only)

Django-Web-Framework

To provide all of the basic functionalities required to build a web platform, the Django-Web-Framework (see Section 5.2.1) is used. It forms a base for the implementation of the web platform. These functionalities include features like handling the connection to the database or security mechanisms to prevent common risks related to web applications. In addition, some further libraries are used to implement features which are not supported by the framework. For instance, a library is used to offer an LTI[8] provider as shown by Section 5.8.

LIVE

The very basic part of the Django Project LIVE is a package called *core*. It implements the basic functionalities of the web platform. This includes the user management and the event management (see Section 5.3). As mentioned above, the different methods of interaction are implemented as independent plugins (see Section 5.6). The interface for these plugins is provided by the package *interactions* and explained by Section 5.4. This package additionally implements the framework for the analysis for the attention of the students (see Section 5.5). In addition to these methods of interaction, which are available during the events, there is some interactivity provided after the event. This means that it is possible to create polls after each event (see Section 5.7).

As mentioned earlier, there is the possibility to download all the data generated by LIVE for research purposes. This feature is implemented in the package *research* and is explained by Section 5.9. Finally, there is a panel for the administrators of the web platform. This panel provides access to the configuration of the web platform and is used to manage the privileges of the users (see Section 5.10).

## 5.2 Technologies

To implement the web platform, different technologies are used. In the case of a web application, it is valid to separate between server-sided

[7] https://www.sqlite.org/, last accessed October 30, 2018
[8] short for Learning Tools Interoperability

technologies and their client-sided counterparts. Furthermore, videos and live-broadcastings are provided in different formats by different protocols. Because of that there are some libraries required to handle them. This sections explains all of the used technologies in detail.

### 5.2.1 Server: Django Web-Framework

As mentioned by the previous section the Django-Web-Framework is used on top of a web-server and a database-server. Django is a highly developed framework for web development. It uses the programming language python.

#### History

Django is a web framework of the third generation of interactive web applications. This definition is based on the categorization of web applications. These three dynamic categories are preceded by a category for static web pages: (Holovaty & Kaplan-Moss, 2012)

0. static web pages
1. CGI[9]
2. PHP[10]
3. dynamic web frameworks

`Generation 0` In the early days of web development there were only static web pages. This means that the complete web page was directly written in HTML[11]. This led to several problems. If something had to be changed, it was required to edit the file directly. Furthermore, a change in the user interface design of the web page required an adaption in all files, because the user interface design was hard coded in each of them.

`Generation 1` It is clear that these problems were not acceptable for a long time. Due to that, a technology called CGI, was invented. This invention is considered

[9]short for Common Gateway Interface
[10]short for PHP: Hypertext Preprocessor
[11]short for HyperText Markup Language

as the first generation of dynamic web applications because it enabled the usage of programs which are creating the web page in a dynamic way. With these programs it was possible to avoid the problems of the static web pages. However, it led to the creation of complex programs. Due to that, the maintenance of these programs became quite difficult.

`Generation 2`

The second generation of dynamic web applications was started by the invention of PHP. This programming language became the most popular tool for creating dynamic web applications very soon, because it is very easy to learn. A further reason for this popularity is that it is directly embedded in the HTML files. This simplicity is also the biggest problem of this language because nearly everything is allowed. Due to that, the programmer has to take care of all relevant issues like security and error handling.

`Generation 3`

To address these problems, web frameworks of the third generation have been developed. Such frameworks are providing a clean interface for developing web applications on a high level of abstraction. They provide an implementation of many functionalities which are typically part of a web application. This includes the handling of the database access. Furthermore, they offer mechanisms to avoid common security issues which are occurring when developing for the web. As mentioned, Django is a member of this generation.

The roots of Django were formed in 2003 when developers of a newspaper named "Lawrence Journal-World" started to develop web applications using python. More and more, they transferred functionalities from their applications to a library. Over time, these efforts resulted in a framework, called Django, which was released as open source in 2005. The name of the framework is often confused with the famous western hero however the framework is named Django to honor the guitar player Django Reinhardt[12].

### Principles and Features

`Principles`

The development process of Django and of projects implemented with the

[12] http://www.redhotjazz.com/django.html, last accessed October 16, 2012

framework, are required to follow some principles. The following list states the most important ones: (Django Team, 2018b)

- **Lose Coupling**: A module or a package which is part of the framework or part of the developed application should not know more about other modules or packages than absolutely required.
- **Less Code**: Writing source code should be minimized. This means that a reduced amount of source code helps to increase the readability.
- **Rapid Development**: The requirements of a web application or of a framework used to develop a web application, are typically changing fast. Because of that, the source code should be open for changes.
- **DRY**: The principle called "Don't repeat yourself" states that everything should be written only once. This encourages the development of reusable components.
- **Explicit before Implicit**: Explicit definitions should be used whenever possible. This suggests that implicit definitions should be used only if they are providing a huge advantage.
- **Consistency**: All of the parts of the framework and the applications developed with the framework should be consistent on all levels of abstraction.

Features

Django offers many functionalities which are typically used when developing web applications. In summary, the following list presents the most prominent features: (Django Team, 2018a; Holovaty & Kaplan-Moss, 2012)

- It follows the MVC design pattern in an adapted way (see Section 3.3.1).
- The complex database API provides access to the data in a way like working with normal classes and objects. This indicates that an ORM is used. Such a system maps classes to tables in a relational database.
- The URLs are explicitly configured using regular expressions.
- A template system is used to render the user interface.
- Internationalization and localization is supported in all components of the framework.
- There exists a library to manage the user accounts and their privileges. Furthermore, the process of authentication is handled by this library.
- The protection from common security risks is provided by the framework. This includes the following ones:

  - SQL-Injections
  - Cross-Site-Scripting

- A simple and lightweight web-server for development is provided.
- Forms are handled by using classes. This simplifies the process of validation, because it is enough to define the constrains and the required checks are performed automatically.

### Third Party Libraries

As mentioned above, Django has a lot of features already built in. However, it is necessary to use some further libraries to implement all of the required features of the web platform. In summary the following third party libraries are in use:

- Modeltranslation[13]
- Simple CAPTCHA[14]
- Timezone Field[15]
- Widget Tweaks[16]
- Bootstrap Admin[17]
- LTI Provider[18]
- Debug Toolbar[19]

The first additional library is used to save translated content in models. This means that a value could be saved in all of the languages provided by the web platform. On accessing such a translated value, it is returned in the currently active language. To increase the security of the registration form and to implement the corresponding method of interaction (see Section 5.6.2), the second library provides functionalities to create and validate a

[13] https://django-modeltranslation.readthedocs.io/en/latest/, last accessed October 30, 2018
[14] https://django-simple-captcha.readthedocs.io/en/latest/, last accessed October 30, 2018
[15] https://github.com/mfogel/django-timezone-field, last accessed October 30, 2018
[16] https://github.com/jazzband/django-widget-tweaks, last accessed October 30, 2018
[17] https://github.com/douglasmiranda/django-admin-bootstrap, last accessed October 30, 2018
[18] https://github.com/wachjose88/django-lti-provider-auth, last accessed October 30, 2018
[19] https://django-debug-toolbar.readthedocs.io/en/latest/, last accessed October 30, 2018

CAPTCHA. Because the web platform operates heavily on date and time values which could be produced by users from different timezones, it is required to save the timezone of a user. For that, the third library offers corresponding fields for the models and forms. Django displays the fields of a form using widgets. Unfortunately, the possibilities to manipulate these widgets are limited. To match the requirements of the user interface for the widgets, the fourth library is used. The built-in admin panel of Django provides templates which are not responsive and because of that, the panel is not usable on mobile devices. To address this issue, the fifth library overwrites these templates with a fully responsive design using Bootstrap[20] (see Section 5.10). As mentioned earlier, the web platform provides an LTI provider. This provider is implemented as a re-usable library (see Section 5.8). Finally, a library is added to show a toolbar when running the development server. This toolbar displays a lot of different information ranging from a visualization of the performed database queries to statistical data regarding the loading time.

### 5.2.2 Client: Html5, jQuery and Bootstrap

Overview

When implementing a web platform, it is a common practice to develop the user interface by using different technologies which are running at the client. In the case of LIVE this includes the following technologies:

- HTML5[21] and CSS[22]
- JavaScript and jQuery[23]
- Bootstrap

#### HTML5 and CSS

HTML5

To develop a user interface of a web platform, a markup language, which is

[20] https://getbootstrap.com/, last accessed October 30, 2018
[21] https://www.w3.org/TR/html5/, last accessed October 30, 2018
[22] https://www.w3.org/Style/CSS/, last accessed October 30, 2018
[23] https://jquery.com/, last accessed October 30, 2018

supported by the different web browsers, has to be used. The typical choice in this case is HTML in its current major version, namely version 5.

In 2004 (Faulkner, Eichholz, Leithead, Danilo, & Moon, 2017), the work on the new version 5 has begun because the last update of the previous version dated back to 2000. This work was led by WHATWG[24] in order to develop a new standard. This was done because the W3C[25] placed its focus on the development of a separate branch of HTML, which focused on improving the currently used version, named XHTML in version 1.1. This new version 2.0 never reached the status of an official recommendation because the focus completely shifted to HTML5. After the first draft which was published in 2007, HTML5 became the official recommendation in 2014.

HTML5 uses most of the common parts of the previous versions of HTML which are required to develop a user interface of a web platform. In addition, it defines some new features or integrates other existing technologies in the specification of the markup language. This includes the following ones:

- DOM[26]
- new semantic elements
- new multimedia tags
- different APIS

The specification of HTML5 now integrates the definition of DOM. This API is a technology which represents the structure of the whole document written with HTML5 as a logical tree. Each node of the tree represents a individual part of the document. In its pure form DOM is completely independent from the platform and the language. To improve the structure of the created documents some new elements have been introduced to define the semantics of the associated part. This includes elements to label the navigation, the header and the footer as well as the main content. Before introducing these elements generic elements had to be used. Furthermore, the increasing need for multimedia content lead to the addition of corresponding elements to the specification of HTML5. Such multimedia elements are providing the possibility to directly embed videos or audio files which could be played

[24] short for Web Hypertext Application Technology Working Group
[25] short for World Wide Web Consortium
[26] short for Document Object Model

when viewing the document without the need of an additional piece of software (e.g. Adobe Flash[27]). As mentioned above, the integration of DOM is part of an effort to include different APIs which are typically useful when developing web applications. For instance, such APIs are providing functionalities like drawing in so-called canvas elements or they offer the support for an advanced storage system for saving values identified by a key.

CSS Because HTML5 defines the structure and the content of a document only, there is the need for a possibility to style this document in a proper way. For that, the stylesheet language named CSS has been invented. It holds declarations of how a part of the document should look.

The development of CSS dates back to 1993 where different proposals for such stylesheet languages were published. Two of the authors of these proposals were combining their efforts. Together they invented the very first version of CSS. In the years 1994 and 1995 they presented their language at different conferences which brought them the attention of the W3C. This connection resulted in an official recommendation in the year 1996. In 1998 the second version of CSS became the new recommendation. The third version is still under development but many of its features are already supported by the major web browsers because of the requirements of the modern web applications.

### JavaScript and jQuery

The possibilities of HTML and CSS are limited to structuring the content of a document and to styling it according some declarations. To add dynamic possibilities, a complete programming language is required. For that, it is common to use JavaScript in the field of web applications.

JavaScript The very first version of JavaScript was published with the release of version 2.0 of the Netscape Navigator in the year 1995. At this time the scripting language was called LiveScript. Later this year Netscape cooperated with Sun Microsystems (the inventor of the programming language Java). This cooperation led to the re-naming of the scripting language to JavaScript.

[27] https://www.adobe.com/products/flashplayer.html, last accessed October 30, 2018

In the following years JavaScript was improved and it received a lot of functionalities. In the year 1997 the ECMA[28] released their own standard together with Netscape to formally specify the scripting language. This effort finally led to the ISO standard ISO/IEC 16262:1998. From this point the language itself is developed in a continuous way. However, the major web browsers enriched the language with different extensions. This led to incompatibilities between the browsers. This problem was resolved over time with the growing number of built-in functionalities.

The field of usage for JavaScript ranges from simple tasks to complex applications. This means that JavaScript is typically used to validate data from the user before sending it to the server or to show dialogues. In addition to these simple tasks, JavaScript allows the manipulation of the DOM and to send and receive data without reloading the page. With these functionalities it is possible to implement complex web applications with JavaScript.

`jQuery`

To simplify the usage of JavaScript, a library has been developed. It is called jQuery and is the most used library in 2018. The functionalities of jQuery (jQuery Team, 2018) are providing simple and consistent functions of the features of JavaScript which are manipulating the DOM. Furthermore, it simplifies the implementation of methods known as AJAX[29]. This technology enables the developer to send data between the browser and the server while a document is displayed. This means that for the sending process, it is not required to reload the complete document.

jQuery was presented in 2008 in its first version. Soon the library became very popular and the jQuery Foundation was founded to drive the development forward. A main reason for its popularity is based on the mentioned incompatibilities of the major web browsers regarding their implementation of JavaScript. jQuery addresses these problems and provides a consistent interface which is independent from the differences of the browsers.

[28] short for European Computer Manufactures Association
[29] short for Asynchronous JavaScript and XML

### Bootstrap

The user interface of the web platform is developed using Bootstrap. This is a library to simplify the implementation of complex designs for a user interface (Bootstrap Team, 2018). The roots of Bootstrap are in an internal project of Twitter. The goal was to develop a simple library to implement user interfaces for the different tools used by Twitter. The very first versions were used in 2010. With the beginning of the year 2011 Bootstrap was used quite heavily by the developers associated with Twitter. This led finally to the release of the library as an open source project. Since then, it is under constant development to keep track with the new possibilities offered by the new versions of HTML5 and CSS3.

The advantage of using Bootstrap is that it offers templates for different elements of the user interface. This includes the following examples:

- forms
- buttons
- tables
- typographic elements
- grid systems
- navigation bars

In addition to these elements, there are also optional JavaScript extensions. They are mainly adding functionalities to these elements or combining them to more complex elements. An example for such an extension is an element which provides a selector for the current date and time. The selected value is automatically inserted into a form field or could be used directly from JavaScript code. A further extension is the handling of dialog boxes. This means that it is possible to show and to hide dialogues which are part of the user interface by using JavaScript.

Because of the growing usage of mobile devices for browsing the web, Bootstrap is fully responsive beginning with its version 2. This means that all of the elements implemented with Bootstrap are automatically adapting themselves according to the size of the screen on which the document is displayed.

### 5.2.3 Playing Videos: Protocols, Formats and Libraries

The developed web platform supports videos and live-broadcastings with different methods of interaction. This indicates that the platform has to be capable of playing video content in different protocols and formats. Currently, the following methods are supported:

- embedded YouTube player[30]
- RTMP[31][32]
- HLS[33][34]
- conventional playback over HTTP using HTML5 video features

YouTube Player

The first possibility to play a video with LIVE is the usage of a YouTube video. For that, the YouTube player is integrated by using the offered library. The API of this library offers most of the required functionalities. However, to fully monitor the progress of the video it is necessary to implement some additional events in order to show the required interactions.

Playing over RTMP

When the development of LIVE began it was required to support RTMP. This protocol supports the streaming of multimedia data between a server and the flashplayer. It was initially invented by Macromedia as a proprietary protocol. After Adobe took control of Macromedia the protocol was released for public usage. The protocol establishes a persistent connection over TCP[35]. This connection is used to transport streams in a smooth way. For that, the stream is split into fragments which size is defined in a dynamic way during the streaming. It supports the playback of all of the formats which are supported by the flashplayer. This includes FLV and F4V[36] as well as MP4[37].

Playing over HLS

With the growing support of HTML5 based video playing, new methods were developed. The protocol HLS was first released in 2009 by Apple. It

[30] https://developers.google.com/youtube/v3/, last accessed October 30, 2018
[31] short for Real Time Messaging Protocol
[32] https://www.adobe.com/devnet/rtmp.html, last accessed October 30, 2018
[33] short for HTTP Live Streaming
[34] https://developer.apple.com/streaming/, last accessed October 30, 2018
[35] short for Transmission Control Protocol
[36] https://www.w3.org/TR/html5/, last accessed October 30, 2018
[37] https://mpeg.chiariglione.org/standards/mpeg-4, last accessed October 30, 2018

provides the streaming of multimedia content by using a standard web server. Today this protocol is supported by all of the major web browsers. In contrast to RTMP, the video is still divided in small fragments but they are distributed using a conventional web server. To achieve this behavior the protocol consists of three parts:

1. server
2. distributor
3. client

The server is responsible for creating the content in the required formats. For that, it has to encode the video and the audio in one of the supported formats. The video stream is encoded using the H.264[38] algorithm and for the audio part it is possible to use MP3[39][40], AAC[41][42], AC-3[43] or EC-3. The encoded video is packaged into a MPEG-2 Transport Stream[44]. This transport stream is split into fragments of equal length by the segmenter. It additionally creates a m3u8 file which saves the references to the individual fragments. Now the distributor serves the fragments and the reference file. For that, a conventional web server is used. When the client plays a video it starts by reading the reference file and based on that, the individual fragments of the video are downloaded by a standard connection over HTTP. The client dynamically connects the fragments together and plays them as a single video.

`Playing over Standard HTML5`

To support a broader range of video serving technologies it is possible to simply serve a video in its complete form from a HTTP or FTP[45] server. These files are played by the video element of HTML5. Currently the following formats are supported:

[38] https://www.itu.int/rec/T-REC-H.264, last accessed October 30, 2018
[39] short for Motion Picture expert group audio layer 3
[40] Standards: ISO/IEC 11172-3, ISO/IEC 13818-3
[41] short for Advanced Audio Coding
[42] Standards: ISO/IEC 13818-7, ISO/IEC 14496-3
[43] https://www.atsc.org/standard/a522012-digital-audio-compression-ac-3-e-ac-3-standard-12172012/, last accessed October 30, 2018
[44] Standards: ISO/IEC 13818-1, ITU-T H.222.0
[45] short for File Transfer Protocol

- OGG[46] using the codec Theora[47]
- WEBM[48]
- MP4 using the codec H.264

Player requirements

As mentioned above, the early versions of LIVE required the flashplayer to play videos. Later versions switched to the usage of HTML5 videos. However, the flashplayer is still supported because of backward compatibility. To display interactions during the video, some functionalities are required from the player software. In summary these are the following ones:

- playback of flash and HTML5 videos or live-broadcastings
- catchable events on user actions like play or pause
- API to access the current status of the video or live-broadcasting
  - current position
  - playing or paused

To achieve these goals a software library called flowplayer[49] is used. It supports all of these requirements. However, the usage of RTMP forced the usage of two different versions of the library. The first one supports HTML5 only and the second branch[50] is responsible for providing the support of the historical flash videos.

## 5.3 Core-Components

Overview

As mentioned above, the basic functionalities of LIVE are implemented in a package called *core*. In summary this package provides the following components.

- The user interface, which implements a design for the platform with the technologies mentioned earlier.

46 https://www.xiph.org/ogg/, last accessed October 30, 2018
47 http://www.theora.org/, last accessed October 30, 2018
48 https://www.webmproject.org/, last accessed October 30, 2018
49 https://flowplayer.com/, last accessed October 30, 2018
50 http://flash.flowplayer.org/, last accessed October 30, 2018

- Because LIVE is available for registered and authenticated users only, a user management is provided.
- All videos or live-broadcastings are handled as events. Due to that, the package *core* manages the events in all phases of their life cycle.
- This includes the joining of the students to the events. After joining they are able to watch the video.
- Finally, the basic package provides a mechanism to save and access settings.

### 5.3.1 User Interface

It is common knowledge that the design of a product or a platform is from high importance. This means that the success depends to a huge degree on the design. Furthermore, the satisfaction of the users is heavily influenced by the user interface design of the product or service. Because of that, "thinking about design is hard, but not thinking about it can be disastrous" (Caplan, 2018). It seems to be obvious that the development of functional and reliable applications is not sufficient. It is important to develop a UI[51] with the following goals: (Wachtler, Geier, & Ebner, 2015)

- user satisfaction
- user experience

UI of first Prototype

The very first prototype of LIVE, as introduced by Section 4, used a very basic user interface for development purposes. Evaluations pointed out that this user interface was not very friendly in terms of usability and design. Because of that, a new one is developed. For this new user interface the web framework Bootstrap is used as explained by Section 5.2.2.

Color and Typography

The very basic components of a user interface used for web platforms are the colors and the typography. The logo in the header and the main elements of the platform are in the same color. For that, a variant of red is used. In terms of typography the font is set to *Helvetica Neue*. To support clients which are not able to display custom fonts, there is a fallback to the standard sans-serif font of the client.

[51] short for User Interface

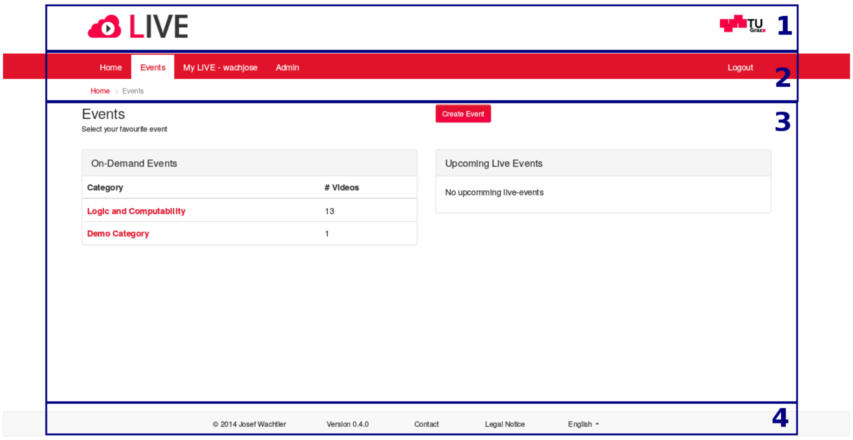



Figure 5.2: The desktop version of the user interface of LIVE

In Figure 5.2 the basic layout of the desktop version is shown. It can be seen that it consists of four parts: `Desktop UI`

1. header
2. main navigation
3. content area
4. footer

The header contains the logo of the platform in combination with its name on the left hand side. In addition, the logo of Graz University of Technology is included at the right hand side of the page. Below that, the main navigation is located. It is displayed as a continuous red bar which states the main areas of the platform. The current active area is highlighted with a white background. At the bottom of the site, there is the footer which contains some links to additional information about the platform. Furthermore, it offers the possibility to change the language which is used to display the platform. These three elements (1, 2 and 4) are fixed for each side because of continuity reasons. This enables the user to reach the basic areas of the platform in the same way on every page.

Between the navigation bar and the footer, the main content is shown. The basic design concept for the different elements of the content is the application of boxes with rounded corners. All parts of the content are presented by such boxes. These boxes are serving two purposes. First, they should group the presented information in a logical form. Second, the boxes should deliver a feeling of togetherness to the users.

Mobile UI Today, it is from high importance to consider the needs of the users with a mobile device. Because of that, the requirements of the mobile users have to be identified and taken into account along the whole development process (Ebner, Stickel, & Kolbitsch, 2010). It seems to be obvious, that on the screens of mobile devices dramatically less space is available in contrast to desktop screens. To address this issue, it is required that the content is displayed using one column only. The grid system provided by the used framework Bootstrap helps to implement such a behavior. This means that it automatically places the content according to the available screen space. The only part of the user interface which needs special treatment is the navigation bar. As shown by Figure 5.3 the elements of the navigation bar are not displayed in its basic form (left screenshot). To access the menu items of the navigation, it is required to use the so-called "Hamburger Icon" (1). On touching this icon, the navigational elements are shown as a list (right screenshot). The figure additionally illustrates the mentioned single column layout used for the main content at mobile screens. This means that the two boxes which are placed side by side at the desktop version are now displayed on top of each other.

Interaction Boxes In contrast to the general layout of the web platform, there is a special part with increased needs for an advanced user interface design. As mentioned, LIVE implements different methods of interaction for videos and live-broadcastings. These methods are implemented as independent plugins and the teacher is able to select the methods she/he wants to offer while creating the event. The offered methods of interaction are shown at the backend to the teacher. In addition, the shown methods provide links to further actions regarding the interaction method in question. To address these requirements, the offered interaction methods are shown by boxes as illustrated by Figure 5.4. It can be seen that for each method of interaction there is a box. The colors of the boxes are indicating the state of the corresponding method. This means that the color green suggests that the interaction method is ready to use. In contrast to that, the color orange states that some additional settings are required to use the interaction method. In this case an icon showing a pencil is printed. By clicking this icon the

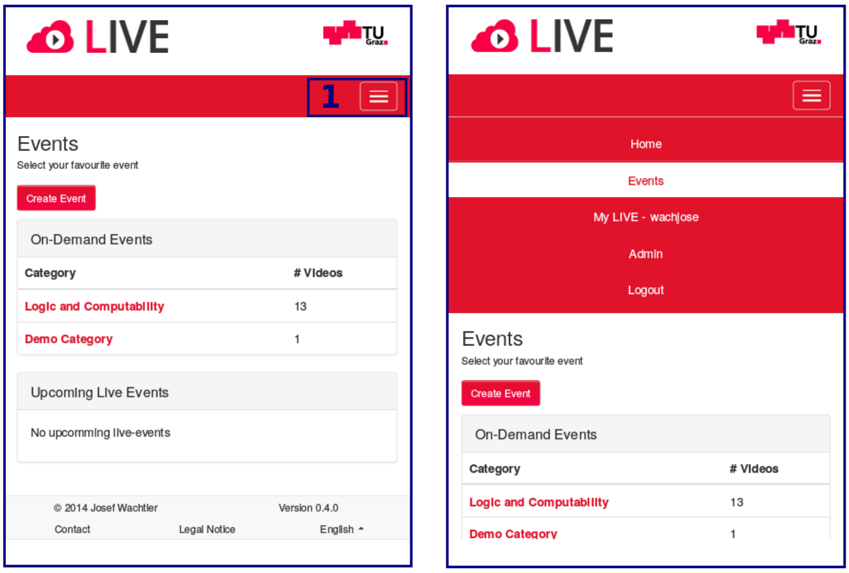



Figure 5.3: The mobile version of the user interface of LIVE

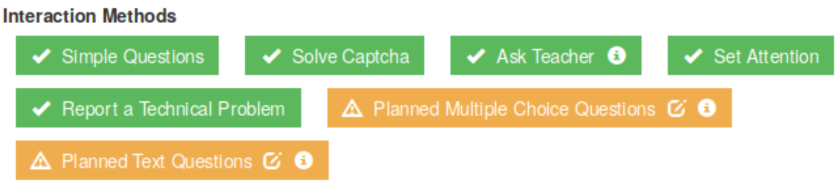



Figure 5.4: The boxes are showing the offered interaction methods.

teacher is able to perform the required settings. After finishing them, the color switches to green and the settings icon is no longer visible. Some of the interaction methods are offering additional information. In most of the cases, these information consists of an evaluation of the performance of the students. For that, an icon printing an information sign is shown by some boxes. Clicking such an icon displays a so-called tooltip which shows the actual links to the detailed information.

Further graphical components of the user interface are explained in the related sections. This includes mainly presentations of the evaluation of the performance of the students at the events.

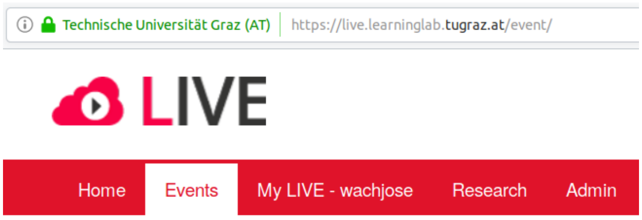



Figure 5.5: The URLs are related to the main navigation.

URL Design

It is obvious that the presented graphical parts are considered as the user interface of a web platform. However, the URLs are also part of the UI. This means that it is required to design the URLs too. For that, the following aspects should be taken into account: (Andrews, 2012)

- An URL could be seen as a structure of directories. This structure should reassemble the navigation hierarchy presented by the graphical user interface.
- It should be possible to move upwards by removing the end of the URL. For that, a hierarchical design is required.
- Users should be able to guess the URL. Because of that, meaningful naming is important.
- To allow easy typing of the URLs, the usage of special characters should be avoided.

Figure 5.5 shows the main navigation bar of the web platform in combination with the URL. It can be seen that the URL is related to the navigational elements.

The URL design of the web platform offers four top-level directories to match the elements of the navigation bar. For that, the following elements are available:

- *event/*: all pages related to the events
- *accounts/*: managing the user accounts
- *research/*: download all data for research purposes
- *admin/*: admin panel

As an example, the URL structure of the research directory is listed. It can be seen that the requirements for the usability of URLs are fulfilled.

- *research/*
- *research/overall/*
- *research/overall/user*
- *research/overall/registration*
- *research/overall/categories*
- *research/overall/events*
- *research/overall/events/ondemand*
- *research/overall/events/live*
- *research/event/ID/TYPE*
- *research/event/ID/TYPE/watch/statistics*
- *research/event/ID/TYPE/watch/csv*
- *research/event/ID/TYPE/watch/csv/statistics*
- *research/event/ID/TYPE/...*

### 5.3.2 User Management

The developed web platform is available for registered and authenticated users only. Due to that, a management of the accounts, the privileges and the processes of registration and authentication is required. To implement these features, the built-in functionalities of Django are used. For a better understanding, these functionalities are explained at first. After that, it is shown how they are used to provide the features required by LIVE.

User Management Models

Django provides a mechanism to save user accounts and their permissions. For that, it offers the possibility to assign a user to a group which has certain permissions. In addition, it is possible to grant permissions to the user directly. The mentioned user accounts and their related groups and permissions are handled by models. In Figure 5.6 these models are shown. It can be seen that the model *User* represents a complete user account. The fields of this model are derived from the model *AbstractUser* which derives from *AbstractBaseUser* and *PermissionsMixin*. The very basic permissions (*is_superuser*, *is_staff* and *is_active* are used mainly by the admin panel to state the privileges to access it or not. From the model *PermissionsMixin* the user account gets its group(s) and permissions. For that, the model holds corresponding lists of the related models. In addition to these models

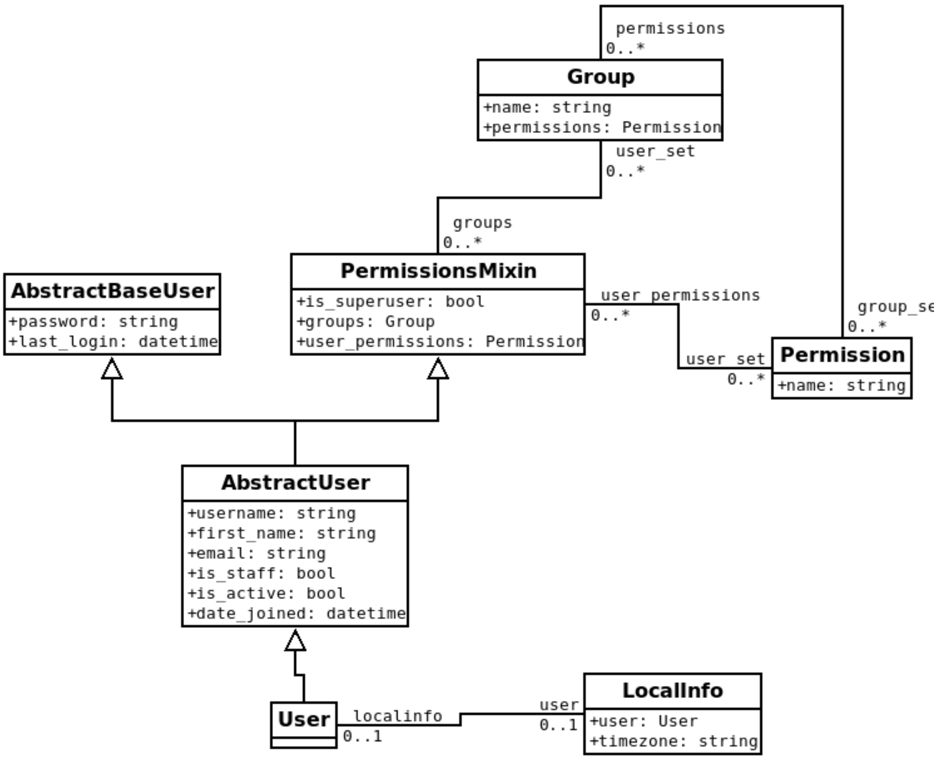



Figure 5.6: The models of the user management.

provided by Django, the model *LocalInfo* is implemented by the *core* package of LIVE. This model assigns a timezone to a user account.

User Management Views and Templates

The explained models of the user management provided by Django, are supported by offered views and methods to handle the registration and the authentication. It is only required to provide the templates for these views in order to make the user interface of these parts related to the rest of the platform. The user registration form shown by Figure 5.7 is an example for such a template. It can be seen that all of the required fields are present. In addition to that, similar views and templates are used to provide functionalities to edit an account or to reset the password.

User Group

In the case of LIVE, users with different privileges are available. This includes the following:

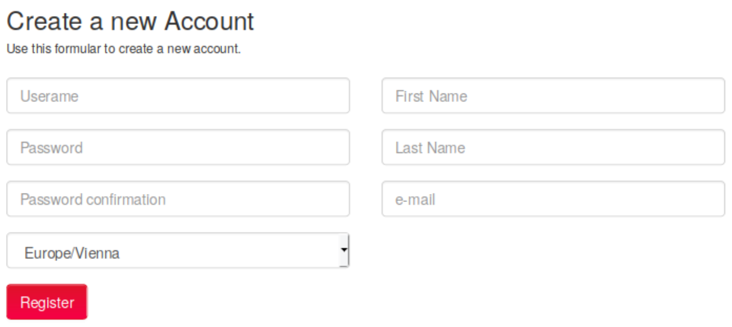



Figure 5.7: The user registration form.

- normal users
- teachers
- researchers
- administrators

The first user group is representing the students. This means that these users are only able to join to the events and to participate at the offered methods of interaction. They are additionally allowed to see the analysis of their own performance. From a technical point of view these users are part of no group and have no special permissions. In contrast to that, the teachers are part of the corresponding group. The permissions of this teacher group enable them, in addition to the normal users, to create events which are supported by interaction methods. Furthermore, they are able to evaluate the performance of the students by viewing the analysis of all of them. A different group, called researchers, could be assigned to the user accounts. With this permission it is possible to download all of the data generated by LIVE for research purposes. Finally, there are the administrators. They are able to access the admin panel to distribute the permissions of the other user accounts. Users are called administrators if the *is_superuser* flag is set.

### 5.3.3 Category- and Event-Management

As mentioned earlier, the videos and live-broadcastings are organized as events. Because of that, the *core* package provides all of the required function-

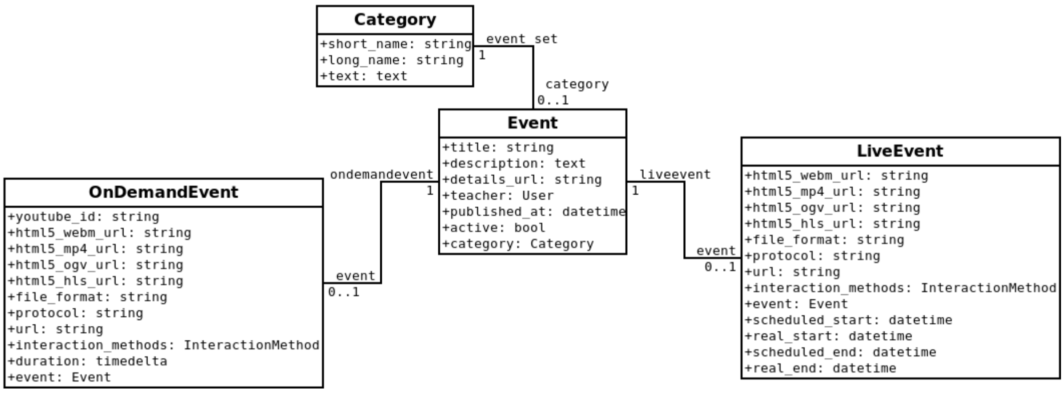



Figure 5.8: The models of the event management.

alities to manage the tasks related to the events. This includes the creation of the events and their grouping into categories. Furthermore, it shows them to the students and offers the teacher area (backend). At this area the teachers are able to access the settings of the event and the different possibilities of analysis.

`Event Management Models`

To save all the data required to handle the events, some models are used. As shown by Figure 5.8, the central element is the model *Event*. This model saves the metadata of an event. It includes a title, a description and an URL to a website related to the event (e.g. the course website). In addition, it links to the user account of the teacher of the event which is represented by the corresponding model presented by the previous section. Because each event belongs to a category, there is a relation to the model *Category*. This model holds a name and a description of the category.

Each event could provide a video, a live-broadcasting or both. For that, there are corresponding models. The video part is represented by the model *OnDemandEvent*. This model mainly saves the information regarding the video. For that different URLs or similar things are saved. Which field is actually used depends on the method used to show the video. It is possible to select from the different video playback methods presented by Section 5.2.3. Additionally, the interaction methods, which are offered at the video, are saved by building a relation to other models (see Section 5.4.3).

In contrast to that, the model *LiveEvent* saves all the data required for a live-broadcasting. It can be seen that it is basically the same like the model for videos. However, it additionally saves the date and time of the start and the end of the live-broadcasting. This is done, because such a live-broadcasting is only available at the scheduled time.

The explained models are forming the base for all the actions related to an event. The typical life cycle of an event consists of the following steps:

`Life Cycle of an Event`

1. create events metadata
2. add video or live-broadcasting
3. configuring the methods of interaction
4. releasing the event to the students
5. analyzing the performance of the students

At first, a generic event has to be created to hold the metadata. For that, a form is offered which has fields to input the title, a description and the URL to a details website. Furthermore, a category has to be selected from a list.

`Create generic Event`

Now it is required to add settings for a video or a live-broadcasting. In the case of a video, the form shown by Figure 5.9 is used. It can be seen that the form offers the possibility to input the information required to play the video according to the fields of the corresponding model (*OnDemandEvent*). The form to add the settings for a live-broadcasting looks very similar to the form for videos. The only difference is that it additionally offers fields to enter the scheduled date and time of the beginning and the end of the event.

`Playback Settings`

In the form the teacher has to select the methods of interaction she/he wants to offer at the concrete event. Based on the selected methods of interaction, the teacher is required to further configure one or more of the selected interaction methods. For that the interaction methods offer their own dialogues (see Section 5.6).

After that, the event is ready to be released to the students. In the case of a video this means that from this time on the students are able to join the video. Due to the fact that a live-broadcasting is scheduled to a specific date and time, it is required that the teacher starts the live-broadcasting manually. More details of the process of joining and watching events are presented by the following section.

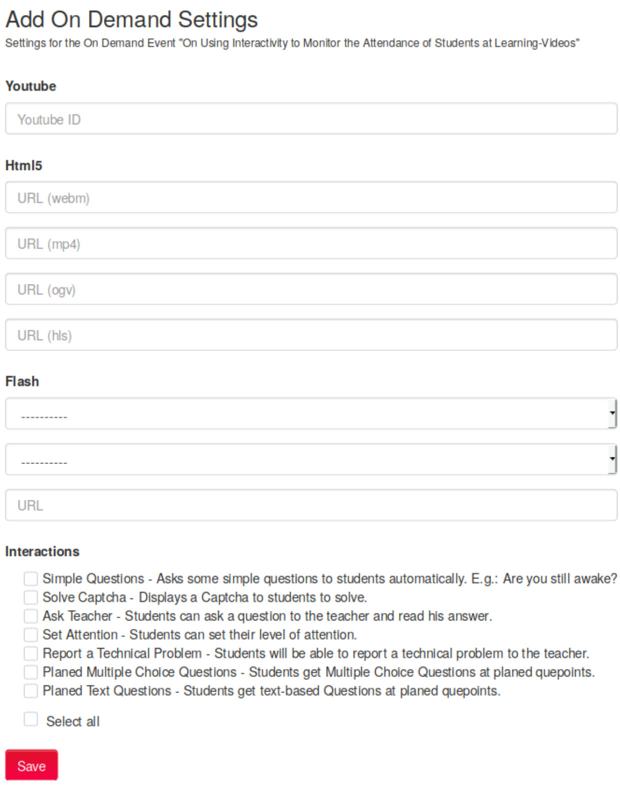



Figure 5.9: This dialog creates an on-demand event.

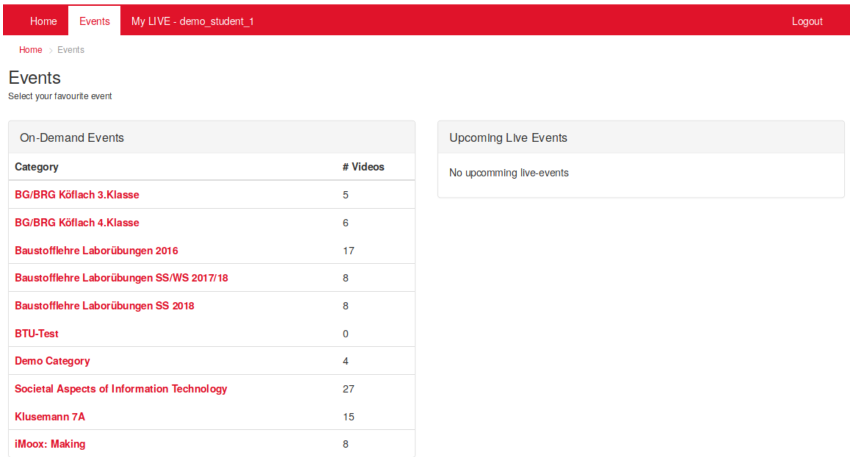



Figure 5.10: The categories and the upcoming events are shown.

List of Events

From the students point of view, the events are brought to them by a listing shown by Figure 5.10. As indicated by this figure, there are two lists. On the right hand side is a list of upcoming live-broadcastings. This list only shows an event if there is a live-broadcasting scheduled to start in the future. In contrast to that, the videos are shown at the left hand side. For that, the different categories of events are listed. Additionally, the number of videos of each category is printed. On clicking the name of a category, all of the videos related to the category are presented.

Performance Analysis

The final action in the life cycle of an event is the analysis of the performance of the students. For that, there are three points of view:

- overall analysis of all students
- analysis of the attention of a single student
- evaluations offered by the methods of interaction

For all of these possibilities of analysis the event management provides the entry point to them. The overall analysis of the students consists of several features to evaluate the attendance of the students. In contrast to that, there are additional methods of analysis to assess the performance of a single student. These possibilities are presented by Chapter 6 from a researchers point of view. The technical aspects are explained by Sections 5.3.4 and

5.5. Some methods of interaction are presenting questions to the students. That's why these interaction methods are offering different possibilities to evaluate the answers of the students.

### 5.3.4 Joining, Watching and Leaving Events

The previous section stated that users are joining to the events in order to watch the content provided by the event. The students use the presented list of events to join to them. While they are joined, they are watching the video or the live-broadcasting. In this watching phase all of the activities of the students are recorded. This means that the recorded values are enabling the platform to state when the students watched which part of the video or live-broadcasting. After each phase of watching, the students are required to leave the event.

`Recording Models` To save all of the required values of the process of joining, watching and leaving, there are some models. As shown by Figure 5.11 the central element is the model *JoinedUser*. Such a model binds a user account to an event. For that, it holds a relation to the model *User* indicating the student as explained by Section 5.3.2. In addition, there is a relation to the model *Event* to state the event to which the user joins. This model is explained by the previous section. Due to the fact, that an event could have both, a video or a live-broadcasting the model *JoinedUser* defines the mode (video or live) to which the user joins. The process of joining and leaving is handled by the corresponding methods.

In order to understand the meaning of the model *History* it is required to examine how a video or a live-broadcasting is watched. This means that a user watches every time from a specific position in the video to a different position. This happens at given date and time values. It can be seen that the watched phases are forming intervals. Each interval is represented by the model *History*. This model saves the position of joining and leaving in relative and absolute values. Now it is obvious that a joined user has a set of such *History* models stating the watched timespans of the student.

The mentioned method for joining mainly creates a new history for the user. While watching the web platform asks if there are interactions to display

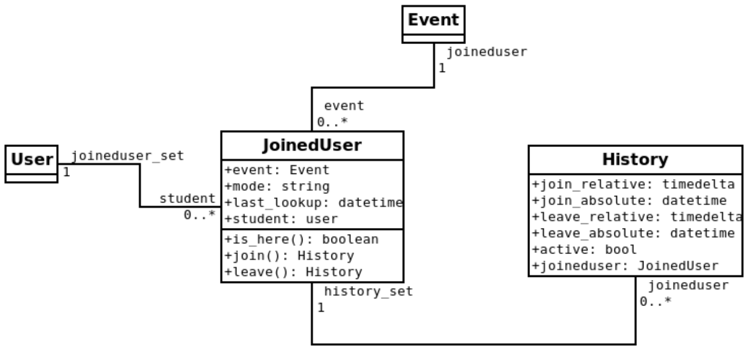



Figure 5.11: The models required to join an event.

(see Section 5.4). This lookup happens every five seconds and the date and time is saved by the member *last_lookup* of the *JoinedUser* model. With this value it is possible for the method *is_here()* to state if the student is still watching. For that, it simply compares the current date and time with the value of the last lookup. If the difference is too big, it can be assumed that the student is not watching anymore. In this case, or if the student manually left the event, the corresponding method is called. This sets the relative and absolute values of date and time in the latest instance of the *History* model.

Teacher's Actions

The technical aspects are mainly driven by the actions of the users, namely the teacher and the students. In the case of a live-broadcasting, the teacher is required to start the event. For that, she/he has to use a button presented at the backend. After that, the teacher is redirected to a special user interface where she/he can monitor the event (see Figure 5.12). This user interface could be divided into four areas. Area number (1) shows some metadata of the event. This includes the name of the event and some information about the scheduled and real begin. In addition, the scheduled end is printed. On the right hand side of this area, there is region (2) for the analysis of the attendees. It shows the number of currently watching students. This value is based on counting the number of *JoinedUser* instances related to the event where the return value of the method *is_here()* is true. Furthermore, a so called attention level is calculated which tries to indicate how attentive the watching students are at the moment (see Section 5.5.2). At the top of the analysis, there is a button to stop the event. This action sets the real end of

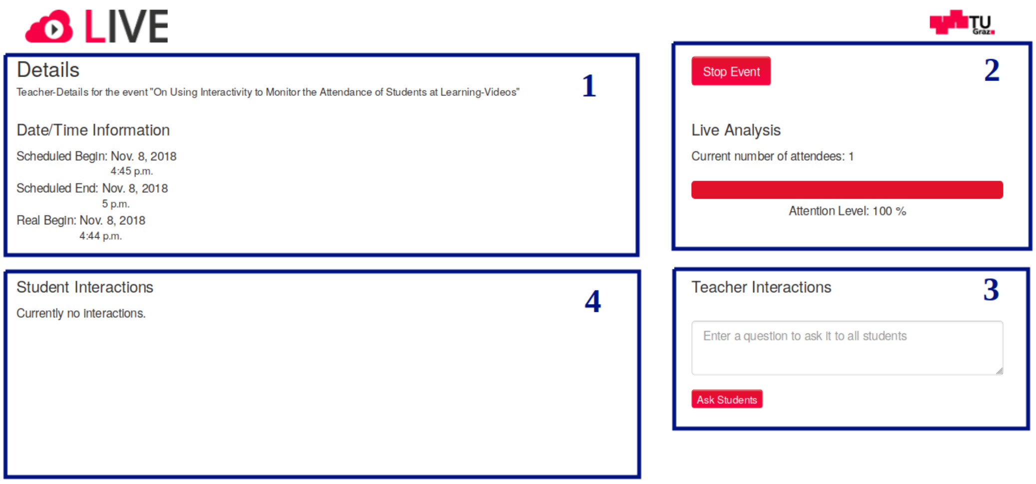



Figure 5.12: The special user interface for the teacher at a live-broadcasting.

the event to the current date and time and forces all of the joined students to leave the event.

Below the analysis area there is some room (3) to place control elements which are enabling the teacher to invoke interactions. These control elements are provided by the different methods of interaction (see Section 5.6). In this example, a form to ask a question to the students is presented. At the left hand side, area number (4) is used to display interactions which are addressed to the teacher. Such interactions are mainly showing the answers of the students to questions asked by the teacher.

Students' Actions

The user interface for the students is the same at a video and at a live-broadcasting. As illustrated by Figure 5.13, there are three important parts of the user interface. The example shows that a video is currently overlaid and therefore interrupted by an interaction. The dialog of the interaction is marked as area number (1). In this example a multiple-choice question with a picture is presented to the students. It can be seen that this dialog is placed partially over the video (2) and the sidebar (3). The screenshot shows that the video is currently paused because of the interaction. Below the video, there is a button to switch to full screen mode. This mode shows the video and the sidebar covering the whole screen.

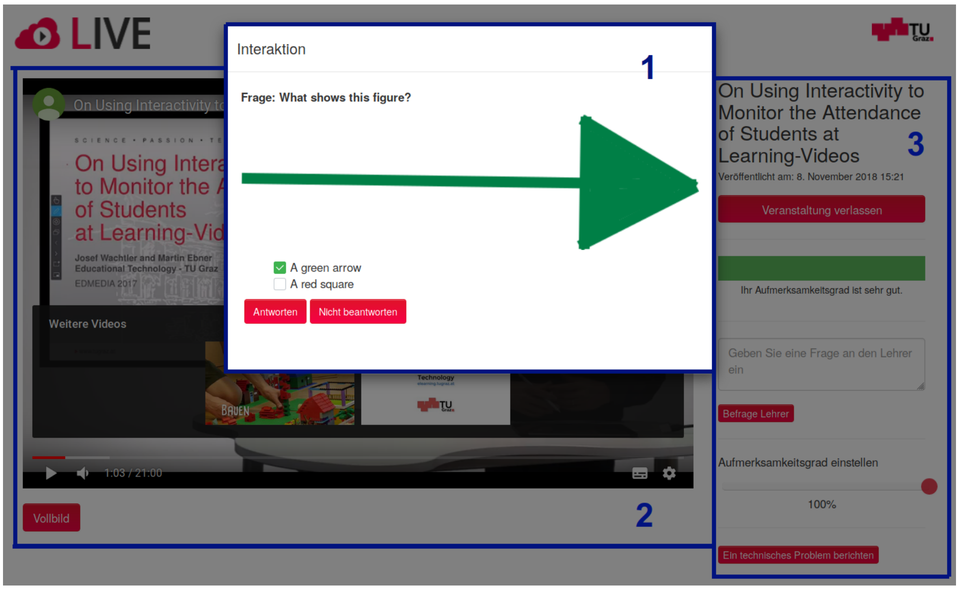



Figure 5.13: The special user interface for the students while watching a video or a live-broadcasting.

The sidebar presents multiple information to the students. At the top, there is some metadata including the name of the event and its date and time of publishing. Below that, a button is visible which enables the students to leave the event. Clicking this button redirects the student away from the watching user interface and calls the corresponding method of the *JoinedUser* model to perform the actions required to properly save the date and time of leaving in absolute and relative values. The next part of the sidebar is a visualization of the attention level of the student (see Section 5.5.2). For that, the colors of a traffic light are used. The rest of the sidebar is used to display some control elements which are enabling the students to start interactions by themselves. They are provided by the interaction methods in the same way as at the user interface for the teacher presented above.

### 5.3.5 Settings

The package *core* provides a mechanism to save settings which could be changed at the admin panel. It is important that these settings are not confused with the basic settings of Django. Because of that, the first ones are called platform-settings and the second ones are referred as framework-settings.

Framework-Settings The framework-settings are used to configure the very basic components of the whole project. This includes the configuration of the database and other similar values. For that, a python module is used which holds the different settings values as constants. These settings could be used by importing the corresponding module in the python code. It seems to be obvious that such settings are defined during the development or at the deployment of the platform. This fact makes them not very useful for settings which are prone to be changed during the runtime of the platform.

Platform-Settings Because of that, platform-settings have been introduced. On developing this settings mechanism the interaction methods have been taken into account. This means that each setting is represented by the following items:

- *key*
- *value*
- *interaction_method*

This key/value store is provided by a model with the listed fields. The key is a name for the settings value. The field *interaction_method* represents the interaction method to which the setting belongs. To implement global settings this field is optional. To access each value, one might assume that the key has to be unique. This is only partially true because it is defined that the combination of the key and the interaction method has to be unique. With this constrain it is possible for the different interaction methods to use the same keys for their settings.

To access the settings, some helper functions are available. There are different functions for the different types of the value. This means that there is one for a string, an integer, a boolean and bytes. The function for the string is shown by the Listing 5.1. It can be seen, that the function takes the key as a required parameter. Additionally, there is an optional parameter for the interaction method. Based on these parameters the corresponding value is returned as a string.

```
from core.models import ConfigEntry

def get_config_string(key, interaction_method=None):
    try:
        if interaction_method:
            c = ConfigEntry.objects.get(
                key=key, interaction_method__id=interaction_method.id)
            return str(c.value)
        else:
            c = ConfigEntry.objects.get(key=key,
                                        interaction_method__isnull=True)
            return str(c.value)
    except:
        return None
```

Listing 5.1: The helper function to get a settings value.

## 5.4 Interaction Framework

Overview

The different methods of interaction, provided by LIVE, are implemented as plugins. These plugins are using the so called interaction framework to be integrated. For that, the framework consists of the following parts:

- rules for the creation and registration of the package
- the definition of the interaction types
- models to save all the data related to interactions
- algorithm to schedule the interactions
- a library to simplify the implementation of a view to show an interaction
- guidelines for the detail pages for the users

### 5.4.1 Package Creation

An interaction method is a Django app

Each plugin representing an interaction method has to be a Django app. Such an app is a python package providing the different parts of the app. This includes the models, the views and the configuration of the URLs (Django Team, 2018a). Because of that, the first step to create a new method of interaction is the creation of such an app according to the standard principles valid for a Django app.

After that, it is necessary to register the app as an interaction method. For that, a new instance of the model *InteractionMethod* has to be created (see Section 5.4.3). This creation sets the following metadata:

- the title of the interaction method
- a detailed description
- the type (see Section 5.4.2)

`Requirements Type 1` The most important part of this step is the specification of the type. Depending on the type, the interaction method is required to provide certain components. If the type is set to automatic the interaction method has to provide the following platform-settings:

- *TRIGGERS_PER_HOUR*
- *STUDENT_CALLBACK_VIEW*

The first setting is responsible for stating how often the interaction should be displayed per hour in an automatic and random way. The second one has to provide the name of the view which should be used to display the interaction. With these values the scheduling algorithm is able to handle the interaction methods of this type.

`Requirements Types 2 and 3` If the interaction method is of type 2 or 3, it seems to be obvious that the control elements which are required to invoke the interactions have to be provided. For that, a Django template has to be used. The name of this template should be provided by a platform-setting with a specific key. In the case of a template used for the user interface of the teacher, the key has to be *TEACHER_TEMPLATE*. In contrast to that, the key is named *STUDENT_TEMPLATE* if the template should be part of the user interface of the students.

`Requirements Type 4` The requirements for the type of planned interactions are stating that there has to be a mechanism to actually plan the interactions. For that a view has to be implemented. The name of this view must be registered by providing it through a platform-setting with the key *INITIAL_SETTINGS_VIEW*. This view should display the video to select the position for the interaction.

For that, a template is offered to include. With the selected position as a parameter, a dialog should be implemented to create the interaction and save it using the corresponding model.

### 5.4.2 Types

As indicated by the previous section, there are different methods of interaction which are implemented as independent plugins. Such a method of interaction has to be of a given type. Currently there are four types:

1. *Automatic Interactions*
2. *Student Triggered Interactions*
3. *Teacher Triggered Interactions*
4. *Planned Interactions*

Type defines way of invoking

It can be seen that the type is defining how the interactions of the method of interaction are triggered. At the first type all of the interactions are started automatically. They are additionally distributed in a random way across the video or the live-broadcasting (see Section 5.4.4). In comparison to that, the interactions related to the types 2 and 3 are triggered on an action performed by the user. This means that in the case of *Student Triggered Interactions*, the student has to use control elements provided by the interaction methods to invoke an interaction. These control elements are placed on the right hand side of the video or the live-broadcasting. The same is true for *Teacher Triggered Interactions*. However, it seems to be obvious that this is only possible at live-broadcastings because the teacher is not instantly available at on-demand videos. Finally, there are the *Planned Interactions* which are showing their interactions at specified positions in the video. This means that the teacher has to create the interactions before releasing the video to the students. During the creation of the interactions, the teacher has to set the position where the interaction should occur.

### 5.4.3 Models

The very basic components of the interaction framework are some models. They are saving all of the data required to handle the interactions in all phases of their life cycle.

`Interaction Model` It can be seen in Figure 5.14 that there is a model called *Interaction* which forms a base for each concrete interaction which could be displayed by LIVE. This model holds the name of a view which is used to display the interaction. If such an interaction has to be shown the mentioned view is displayed to the teacher or to the student. Because of the fact that interactions are shown at specific positions in the video or the live-broadcasting, the model saves the scheduled starting position of the represented interaction. As indicated by the previous sections, each interaction is part of a plugin which is called method of interaction. Because of that, there is a relation to the corresponding model to state to which method of interaction the interaction belongs.

At the description of the model representing an interaction, two remarks were given:

- interactions are bound to a destination
- interactions are part of a method of interaction

`Concrete Interaction Models` The mentioned model is a base for further concrete models which are defining the destination of the interaction. This means that there is a model called *StudentInteraction* which represents interactions shown to a student. This model derives from the *Interaction* model and additionally has a relation to a joined user. In contrast to that, the model *TeacherInteraction* also derives from *Interaction* and has a relation to an event of which a user is a teacher. The third sub model of *Interaction* is called *ErrorToTeacher*. It represents an error shown to a teacher. In the case of a live-broadcasting this error is displayed to the teacher instantly and at a video the errors are listed at the backend of the event for the teacher. Such an error could be triggered by an interaction method or by the platform itself. It contains a subject and a question stating the error. Furthermore, it holds a relation to the corresponding event which is of a given mode. If the error is related to a joined user, an associated relation is saved. The teacher is able to submit an

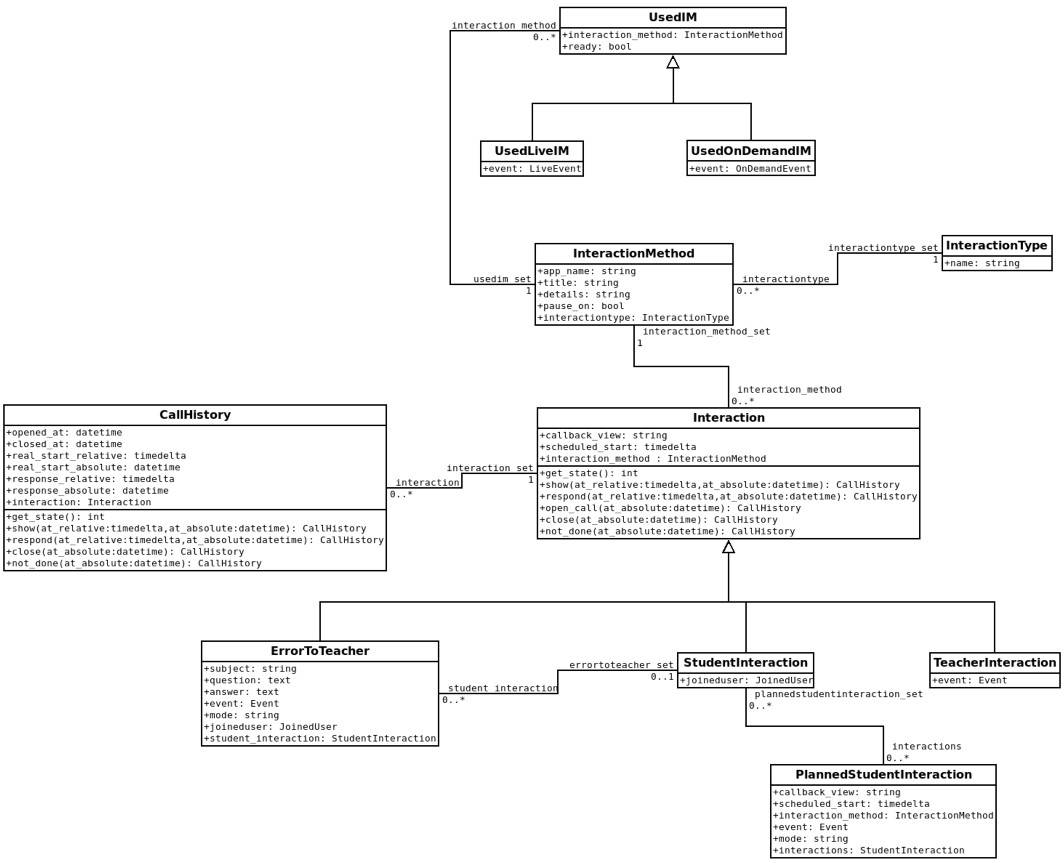



Figure 5.14: The models required to handle the interactions.

answer to the error which is stored by the corresponding field. Because such an answer is directed to a student, a *StudentInteraction* is used to display the answer to the student.

`Planned Interaction Models` Some of the interaction methods are providing interactions shown at specific positions in the video which are planned before the event is released to the students. For that, there has to be a mechanism to save these planned interactions (*PlannedStudentInteraction*). This model offers mainly the same fields as a normal interaction however it adds a relation to an event which is of a given mode. For each interaction, which is planned by the teacher, an instance of this model is used. If a student joins to an event, all of the planned interactions are translated to instances of a real *StudentInteraction*. These student interactions are listed by the corresponding field in the model of the planned interactions.

`Methods and Types` The second remark stated during the description of the model representing an interaction indicated that an interaction is part of a method of interaction. Due to that, there is a corresponding model (*InteractionMethod*) to represent this connection. It saves the name of the plugin holding the interaction method in the field *app_name*. Each interaction method has a title and a detailed description. Furthermore, it states if the video should be paused if an interaction connected to this method is displayed. As suggested, each method of interaction is of a given type (see Section 5.4.2). To define this relation there is a model called *InteractionType*. It has a name and is related to the model representing interaction methods.

`Connect Interaction Methods to Events` Section 5.3.3 states that on creating an event, the teacher has to select the methods of interaction she/he wants to offer. For that, a list is displayed at the creation dialog. This list is based on the interaction methods saved by the *InteractionMethod* model explained above. To finally save the selected methods some helper models are required. The base for them is the model *UsedIM*. It holds a relation to the selected interaction method and a flag indicating if the interaction method is ready to be used. From this base model two concrete ones are inheriting. The first one (*UsedLiveIM*) represents a connection to the corresponding live-event and the second one (*UsedOnDemandIM*) holds the connection to an event showing a video. It can be seen that with these helper models it is possible to state that a concrete event could have multiple interaction methods and vice versa. In addition, it helps to save additional information like the *ready* field.

`CallHistory`

Because all of the interactions are shown to the teacher or to a student, it is required to record when the interaction is shown and when the reaction of the user happens. This recording is important for the algorithm scheduling and displaying the interactions (see Section 5.4.4). For that, there is the model *CallHistory*. It has a relation to the corresponding interaction model and because of that, each interaction could have multiple recordings of its displaying. This is required because in the case of a video, it is possible that an interaction is shown multiple times because a video could be watched more often than once. The fields saving the values required for the recording are the following: (Wachtler & Ebner, 2019)

- *opened_at*: Saves the date and time of the creation of the *CallHistory*.
- *closed_at*: Holds the date and time of closing the *CallHistory*. This happens when the interaction is never displayed.
- *real_start_relative*: position in the video of the displaying of the interaction
- *real_start_absolute*: date and time of the displaying of the interaction
- *response_relative*: position in the video of the user's response
- *response_absolute*: date and time of the user's response

These values are not accessed directly. Instead of that, the offered methods of the model *Interaction* are used. These methods are mainly delegating to the corresponding methods of the model *CallHistory* after performing some checks.

All of the mentioned related models which are not shown by the figure are the same as explained by Sections 5.3.3 and 5.3.4.

### 5.4.4 Scheduling Algorithm

The models presented by the previous section are saving the interactions with all of the required data. To handle the models, there is an algorithm which is responsible for planning and loading the interactions (Wachtler & Ebner, 2019).

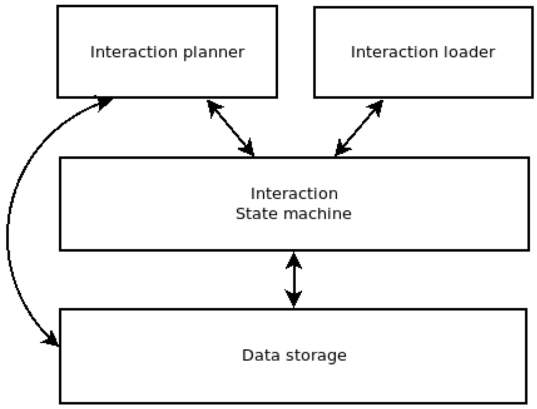



Figure 5.15: The components of the scheduling algorithm.

Overview The different components of this algorithm are shown by Figure 5.15. It can be seen that the models are forming the base. While the video or the live-broadcasting runs, the state machines of the interactions are tasked with the management of the interactions. This means that based on the state, interactions are shown or ignored. To achieve these actions the state machines interact heavily with the models.

In addition to the state machines of the interactions, there are two supporting components. The first one is the interaction planner which is responsible for scheduling the interactions for the users when they join to the event. For that, the planner has to directly access the models. Finally, there is the interaction loader. This helper module handles the displaying of the interactions at their scheduled time. For that, it uses the mechanisms of the state machine.

## State Machine

Transitions The field of operation of the state machine is a single interaction. This means that each interaction is in a state which is defined by the values of the model *CallHistory*. The state is defining the actions which are performed on the interaction. Figure 5.16 shows the states and the transitions between them. It can be seen that the initial state is called "open". This state is reached if a user joins to an event the first time and all of the state machines are created

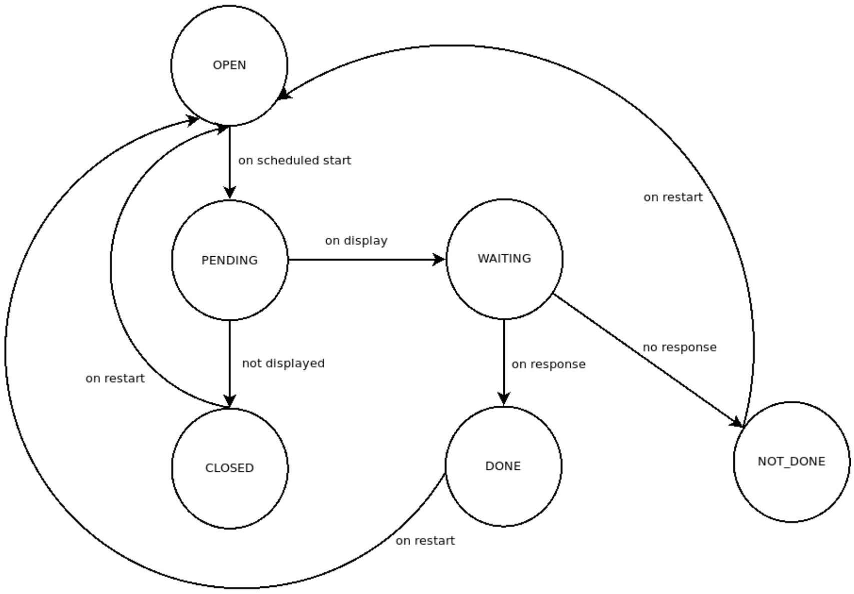



Figure 5.16: The states of the interaction state machine.

by planning the interactions (see Section 5.4.4) where a new *CallHistory* is added to the interaction.

Now the video or live-broadcasting proceeds and during that, the current position is monitored. It seems to be obvious that after some time the scheduled starting position of an interaction will be reached. If this is the case the state of this interaction switches to "pending". From this state on there are two possibilities:

- The interaction is not shown.
- The interaction is displayed.

At the first possibility, the interaction loader (see Section 5.4.4) transfers the interaction to its displaying queue. However, it is never shown because an other interaction is currently on display in its waiting state. In this case the state switches directly to "closed". The second possibility happens if no other interaction is on display and the interaction in its pending state will be displayed. This initiates a transition to the state "waiting". As indicated by the name of this state, the interaction is now waiting for a response from the

| State | opened _at | closed _at | real_start _relative | real_start _absolute | response _relative | response _absolute |
|---|---|---|---|---|---|---|
| OPEN | x | | | | | |
| PENDING | x | | x | | | |
| WAITING | x | | x | x | | |
| DONE | x | | x | x | x | x |
| NOT_DONE | x | x | x | x | | |
| CLOSED | x | x | | | | |

Table 5.1: The states are defined by the fields of the *CallHistory* model. An "x" indicates that the related field is set.

user. If such a response finally comes the state will be changed to "done". In contrast to that, the state is supposed to switch to "not_done" if no response of the user happens.

In the case of a live-broadcasting, it is clear that the whole event could be watched only once. However, at a video it is possible to watch it multiple times. In this case the interaction planner which runs when a user joins to a video, resets the state of all interactions to "open". This is done by creating a new *CallHistory* for each of the interactions.

State depends on Model Fields

As indicated, the state of an interaction is defined by the values of the fields of the *CallHistory* of an interaction. This means that the state depends on the distribution of the set fields. Table 5.1 lists all of these fields and their relation to the state.

### Planning Interactions

As mentioned by the previous section, there is a helper module to create the plan of interactions. This happens if the user joins to an event the first time. Because each interaction is represented by a concrete instance of the model *Interaction*, such an instance is created by the planner. Which concrete model is used depends on the role of the user (*StudentInteraction* or *TeacherInteraction*). Depending on the type of the interaction method the interactions are created in different ways:

- planned: type 4
- automatic and random: type 1
- student- or teacher-triggered: type 2 and 3

Planned Interactions

The first task of the interaction planner is the creation of the planned interactions. As explained by Section 5.4.3, a model called *PlannedStudentInteraction* is used to save the interactions created by the teacher before the event starts. Now, if a student joins to an event, all these planned interactions are translated to instances of a *StudentInteraction*. These created interactions are placed in the corresponding relation of the *PlannedStudentInteraction* to have a link between the new concrete interaction and its planned template.

Automatic Interactions

After that, the interactions which are part of an interaction method of type 1, are created. This means that they are distributed throughout the timeline of the video or live-broadcasting in a random way. For that, the interaction method is required to provide a platform-setting named *TRIGGERS_PER_HOUR*. This setting has to specify how often the interactions of its interaction method should be displayed per hour. With this value a slot length is computed. This is done by dividing the seconds of an hour (3600) by the value of this setting. Now the event consists of some slots. In each slot an interaction is placed at a random position. In addition, there are two restrictions in order to avoid that interactions are placed too close to each other:

- The first and the last 10% of the slots are not used for an interaction. This is done to avoid that an interaction is placed at the beginning in a slot and at the end in the preceding one which will lead to the fact that these two interactions are very close.
- If there are other interactions located in the slot, it is tried to leave at least a free space of +/- 10% of the slot length.

User Triggered Interactions

The interactions which are part of interaction methods of type 2 and 3 are not created by the interaction planner. This is the case because such interactions are created on actions performed by the users during the event. As explained above, the related interaction methods are required to provide control elements to perform the actions. Due to that they are also required to create the correct concrete instance of *Interaction* when the action is performed.

```
if users_first_join() {

    for pi in set(PlannedStudentInteraction) {
        interaction = create StudentInteraction from pi
        pi.interactions.add(interaction)
    }

    for each interaction_method of type automatic {
        slot_length = 3600 / interaction_method.interactions_per_hour
        slots_per_video = video_length / slot_length
        for each slot in slots_per_video {
            do {
                pos = random(slot.begin + 10% of slot_length,
                             slot.end - 10% of slot_length)
            }
            while pos is not around an existing interaction

            automatic_interaction = StudentInteraction at pos
        }
    }
}
for each StudentInteraction {
    create a new CallHistory
}
```

Listing 5.2: The interaction planner creates the interactions when a student joins a video or a live-broadcasting.

Interaction Planner

The interaction planner is shown by Listing 5.2 in pseudo code. It can be seen that the creation of the interaction happens only if the user joins the first time. Within this block there are two for-loops. The first one is responsible for creating the planned interactions as explained above. The second one creates the automatic and random interactions. This is done for each interaction method of this type. At first the slots are computed and after that it is tried to find a position for the interaction in the slot which is not too close to another already existing interaction. Finally, the interaction planner (re)sets the state of all interactions to their initial state. For that, a new instance of the model *CallHistory* is created for each interaction every time a user joins an event.

### Showing Interactions

To show interactions two additional components of the interaction loader are required:

- a monitoring of the current position
- a library to implement a view to show an interaction

Position Monitoring

The interaction loader monitors the current position of the video or the live-broadcasting during the event. It continuously reports this position to the state machines of the interactions. With this information, they are able to switch their state from "open" to "pending". If such an interaction is in the state "pending", the interaction loader tries to show it to the user by calling the corresponding view and changing the state to "waiting". This happens only if no other interaction is in its waiting state. In this state the interaction loader waits for a response from the user. If this response finally happens, the state machine of the associated interaction is informed so that it can switch to its end state.

Library for Views

To implement a view which is able to display an interaction, a library is offered. This library handles all the related state changes and gives access to the interaction represented by the view. Listing 5.3 shows an example of a view presenting an interaction to a student. At first some libraries are loaded. This includes the mentioned interaction loader. After that, the view is defined as a simple python function (line 11). The view begins with getting the current date and time (line 12) and retrieves the current position in the video or the live-broadcasting in line 13. Now, the interaction loader is used to check the interaction which should be displayed. If something went wrong, an error is raised. If the view is displayed to the user the first time, a HTTP-GET request is used. Because of that, the "if" block starting at line 20 never happens. Finally, a method of the interaction loader is called in line 29 which indicates that the interaction is now shown. This initiates the state change from "pending" to "waiting". If a response from the user is submitted using HTTP-POST the corresponding "if" block is taken. After some handling of the response according to the duties of the interaction, the interaction loader is informed that the response happened (line 25). This leads to the final switch of state to "done".

```
from django.shortcuts import render
from django.contrib.auth.decorators import login_required
from django.http import HttpResponseRedirect, Http404
from django.core.urlresolvers import reverse
from django.utils import timezone

from interactions.student import StudentInteractionLoader
from interactions.utils import get_at_relative

@login_required
def index(request, student_i_id, call_id):
    n = timezone.now()
    d = get_at_relative(post=request.POST, get=request.GET)
    l = StudentInteractionLoader(user=request.user)
    ok = l.load(student_i_id=student_i_id, call_id=call_id,
                at_absolute=n, at_relative=d)
    if not ok:
        raise Http404

    if request.method == 'POST':

        # Here is the place to validate a form or react to
        # the students response.

        l.respond()
        return HttpResponseRedirect(
            reverse('interactions.views.no_interactions'))

    l.show()
    params = {
        'cid': int(call_id),
        'siid': int(student_i_id)
    }
    return render(request,
                  'interactions/appname/index.html', params)
```

Listing 5.3: An example of a view to show an interaction.

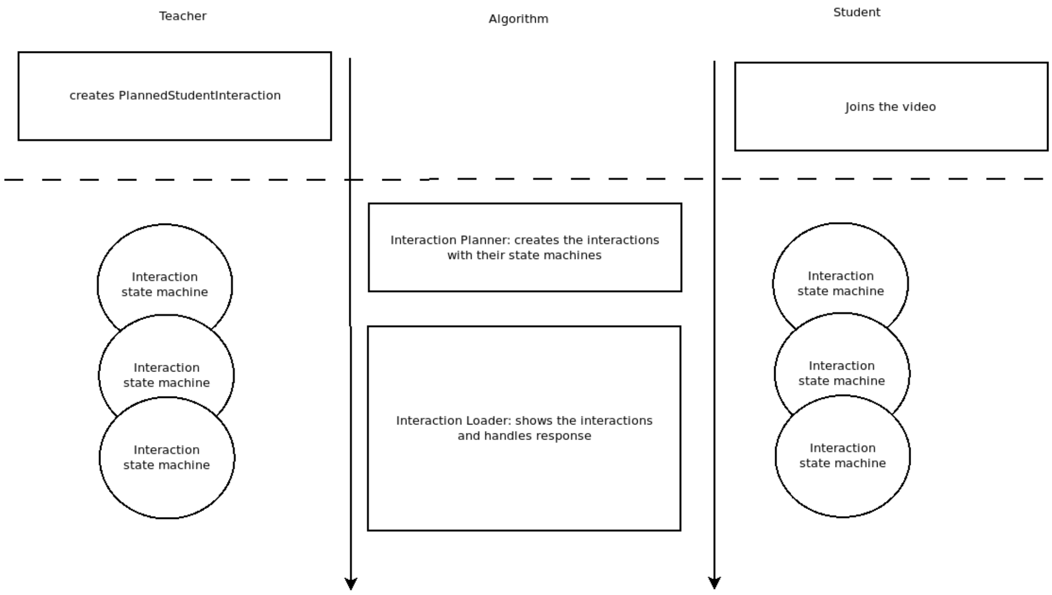



Figure 5.17: The flow of events of the scheduling algorithm.

### The Algorithm

With all parts of the algorithm explained, it is possible to combine them together as shown by Figure 5.17. It can be seen that at first the teacher has to create the planned interactions. From a technical point of view this creation is not part of the algorithm but it is required for a better understanding. Now it is possible for the students to join to the event. This invokes the interaction planner which creates the interactions for them. For that, the planned interactions created by the teacher are translated to real interactions and the automatic ones are placed throughout the event. Now, the interaction loader starts running. As mentioned above, it reports the current position to the state machines of the interactions in order to enable them to switch their state if required. It can be seen that interactions are displayed and answered during the whole event. In the case of a video, the interaction loader only handles interactions for students. At a live-broadcasting the same happens for a teacher because in this case she/he has to be present to react to the interactions requiring her/his attention.

### 5.4.5 Detail Sites

Each interaction method generates data. In most of the cases this includes questions and related answers. Because of that, there is a mechanism to present this data to both, the teacher and the student.

Requirements It is possible to create multiple versions of such detail sites. Each site has to be a view which takes some specific parameters. In the case of a detail site displaying information about a single student the corresponding *JoinedUser* model is used as the parameter to identify the student of which the information should be displayed. In contrast to that, a view presenting information about all joined students the corresponding event and mode are passed as parameters.

It seems to be obvious that these views have to be registered at the platform so that it can display the links to them. For that, two helper functions are required. The first one is responsible for creating the links to the views presenting information about a single student. In contrast to that the second one does the same for the views presenting details about all students. It is clear that views of the latter group are only available to the teacher. The helper functions are required to return a list of the links in combination with a title. To give the platform access to these functions, the names of them have to be supplied using the following platform-settings:

- *STUDENT_DETAILS*
- *TEACHER_DETAILS*

## 5.5 Attention Analysis Framework

LIVE provides a framework to enable the interaction methods to be part of the analysis of the attention of the students. This framework consists of two parts: (Wachtler & Ebner, 2014a)

- a detailed recording of the watched timespans
- a calculation of an attention level for each timespan

### 5.5.1 Watched Timespans

The detailed recording of the watched timespans is implemented by the platform using the models presented by Section 5.3.4. As explained by the mentioned section, there are some models for this recording. One model indicates, that a user has joined to an event and the other one represents a single watched timespan. Such a timespan is called *History* because a list of such timespans resembles the complete watching history of a joined user.

With this recording it is possible to calculate some statistical values like the following:

Statistical Values

- longest watched timespan
- shortest watched timespan
- average length of a timespan

In addition to these values, some further analysis features could be implemented. This includes a list of all joined users and their complete watched timespan (see Section 6.1.1) as well as a complete visualization of the watching history of a joined user as shown by Section 6.1.3.

### 5.5.2 Attention Level

The second part of the attention analysis consists of the calculation of a so-called attention level. This value ranges from 0% to 100%. The meaning of the lowest value is that the student is considered completely absent. In contrast to that, the highest value states that the student is fully attentive.

The attention level is computed at two different times. The first one is the computation while watching the event. In contrast to that, the final attention level of a watched timespan is calculated after watching it.

The actual calculation is mainly done by the interaction methods themselves. For that, they are required to implement the offered interface as explained by Section 5.5.2. Because of that, the calculation is divided into three rounds as shown by Figure 5.18.

Three Rounds of Computation

The first round consists of the calculation of an attention level based on the

First Round

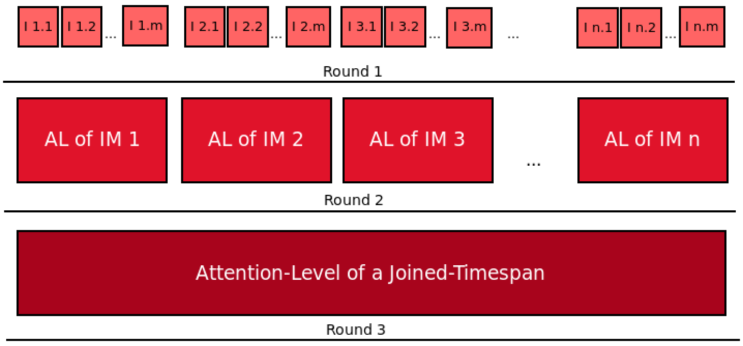



Figure 5.18: The attention level consists of the attention levels of each interaction method.

reaction times to the interactions (I). For that, a function is offered which could be used by the interaction methods (see Section 5.5.2). This function is used by all interaction methods which are representing a classic question answering approach. Other interaction methods are not using this function and because of that, they are completely skipping round 1.

`Second Round` The second round represents the calculation of an attention level (AL) of each interaction method (IM). If the reaction time based approach is used, this round consists of grouping the attention levels of the interactions of this interaction method to a single attention level. In contrast to that, if the reaction time based method is not used, the calculation is done by the interaction method in its own way.

`Third Round` Finally, the complete attention level of a watched timespan is computed. For that, the mean of the attention levels of the different interaction methods is calculated. In the following equation $a_i$ represents the attention level of the $i$-th interaction method and $n$ the number of interaction methods.

$$attention = \frac{\sum_{i=0}^{n} a_i}{n} \tag{5.1}$$

### Interface

`Command Pattern` As mentioned above, each interaction method could compute their own level of attention. For that, a variant of the command pattern (see Section 3.3.3) is used. Because of that, an interface is defined by the module *interactions.analysis*. Mainly, this interface defines the following two methods which must be implemented:

- *get_attention_level(joineduser, at_relative, at_absolute)*
- *get_attention_level_per_history(history)*

The first method has to return the attention level of the given joined user at the date and time provided in absolute and relative values. It is called during the event while the student watches the video or the live-broadcasting. In contrast to that, the second method is responsible for computing the attention level of an interaction method of a watched timespan represented by the history. It is clear that this method is called only after watching. When comparing this interface with the command pattern it can be seen that it is the equivalent of the *Command* interface. The implementation of this interface resembles the *ConcreteCommand* which implements the *Command* interface. The passed parameters are defining the link to the *Receiver* and setting the state.

`Settings`

According to the specification of the command pattern the concrete commands are required to be registered at the *Client*. For that the following platform-settings must be used:

- *LIVE_ANALYSIS_CALLBACK*
- *ANALYSIS_CALLBACK*

The first one should be used for the method returning the attention level during the event. In contrast to that, the second one is responsible for registering the method providing the attention level of a watched timespan. In both cases, the name of the methods has to be saved by the setting.

In the case of LIVE, the *Client* is the platform itself. It passes the registered concrete commands which are providing the calculation of the attention level of an interaction method to the corresponding callers. This means that the method used to calculate the attention level during the event is added to the lookup function which looks for new interactions to display (see Section 5.3.4). This lookup function computes the overall attention level during the event each time it is called. Because of that, it is the caller of the according methods. In contrast to that, the method responsible for calculating the attention level of a watched timespan is linked to the backend site where the analysis is displayed. Due to that, the caller of these methods is the view of the backend site.

### Computation Based on Reaction Time

As mentioned by the previous sections, there is a function which could be used by the interaction methods to calculate an attention level based on the reaction times to the interactions. For that, this library function needs the following parameters:

- all of the interactions of the interaction method
- the watched timespan to analyze represented by its history
- the current date and time
- platform-settings
  - *MAX_LOOKBACK*
  - *LIVE_SUCCESS_UNTIL*
  - *LIVE_FAILED_AFTER*
  - *SUCCESS_UNTIL*
  - *FAILED_AFTER*

`Parameters` The first required parameter is a list of the interactions of which the attention level should be computed. To decide of which call the attention level should be calculated the second parameter is the history to which the call in question is related. The third parameter is required only at a live-broadcasting to state the current date and time as well as the current position. It helps to define which interactions should be considered in the calculation because the first platform-setting states how many seconds the calculation should look back from the current position.

`Calculation` The last four platform-settings are responsible for the actual calculation of the attention level. The two ones with the prefix *LIVE_* are used at the calculation during the event. The latter ones are the same for the calculation for a watched timespan. The actual calculation follows the maxim "if the student reacts slower, the attention level decreases". Figure 5.19 illustrates this decreasing attention level. It can be seen that up to a reaction time, defined

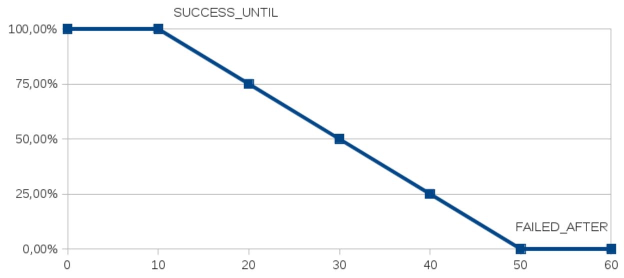



Figure 5.19: The attention level decreases with the growing reaction time.

by the parameter *SUCCESS_UNTIL*, an attention level of 100% is returned. If the reaction time is longer than the value provided by *FAILED_AFTER*, the attention level is set to 0%. Between these two points, the attention level decreases in a linear way.

In detail, the attention-level of the $j$-th interaction of the $i$-th interaction-method is calculated by the following equation. For that, it has to be assumed that $t_{ij}$ is the corresponding reaction-time:

$$f(t_{ij}) = \begin{cases} 100 & \text{if } t_{ij} \leq \text{SUCCESS\_UNTIL} \\ 0 & \text{if } t_{ij} > \text{FAILED\_AFTER} \\ g(t_{ij}) & \text{else} \end{cases} \tag{5.2}$$

Where $g(t_{ij})$ is:

$$g(t_{ij}) = 100 - \left( \frac{t_{ij} - \text{SUCCESS\_UNTIL}}{\text{FAILED\_AFTER} - \text{SUCCESS\_UNTIL}} * 100 \right) \tag{5.3}$$

With these two equations, the attention level of a single interaction is computed. Now, the attention levels of all of the interactions of an interaction method have to be grouped to a single attention level. For that, the following equation forms the mean over all of them, where $a_i$ is the attention level of the $i$-th interaction-method and $m_i$ the number of its interactions:

$$a_i = \frac{\sum_{j=0}^{m_i} f(t_{ij})}{m_i} \tag{5.4}$$

Now, the calculation of the attention level of an interaction method is finished. It can now become part of the overall attention level as explained above.

## 5.6 Interaction Methods

Now, with all parts of the platform and all of the interfaces explained, it is possible to present the different methods of interaction. As mentioned earlier, each interaction method is implemented as an independent plugin. Furthermore, each plugin has to be of a given type.

Overview The first type consists of interaction methods presenting their interactions in a random and automatic way. Of this type, there are two interaction methods:

- **Simple Questions**
  Simple questions which are not related to the content of the event, are asked to the students.
- **Solve CAPTCHA**
  A CAPTCHA is shown to the students.

To indicate interaction methods which are allowing the users to invoke interactions, there are two types. The first one represents the interactions triggered by the students.

- **Ask Teacher**
  The students are able to ask questions to the teacher during the event.
- **Set Attention**
  Students are offered a slider to indicate how attentive they are.
- **Report Problem**
  A dialog is provided to report a technical problem to the teacher.

The second type of user triggered interactions consists of the interactions started by the teacher. Currently, there is only one such interaction method:

- **Ask Students**
  The teacher could send questions to the students at live-broadcastings.

Finally, there is a type representing the interactions which are occurring at planned positions during the event. Based on this type, two interaction methods are implemented:

- **Planned Multiple-Choice Questions**
  Multiple-choice questions are presented to the students at fixed positions.
- **Planned Text Questions**
  In a very similar way text questions are shown to the students at planned positions.

### 5.6.1 Simple Questions

The first method of interaction is of the type 1. This means that its interactions are presented to the students completely automatic and at random positions according to the scheduling algorithm presented above. As suggested by the name, this interaction method shows simple questions to the students. It is defined by the corresponding platform-setting that these interactions are triggered seven times per hour. `Type 1: Automatic`

The example presented by Figure 5.20 shows such a question. It could be seen that it is not related to the content of the event. The questions are selected from a pool of questions in a random way. To answer the question, the students are required to click a single button. This indicates, that the students are not forced to think about an answer. The task of answering consists only of clicking the only available button. Because of that, this interaction method has the purpose to keep the students busy in order to support their attention. `Showing Question`

This interaction method is part of the analysis of the attention. Because of `Analysis`

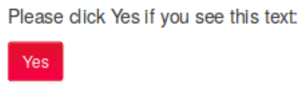



Figure 5.20: Simple questions are shown to the students.

that, it implements the corresponding interfaces. It seems to be obvious that, the criteria to measure the attention is the reaction time. To compute the attention level based on the reaction time, the offered function (see Section 5.5.2) is used. For that, the following platform-settings are provided:

- *(LIVE)_SUCCESS_UNTIL*: 20 seconds
- *(LIVE)_FAILED_AFTER*: 65 seconds
- *MAX_LOOKBACK*: 600 seconds

The first parameter states that the first 20 seconds of the reaction time still leads to a level of attention of 100%. In contrast to that, the second parameter states that after a reaction time of 65 seconds the attention level is set to 0%. For the calculation of the attention level during the event, the interactions of the last 10 minutes are considered. For that, the last parameter is set to 600 seconds.

### 5.6.2 Solve Captcha

`Type 1: Automatic`

The second method of interaction of type 1 is very similar to the one presented by the previous section. The major difference is that it displays a CAPTCHA to the students. According to the type, this happens completely automatic and random. To indicate how often such an interaction should be displayed, the corresponding platform-setting is used. It defines that a CAPTCHA is shown seven times per hour.

`Showing CAPTCHA`

An example of such an interaction is shown by Figure 5.21. It can be seen that a very simple version of a CAPTCHA is used. It consists of 4 letters and uses some dots as background noise. From this simplicity it could be derived that the reason of the CAPTCHA is not the blocking of robots. The

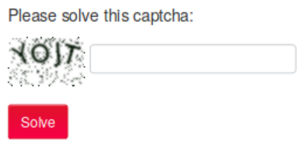



Figure 5.21: A simple CAPTCHA is displayed to the students.

main purpose could be summarized to support the attention of the students by providing them with something to do.

Analysis

As the previous interaction method, this one is also part of the analysis which calculates a level of attention. For that, the time is measured which was required by the students to solve the CAPTCHA. Because of that, the function for a time based calculation provided by the library is used and configured with the following parameters:

- *(LIVE)_SUCCESS_UNTIL*: 20 seconds
- *(LIVE)_FAILED_AFTER*: 85 seconds
- *MAX_LOOKBACK*: 600 seconds

It can be seen that the only difference to the simple questions is that the students are granted more time to solve the CAPTCHA by setting the second parameter to 85 seconds. This is done because a CAPTCHA is more difficult to solve than the simple click on a button.

### 5.6.3 Ask Students

Type 3: Teacher Triggered

This is the only method of interaction of type 3 which enables the teacher to ask questions to the students in real time. It seems to be obvious, that this is only possible at live-broadcastings because the teacher has to be present to ask the question which is not the case at videos. To achieve these functionalities, the interaction method provides the following components:

- models to save the questions and answers
- a template to provide the dialog to ask a question
- interactions to show the questions and the answers to the students and the teacher

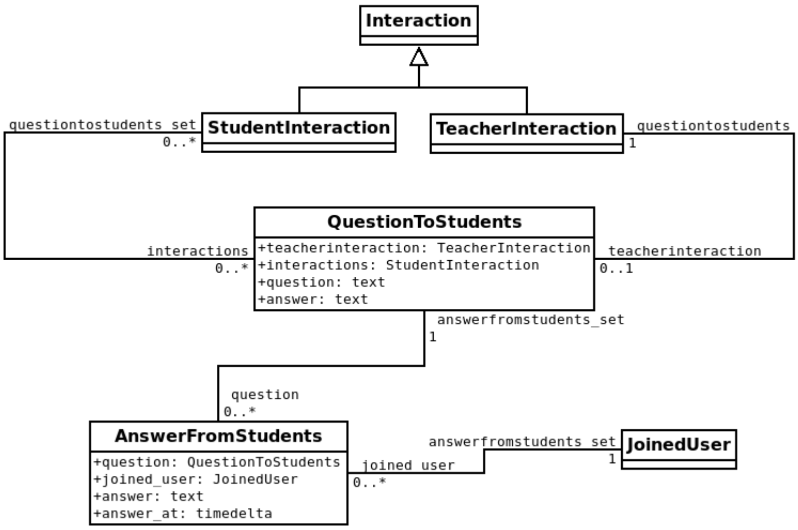



Figure 5.22: The models of the interaction method to ask real time questions to the students.

`Models` It can be seen by Figure 5.22 that the model *QuestionToStudents* is the central element. It saves a question which is asked by the teacher to the students. In addition, it holds the reference answer given by the teacher. To display the question to the students, a *StudentInteraction* is created for each joined student. These interaction models are related to the model holding the question. To present the list of the answers of the students, an interaction is created for the teacher too. This instance of *TeacherInteraction* is also related to the question model. The mentioned models of the interactions are the same as explained by Section 5.4.3.

To save the given answers of the students to the questions from the teacher, there is a model called *AnswerFromStudents*. Obviously, it has a relation to the answered question and to the joined user known as the student who answered the question. In addition, it saves the text of the answer and the position in the live-broadcasting where the question is answered.

`Creating and Showing Questions` To ask a question to the students, the teacher is required to enter it in a text box provided by the interaction method in the bottom right corner of the user interface which is shown to the teacher during a live-broadcasting (see Section 5.3.4). After submitting the entered question, interactions are created. On the one hand, instances of the model *StudentInteraction* are used to show

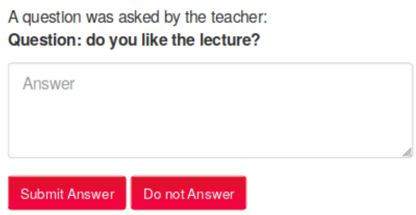



Figure 5.23: The question asked by the teacher is presented to the students.

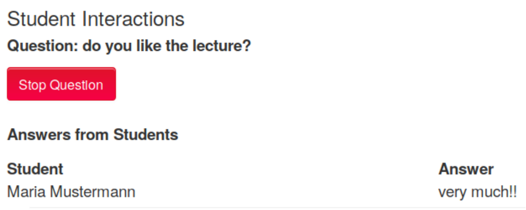



Figure 5.24: The answers of the students are listed for the teacher.

the question to the students. This results in the dialog, which is shown by Figure 5.23. It can be seen that the question is shown in combination with a dialog to answer it.

`Students' Answers`

On the other hand, a *TeacherInteraction* is created to load a view at the user interface of the teacher. This is done to present the answers to her/him. As shown by Figure 5.24, this is done by displaying the question and a list with the answers. The list consists of two columns where the first one holds the name of the answering student and the second one the given answer. In addition to this list, there is a button to stop the questioning round.

`Reference Answer`

Clicking this button redirects the teacher to a further view as presented by Figure 5.25. It can be seen that again the question is printed. In addition to that, a dialog is shown which could be used to submit a reference answer. The teacher could use this dialog to provide feedback and the correct version of the answer to the students. This is done by using a *StudentInteraction* for each watching student. The submitted reference answer is displayed to the students using the interaction (see Figure 5.26). As pointed out by the figure, the question is shown together with the reference answer. Furthermore, the students are required to acknowledge this interaction by clicking the offered

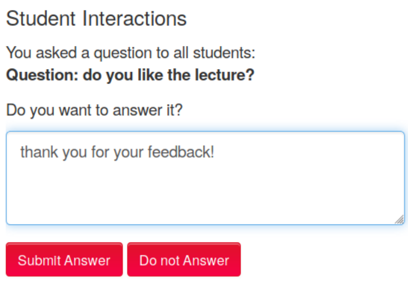



Figure 5.25: The teacher can provide an answer to his question as a reference.

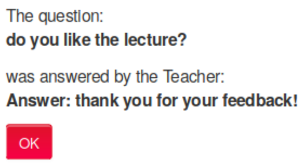



Figure 5.26: The reference answer is shown to the students.

button.

`Analysis` To calculate the attention level of the interactions of this interaction method, the provided library function is used. This means that the calculation is based on the reaction times of the students to the interactions. In contrast to the first two interaction methods, the boundaries set by the following parameters, allowing a much longer reaction time:

- *(LIVE)_SUCCESS_UNTIL*: 45 seconds
- *(LIVE)_FAILED_AFTER*: 110 seconds
- *MAX_LOOKBACK*: 600 seconds

### 5.6.4 Ask Teacher

`Type 2: Student Triggered` The opposite version of the previous interaction method is implemented by this one which is of type 2. This means that with this method of interaction the students are able to ask a question to the teacher. For that, there is a model to represent the question and some dialogues to handle the processes of asking and answering.

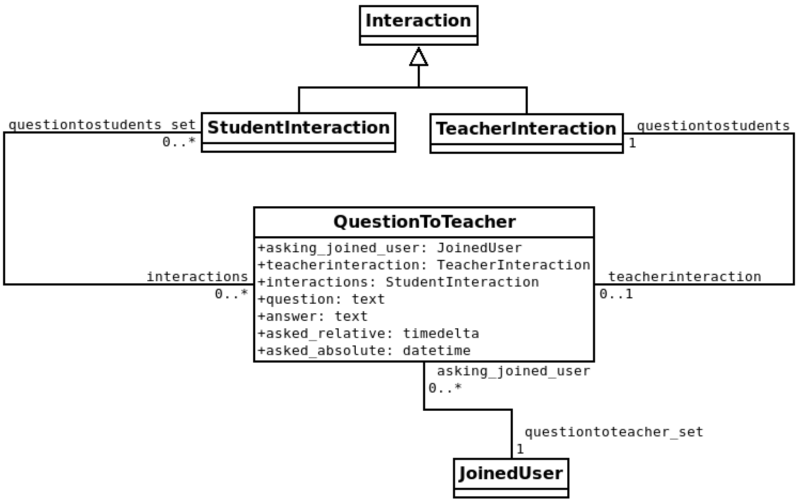



Figure 5.27: The models of the interaction method to ask real time questions to the teacher.

As indicated by Figure 5.27, in addition to the models of the framework, there is only one model used for this interaction method (*QuestionToTeacher*). It seems to be obvious, that it has a relation to both, the asking joined user and the *TeacherInteraction*. The first points out which student submitted the question. In contrast to that, the second one is the interaction which is used to deliver the question to the teacher in the case of a live-broadcasting. In addition to these two relations, there is a third relation to instances of *StudentInteraction* because an asked question is not only presented to the teacher, it is shown to all other students too. The asked question and the answer of the teacher is stored by the corresponding fields. Finally, there are fields to save the absolute and relative date and time of asking the question for statistical reasons. Models

The students are enabled to ask a question to the teacher by using offered control elements. This means that in the sidebar on the right hand side of the video or the live-broadcasting a text box is visible which could be used to ask the question (see Figure 5.28). Asking Question

When the student submits a question, two possible scenarios are arising. The first one is, that the event transports a video. In this case no teacher is available to answer the question instantly. Because of that, no according interaction is created. The question is simply presented to the teacher at the

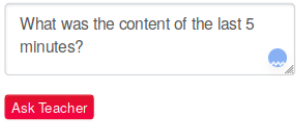



Figure 5.28: Students could enter a question to the teacher in the sidebar.

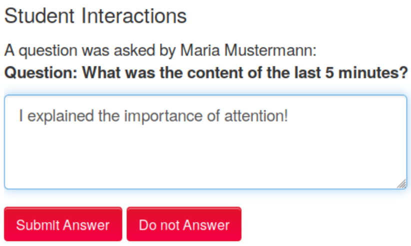



Figure 5.29: The question from the student could be answered by the teacher.

backend of the video where she/he has the possibility to answer it in an asynchronous way. For that, a detail site for the teacher (see Section 5.4.5) is provided.

`Question is shown to Teacher`

The second scenario resembles a live-broadcasting. Here it is possible to present the question to the teacher immediately. In addition, there are typically other students simultaneously watching. Because of that, the question is presented to them as well. For both recipients the according interactions are created. From a teachers point of view, the asked question is presented to him/her by the interaction shown by Figure 5.29. It can be seen that the question and the name of the asking student is printed. Furthermore, a dialog to answer the question is provided.

`Answer of the Teacher`

If the teacher submits an answer using the presented dialog at a live-broadcasting it is instantly displayed to the students. For that, an according interaction is used which will result in the view presented by Figure 5.30. It simply displays the name of the asking student and the question in combination with the answer of the teacher. In contrast to that, the answer to the question could not be displayed to the students using interactions at a video. Due to that, a detail site for students is implemented which lists the asked questions and the answer of the teacher if she/he submitted one.

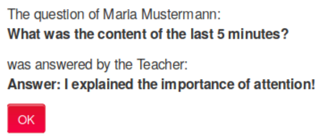



Figure 5.30: The answer of the teacher is displayed to the students.

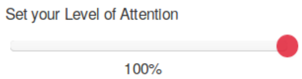



Figure 5.31: A slider to set the current level of attention is located in the sidebar.

Due to the fact that this interaction method is very similar to the previous one, it uses the same method for calculating the attention level. For that, the following parameters are provided again: Analysis

- *(LIVE)_SUCCESS_UNTIL*: 45 seconds
- *(LIVE)_FAILED_AFTER*: 110 seconds
- *MAX_LOOKBACK*: 600 seconds

### 5.6.5 Set Attention

In contrast to the interaction methods presented above, this one consists not of questions and answers. It provides the possibility to students to set their current level of attention. This means that students are able to express how attentive they are at the current moment. For that, an according control element is located in the side bar at the right hand side of the video or the live-broadcasting. As pointed out by Figure 5.31, the students could use a slider to define the mentioned level of attention. Because the attention level is measured with a value in percent, this mechanism is used for the values of the slider too. Type 2: Student Triggered

To save the set values of attention, a model is used which is named *SetAttention* (see Figure 5.32). It has a relation to the joined user and to the current active watching history. In addition, it saves the submitted attention level and the date and time when the value is set. Models

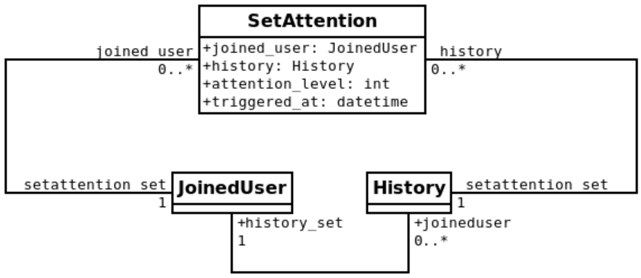



Figure 5.32: The models of the interaction method to set the current level attention.

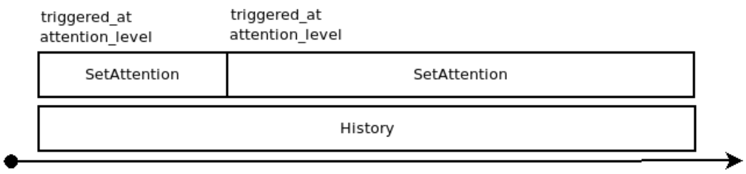



Figure 5.33: The attention level is calculated according to the set values.

`Analysis` It seems to be obvious, that this interaction method could not use the calculation of the attention level based on the reaction times to interactions because no such interactions are occurring. Instead of that, the attention level is calculated using the set values as shown by Figure 5.33. It can be seen that, a saved value of attention starts at the beginning of a watching history. This value is set automatically to the value of the last set value which is part of the previous watching history. If the student sets a new value using the slider, this set value is active for the rest of the history or until the student changes the value again.

To actually compute the attention level the following equation is used:

$$attention = \left( \sum_{i=0}^{n-1} a_i * \frac{t_{i+1} - t_i}{l} \right) + a_n * \frac{he - t_n}{l} \qquad (5.5)$$

It can be seen that the attention level of this interaction method is the sum of all set values. For that, the first part in the bracket is the sum of all this values excluding the last one. Each element ($a_i$) of the sum is weighted with its length relative to the length of the complete history. The weighting factor is computed by calculating the difference between the time stamp of

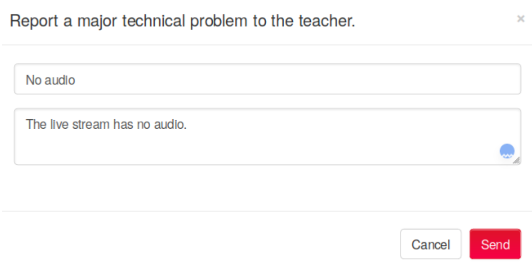



Figure 5.34: A dialog to report a technical problem to the teacher.

the next ($t_{i+1}$) and the current ($t_i$) set value. This difference is divided by the length ($l$) of the history. To cover the last set attention level, the same weighting is done where the length is computed using the time stamp of the end ($he$) of the history.

### 5.6.6 Report a Technical Problem

This interaction method offers the possibility to report a (technical) problem to the teacher. It is valid to state that this method of interaction is mainly used by the students at live-broadcastings to report issues with the live stream.

`Type 2: Student Triggered`

From a technical point of view, this interaction method provides no own models. This is the case because the interaction framework already offers a model named *ErrorToTeacher* as explained by Section 5.4.3. With this model it is possible to bring errors to the teacher in a prominent way.

To report such a problem, the students are required to click a button which is placed in the side bar at the right hand side of the video or the live-broadcasting. Clicking this button displays the dialog presented by Figure 5.34. It can be seen that the students are forced to enter a subject for the error. In addition, a meaningful description of the problem has to be provided.

`Report an Error`

After submitting the problem report, the error is shown to the teacher in two different ways depending on the mode of the event. In the case of a video, the error is displayed to the teacher at the backend in a list of all errors. In contrast to that, at a live-broadcasting the error is shown to the teacher

`Error shown to the Teacher`

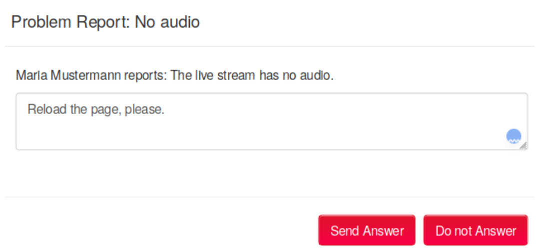



Figure 5.35: The teacher could answer to the problem report.

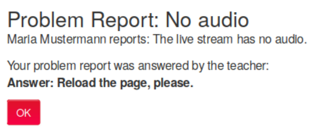



Figure 5.36: The answer from the teacher to the problem report is shown to the student.

instantly using a pop-up dialog as shown by Figure 5.35. This dialog prints the subject and the description of the error in combination with the name of the reporting student. Furthermore, a text box is shown which could be used by the teacher to provide a reaction to the error.

`Reaction of the Teacher`

If the teacher submits a reaction to the error, it is displayed to the asking student. For that, a *StudentInteraction* is used. This results in the interaction, shown by Figure 5.36. It reiterates the reported problem and displays the reaction of the teacher in a highlighted form. Below that, a button is displayed which forces the student to acknowledge the interaction.

It seems to be obvious that this method of interaction is not part of the calculation of the attention level. This is done because of two reasons. First, the error reporting is not used very frequently in general. Second, only one interaction is used to display the reaction of the teacher. This sole interaction is not a very resourceful representation of the attention of the student because it is only used if something went wrong.

### 5.6.7 Planned Multiple-Choice Questions

`Type 4: Planned`

The interaction methods of type 4 are providing interactions at planned positions during the event. This interaction method is of this type and because of that, it offers multiple-choice questions at planned positions. For that, it consists of the following parts:

- the models
- a dialog to plan the questions
- interactions to show them
- detailed analysis of the answers of the students (see Section 6.1.4)

Models

The models which are required to handle the multiple-choice questions are shown by Figure 5.37. When the teacher plans the questions, a new instance of *PlannedQuestionToStudents* will be created in combination with an instance of the related model *PlannedStudentInteraction*. The latter one is explained by the previous section regarding the interaction framework. The first one holds the text of the question and an optional image. In addition, it states if the correct answer should be displayed to the students after answering the question. In the case of a live-broadcasting a relation to an instance of a *TeacherInteraction* is created. This is done to display the answers of the students instantly to the teacher. Because a multiple-choice question consists of several possible answers, a corresponding model (*PossibleAnswer*) is used. This model additionally states if the possible answer is a correct one. The creation of such possible answers leads to a set which is held by the model saving the question.

When a student joins to an event all instances of the model *PlannedStudentInteraction* are translated to real instances of *StudentInteraction*. This mechanism is explained in detail by Section 5.4.4. With the creation of a real interaction, a multiple-choice question could be displayed to the students. If a student submits an answer by selecting one or more of the possible answers an instance of *ChoiceOfStudent* is created for each of them. This model holds a relation to the selected possible answer and the real interaction showing the question. Because an interaction could be shown multiple times in the case of a video, each showing is represented by the model *CallHistory*. To bound the selected answers to a specific displaying of the interaction, the model *ChoiceOfStudent* additionally has a relation to the model representing the call.

Question Planning

To plan the questions, a dialog is implemented. For that, a special view

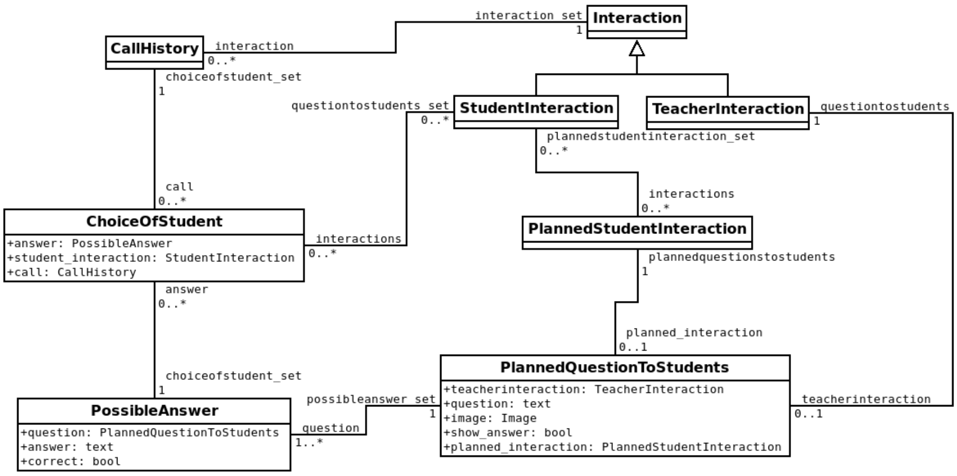



Figure 5.37: The models of the interaction method to ask mulitple-choice questions at pre-defined positions to the students.

is provided and registered at the platform using the according platform-setting named *INITIAL_SETTINGS_VIEW* (see Section 5.4.1). This view is shown by Figure 5.38. It can be seen that at the top, the video is presented. In the case of a live-broadcasting, a slider is printed instead of the video. With that, it is possible to select a position for the question. Below that, a field is offered where the teacher could enter the position manually. In both cases the selection of a position leads to a further dialog which asks for the question and the related possible answers.

After these control elements to create a question, a list of the created ones is printed. This list states the created question, the position and the number of possible answers. It additionally provides the possibility to edit or to delete a question. Below this list, there is a button to mark the questions as ready. Clicking this button indicates that the creation of the question is finished and that they are ready to be displayed to the students. After marking the questions as ready, it is no longer possible to edit them. This is done to ensure that all students are receiving the same questions.

Showing Questions

Now the multiple-choice questions will be shown to the students. For that, the created interactions are used. When the position of a question is reached, the interaction shows the question as presented by Figure 5.39. At first, the question is printed and below that, the optional image is shown. This

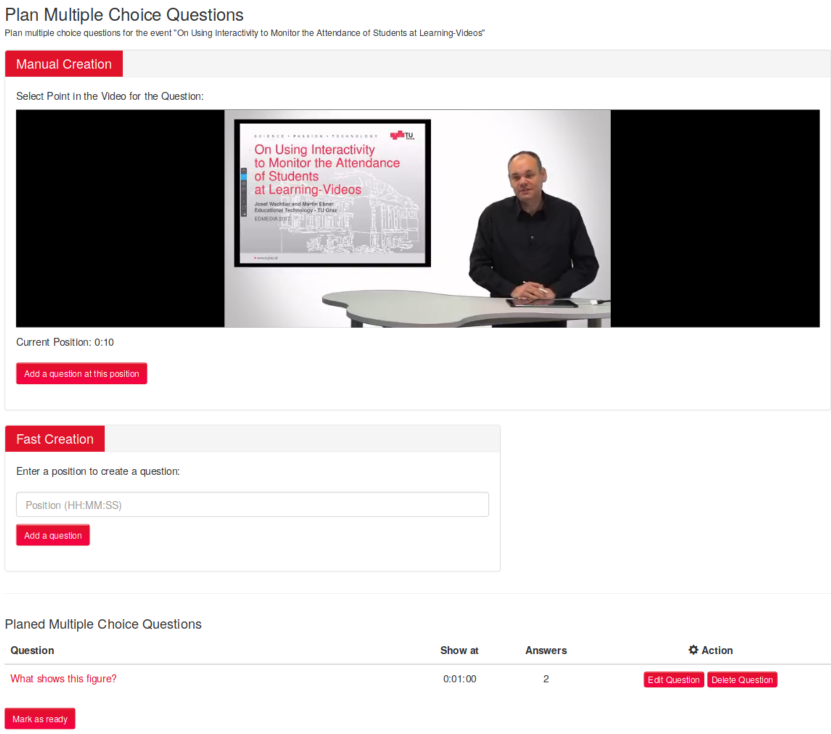



Figure 5.38: The dialog to plan the multiple-choice questions.

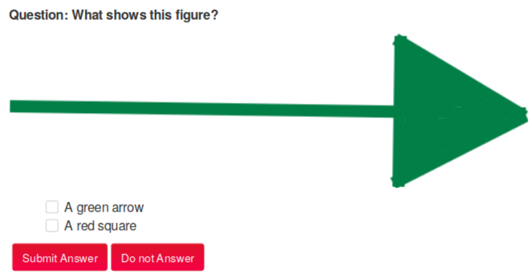



Figure 5.39: A planned multiple-choice question is shown to the students.

is followed by a list of the possible answers. The students are required to select one or more such answers by marking the offered check boxes. After submitting the answer, the correct one is printed if the teacher enabled this feature during the creation of the question.

`Analysis` Because this interaction method is presenting questions which require an answer, it is part of the calculation of the attention level. For that, the reaction time based method provided by the library is used with the following parameters:

- *(LIVE)_SUCCESS_UNTIL*: 45 seconds
- *(LIVE)_FAILED_AFTER*: 110 seconds
- *MAX_LOOKBACK*: 600 seconds

### 5.6.8 Planned Text Questions

`Type 4: Planned` In contrast to the previous interaction method which presents multiple-choice questions to the students at planned positions during the event, this one does the same with text-based questions. For that, it provides the required models and the dialogues to create and to answer the questions. The implemented features of analysis are presented by Section 6.1.5.

`Models` Because of the mentioned similarity to the interaction method presented by the previous section the models shown by Figure 5.40 are very similar. The main model is named *PlannedTextQuestionToStudents* and represents a question planned by the teacher. It saves the question and a supporting

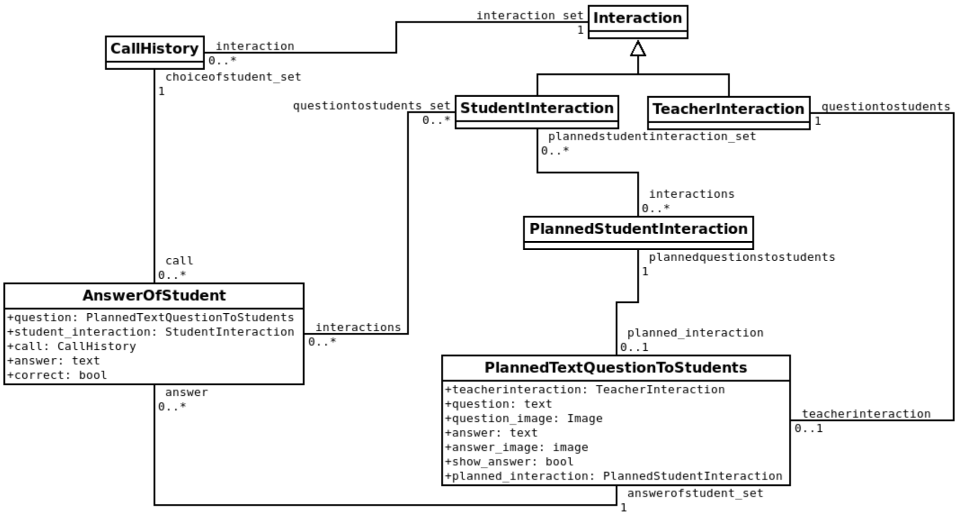



Figure 5.40: The models of the interaction method to ask text-based questions at pre-defined positions to the students.

image which could be added optionally. This model saves the reference answer of the teacher and an optional image for that answer too. To indicate if the reference answer of the teacher should be shown to the students after answering the question, the corresponding field has to be set to "true". In the case of a live-broadcasting, an instance of the model *TeacherInteraction* is created in relation to the model holding the question. This is done because the interaction is used to show the answers of the students to the teacher instantly. In order to show the question to the students, the related model *PlannedStudentInteraction* ensures that real instances of the model *StudentInteraction* are created for the question when the students join to the event containing the question.

When a student finally submits an answer the model *AnswerOfStudent* is used to save it. For that, it has a relation to the question which is answered. In addition, it holds a relation to the interaction presenting the question to the students to indicate which student submitted the answer. Because a question could be presented more often than once in the case of a video, the history of each calling of the interaction is recorded by the model *CallHistory*. There is a relation between the model saving the answer and the one representing the call of the interaction because an answer is always

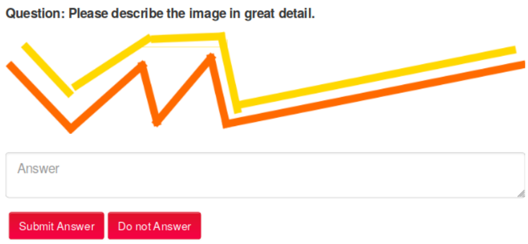



Figure 5.41: A planned text-based question is shown to the students.

unique to a single call. It seems to be obvious, that the model additionally saves the answer of the student. To indicate the correctness of the answer, the corresponding field has to be manually set by the teacher because it is not possible to evaluate a text-based answer automatically.

`Question Planning` As mentioned, the teacher has to create the questions before the event. For that, a dialog is implemented in the same way as explained by the previous section about the multiple-choice questions. This dialog provides the possibility to select a position for the question and the dialog to enter the question and the reference answer. It additionally lists all created questions with the possibility to edit or to delete it. Finally, a button is provided to mark the question as ready. After clicking this button, the questions are locked and will be shown to the students.

`Showing Questions` To actually show a question to the students the corresponding interaction is used. Because of that, a view is implemented to display the question as an interaction. As shown by Figure 5.41, the question is presented to the students in combination with the image if the teacher uploaded one. Below that, a text box is located to enter the answer. In the case of a live-broadcasting, the given answer will be shown to the teacher in real time. In contrast to that, the answer is listed at the backend for the teacher at a video. In both cases the teacher is required to manually assign the correctness of the answers (see Section 6.1.5).

`Analysis` In order to evaluate the attention of the students this interaction method is also part of the calculation of the attention level. It uses the provided function which implements the reaction time based approach. For that, the following parameters are set:

- *(LIVE)_SUCCESS_UNTIL*: 45 seconds
- *(LIVE)_FAILED_AFTER*: 110 seconds
- *MAX_LOOKBACK*: 600 seconds

## 5.7 Post-Event Polls

Until now, only methods of interaction which are occurring during the event were discussed. To provide interactivity also after the events, the concept of post-event polls is introduced. This feature enables the teacher to place a survey immediately after the students left the event. This feature is mainly used to ask questions regarding the satisfaction of the students with the content of the event.

To save all the data required to handle the polls, the models shown by Figure 5.42 are used. A poll is represented by the model *PostEventPoll*. It is clear that it is required to have a relation to an event and the corresponding mode. This means that the poll belongs to a specific event and to its video or its live-broadcasting. The field *show_after* holds a percentage value indicating how much of the event the student has to watch, in order to receive the poll on leaving. It is important to use this parameter depending on the content of the question presented by the poll. This means that if the poll asks a question regarding to the whole video, a value near 100% is recommended to ensure that the students viewed most of the video before answering the poll. Additional fields are saving the date and time of the creation of the poll and if it is currently active. Finally, the text of the question is saved. It could be supported by an optional image and an optional text. If such an optional text is provided, the students are required to enter a text-based answer to the poll in addition to the selection of one of the possible answers. `Models`

These possible answers are saved by the model *PEPPossibleAnswer*. This means that each poll holds a set of such possible answers representing the answers which could be selected by the students. The actual answers of the students are saved by the model *PEPAnswer*. It has a relation to the corresponding question and to the selected possible answer. Additionally, it holds the text entered by the student if such a text-based answer is requested by the poll. To state the date and time of answering to the poll the

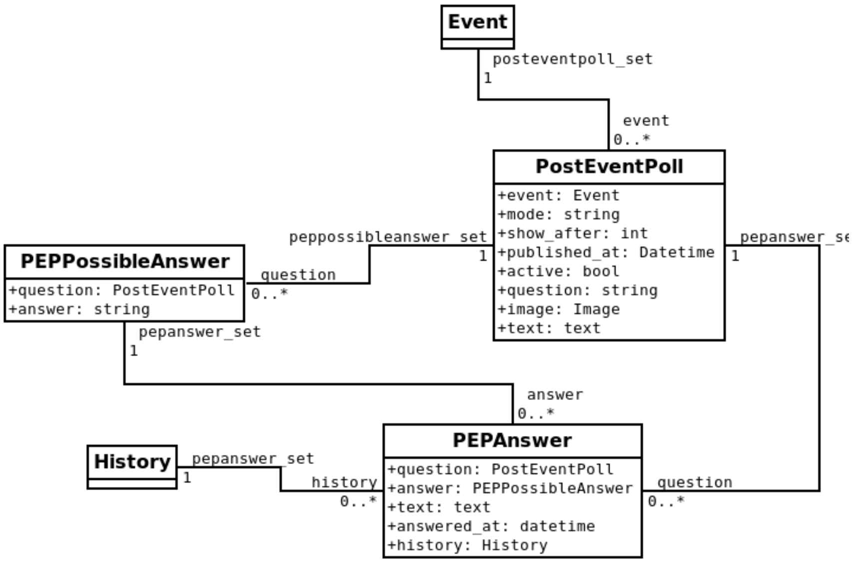



Figure 5.42: The models of the post event polls.

corresponding field is used. Finally, there is a relation to a history model which is explained earlier. This is done because of two reasons. First, this relation leads to the answering joined user. Second, it indicates the watched timespan after which the poll was shown.

`Poll Creation` Based on this presented way to save the polls, the teacher has to use a dialog to create a poll (see Figure 5.43). It can be seen that this dialog is related to the model representing a poll. At first, the question of the poll has to be entered. It should somehow indicate that the answer has to be selected from a list. After that, it is possible to upload an image which will be shown when the poll is presented to the students. The following text box holds the text which could be used to encourage the students to provide an additional text-based answer. Finally, it has to be stated how much of the event the student has to watch in order to see the poll.

`Poll Activation` After submitting this form, a further dialog asks for the possible answers. At this dialog, the teacher is required to enter all of them. If a poll is created, it is still not active and listed by the corresponding table (see Figure 5.44). At this stage, it is still possible to edit or to delete the poll. If everything is correct, the teacher has to activate it. For that, the offered button has to be

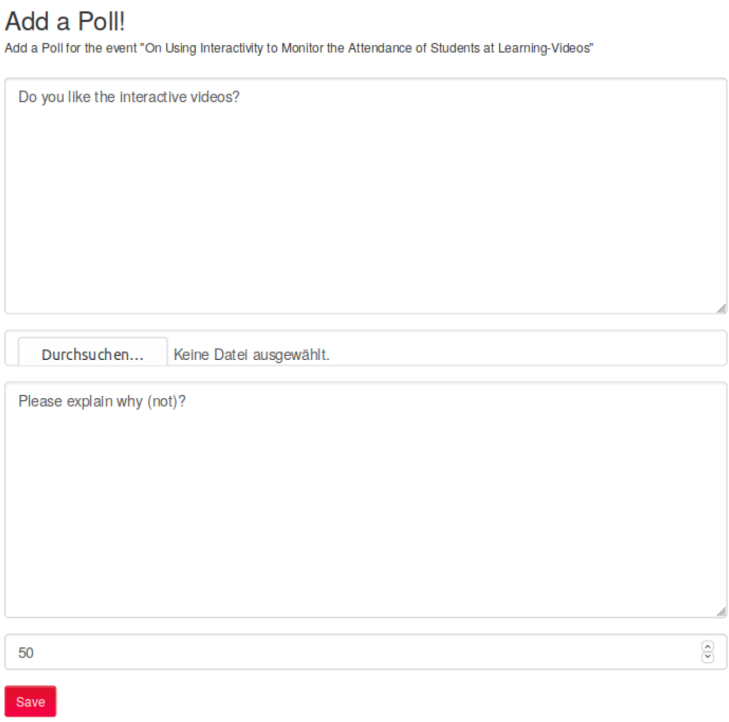



Figure 5.43: The dialog to create a poll.

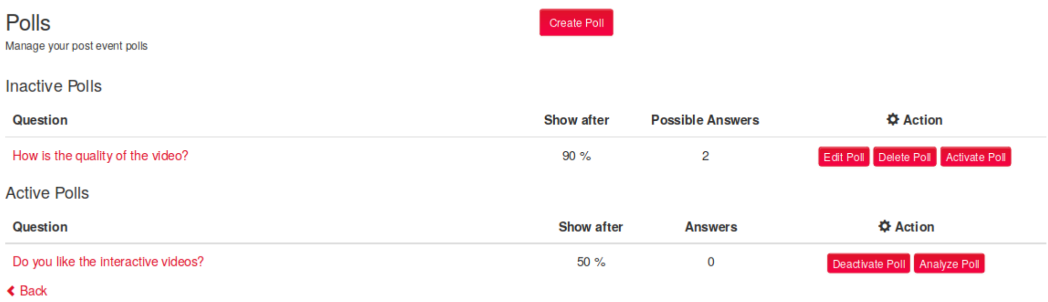



Figure 5.44: The overview for the teacher of the polls.

used. This moves the poll to the list of active ones.

Now, the poll will be displayed to the students when they leave the event and watched enough of the video according to the specified value. This looks like as shown by Figure 5.45. It can be seen that the question of the poll is printed along with the possible answers. Here, the student has to select one of them. Below that, the text which prompts the students for an additional text-based answer is shown together with a text box to enter such an answer.

All answers are presented to the teacher in a detailed form. This evaluation of the students is explained by Section 6.1.6.

## 5.8 LTI Provider

To connect the platform providing the interactive components to other learning platforms, an LTI provider is implemented. Such a provider offers the following functionalities: (Wachtler, Scherz, & Ebner, 2019)

- registration and authentication of the students at LIVE
- redirecting to the requested content.

Vocabulary

At first, this section explains the basic concept of LTI and after that, the implementation is presented. For that, it is required to define some vocabulary. The active component is a learning platform which is called LTI

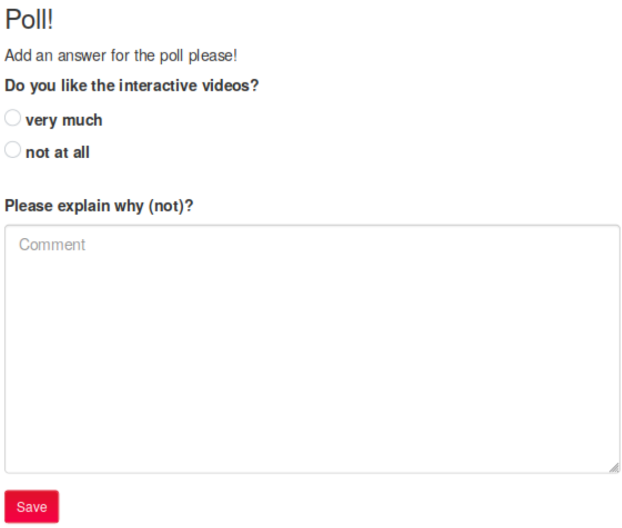



Figure 5.45: A poll is presented to a student.

consumer. Such a platform provides the content and materials of a course to the students. It is common that the students are registered and authenticated at such platforms. In contrast to that, the learning tool is named LTI provider. This provider is offering content for the students using the learning platform. In the case of this work the LTI provider is the tool named LIVE.

Specification

The general principle of LTI is illustrated by Figure 5.46. It can be seen that the students are required to register and to authenticate at the LTI consumer which is implemented by a learning platform. When a student requests a content offered by a learning tool through an LTI provider, the learning platform automatically authenticates the student at the learning tool. From a technical point of view, the account details which represent a user are sent to the learning tool in combination with a signature. These details consist of username, given name and family name as well as the e-mail address. Further parameters are possible to state the requested content. To verify the correctness of the sent data, a signature is computed using a shared secret. For that, the mechanisms of OAuth[52] are used. This signature is transmitted to the learning tool too. Now, the LTI provider has to compute the signature

[52]short for Open Authorization

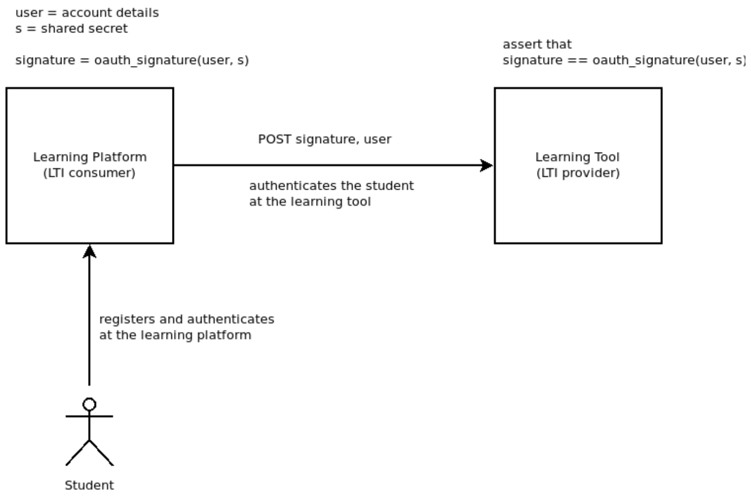



Figure 5.46: A student is authenticated at the learning tool through the learning platform using LTI.

by itself and to compare it with the received one. If they are the same, the authentication request is valid and the requested content is displayed to the student.

`Library` The implementation of the LTI provider is separated from LIVE. As stated by Section 5.2.1, a library which provides the LTI provider is used. This library is implemented to add such a provider to projects based on Django in a simple but highly configurable way. Because this library could be used for any project, it has been released under the terms of the MIT license[53].

The mentioned library uses a further library[54] which provides the basic LTI components. The missing parts are implemented to be a complete LTI provider. This includes the following components:

- models to save the data
- request validator
- authentication backend
- redirection logic

`Models` The basic part of the implemented library consists of some models (see

[53] https://opensource.org/licenses/MIT, last accessed October 30, 2018
[54] https://github.com/pylti/lti, last accessed October 30, 2018

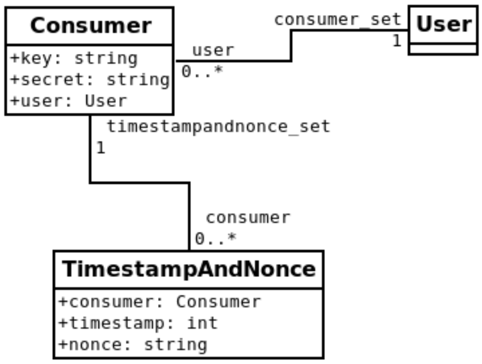



Figure 5.47: The models required by the LTI provider.

Figure 5.47). The central model is called *Consumer* which represents a learning platform which wants to use the LTI provider. It holds a key to identify the consumer and the related secret which is used to sign the authentication request. In addition, this model has a relation to the model *User* to state the teacher responsible for the consumer. On each authentication request a timestamp and a nonce is transmitted. This is done to ensure that a single request is only valid once. For that, the submitted timestamp and the related nonce is saved by the model *TimestampAndNonce* which additionally has a relation to the requesting consumer.

Figure 5.48 shows the validator which is required to check the signature of the authentication request. As mentioned above, the mechanisms of the OAuth library are used. For that, the interface called *RequestValidator* is implemented. Because only a limited subset of the OAuth protocol is required, only the corresponding parts are implemented. This is done by the class *LTIValidator*. The member variables of this class are setting some basic parameters. The first method returns the secret of a consumer identified by its key. For that, the corresponding model is queried. The second method validates the key of a consumer. This is done by checking if the key exists in the database represented by the models. The final method is responsible for validating a combination of a timestamp and its nonce. This validation is also done by querying the database. If the timestamp and nonce, which should be validated, exist they are invalid because such a combination is only valid once. Due to that, this method also saves the provided values to ensure that they are invalid, if it is tried to validate them again. `Validator`

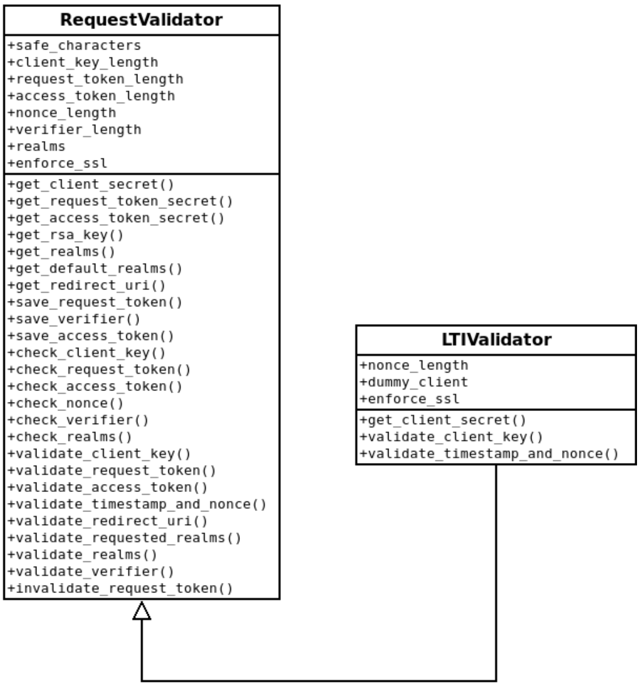



Figure 5.48: The validator which checks the LTI request.

With the presented models and the validator it is possible to verify the correctness of an authentication request. To actually perform the authentication, it is required to provide a backend for Django which handles the authentication of the users through an LTI request from a consumer. This backend uses the validator and the OAuth library to validate the signature with the shared secret. Furthermore, a Django user is created from the transmitted parameters (name, e-mail, ...). `Authentication`

The final step after authenticating the user using the backend is the redirection to the requested content. This is done by evaluating the transmitted custom parameters. This means that a set of parameters corresponds to a specific view. To configure this behavior, some application-settings are required. These settings are defining a relation from a set of parameters identified by name to a view which takes the same parameters as arguments. Based on this configuration, the redirection is performed. `Redirection`

## 5.9 Data Download for Researchers

It seems to be obvious that the presented components of LIVE are generating a lot of data. Most of these data are presented to the teacher by different possibilities of learning analytics which are built-in (see Section 6.1). To enable further possibilities of evaluation, it is possible to download all of the data in the form of spread sheets. These possibilities include evaluations presented by Section 6.2.

To control the access to the downloads, the user management defines an own group for this purpose (see Section 5.3.2). This group is named "researchers" because the downloads are intended to be used for research purposes. Due to that, all the downloads are available in an anonymous form only.

Figure 5.49 illustrates the overview of all available downloads. It can be seen that at the left hand side the overall downloads are located. This area offers downloads of two different types. At first, there are some statistical values about the users and the registration. This includes the number of users of the different groups as well as a statistic showing the number of registrations grouped by the months. The second type of data of the overall `List of Downloads`

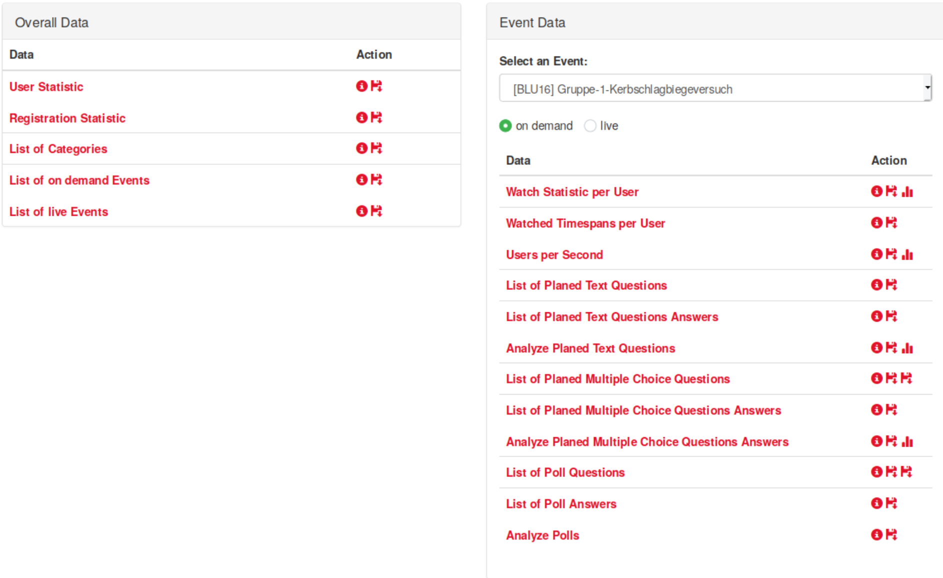



Figure 5.49: Researchers could download all of the data.

section consists of lists of the categories and the events in their mode. These lists are required because the other downloads which are related to a specific event are only using the ID of the event to identify it.

At the right hand side, the downloads which are related to a specific event in a specific mode are listed. At the top of this section, there is a drop down field to select the event of which the downloads are requested. Below that, radio fields are forcing the selection of the mode (on-demand or live). The first three elements in the list of the downloads are different evaluations of the watching history of the students. The other downloads are related to the interaction methods which are providing questions and to the post-event polls.

## 5.10 Admin Panel

As indicated by the previous sections, there is an admin panel. This panel could be accessed only by the users which have administrator privileges. It

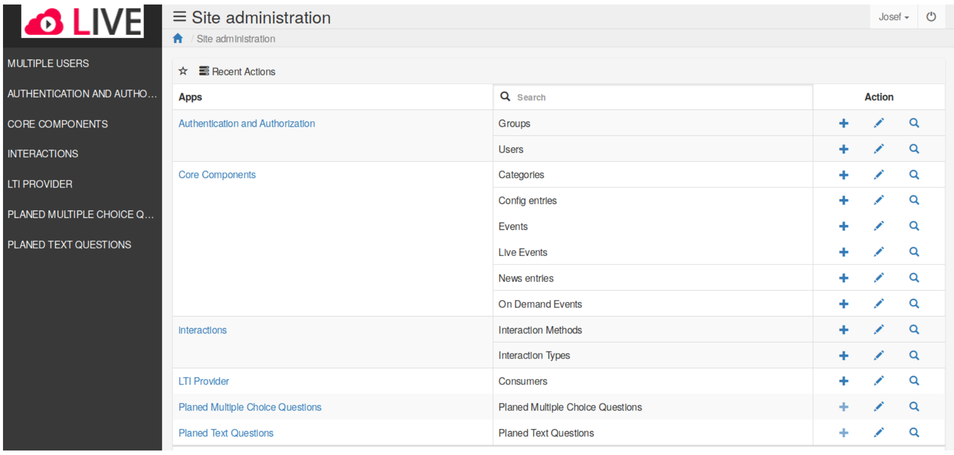



Figure 5.50: The main site of the admin panel.

provides access to the following components as shown by the main page of the admin panel (see Figure 5.50):

- management of the users and their privileges
- platform-settings
- LTI configuration
- edit all values of the
  - categories
  - events
  - concrete events
  - interaction types
  - interaction methods
  - planned multiple-choice questions
  - planned text questions

It seems to obvious, that the admin panel gives access to nearly all components which are required to run LIVE. Because of that, to access these sensitive information, it is required that the administrator privileges are distributed with caution.

# 6 Learning Analytics

It seems to be obvious, that the platform for learning videos or live-broadcastings, which are supported by interactive components presented by the previous section, generates a lot of data. It is possible to use these data to evaluate the performance of the students in detail. These evaluations could be divided in two categories: Overview

- built-in possibilities of evaluation
- external analysis of the data

The possibilities of the first category are implemented by LIVE and available to the teacher of an event. In contrast to that, the presented external possibilities of analysis could be performed by researchers using the data offered as spread sheets for downloading.

## 6.1 Built-in Possibilities

As mentioned above, the built-in possibilities of analysis are presented to the teacher of an event. As suggested by the name of the category, it is clear that these features of analysis are implemented by the platform itself. There are two different types of these features of analysis.

The first type includes possibilities of evaluation based on the values recorded by the models presented by Section 5.3.4. Currently, the following analysis features are available of this type: Watching History

- **Watched Timespans**
  All watching students are listed to the teacher together with their watched timespan.

- **Timeline**
  A diagram prints a timeline indicating how many students watched which part of the event.
- **History per user**
  For each student, it is illustrated when she/he watched which part of the event.

Interactive Components

In contrast to that, the second type of analysis features consists of the evaluations of the interactive components (see Sections 5.6 and 5.7):

- **Planned Multiple-Choice Questions**
  The results of the multiple-choice questions are listed.
- **Planned Text Questions**
  Lists of the answers to the text-based questions are printed.
- **Post-Event Polls**
  The results of the polls are presented to the teacher.

### 6.1.1 Watched Timespans

In order to see which students watched how much of the video or the live-broadcasting, a corresponding analysis feature is implemented by LIVE. For that, a list is generated which includes the watching students with the associated watched timespans.

List of Students

As shown by Figure 6.1, such a list consists of two columns. The first column prints the name of the joined user and after that, the username is shown in brackets. On clicking on a name, a detailed analysis of the student is loaded (see Section 6.1.3). In contrast to that, the second column states how much of the event the corresponding student watched. For that, a red bar is printed. This bar shows how much was watched. Below the bar the exact length of the watching is presented. After that, a value in percent indicates how much was watched in relation to the length of the whole event.

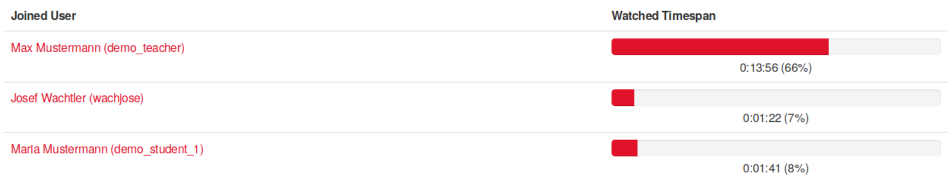



Figure 6.1: A list of all students is presented in combination with their complete watched timespans.

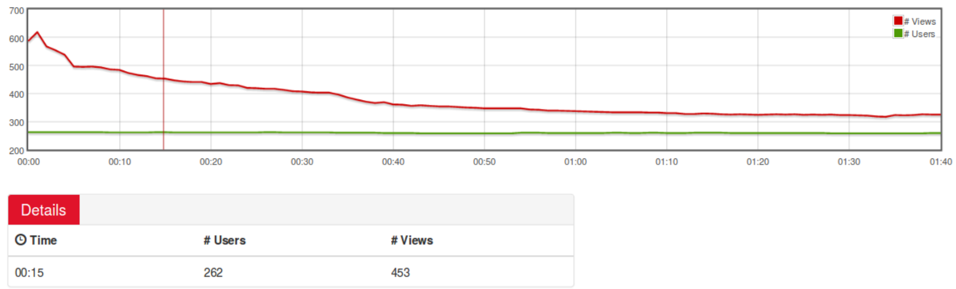



Figure 6.2: The timeline analysis states how many students watched which part of the event.

## 6.1.2 Timeline Analysis

To gain an overview which part of the video or the live-broadcasting was watched how many times, the so-called timeline analysis is available. It enables the teacher to indicate the parts of the event which were more interesting than others.

Timeline Diagram

Figure 6.2 shows this analysis. It can be seen that a diagram is printed. On the x-axis the timeline of the video is represented. In contrast to that, the y-axis states the number of students. The red line which is drawn from the left to the right, visualizes how often the event has been watched at each position. The green line shows how many different users are responsible for the watching. It seems to be obvious, that these two lines are only different in the case of a video because a live-broadcasting could only be watched once.

To see exact values, the teacher is required to move the vertical dark red cross-hair along the x-axis. This movement prints the exact values of the

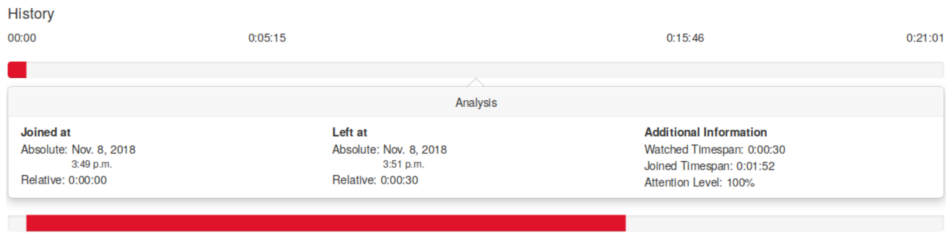



Figure 6.3: A detailed history of the watched parts is generated for each student.

selected position in the box below the diagram. The first column in the box states this position and the following two columns are printing the values. The first of these values is related to the green line stating the number of different users. In contrast to that, the second value represents the red line and because of that, it states the number of views.

### 6.1.3 History per User

As mentioned above, there is a detailed analysis of the watching history of each student. This analysis states in all details, when the corresponding student watched which part of the video or the live-broadcasting. Because this evaluation is related to a single student, it is not only shown to the teacher, it is additionally available to the student her-/himself.

`History Timeline` It can be seen by Figure 6.3, that a horizontal timeline of the event is printed. Below this timeline a red bar marks the watched part for each joined timespan. This means that the number of such red bars is equal to the number of watched timspans. On hovering such a bar with the mouse pointer, detailed information is shown. This includes the absolute and relative date and time of joining and leaving the event. In the third column of this details box, the length of the watched and the joined timespan is printed. The first value states how much of video content was watched. In contrast to that, the second value expresses how long the student needed for watching this part of the video. The values are different if an interaction happened during this watched timespan because such an interaction pauses the video. Below these two values regarding the timespan, the calculated attention level is shown (see Section 5.5.2).

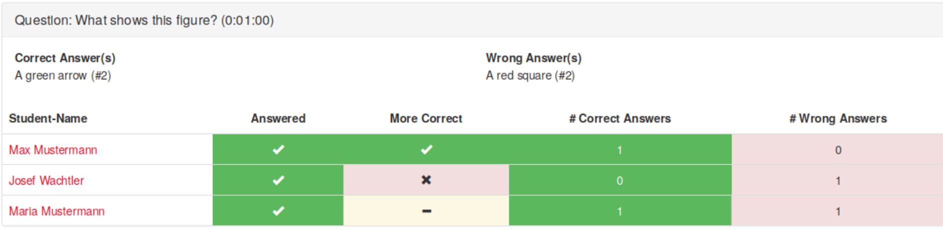



Figure 6.4: The analysis of the multiple-choice questions is shown to the teacher.

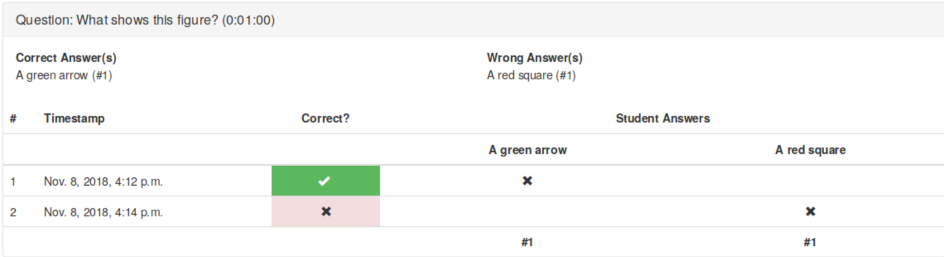
### 6.1.4 Analysis of Planned Multiple-Choice Questions

The interaction method presented by Section 5.6.7 shows multiple-choice questions to the students while watching. To evaluate the performance of the students at the questions, there are two visualizations of the results. The first one is an overview for the teacher containing all students and the second one is a detailed evaluation of a single student.

Teacher Analysis

As shown by Figure 6.4, the question is printed at the top. After the question, the position in the event is shown in the brackets. It is clear that a multiple-choice question could have more than one correct or wrong answer. Because of that, they are listed below the question. In the brackets behind the possible answers it is printed how often the corresponding answer was given by the students.

The actual evaluation is printed as a list of four columns. The first column states the names of the answering students. On clicking such a name, the detailed analysis is loaded (see below). The second column states if the corresponding student tried to give an answer to the question. It is clear that, at a live-broadcasting a question could be answered only once. However, at a video each question could be displayed multiple times because a video could be watched as often as liked. Because of that, the last two columns are counting the number of correct and wrong attempts. As a summary, the third column counts if there are more correct answers than wrong. If this is the case, the field is green with a white checkmark. In contrast to that, the scenario with more wrong answers is indicated by a light red field with a black “x”. If the number is even the field is colored in a light yellow in combination with a black “-”.



Question: What shows this figure? (0:01:00)

Correct Answer(s)
A green arrow (#1)

Wrong Answer(s)
A red square (#1)

| # | Timestamp | Correct? | Student Answers | |
|---|---|---|---|---|
| | | | A green arrow | A red square |
| 1 | Nov. 8, 2018, 4:12 p.m. | ✔ | ✖ | |
| 2 | Nov. 8, 2018, 4:14 p.m. | ✖ | | ✖ |
| | | | #1 | #1 |

Figure 6.5: The answers to the multiple-choice questions of a single student.

`Student Analysis` As mentioned above, there exists a detailed analysis of a single student too. It can be seen by Figure 6.5 that at the top, the question and the possible answers are printed in the same way as by the analysis regarding all students. Below that, a list is printed which shows a row for each given answer. The first column assigns a running number to the answers and the second one states the exact date and time of answering. The third column indicates if the answer has been correct or not. For that, a correct one is visualized by a green field with a white checkmark. In contrast to that, a wrong answer leads to a light red field with a black "x". This column is followed by columns for the different possible answers. In this example there are two. They are used to state which one was selected. For that, the selected answers are marked with an "x". At the bottom of the list, the "x" signs are counted.

### 6.1.5 Analysis of Planned Text Questions

The analysis of the planned text-based questions (see Section 5.6.8) is very similar to the evaluation of the mulitple-choice questions, presented by the previous section. It also consists of an analysis containing all students and a detailed one for each student.

`Teacher Analysis` The analysis of all students is available for the teacher only. As shown by Figure 6.6, the question and its position in the event is displayed at the top. In addition, the reference answer of the teacher is shown below. At the right hand side, the image related to the question is printed. This is followed by a list of the answers of the students. In the first column of this list, the name

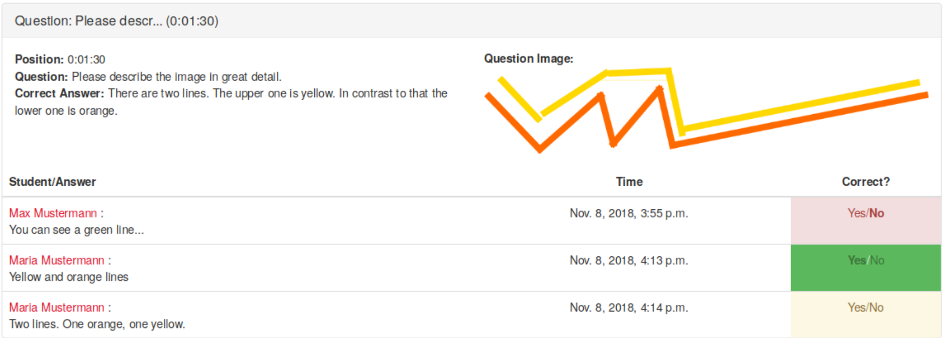



Figure 6.6: The answers of the students to the text-based questions are listed to the teacher.

of the answering student is shown. This name could be clicked to display more details of the student (see below). In the line below the name, the given answer is presented. The second column states the date and time of answering.

It seems to be obvious that it is not possible to evaluate the correctness of a text-based question automatically. Because of that, the third column is used for two functionalities. On the one hand, it indicates if the answer is marked correct or wrong. On the other hand, it offers the possibility to perform the mentioned marking. If a question is still unmarked the field is colored light yellow and provides the possibility to state if the answer has been correct. For that, the words "Yes" or "No" could be clicked. After clicking "Yes" the color of the field switches to green. In a similar way, the selection of "No" leads to a change of color to a light red.

Student Analysis

The analysis of the performance of a single student looks very similar (see Figure 6.7). At the top, the question and the related data are printed in the same way as at the analysis of all students. Below that, a list prints a row for each answer. In the case of a live-broadcasting it is clear that there will be only one row. However, at a video multiple attempts of answering could happen because a video could be watched more often than once. The list prints a running number in the first column and the second one states the submitted answer. Behind that, the third column holds the exact date and time of the submission of the answer. The final column displays if the answer is correct or not. The displayed state depends on the selection of

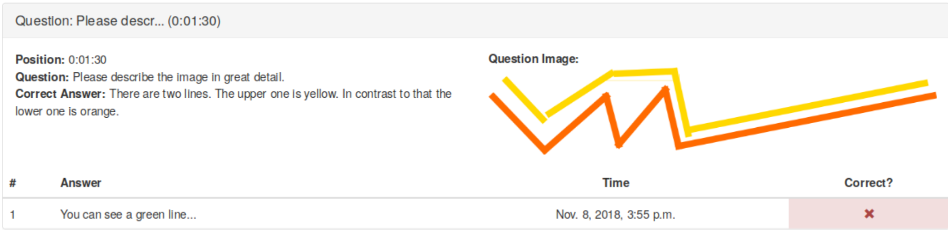



Figure 6.7: The answers of a single student to the text-based questions are listed.

the teacher presented above. If the teacher marked the answer correct, the field is green with a white checkmark. In the case of a marking indicating a wrong answer, the field is colored in a light red displaying a "x". If the rating of the teacher is still not present, the field is displayed in a light yellow printing a "-".

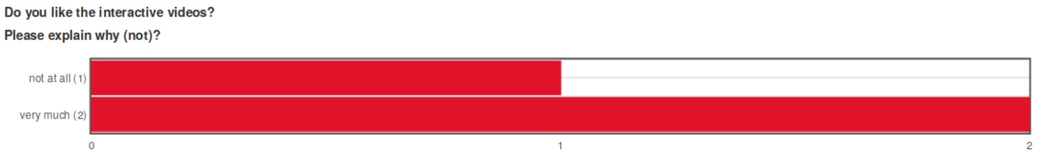
### 6.1.6 Analysis of Post-Event Polls

When students leave an event it is possible to show a poll to them. As explained by Section 5.7, a poll consists of a question and several possible answers. Students are able to select one of them and to enter a text-based explanation additionally.

`Bar Chart` Figure 6.8 presents an analysis of a poll, where the question is printed at the the top. It consists of two parts. The first one is an overview of the number of selections of the possible answers. For that, a horizontal bar chart is used, where a bar represents a possible answer. The length of a bar visualizes the number of selections. In front of each bar, the corresponding possible answer is printed in combination with the exact number of selections.

`Detailed List` The second part is a detailed list of the answers of the students. It shows four pieces of information:

- the name of the answering student
- the date and time of submitting the answer
- the selected answer
- the text-based answer




Details

| User | Time |
|---|---|
| Choice | Answer |
| Max Mustermann (demo_teacher) | Nov. 8, 2018, 4:35 p.m. |
| very much | I like the colors! |
| Max Mustermann (demo_teacher) | Nov. 8, 2018, 4:36 p.m. |
| not at all | Interactions are requiring me thinking! |
| Max Mustermann (demo_teacher) | Nov. 8, 2018, 4:36 p.m. |
| very much | interactions help me better understand. |

Figure 6.8: The results of a poll is shown to the teacher.

These information are shown as a list, where one entry consists of two rows. The first row prints the student's name and the date and time. The second row states the selected and the text-based answer.

## 6.2 External Possibilities

As explained by Section 5.9, it is possible to download all the data of LIVE in the form of spread sheets. Based on these downloads, it is possible to plot the following visualizations:

- **Box-Plot**
  The reaction times to the interactive components could be plotted using boxes.
- **Violin-Plot**
  A different possibility to plot the reaction times resembles the form of a violin.
- **Line-Plot**
  The points of leaving the event could be plotted in conjunction with the position of the question.

- **Dot-Plot**
  The number of different actions performed by the students could be plotted using dots.

To create the different plots it is possible to use according statistical tools. The examples presented by the following sections are created using the "free software environment for statistical computing and graphics" called *The R Project*[1].

### 6.2.1 Box-Plot: Delay of Response

The interactions of LIVE are providing questions to the students which should be answered. As mentioned, the reaction times are recorded. With these recorded reaction times, it is possible to plot the delay of the response to the questions.

`Reaction Time`

An example of such a plot is shown by Figure 6.9. It can be seen that on the y-axis the reaction time in seconds is printed. On the x-axis different videos or other groups of users could be placed. The example shows the reaction times of all the interactions of two videos. Each reaction time is plotted as a point at the position related to the delay of response. The median is visualized as a horizontal black line in the boxes. The boxes themselves are illustrating the standard deviation. It can be seen that the mentioned points indicating the reaction times are forming a line if the distance to the median becomes smaller. This happens because in these areas, the number of reactions is that high that it is not possible to see individual points anymore.

The two examples presented by the plot are showing that most of the reactions are happening around 12 seconds delay. The red example suggests that fewer extreme reaction times happened because the standard deviation is smaller at the upper end. In comparison to that, the blue example has a higher standard deviation at the upper end which indicates that the reactions are not occurring very near to the median. This observation is supported by the individual reaction points which are visible in the field

[1] https://www.r-project.org/, last accessed October 30, 2018

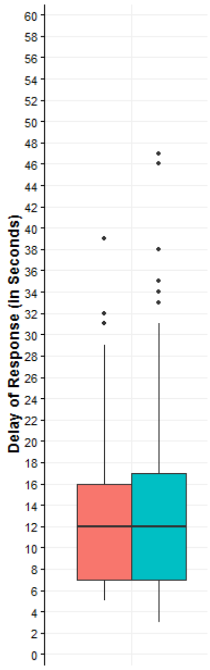



Figure 6.9: The box-plot analysis.

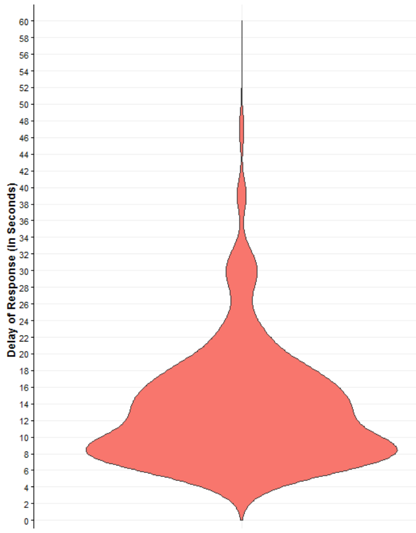



Figure 6.10: The violin-plot analysis.

of higher delays. It can be seen that the longest reaction time at the red example is located at 39 seconds and at the blue example at 47 seconds.

### 6.2.2 Violin-Plot: Reaction Time

In comparison to the plot presented by the previous section, the violin-plot prints the reaction times in a different form. Because in most of the cases the mentioned form resembles a violin, the plot is called accordingly.

`Reaction Time`

Figure 6.10 presents an example of such a plot. Again, the y-axis states the reaction time in seconds. On the x-axis it is possible to print multiple such forms. Each form is plotted in a mirrored way and it becomes wider at positions with a higher number of reaction times. This means that the broadest part of the form is reached at the reaction time which occurred most often. The violin-plot gives a better overview of the real number of reactions at each reaction time than the box-plot presented by the previous section.

An interpretation of the presented example points out that the number of reactions reaches its maximum between 8 and 10 seconds delay. Moreover,

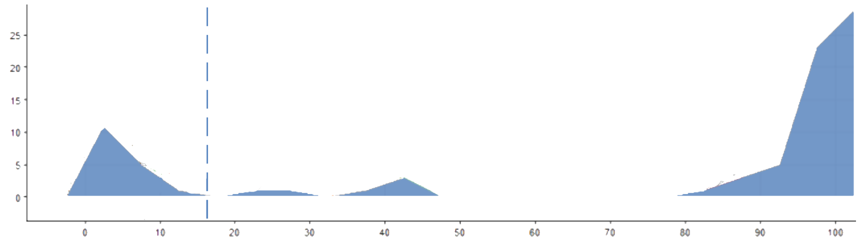



Figure 6.11: The line-plot analysis.

it is possible to state that most of the reactions are happening between 4 and 22 seconds delay.

### 6.2.3 Line-Plot: Drop-Out Point and Question Placement

To study the conjunction of the position of the questions to the point of leaving the event, a line-plot could be used. For that, the data regarding the watched timespans are combined with the positions of the question.

The example shown by Figure 6.11, presents such a plot. It can be seen that on the x-axis the timeline of the video or the live-broadcasting is printed in percent. In contrast to that, the y-axis states the number of students. Now, the line diagram indicates how many students left the event at each position. This means that the line is higher if more students stopped watching at the related point on the timeline. To indicate the position of a question, a dashed vertical line is printed. `Drop-out Rate`

The presented example plot suggests that many students are dropping out during the first 15% and at the end of the event during the last 20%. In between the dropout rate is not very high or even zero (50% to 80%). Furthermore, it is visible that a question is placed approximately after 17% of the event.

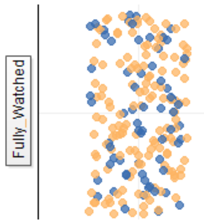



Figure 6.12: The dot-plot analysis.

### 6.2.4 Dot-Plot: Number of Actions

The plots presented by the previous sections have their focus on the reaction time to interactions or on some actions along the timeline of the video or the live-broadcasting. In contrast to that, the dot-plot is a simple possibility to count different things.

Action Counting

An example of such a plot is shown by Figure 6.12. It simply prints some dots in a square. Each dot represents an item of the elements to count. Different colors could be used to represent groups of a property to compare. The example plot counts the number of students who watched the complete video. This is done for two videos, where one video is represented by the blue dots and the other one by the orange dots. It can be seen that there are more orange dots which indicates that the related video was watched completely by more students.

# 7 Evaluation

According to the methodology explained by Chapter 3, the platform for interactive videos or live-broadcastings is developed using the mechanisms of test driven development and rapid prototyping. This means that the requirements are modeled using automatic tests and the platform is implemented by fulfilling the tests. After the implementation the platform is used in a productive environment for evaluation. At this step, the means of qualitative and quantitative research are used.

`Categorization of Usages`

This chapter presents the results of the mentioned evaluations of the platform. It is clear that the different phases of evaluation were performed with different versions of LIVE. The gathered results were used to redefine the requirements and because of that, the platform has been improved multiple times. In summary, it is possible to group the evaluation in three categories of usage:

- live-broadcastings or recordings of lectures
- massive open online courses
- different learning scenarios

The first usage scenario consists of the support of a live-broadcasting or of a video of a lecture held in classroom. It seems to be obvious that such events are typically of a longer length because a unit in the class has a typical length of 1.5 hours. In contrast to that, the usage at massive open online courses has only videos of a shorter length. This means that the videos at this scenario are only of a length between 10 and 20 minutes. The final category includes different forms of usages like a flipped-classroom concept used for videos at school.

`Chronological Order of Usages`

In contrast to the presented categorization of the usages, it is possible to order them in a chronological way. Such an ordering is presented by Figure

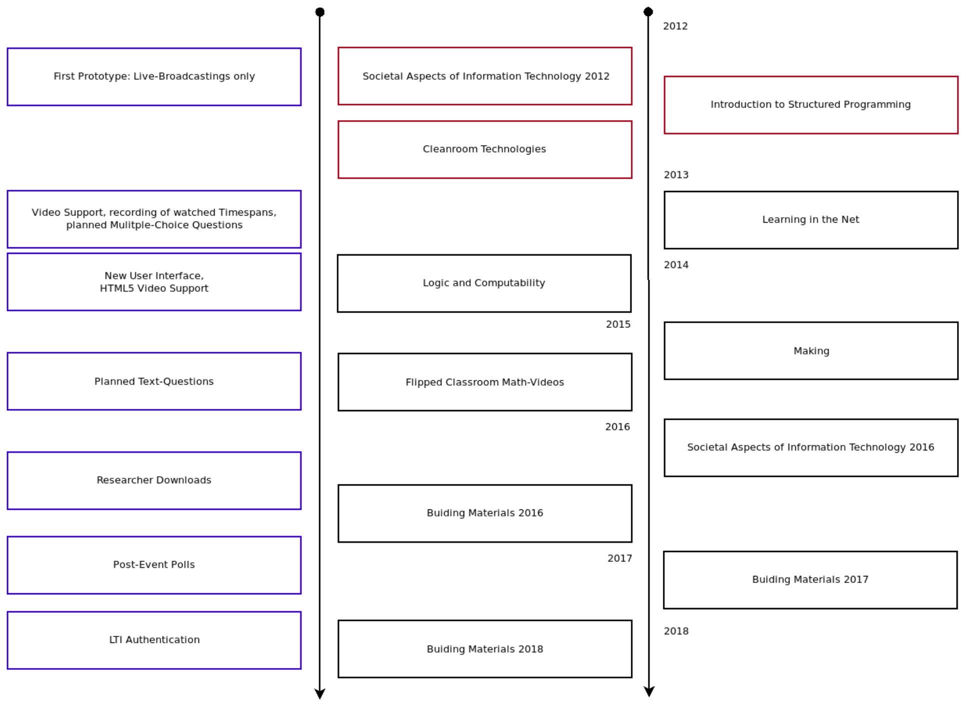



Figure 7.1: The timeline shows the usages of LIVE in chronological order.

7.1. It can be seen that a vertical timeline is printed and all of the evaluations are printed as boxes. The boxes with a red border are representing live-broadcastings and the black boxes are the videos. It can be seen that at the beginning LIVE was used for live-broadcastings only and switched to an exclusive usage for videos later. A reason for that is that at the beginning, only live-broadcastings were possible. With the development of the support of videos the teachers as well as the students favored videos more than live-broadcastings.

In addition to the boxes representing usages of LIVE there are blue boxes highlighting the different milestones of the development process. As mentioned, at the beginning only live-broadcastings were possible with the first prototype explained by Chapter 4. Because of the limitations of the first prototype, the web platform was mostly rewritten to support videos and to record the watched timespans of the students. In addition, the feature to add multiple-choice questions at planned positions in the video was implemented. The next step was the development of a new user interface. This was done to address the needs of the growing usage of mobile devices. Due to that, the support of videos played by using the technologies offered by HTML5 was added. In order to provide additional types of planned questions, the next milestone supported the usage of planned text-questions. To simplify the evaluation of the performance of the students, downloads of all of the data generated by LIVE were implemented. The last two milestones included the implementation of post-event polls and the automatic authentication using LTI. The requirements of the different milestones were mostly identified at the usages presented below. These requirements were modeled using automatic tests and implemented based on them. Each new milestone was used in a productive way again. It can be seen that the mechanisms of rapid prototyping and TDD have been applied in conjunction.

## 7.1 Lecture Live-Broadcastings or Recordings

In its early days, the platform was used mainly at lectures. This means that on the one hand, LIVE was responsible for supporting the live-broadcasting of a lecture with interactive components. On the other hand, the recordings of these lectures were additionally provided to the students and enriched with the interactions offered by the platform.

### 7.1.1 Societal Aspects of Information Technology 2012

The very first usage of the platform at live-broadcastings was performed at *Societal Aspects of Information Technology*[1] at Graz University of Technology (Wachtler, 2012; Wachtler & Ebner, 2014a). This lecture was part of the bachelor program in computer science and consisted of several presentations of guest lecturers. At each unit of the lecture, two such presentations took place. LIVE was used at one of these units. Because the attendance at the

[1] https://online.tugraz.at/tug_online/lv.detail?sprache=1&clvnr=162241, last accessed January 15, 2013

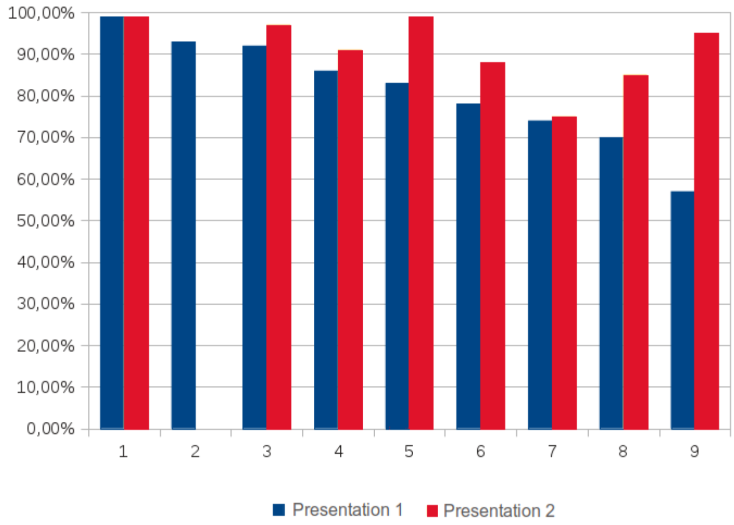



Figure 7.2: The calculated attention levels at GADI 2012.

presentations was compulsory, nine students were recruited to watch the live-broadcasting using LIVE instead of attendance.

Attention Analysis

Figure 7.2 shows the calculated attention levels of these nine students. It can be seen that the attention level reached a value of 75% for most of the students. Only two of them reached a smaller value at the first presentation. Furthermore, all but one of the students reached a higher value at the second presentation.

Feedback

The recruited students were additionally asked to provide feedback regarding the application of interactive components during a live-broadcasting. For that, they were required to write down a list of advantages and disadvantages. In summary, the reports presented the following positive statements:

- The content related questions are helpful to understand the presentation better.
- Asking questions to the teacher increased the feeling to be part of the lecture.
- The possibility to monitor the attendance of the students even at a live-broadcasting is helpful because it is not required to go to the lecture

theater.

In contrast to these advantages, the following issues were reported by the students:

- Questions which are not related to the content of the live-broadcasting are considered irritating.
- The fullscreen feature could not be used because the interactions were not visible in fullscreen mode.
- The interactive components are only available during the live-broadcasting and not after it.

### 7.1.2 Cleanroom Technologies 2012

A further very early usage of the platform took place at the lecture *Cleanroom Technology*[2] which was offered by the *Life Long Learning* department at Graz University of Technology (Ebner, Wachtler, & Holzinger, 2013). The content of this lecture focused on cleanroom technology which could be seen as a core technology. This statement seems to be valid because these technologies are influencing many fields of production. The attendees received a specialized training to the field of cleanroom technology from a scientific point of view.

LIVE was used to present the live-broadcastings of the first five lectures. Each of the lectures had a length of two hours. The number of attendees ranged from seven to twelve. During the lecture, text-based questions were asked to the students. For that, the interaction method presented by Section 5.6.3 was used.

Attention Analysis

It was observed that the calculation of the attention level returned a very low value at the first lecture. This means that the calculation reported that the attention was located at 40% or less. An analysis pointed out that the difficulty of the questions was quite high and because of that, the students required a considerable time to answer them. This issue was further

[2] http://portal.tugraz.at/portal/page/portal/TU_Graz/Studium_Lehre/Life_Long_Learning/ULG%20Reinraumtechnik, last accessed January 15, 2013

complicated by the fact that the teacher did not pause the lecture in the case of a question.

To address these problems of the first lecture, the teacher was advised to stop presenting if a question was asked. With this improvement the calculation of the attention level increased during the second and third lecture. At the last two lectures, all of the students achieved an attention level of 75% or higher.

Feedback

The students were asked to provide feedback regarding the usage of the platform. Most of the students claimed after the first lecture, that they are feeling uncomfortable with the calculated attention level. With the improvement mentioned above, they felt that the calculation is now more realistic. In addition to that, students also reported that they had liked questions which are related to the content of the lecture more than general ones.

It was noted that the workload of the teacher is increased at such a setting in comparison to a standard lecture. This means that she/he is required to craft the questions and to ask them to the students. In this case these tasks were handled by an assistant.

### 7.1.3 Introduction to Structured Programming 2012

The web platform was regularly used at the lecture called *Introduction to Structured Programming*[3] at Graz University of Technology (Wachtler, 2012; Wachtler & Ebner, 2014a). This lecture was part of the bachelor program in computer science and located in the first semester according to the curriculum. Because of that, the lecture was attended by a large number of students. To avoid crowded lecture theaters, the six units of the lecture were broadcast as a live-stream. The students were able to watch a normal live-stream or the one supported with interactive components by LIVE.

Counting Students

The number of students using the platform is shown by Figure 7.3. It can be seen that on average the number of students is located at 14. The first

[3] https://online.tugraz.at/tug_online/lv.detail?sprache=1&clvnr=162268, last accessed January 15, 2013

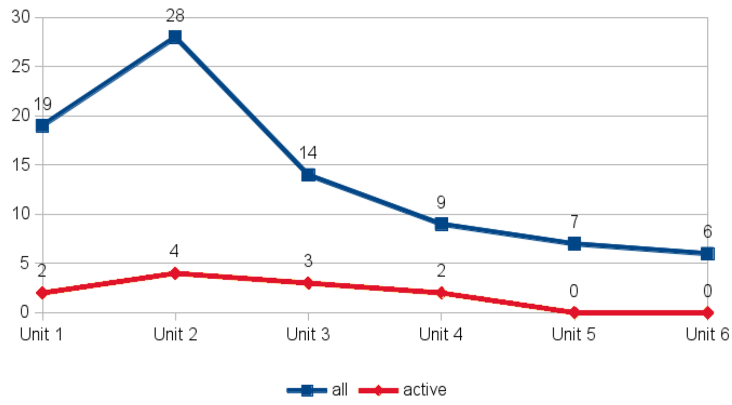



Figure 7.3: The number of students at the units of the lecture.

two units were attended by the most students and after that, the number decreases. In contrast to the watching students, the active ones were only a small group (2-4). This means that many students simply ignored the offered interactive components and only watched the live-broadcasting without reacting to the questions.

It was observed, that the number of different interaction methods was limited. This means that at this version only text-based questions were possible. Further methods of interaction would be helpful to ask questions which are resembling the content of the lecture in more detail. In this case this means that small programming exercises could be provided by interactions.

### 7.1.4 Logic and Computability 2014

In comparison to the previous usages of LIVE, where live-broadcastings were supported with interactive parts, the platform was used to enrich recordings of a lecture. The mentioned lecture was part of the bachelor program in computer science and was named *Logic and Computability*[4] (Wachtler & Ebner, 2015). In the course of this lecture different types of logic were

[4] http://www.iaik.tugraz.at/content/teaching/bachelor_courses/logik_und_berechenbarkeit/, last accessed November 15, 2014

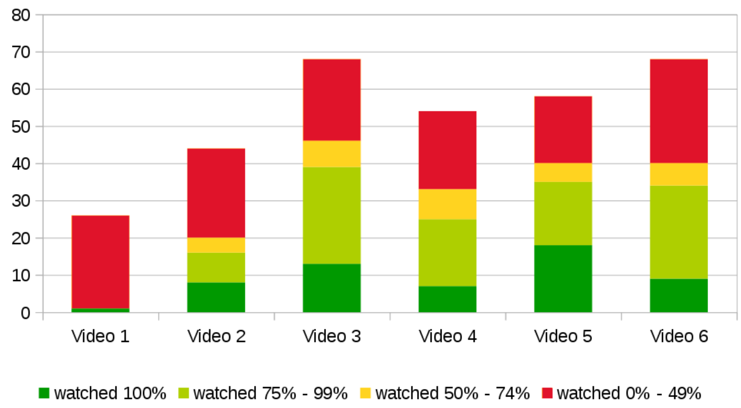



Figure 7.4: The number of students at the units of the lecture Logic and Computability.

explained. This includes propositional logic and predicate logic as well as temporal logic. In addition to the explanation of the syntax and semantic of the different logics, the decidability of them was discussed. The platform was used to present six recordings of the units of this lecture to the students. Each unit had a length of at least 1.5 hours.

The evaluation of the platform at this lecture consisted of three parts:

- analysis of the number of students
- connection of the placement of the interactive components to the correctness rate
- user interface field study

Number of Students

The number of students at the six units of the lecture is visualized by Figure 7.4. It can be seen that the number of students who watched more than 75% (green and light green) is quite high beginning with the third video. This lower acceptance of the first two videos could be explained by the fact that the first one mainly covered topics of organizational and the second one introduced very basic parts only. Furthermore, it can be observed that the number of students, who watched less than 50%, was constant across all videos.

Question Placement affects Success

The second evaluated factor at this lecture is, if there is a connection from the placement of the questions in the video to the correctness rate of them. This analysis pointed out that the results are very similar at each video.

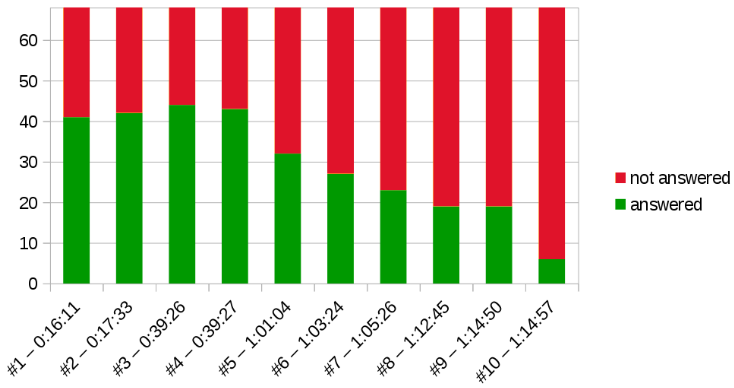



Figure 7.5: The number of students who answered the questions at the lecture Logic and Computability.

Because of that, the following explanations are focusing on the third video which acts as a representative for the remaining ones.

Figure 7.5 prints the number of students who provided an answer to the multiple-choice questions or not. Each bar represents a question and below such a bar, the position in the video is printed. It can be seen that the number decreases considerably beginning with the fifth question. It seems to be obvious that the reason for this is the fact that not all students watched the full video and because of that, they never progressed far enough to reach the later questions.

The correctness rates at the multiple-choice questions are shown by Figure 7.6. Due to the fact that the video could be watched more often than once, it is necessary to evaluate how often a question was answered correct or not. Because of that, the green part of the bar represents the students who answered the questions more often correct than wrong. The opposite scenario (more false answers) is expressed by the red part of the bar. If the number is even, the color yellow is used. Again the position in the video is printed below the bars.

Based on the correctness rates to the questions, it is possible to state some observations which could be made at all videos of the lecture in a similar form. A further requirement for the accuracy of the following observations is the assumption that the difficulty of the questions is equal.

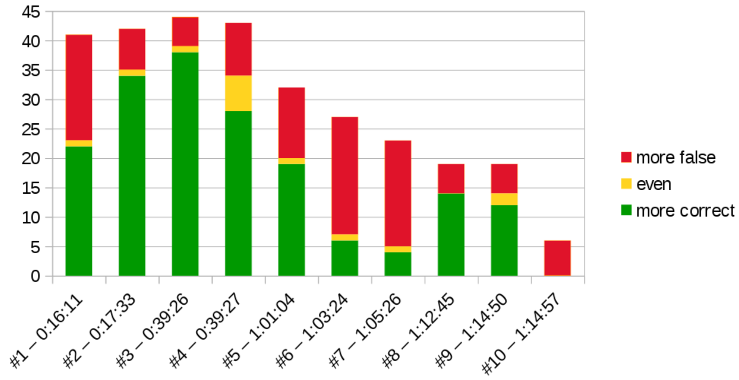



Figure 7.6: The correctness rate at the questions of the lecture Logic and Computability.

- **Lazy Start**: The number of correct answers to the first question is not very high.
- **Correct after Question Pause**. Correct answers are numerous at the third question despite the fact that the timespan since the last question is quite high.
- **Tight-Placed Errors**: If questions are placed very tight the number of correct answers is decreasing. This effect is shown by the questions number three and four as well as by number five to seven.

In order to explain these observations, the exact times of occurrence of the interactions in the video are required. Because of that, Table 7.1 prints for each multiple-choice question the position in the video and the timespan elapsed since the last multiple-choice question. In addition, the maximum possible timespan since the last random interaction is listed as well.

`Lazy Start` If it is assumed that the distribution of the interactions has an impact, the observation named "Lazy Start" was made because the number of students who had answered the first question more often wrong than correct is quite high (18 out of 41). For that, two reasons are possible. The first one is, that because this is the first content related question. This means that until the occurrence of this question, the students were not required to answer a question with a connection to the content of the video. The second reason might be that the last random interaction occurred a long time (max. 16 minutes and 11 seconds) ago. Because of that, the students simply had been

| No. | Time | Timespan since last MC question | Max. timespan since last random interaction |
|---|---|---|---|
| 1 | 00:16:11 | - | 00:16:11 |
| 2 | 00:17:33 | 00:01:22 | 00:08:59 |
| 3 | 00:39:26 | 00:21:53 | 00:13:44 |
| 4 | 00:39:27 | 00:00:01 | 00:13:45 |
| 5 | 01:01:04 | 00:21:37 | 00:09:40 |
| 6 | 01:03:24 | 00:02:20 | 00:12:00 |
| 7 | 01:05:26 | 00:02:02 | 00:14:02 |
| 8 | 01:12:45 | 00:07:19 | 00:12:47 |
| 9 | 01:14:50 | 00:02:05 | 00:14:52 |
| 10 | 01:14:57 | 00:00:01 | 00:14:53 |

Table 7.1: The occurrence of the multiple-choice questions in comparison to the random interactions at the lecture Logic and Computability.

unprepared for such a question.

`Correct after Question Pause`

On examining the observation named "Correct after Question Pause", it can be seen at the third question that the number of students who answered it more often correctly is high (34 out of 41). However, the last multiple-choice question happened more than 20 minutes ago. A reason for the high correctness rate might be that the timespan since the last random interaction was not very long. Because of that, it could be that the students were ready for a question because of the random interaction which happened not so long ago.

`Tight-Placed Errors`

As mentioned above, the observation called "Tight-Placed Errors" states that the correctness rate at the multiple-choice questions is decreasing if there are too many questions within a short period. It is possible to observe this effect at the questions number three and four as well as at the questions number five to seven. In a minor form the effect is also present at the questions number eight to ten. In summary, it is possible to state that a gap of two minutes and twenty seconds or less between the questions could lead to a decreasing number of correct answers. In comparison to that, a gap of seven minutes and 19 seconds is prone to lead to acceptable results of the students at the questions. The effect could not be observed at the first two questions. A reason for that might be that the first one is influenced by the issue named "Lazy Start". This means that the unexpected occurrence of the first question is helpful for the following questions.

UI Design

In addition to the analysis of the interactions and their impact on the correctness rate of the questions, a field study evaluating the user interface was performed at this lecture (Wachtler et al., 2015). For that, the teacher as well as the students were asked to provide feedback. This feedback pointed out that the coloring and the styling of the fonts is well readable and not offensive. In addition, it was stated that the navigation bar is well organized and because of that, the searched pages are easy to find.

In contrast to the positive remarks some issues were reported too. It was stated that the history timeline per user (see Section 6.1.3) sometimes had not shown a bar. This had happened if the length of the watched timespan was very small. This had led to the problem that because of this short length, the bar was too small to be visible. Furthermore, it was reported that when creating planned questions, it had been difficult to find the correct position in the timeline of the video.

## 7.2 Massive Open Online Courses

In contrast to the live-broadcastings or recordings of lectures, this section evaluates the usage of LIVE at massive open online courses. It is typical for such MOOCs that only videos are used. In addition, these videos are of much shorter length. Because of that, different results of the evaluation are possible in contrast to the evaluations presented above.

| | Video 1 | Video 2 |
|---|---|---|
| Length [min] | 10:13:00 | 08:32:00 |
| Attendees | 32 | 20 |
| 100% watched | 20 (62,5%) | 9 (45%) |
| 75% - 99% watched | 7 (21,88%) | 5 (25%) |
| 50% - 74% watched | 1 (3,13%) | 1 (5%) |
| MC-Questions | 2 | 2 |
| 1. Question correct | 23 (71,88%) | 18 (90%) |
| 1. Question not correct | 5 (15,63%) | 2 (10%) |
| 2. Question correct | 26 (81,25%) | 20 (100%) |
| 2. Question not correct | 2 (6,25%) | 0 (0%) |

Table 7.2: Some statistical values of the first two videos of LVnet.

### 7.2.1 Learning in the Net: From possible and feasible things 2013

The very first usage of the web platform at a MOOC happened at the one called *Learning in the Net: From possible and feasible things*[5] which was provided by the Karl-Franzens University of Graz (Wachtler & Ebner, 2014b). It consisted of the presentation and discussion of different technologies, which are part of the so-called new medias, for the purpose of teaching. For that, videos were offered to the attendees, using LIVE to support them with interactive components. At each video two multiple-choice questions were added. Because of the short length of the videos, only one random interaction occurred.

Number of Students

This evaluation uses the first two videos. For that, some statistical values are printed by Table 7.2. It could be observed that both, the absolute number of attendees as well as the number of students who watched the full video decreased considerably at the second video. In contrast to that, the number of students who watched 50% to 99% is quite the same at the two videos.

Question Success

When examining the multiple-choice questions, it can be seen that the suc-

[5] https://online.uni-graz.at/kfu_online/lv.detail?cperson_nr=63360&clvnr=370548, last accessed January 15, 2013

cess rate is higher at the second video. This means that both questions are answered more often correct at the second video than at the first. Furthermore, it can be observed that at the second video all watching students tried to answer the questions. This was not the case at the first video because of the 32 attendees only 28 answered both questions. This is strange because the number of students who watched the full video is lower at the second video. This indicates that some students jumped from question to question at the second video because if they watched from the beginning not all of them would reach the second question.

### 7.2.2 Making: Creative, digital creating with children 2015

In contrast to the evaluation of the MOOC presented by the previous section, the analysis performed at this one is based on the possibilities shown by Section 6.2. The evaluated MOOC was named *Making: Creative, digital creating with children*[6] and had a duration of seven weeks with at least one video per week (Wachtler, Khalil, Taraghi, & Ebner, 2016). It explained possibilities for creative and digital creating or experimenting with children. For that, different tools and activities were presented. Each video was presented to the attendees of the course using LIVE to support the videos with interactive components.

`Reaction Time`

To evaluate the reaction delay of the attendees, two different plots are used, namely the box-plot and the violin-plot. The first one is shown by Figure 7.7. It can be seen that on the x-axis the different videos are printed. In contrast to that, the y-axis shows the reaction time in seconds with a limit of 60. The different groups of users are represented by the two colors (light red and blue). The first group consists of the students who finished the course successfully and because of that they requested a certificate. The second group represents the un-certified students.

On comparing the two groups, it can be seen that in week 4 and 7 the certified students required more time to answer the questions. This statement is valid because the median of the reaction time in week 4 is located at

[6] https://imoox.at/mooc/local/courseintro/views/startpage.php?id=3, last accessed December 12, 2018

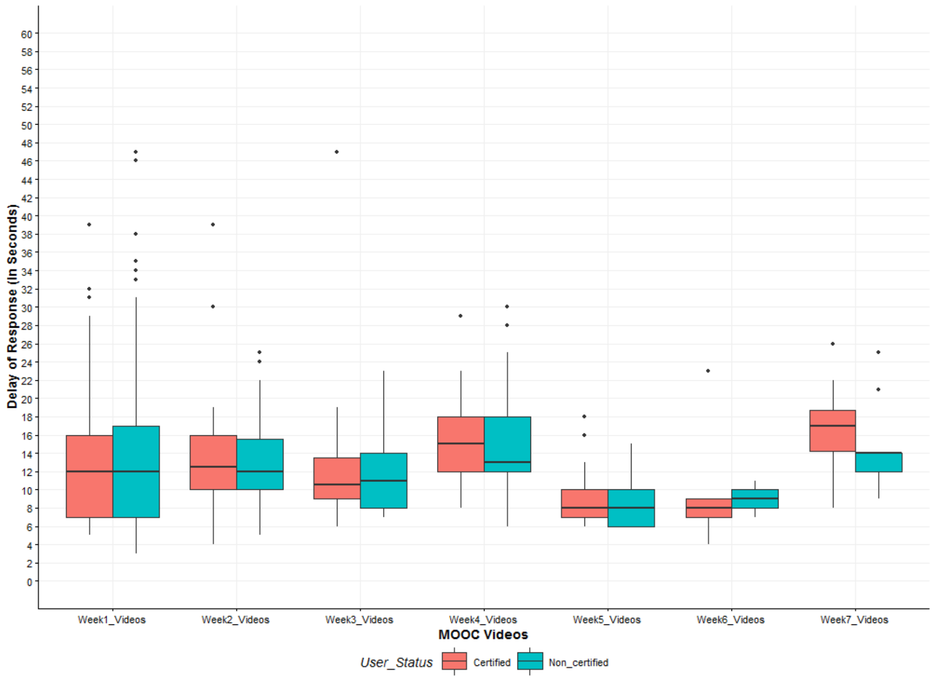



Figure 7.7: To visualize the reaction times to the mulitple-choice questions a box-plot is used at the Maker-MOOC.

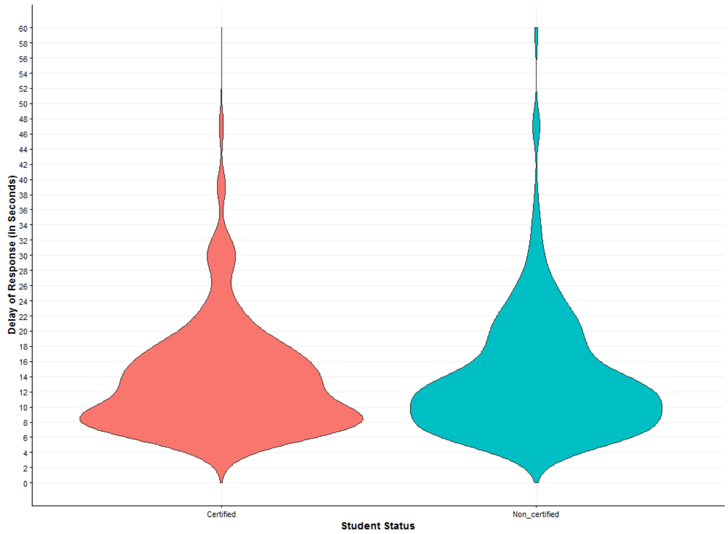



Figure 7.8: The violin-plot creates a summary of the reaction times to the questions at all videos of the Maker-MOOC.

13 seconds for the certified students and 15 for the un-certified ones. In week 7 the difference in the medians is 3 seconds. Furthermore, it could be seen that the medians in week 1 and 5 are more or less the same. It is only possible to observe an insignificant variation located between the first and third quartiles.

The plot presented by Figure 7.8 prints a violin for each group of students (certified or un-certified). As by the box-plot presented above, the y-axis states the reaction time of the students to the multiple-choice question. It is possible to observe that the density of the blue form is a bit wider in contrast to the red one. This suggests that the students took more time for reacting to the questions. Furthermore, it is visible that at the outer ends the blue form has more width than the red one. This indicates that at the group of the un-certified students, some reacted very slow (upper end) or very fast (lower end).

Dropout-Rate and First Question

The line-plot presented by Figure 7.9 visualizes two things. The first one is the dropout-rate. The second one marks the point of the first question. On the x-axis the timeline of the video is printed in percent. This is done

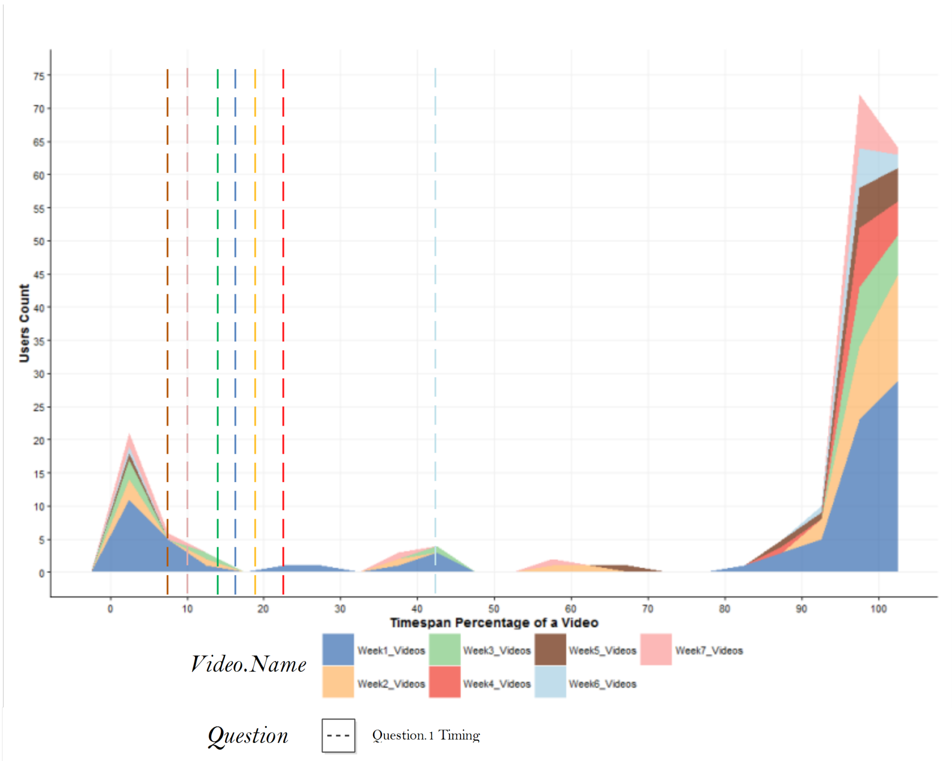



Figure 7.9: The dropout points are visualized by a line-plot at the Maker-MOOC.

because all videos are summarized in one plot and because of that, it is necessary to form a common base for the different lengths of the videos. On the y-axis the number of the dropping out students is counted. This means that if the line reaches a higher value at some point, more students are leaving the video at this position. To show the placement of the first question, a dashed vertical line is used.

It can be seen that the dropout-rate is quite high at the first 15% of the video. Because of that, the questions were placed in this region to grab the attention of the students. In addition to that, it can be seen that only very few of the students are leaving the videos between 20% and 80%.

To count the number of students at the different videos, the plot presented by Figure 7.10 is generated. Again, the timeline of the videos is plotted along the x-axis in percent. Each video is represented by a dotted line in different

`Comparing Number of Students and Views`

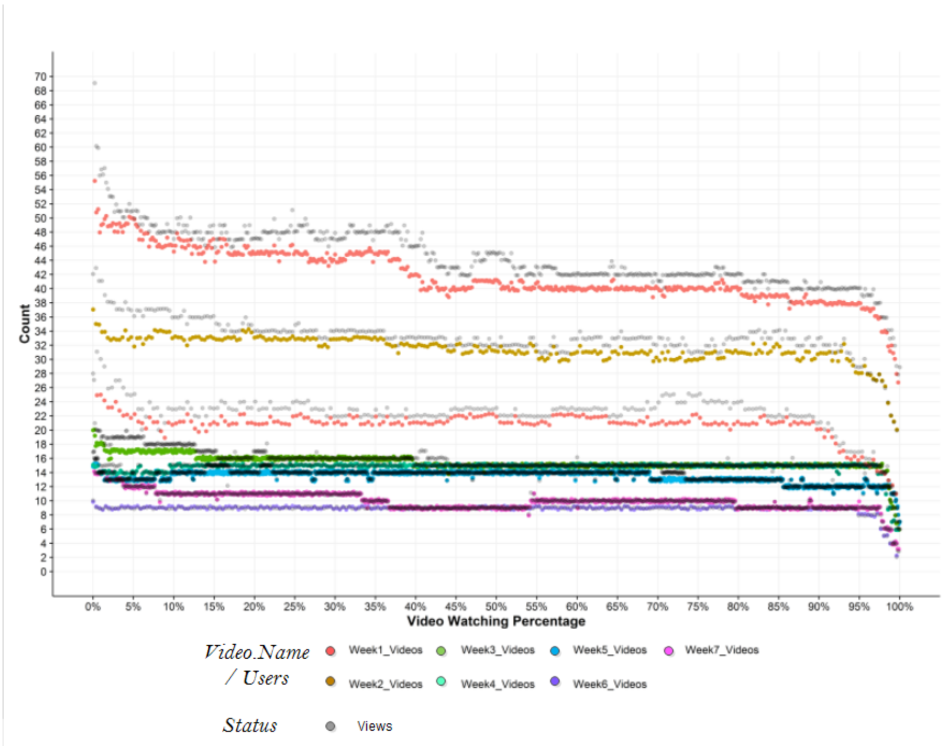



Figure 7.10: The number of students at each video of the Maker-MOOC.

colors. These lines state how many students watched the videos. Each of these lines have a shadow line which indicates how often the students viewed the video. It is clearly visible that in the first three weeks, more views per user happened. In contrast to that, the number of watchers and views is more or less equal at the following videos.

A further observation indicated that the dropout during the videos is not very high. However, at the last 2%-3% the users quit a lot. The same effect could be seen at the line-plot presented by Figure 7.9. This could be explained by the fact that at the end of each video the credits are shown which are not very interesting for the students.

On examining the activity of the students at the videos, the plot presented by Figure 7.11 is used. This plot prints the number of "play" actions, "stop" actions and students who watched the full video. For that, dots are printed

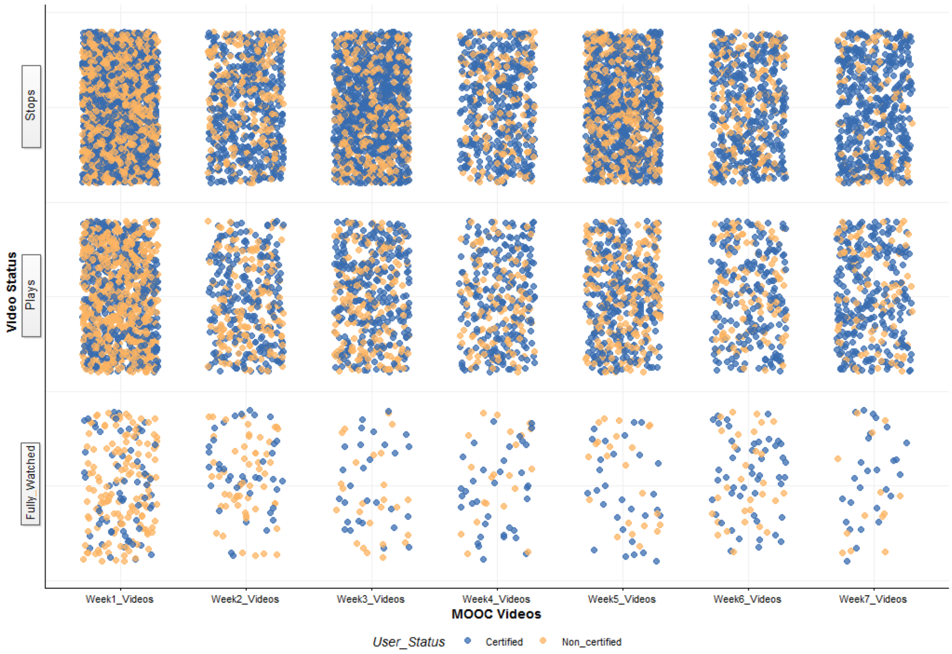



Figure 7.11: The dot-plot of the activities of the students at the Maker-MOOC.

for each item to count in a square. Such a square represents a single video. To distinguish between the certified and un-certified students, different colors are used for the dots. It can be seen that in the first three weeks the number of actions (play and stop) is quite high. In contrast to that, the engagement decreased at the later videos. The same decreasing could be noticed at the number of students who watched the whole video.

### 7.2.3 Societal Aspects of Information Technology 2016

As shown by Section 7.1.1, LIVE was used at the lecture *Societal Aspects of Information Technology* at Graz University of Technology in 2012. After some changes in the curriculum, the lecture was moved from the sixth to the second semester and because of that, more students joined the course.

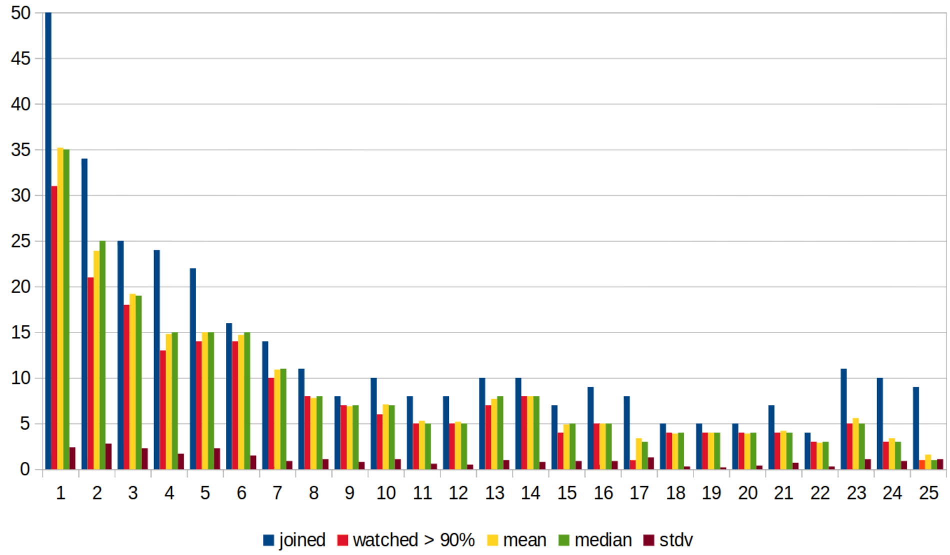



Figure 7.12: Number of students at GADI 2016

Due to that, the didactic concept was changed to a MOOC in 2016[7]. This means that the presentations of the guest lecturers were transferred to short learning videos with a length ranging from 10 to 15 minutes. In summary, the lecture consisted of 25 videos, where 2 were released per week.

Number of Students

Some statistical values regarding the number of students are shown by Figure 7.12. This bar chart prints multiple values per video. This includes the following ones:

- number of students who joined to the videos
- number of students who watched more than 90%
- mean, median and standard deviation

It is clearly visible that all numbers have a decreasing tendency. For instance the number of students, who watched more than 90%, was located at 31 at the first video and decreased to values below 10, beginning with the 8th video. Furthermore, it can be seen that the number of students who joined to a video is considerably higher than the number of students who watched

[7] https://online.tugraz.at/tug_online/wbLv.wbShowLVDetail?pStpSpNr=187268&pSpracheNr=2, last accessed January 15, 2019

more than 90% in most of the cases. If this difference is bigger, it is visible that the mean and the median are higher than the number of students, who watched more than 90%. This observation is supported by the fact that also the standard deviation is higher in this case.

## 7.3 Miscellaneous

This section presents evaluations of the web platform at different scenarios. This includes a flipped-classroom concept implemented at a school as well as different didactic approaches used at a lecture through different years.

### 7.3.1 Flipped Classroom Math-Videos at School 2015

As suggested by the headline, this evaluation (Wachtler, Hubmann, Zöhrer, & Ebner, 2016) was performed at school using the concepts of the flipped-classroom principle. This means that the content of the class was presented to the pupils not at school. They were required to learn the topics at home using interactive videos provided by LIVE. The exercises normally done at home took place at school according to this concept. This concept of learning was tested at a class of an academic high school (BG Klusemann) in Graz. This high school has its main focus on STEM[8]. The mentioned class used the flipped-classroom concept at the subject of mathematics to learn all the required topics regarding differential calculus according to the Austrian curriculum. This included monotonicity, maxima and minima, inflection points, saddle points, finding polynomial functions and the graphical construction of derivatives. The attendance for the 20 pupils was compulsory.

At this usage, three issues were addressed. At first, the observation called **Lazy Start** reported by Section 7.1.4 is evaluated in more details. In addition to that, a further observation named **Tight-Placed Errors** is analyzed. This means that at first it is checked if the correctness rate at the first question of a video is prone to be lower. Second, it is evaluated if the number of correct answers is decreasing if the questions are placed too close to each other. The

[8]short for Science Technology Engineering Math


Figure 7.13: The correctness rates to the first question grouped by their position.

third addressed issue regards the long term learning success. In summary, 11 videos were used to evaluate the usage of the platform used at school. Each of these videos had a length of approximately 12 minutes.

Lazy Start

To evaluate the issue named **Lazy Start**, the first question of a video was placed at different positions in different videos. This means that the first question was placed in the videos according to the following list:

- after 1 minute: used in 3 videos
- after 2 minutes: used in 3 videos
- after 3 minutes: used in 3 videos
- after 4 minutes: used in 2 videos.

With these questions in the videos it is possible to examine the **Lazy Start** observation at shorter learning videos. Figure 7.13 shows the correctness rates at these questions in percent values. This is done because the number of answers varies between the videos. It can be seen that at the questions appearing after one minute, 55% of the answers were correct and 45% wrong. In absolute values this means 29 true answers and 24 false ones. The number of correct answers decreased to 16 (38%) at the videos where the first question pops up after two minutes. The best results were achieved at the videos with a question after three minutes run time. In this case 77% out of the 39 answers were correct. In contrast to that, only 23% were wrong. Finally, the correctness rate at questions appearing after 4 minutes is located at 65%.

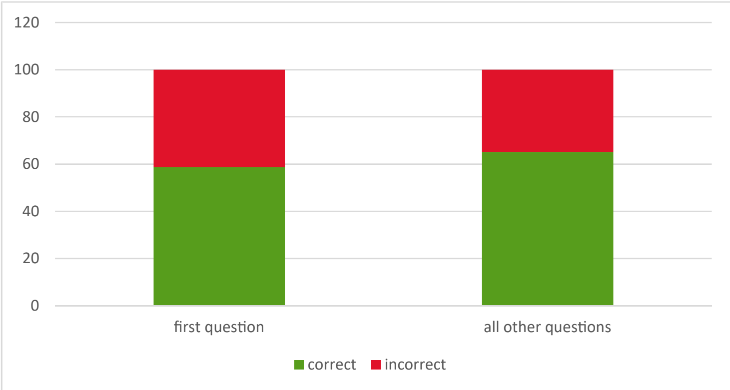



Figure 7.14: The correctness rates to the first question compared with the remaining ones.

The observation named **Lazy Start** additionally includes the assumption that the first question is answered more often wrong than the remaining questions. To address this issue, all questions of all videos are grouped in two categories, namely first questions in the videos and other questions. The results are printed in percent values by Figure 7.14. It can be seen that at the first questions, 59% answered wrong. In contrast to that, the number of wrong answers at the remaining questions is located at 41%. The observation named **Lazy Start** is confirmed by these numbers. However, the results are not statistically significant.

To evaluate the observation called **Tight-Placed Errors** at shorter learning videos, the questions were placed with different intervals between them in the videos. This was done to test if the distribution of the questions throughout the video affected the correctness rates at the questions. In summary, the following intervals were used:

`Tight-Placed Errors`

- 1.5 minutes: used in 2 videos
- 2 minutes: used in 2 videos
- 2.5 minutes: used in 3 videos
- 3 minutes: used in 4 videos.

The correctness rates of the questions are presented by Figure 7.15. An interval length of 1.5 minutes produces a correctness rate of the questions of 71%. In contrast to that, the number of correct answers is located at 58%

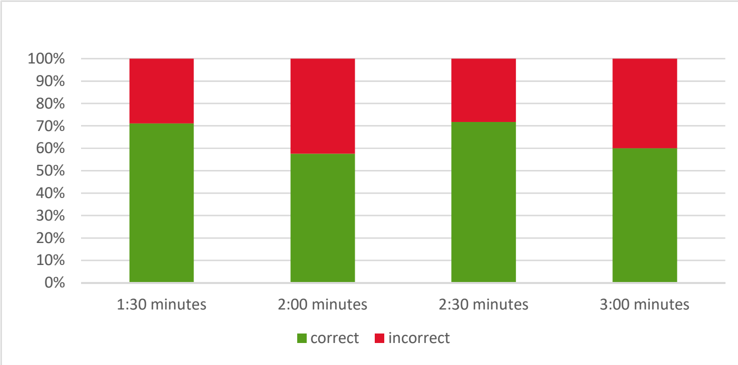



Figure 7.15: The correctness rates of the questions grouped by different intervals between them.

at the videos with a two minutes interval between them. The best results were achieved at the videos with an interval length of 2.5 minutes. This means that the answers were correct in 72% of the cases. Finally, the longest interval length led to a correctness rate of 65%.

It can be noticed that no real tendency could be observed. This means that it is not possible to transform the observation called **Tight-Placed Errors** from longer lecture recordings to shorter learning videos.

Long Term Learning Success

To measure the long term learning success, the results of a test performed in the class were evaluated. Because such tests are strictly regulated in the Austrian curriculum, the test is planned independently from the videos. This means that the test took place after 6 videos and contained not only the topics presented by the videos. The test additionally checked for other topics too, however the amount of such other topics was very small. The results of the test is compared with the results of a different class which was taught in conventional manner without the support of videos.

The mentioned results are shown by Table 7.3. The table lists the possible grades in the first row. They are ranging from 1 being the best, to 5 which is the worst. The second row shows the results of the class (A) using the interactive learning videos. In contrast to that, the results of the class (B), taught using a standard classroom concept, are presented by the third row. It can be seen that class A reached considerably better results than class B. This means that the mean and the median of the grades of class A is located at 3

| Grades | 1 | 2 | 3 | 4 | 5 |
| --- | --- | --- | --- | --- | --- |
| Class A | 2 | 5 | 6 | 3 | 4 |
| Class B | 0 | 1 | 7 | 3 | 8 |

Table 7.3: Test results of a class (A), taught using the flipped-classroom concept, compared with a standard class (B).

with a standard deviation of 1.247. In contrast to that, class B has a tendency towards the negative results. This is expressed by a mean of 3.948 with a median of 4 and a standard deviation of 1.026. These better results at the class implementing the flipped-classroom concept are statistical significant. To explain these significant better results, the grades could be examined. It can be seen that at class A, two pupils managed to achieve the best grade and five the second best. In contrast to that, class B students reached no best grade and only one second best. 4 students of class A and 8 of class B achieved the worst one.

The feedback of the teacher stated that the evaluation of the questions provided a valuable insight in the understanding of the students. However, it was claimed that the available methods of interaction are not fully suited for math topics.

### 7.3.2 Attendance Controlling for Videos of Lab-Experiments 2016

At the university, compulsory attendance is typically applied at practical lectures. Section 2.5 explained the benefits of such compulsory attendance as well as some methods to enforce and to control it. At this usage of LIVE, it was analyzed if it is possible to use the features of the platform to monitor the attendance at online videos (Wachtler & Ebner, 2017). To support the attendance monitoring, the multiple-choice questions placed in the videos were evaluated too. In addition to that, a survey was placed in some videos to ask the students for feedback regarding the platform.

The usage of the platform for this evaluation took place at the lecture *Building Materials Basics*[9] at Graz University of Technology. This lecture was part of the bachelor program for *Civil Engineering Sciences and Construction Management*. This lecture is typically located in the second semester according to the curriculum. This course taught the students the basics of the utilization of building materials and the relevant features and characteristic values of building materials for carried and non-carried components. The course was split into a classroom part and a part held in the laboratory. The laboratory part was used to show practical demonstrations to the students. 304 students enrolled for the course and it was vital for all of them, to see all of the 17 demonstrations. Because it was impossible to place such a large number of students in a laboratory, the students were divided into 17 groups. Each of the groups was required to perform one demonstration and to create a learning video showing the demonstration. The members of all groups were required to watch all of the 17 videos of the demonstrations.

List of Usernames

In summary, the lecture followed the schema presented by Figure 7.16. It can be seen that at first, the students were forming groups by themselves. The remaining group-less students were assigned to existing groups and a final list of groups was created by the teacher. In addition to that, the teacher assigned a unique username to each student which had to be used at LIVE. This was done to identify the students at the analysis of the attendance monitoring. For that, the teacher created a list with usernames. This was done according to the schema, presented by Figure 7.17 (Wachtler et al., 2019). It can be seen that the first names and the last names of the students were used in combination with a preceding running number. The individual parts were separated by an underline.

The students were producing the videos by making their demonstration. For that, they had to write a script at first and after that, the actual filming happened. Finally, some post-production was required to create a meaningful learning video. The videos had to be sent to the teacher until a predefined submission deadline.

After the submission deadline, the teacher released them to the students through the web platform. Furthermore, she/he added some interactive

[9] https://online.tugraz.at/tug_online/wbLv.wbShowLVDetail?pStpSpNr=190463&pSpracheNr=2, last accessed January 15, 2017

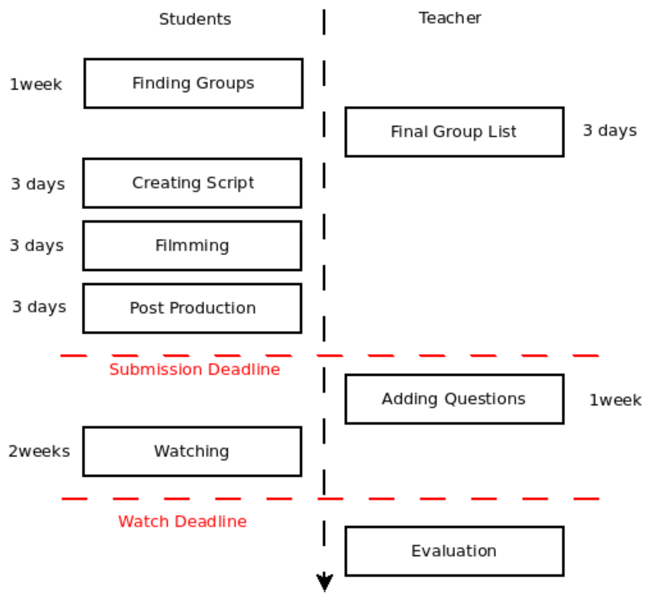



Figure 7.16: The flow of events at the lab experiments 2016.

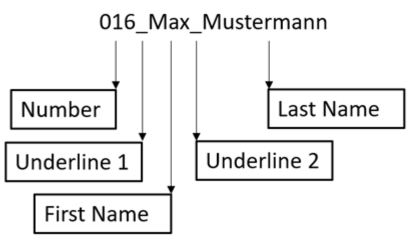



Figure 7.17: The schema of the usernames.

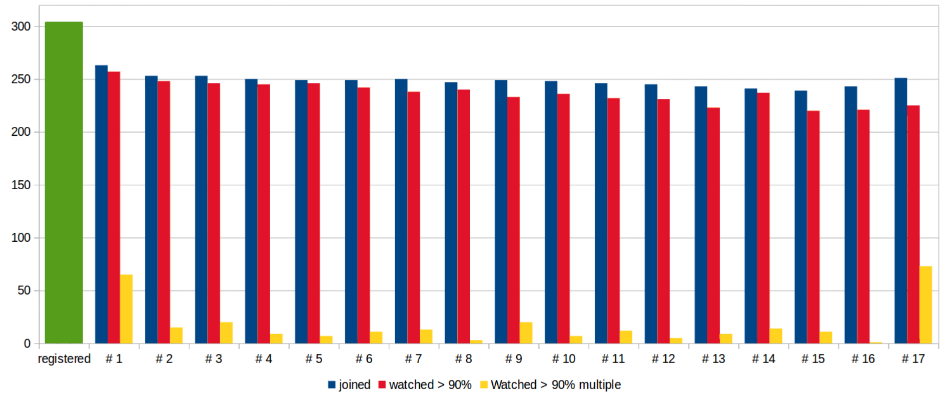



Figure 7.18: The number of students at the lab experiments 2016.

components to the videos to improve the quality of the attendance monitoring. Now, it was time for the students to watch all of the videos and to answer the questions provided by the interactive components. Finally, the teacher was able to evaluate the performance of the students by using the features of LIVE presented by Section 6.1.

Number of Students

The first part of an evaluation of the attendance monitoring is an analysis which states how many students watched how much of the videos. Such an analysis is shown by Figure 7.18. On the left hand side of the diagram, a green bar prints the total number of registered students. After that, three bars are printed for each video. The first bar (blue) shows the number of students who joined to the video. After that, the red bar states how many of these joined students watched more than 90%. Such a threshold is required because the videos are presenting their credits at the end and due to that, many students are not interested in watching these credits. The final bar (yellow) indicates how many students re-watched the video.

On examining the numbers presented by the diagram, several issues are visible. At first, it can be seen that approximately only 250 students joined the videos. This indicates that 50 students from the 304 registered ones are missing. A more detailed analysis points out that most of the missing students are the same at each video. The second issue states that the number of fully watching students had a decreasing tendency at the later videos. This means that at the first 6 videos, only 7 or less students left the video

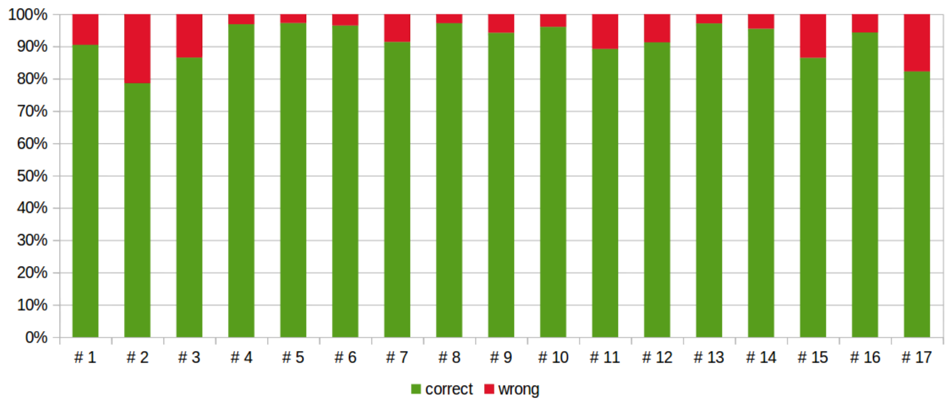



Figure 7.19: The results of the multiple-choice questions at the lab experiments 2016.

early. This number increased at the following videos to at least 12. Finally, the maximum of early leaving students was reached at the last video where 26 students didn't finish watching. The final issue points out that not many students watched the videos more often than once. It is visible that from video 2 to video 16 the maximum of re-watching students is located at 20. Only the first and the last video was watched more often than once by more than 60 students.

`Question Analysis`

The second part of the evaluation of the attendance monitoring is the analysis of the multiple-choice questions. It is shown that nearly all students provided an answer to these questions. The results are printed by Figure 7.19. It can be seen that the correctness rate is very high at most of the videos. There are only exceptions at the first and the last video. At these videos the number of wrong answers is higher than 15%.

`Attendance Monitoring`

With these two parts, it is possible to analyze the attendance monitoring of the students. Because each student could be identified by the assigned username, the teacher is able to state for each student when which part was watched. To support the accuracy of the attendance monitoring, the teacher is additionally able to use the answers to the multiple choice questions as a proof for the seriousness of the attendance. At this course, the high number of students who watched more than 90% and the high correctness rates to the questions are indications for such a serious attendance of most of the students.

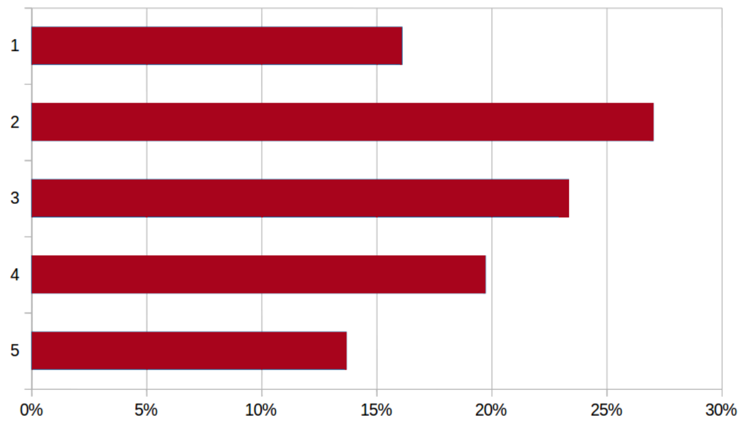



Figure 7.20: The acceptance of LIVE at the lab experiments 2016.

Feedback

In addition to the evaluation of the attendance monitoring, this usage also embedded a text-based question in some videos. These questions were used as a survey to get some feedback about the didactic approach as well as the web platform itself. For that, the students were asked to use the grading system of Austrian schools where 1 is the best grade and 5 the worst. The results of this survey are printed by Figure 7.20. It is visible that in most of the cases, the best or the second best grade was assigned. From a statistical point of view, the mean of the grades is located at 2.92 and the median at 3 with a standard deviation of 1.26.

In combination with the survey regarding the didactic approach, the students were asked to justify their assigned grade by writing a short statement. These statements were mainly variations of the following samples:

- I like interactive videos because with the help of the questions I am able to watch the videos actively.
- I don't have to go to the lab for all demonstrations because of the videos with attendance monitoring.
- The interactive components are stopping me from watching the videos faster.

### 7.3.3 Polls and Assessment at Videos of Lab-Experiments 2017

One year after the usage of LIVE, as presented by the previous section, the platform was used at the same lecture again. This means that the practical part of the lecture *Building Materials Basics*[10] was again supported by the platform presenting interactive components in videos (Wachtler, Scherz, & Ebner, 2018). In the previous year, the platform was mainly used to monitor the attendance of the students. Because the platform was now able to present polls to the students after watching a video (see Section 5.7), the didactic concept was changed as well.

Again, the course was divided into a theoretical part and a practical part. As shown by Figure 7.21, the students were taught by the teacher in conventional manner at the theoretical part. In the practical part, the students were required to perform demonstrations and experiments on different topics after the experiments were explained by the teacher. These topics included the following ones:

- Aggregate
- Binders 1
- Binders 2
- Fresh Concrete
- Hardened Concrete 1
- Hardened Concrete 2
- Steel
- Synthetic Materials

All of the registered 150 students were divided into 16 groups. Each group was responsible for one of the experiments. This suggests that the same experiment was performed by two different groups. After the students created and submitted the videos, the two ones of the same experiment were merged to one video where after the first one, the second one was placed. At the end of each video, the teacher placed some polls, asking the students for a comparative evaluation of the two videos. In addition to

[10] https://online.tugraz.at/tug_online/wbLv.wbShowLVDetail?pStpSpNr=198333&pSpracheNr=2, last accessed January 15, 2017

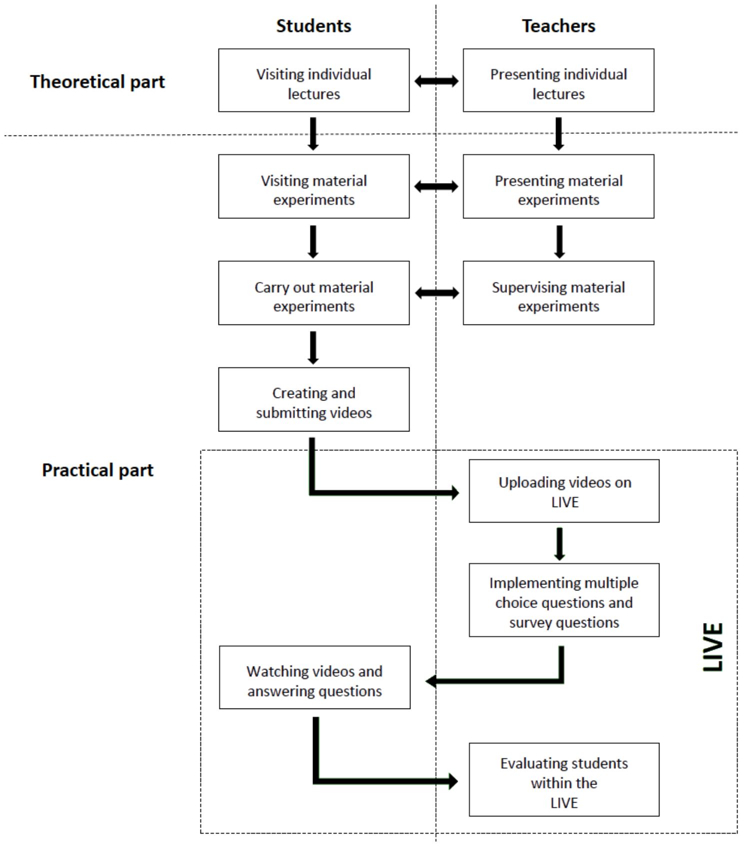



Figure 7.21: The flow of events at the lab experiments 2017.

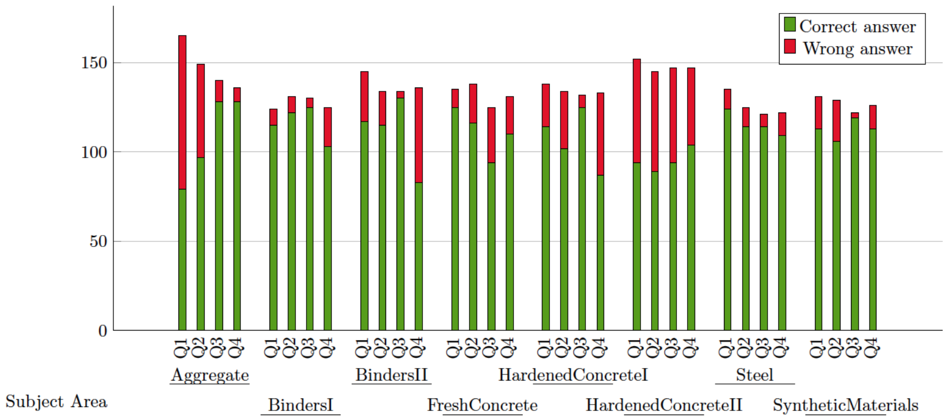



Figure 7.22: The results of the multiple-choice qestions at the lab experiments 2017.

the polls, the teacher also created 4 multiple-choice questions during the videos.

Question Analysis

The evaluation of the results of the multiple-choice questions is shown by Figure 7.22. It can be seen that the number of answering students is varying because not all students watched the whole video. So they simply reached not all of the questions. In most of the cases, the number of correct answers is quite high. The exceptions are the first ones of the first video (Aggregate) and all of the 6th video (Hardened Concrete II).

Attention Level

The reaction times of the students are part of the calculation of the attention level (see Section 5.5.2). Most of the students reached an attention level of 94% or higher. Only a very small number received a value below 70%.

Poll Analysis

As mentioned above, two videos were created for each topic. They were presented to the students in a combined way and after the two videos of a topic, some polling questions were shown to the students. This was done to encourage the students to a comparative competition. The questions of the polls and the results are shown by Figure 7.23. On the x-axis of the diagram of each question, the topics are grouped by the two videos. In contrast to that, the y-axis prints the number of students who provided an answer to the corresponding possible answer of the poll. It seems to be obvious,

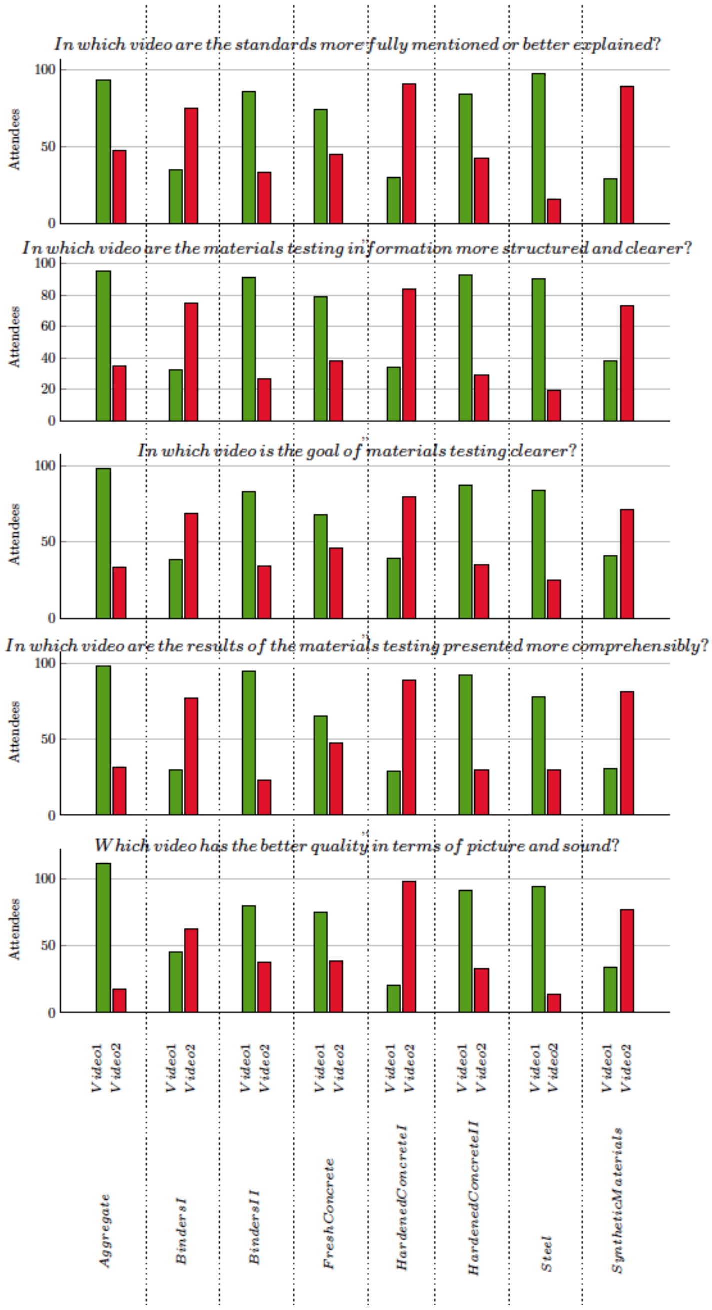



Figure 7.23: The outcome of the polls at the lab experiments 2017.

that the number of answering students is varying because not all students watched enough of the video to see the polls.

It is possible to observe two issues from the results. First, at all questions, the same video is selected more often than the other. This means that for instance at the video "Aggregate", the first video is the favorite of the students at all questions. In contrast to that, video number 2 is selected more often at all questions of the topic "Synthetic Materials". The second issue points out that at most of the topics, the outcome of the polls is very clear for each question. Only at the topics "Binders 1" and "Fresh Concrete" the outcome is narrower.

In addition to these evaluations of the multiple-choice questions and the polls, the teacher provided some feedback about this usage of LIVE. This feedback stated that the workload for evaluating the performance of the students in the laboratory experiments was considerably reduced by using the interactive videos. In addition, it was reported that the students showed a better grade point average with the usage of interactive videos in comparison to the grades at the same lecture before using the explained didactic concept.

### 7.3.4 LTI Login to Videos of Lab-Experiments 2018

Again the web platform was used at the course *Building Materials Basics*[11] at Graz University of Technology (Wachtler et al., 2019). In comparison to the usages, presented and evaluated by the previous sections, the following adaptions were made:

- automatic login
- course design for a small number of students
- suggested multiple-choice questions by the students

The first mentioned adaption consisted of the automatic login of the students at the web platform using the mechanism presented by Section 5.8. The second adaption was required because the course was offered additionally

[11] https://online.tugraz.at/tug_online/wbLv.wbShowLVDetail?pStpSpNr=207235&pSpracheNr=2, last accessed January 15, 2019

a second time in the year because of the huge number of students. However, this second run of the course was attended only by a smaller number of students and because of that, some adaptions to the course design were required. Furthermore, students were asked to submit multiple-choice questions of themselves. This was done to identify the areas of interest of the students.

Different Course Designs

As mentioned, the course was offered twice this year because of the large number of students. At the summer term, 127 students registered for the course and 121 completed the course. In contrast to that, the winter term was only attended by 7 students. A reason for this small number might be that the lecture collided with other major lectures in the winter term. Because of that, an adapted course design was developed. Figure 7.24 compares these two different flows of events at the summer and winter term.

It can be seen that at both variants, the lecture was divided in a theoretical and a practical part. In both cases, the theoretical part consisted of some frontal lectures presented by the teacher to the students. In addition, material experiments were shown to them. As at the lecture of the previous year (see section above), there were 8 different subject areas.

Large Number of Students

At the scenario for a larger number of students, the students were required to perform the material experiment by themselves and to create learning videos from the experiments. Because of that, there were finally 16 different videos (2 per subject area). The teacher combined the 2 videos of the same subject area to one video and published them using LIVE. In addition, she/he added some multiple-choice questions and polls. Now, the students were able to watch the videos and to answer the questions. For that, they were authenticated in an automatic way through the learning platform of the university. Finally, the students were asked to submit questions which are from high interest for them as a bonus task to earn extra points. These submitted questions should help the teacher to better understand the interests of the students.

Small Number of Students

In contrast to that, the concept for a small number of students did not require that the students create videos by themselves. This was done because there were simply not enough students to form enough groups for the different subject areas. Instead of that, the teacher used the videos from the previous summer term. This means that she/he again added some questions but no

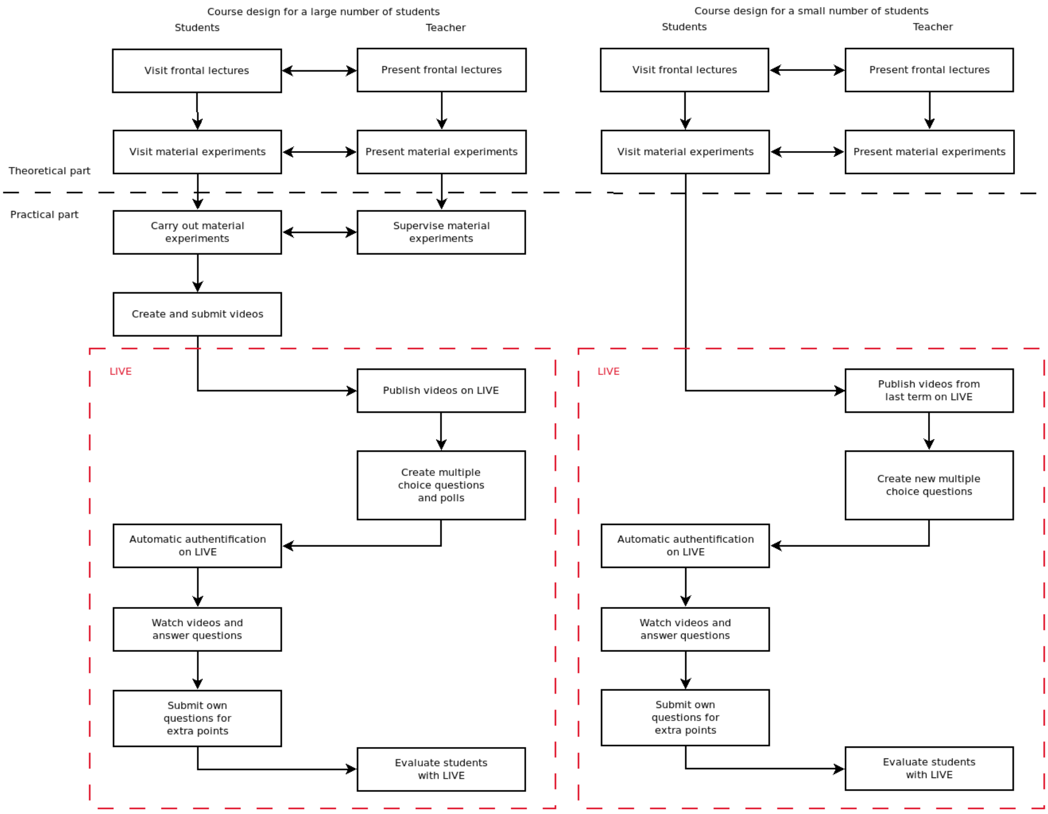



Figure 7.24: The different course designs used at the lab experiments 2018. The left side shows the flow of events for a large number of students and at the right hand side, the flow of events for a small number of students is printed.

polls. This was done because the videos were not created by the students of this year and due to that, the primary function of the polls, namely comparative assessment between the two videos, was not required. The rest of the course worked in a similar way.

Automatic Login

From a technical point of view, the automatic login at LIVE worked without any problems. This means that each student was able to login through the learning platform of the university and to watch the videos. In addition, the teacher was able to identify each student without problems because the data (e-mail and name) from the official student management system of the university were used for the automatic authentication. In contrast to the summer term of the previous year, this is an improvement because no students were unidentifiable. This means that at the summer term 2017, 12 out of 150 students used a user name which was not recognizable for the teacher and in the current term all students were identifiable.

Question Analysis

The results of the multiple-choice questions are shown by Figure 7.25. It can be seen that most of the students answered the questions correctly. The best question was Q3 from the subject area *Steel* where 121 students provided a correct answer and only 2 answered wrong. In contrast to that, the most difficult question seems to be Q2 from the subject area *Aggregate*. In this case 97 students answered correct and 27 wrong.

To make a statement regarding the correctness rates at the different subject areas all answers of all questions from a subject area are combined. As pointed out by Figure 7.26, the questions of the subject area *Steel* led to the highest correctness rate (475 correct, 17 wrong). In contrast to that, the lowest correctness rates happened at the subject areas *Aggregate* (417 correct, 79 wrong), *Hardened Concrete 2* (430 correct, 67 wrong) and *Hardened Concrete 1* (439 correct, 61 wrong).

Submitted Questions

As mentioned above, 121 students completed the course. From these students 31 submitted questions which should be embedded in further videos according to their opinion. These questions are compared to the questions created by the teacher. This is done to get an overview of the interests of the students in comparison to the focus of the teacher. The results are summarized by Figure 7.27. The dark blue bars are stating the complete number of submitted questions and the yellow ones are expressing how many of them addressed a different topic. The three bars printed in shades of green

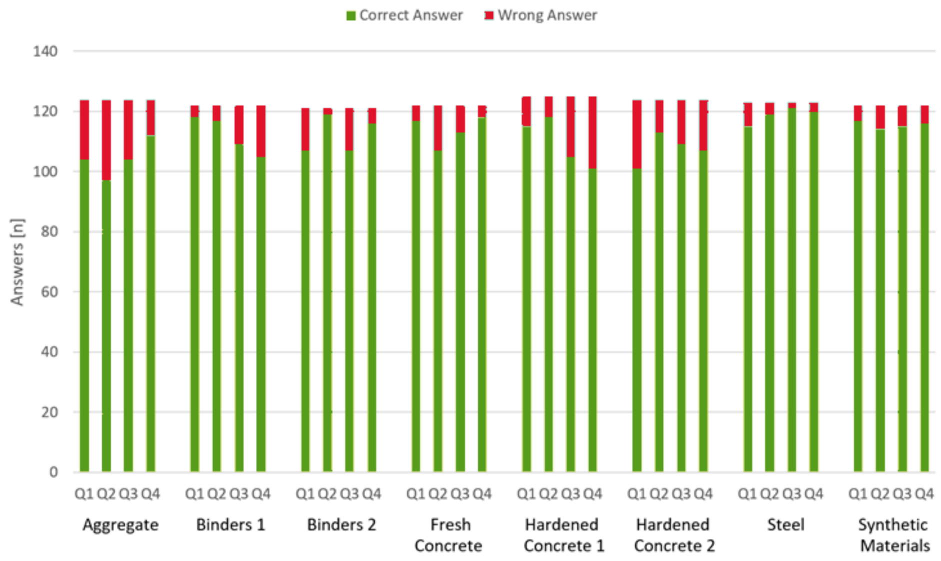



Figure 7.25: The results of the individual multiple-choice questions at the lab experiments 2018.

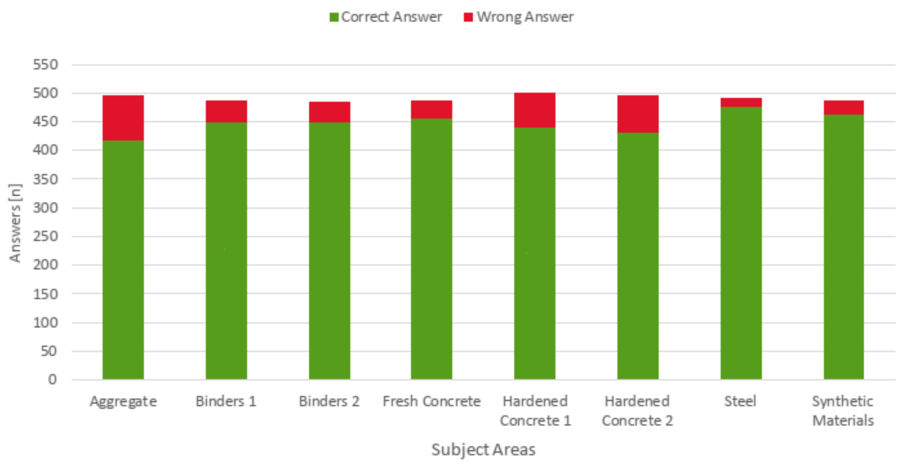



Figure 7.26: The summarized results of the multiple-choice questions at the lab experiments 2018.

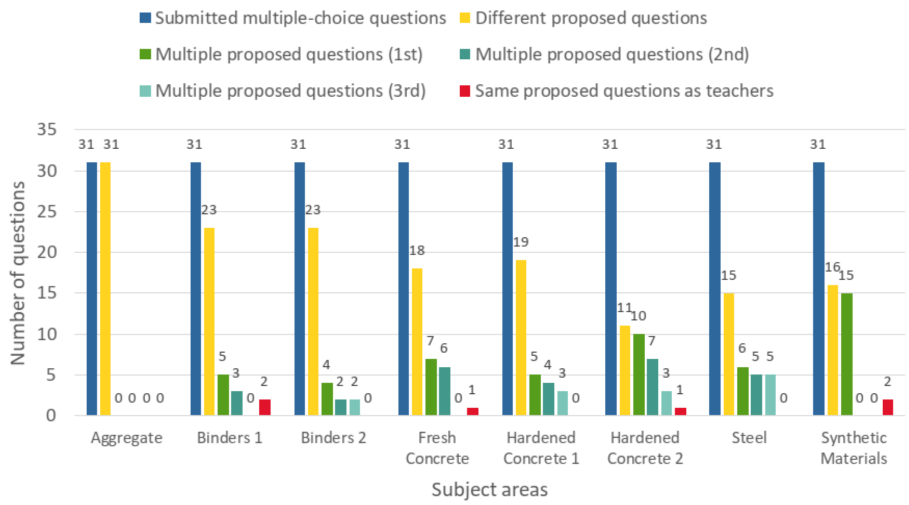



Figure 7.27: The submitted multiple-choice questions at the lab experiments 2018 are compared with the questions of the teacher.

are counting the three most addressed topics. Finally, the red bar states the number of submitted questions which are addressing the same topic as the questions of the teacher. It can be seen that all of the 31 submitting students provided a question for each subject area.

At the subject area *Aggregate*, all questions address a different topic and no question is related to a question of the teacher. In contrast to that, at the subject area *Hardened Concrete 2*, the students submitted 11 different questions and one specific topic is addressed by 10 students. The second mostly addressed topic consists of 7 submissions. 3 students address a third topic. One question is related to the questions of the teacher. In summary, it can be seen that the number of questions which are related to the questions of the teacher is not very high at all subject areas.

For a deeper analysis, the submitted questions are compared with the answers to the multiple-choice questions added by the teacher to the video. It is pointed out that the correctness rate is higher if the question is submitted by some students. This indicates that the submitted questions represent the interests of the students. Based on that, the teacher is able to adapt the focus in the course accordingly to help the students to better understand the relevant but probably not very interesting topics.

Feedback

To better understand the behavior of the students, they were asked to provide feedback regarding their approach for watching the videos and answering the questions. Most of the students operate according a variant of the following procedure:

1. watching the videos
2. answering the multiple-choice questions
3. re-reading the teaching materials, if there was an uncertainty in the given answer
4. re-watching the videos
5. answering the multiple-choice question with additional knowledge

The final results pointed out that the students performed a grade point average better than before using the concept of the videos provided by LIVE. An explanation for this better performance might be that the students are feeling more connected to the content of the course because of the questions in the videos. This seems to be valid because the reported approach to watch the videos (see above), suggests that the students are trying to answer the questions correctly.

# 8 Discussion

This chapter discusses the results of the evaluation of the web platform presented above. For that, the major issues which were observed are tried to explain. This means that at first, it is discussed why the platform is used mainly for videos instead of live-broadcastings. After that, various remarks provided by the students using LIVE are examined. This is followed by the observed fact that students are liking content related questions more than general ones. As stated by different usages, the placement of the questions in the video has an effect on the success rates of these questions. Because of that, the possible reasons and the accuracy of these effects is discussed as well.

In contrast to the success rates of the built-in questions, some of the evaluations pointed out, that the usage of LIVE influences the long-term learning success. Due to that, these influencing factors are examined in more detail. After that, the accuracy and correctness of the attention-profiling algorithm is discussed. The evaluation pointed out some observations regarding the workload of the teacher when using the web platform. These observations are analyzed in detail to understand the conditions where an optimization of this workload could be achieved. The web platform is used to monitor the attendance of the students at the videos or live-broadcastings. That's why it is required to discuss how genuine this monitoring is.

The data generated by LIVE could be used to analyze the performance of the students from a teacher's point of view and from the perspective of a researcher. Because of that, the possibilities of learning analytics are examined in detail. Finally, some technical issues were pointed out during the evaluation which are discussed.

## 8.1 Videos are More Important

At the beginning of the previous chapter, the different usages of the web platform for evaluation are listed in chronological order. It can be seen that live-broadcastings were used at the early usages only. This means that after the third usage only videos were used.

In order to explain this fact some possibilities arise:

- decreasing number of students
- additionally offered recordings
- compulsory attendance

Number of Students

The first possible reason for the reduced usage of LIVE at live-broadcastings might be that the number of students has a decreasing tendency. This was mainly observed at the usage at a larger course (see Section 7.1.3). It was shown that the number of students using the interactive live-broadcasting decreased considerably beginning with the third unit of the lecture. It seems to be the case that students are attending class if they want to be part of the lecture in real time.

Lecture Recordings

If they are not able to attend to the lecture in person, they still have the possibility to watch the additionally offered recordings of the whole lecture. This means that the motivation to watch the live-broadcasting is reduced due to the provided recordings as on-demand videos.

Compulsory Attendance

A further reason for the decreased usage at live-broadcastings might be that in some cases compulsory attendance was applied. This means that the students were required to attend to the lectures and because of that, the participation at live-broadcastings was not possible. This seems to be true because with the earlier versions of LIVE, it was not possible to monitor the attendance of the students in a satisfying way (see below).

In addition, it can be seen that the sole usage of videos started with the support of videos by the platform. This suggests that the feature of supporting videos with interactive components is in general favored in contrast to live-broadcastings. This trend was encouraged by the evolving offering of MOOCs which are requiring videos in many cases.

## 8.2 Student Feedback

At various usages, the students were asked to provide feedback regarding the usage of LIVE as well as the didactic concept applied by the lecture. This means that the feedback mainly addressed the advantages and disadvantages of interactive videos or live-broadcastings.

Advantages

In summary, the students reported mainly the following positive remarks:

- Content related questions placed in videos or live-broadcastings are helpful in multiple ways.
- The possibilities to trigger interactions are increasing the feeling to be part of the lecture.
- The attendance monitoring offers the possibility to fulfill the requirements of the course without attending in person.
- The interactive components are enabling to watch the videos or live-broadcastings more actively.
- Questions should be distributed evenly across the video or the live-broadcasting.

Disadvantages

As a contrast, they also stated some negative opinions which are summarized by the following list:

- Questions which are not related to the content are considered to be useless and disturbing.
- The interactive components are slowing down the process of watching.
- The questions should not be overused.

The first items of both, the positive feedback and the negative one, address the content of the questions. Because these are the most reported issues, the following section discusses them in detail.

Student Triggered Interactions

As explained by the section regarding the different methods of interaction, the students are able to invoke interactions by themselves. This includes the possibility to ask questions to the teacher. Many students reported that these features are valuable for them because they are feeling more connected to the lecture. This means that students are more integrated in the process of learning in contrast to a passive only lecture. Similar findings are reported by Carr-Chellman and Duchastel (2000) who created the statement that interactive components are key factors for successful online courses. In addition, it was pointed out by Lehner (2014) that students are favoring to be part of the course and that they are wanting to be challenged.

Attendance Monitoring

If LIVE was used to apply compulsory attendance the students reported that they considered this possibility to monitor their attendance a huge benefit. This statement was mainly motivated by avoiding the need to go to a lecture theater personally. From the students point of view, the attendance monitoring was seen in a positive way only. This suggests that they are able to understand the positive effects of such a monitoring reported by some studies (Rodgers, 2002) (Devadoss & Foltz, 1996). For a detailed analysis of the accuracy of this way of monitoring the attendance see Section 8.8.

Active Watching

The students additionally reported that they are able to work with the videos or the live-broadcastings more actively if interactive components are offered. A reason for this might be that with the help of the interactions, a passive medium is transferred to an active one. This means that students are now able to participate at the lecture which is helping them to sort out the important parts. A further possibility might be that the students are trying to answer the questions in a correct way (see Section 7.3.4). This might lead to increased attention and activity. The motivations for answering the questions could be grouped in four groups according to Cummins et al. (2016) as explained by Section 2.3.

Watching Speed

As mentioned above, the first negative feedback of the students is discussed by the following section. The second negative remark addresses the issue that the interactive components are slowing down the process of watching. This statement is motivated by different reasons. First, some students are watching videos with increased playback speed. This behavior happens mainly at longer videos (e.g. recordings of lectures). In this case, it is true that such a way of watching is not possible with this platform. However, it stands to be evaluated how effective such a fast watching is for the understanding of the content. In contrast to that, students sometimes also have the opinion that the questions are a slowing factor. Here, it has to be stated that if the students are watching the videos seriously they should be able to answer the questions without real problems very fast. This means that students who were reporting such statements are probably not really interested in the content of the video. To address such a faster watching, a platform was developed by Kovacs (2015). With this platform it is still not possible to run the playback faster, however, the videos are segmented into smaller parts. Each part is related to a question and the students are allowed to skip the segment if they are able to answer the question of the segment in a correct way.

Distribution

The last statements of the students at both, the positive and negative feedback, address the distribution of the questions. It is stated that students are favoring an even distribution in contrast to questions placed too narrow or too far apart. Students mainly supported these statements with the explanation that they want to benefit from the questions during the whole video and not only at some parts of them. Additionally, the tight placement of question was reported to be irritating. This feedback is confirmed by some observations regarding the correctness rates at the questions in comparison to their placement (see Section 8.4).

Rating

In addition to the feedback of the students in a written way, they were asked to rate the usage of interactive videos according to a grading system. The results presented by Section 7.3.2 pointed out that most of the students selected the two best grades. This indicates that the majority of the students liked or at least accepted the approach. Furthermore, it is visible that the assigned grades have a relation to the provided feedback across all usages. This means that the number of students who assigned better grades mainly reported positive remarks. In contrast to that, the negative claims correspond to the number of students who assigned bad grades.

## 8.3 Content-Related Questions are Favored by Students

As stated above, many students reported that they like questions which are related to the content of the video or the live-broadcasting more than general questions. At the content related questions students claimed some reasons for favoring them. These reasons could be summarized to the following ones:

- better understanding the content of the video or the live-broadcasting
- feedback if they understood things correctly
- indicator for the important parts

Recap Content

The first possible reason claimed that questions which are related to the content are helpful for the understanding of the mentioned content. The students are seeing a topic of the video or the live-broadcasting again and because of that, they are required to think about the content again.

Feedback

A further addressed reason was that the questions are providing feedback to the students. This means that they could use the results of the questions as a measurement which indicates how valid their understanding has been. This reason could be considered to be accurate if most of the content of the video or the live-broadcasting is covered by the questions.

Important Parts of the Content

In addition to that, students could use the questions as an indicator for the important parts of the video or the live-broadcasting. This means that the students are able to identify the key content by interpreting the questions as representatives. A platform was developed and evaluated by Kovacs (2015) which focused on the usage of the questions as an indicator for the key parts (see Section 2.2). It seems to be obvious that such an approach only works if the key parts are fully represented by the questions.

Disturbing and Irritating

Students stated that questions which are not related to the content are disturbing or irritating. This indicates that students are not aware of the reasons for these questions. Because of that, the occurrence of such questions is explained to the students at the beginning of a video or a live-broadcasting by a special interaction. It states that all kinds of questions are used to support the attention of the students. In general, it is clear that the questions are disturbing for students who only watch the videos in order to get the task done or for those who want to watch with increased speed (see above).

Live-Broadcastings

In the case of live-broadcastings some further considerations are required (see Section 7.1.2). Because the teacher is responsible for asking the questions during a live-broadcasting, it is important that she/he adapts the way of presenting to the questions. This means that if a question is asked the lecture should be paused while the students are trying to answer it. In addition, the difficulty of the questions should not be too high for some reasons. First, the lecture is slowed down too much if the students require too much time to answer the questions. Moreover, the calculation of the attention level of the students is based on the time required to answer the questions (see Section 5.5.2). It is clear that the attention level is decreasing if the questions are too difficult to be answered in time. Based on that, students are feeling uncomfortable if despite their best effort to answer the questions correctly, the attention level is low because of the longer time required to give an answer.

## 8.4 Question Placement

During the different steps of evaluation, it was observed that under specific conditions, the placement of the questions has an impact on the correctness rate. This means that at videos with a longer length (e.g. recordings of lectures), it is possible to observe some issues (see Section 7.1.4).

`Lazy Start`

The first one is called **Lazy Start** and covers the observation that the first question in a video is prone to be answered more often wrong. A reason for that might be that until this point, the students were not required to answer a question and because of that, they were irritated. However, this is a contradiction to previous usages at live-broadcastings where the correctness rates decreased along the timeline of the live-broadcasting. A different reason for the problems at the first question might be that it is typically placed in the region with a high drop-out rate (see Section 7.2.2). Because of that, it is possible to use the first question to grab the attention of the students even if this might lead to a reduced performance at it. The application of an introduction of the interactions at the beginning helped to address this issue.

`Correct after Question Pause`

A further observation is called **Correct after Question Pause** which means that the correctness rate at questions is quite high even if the timespan since the last content related question is long. To explain this observation the questions which are not related to the content could be evaluated. This means that if such a supporting general question is located between the content related questions the correctness rate remains stable at a high level. This indicates in contradiction to the feedback of the students (see above) that the general questions are of use even if they are not favored by the students.

Tight-Placed Errors

The observation named **Tight-Placed Errors** suggests that questions are answered more likely wrong if the space between them is too narrow. A possible explanation might be that the students are feeling irritated by closely following questions. This is supported by the feedback of the students which states that the questions should be distributed in an even way across the video or the live-broadcasting and that they should not be overused.

Shorter Videos

Some of these mentioned observations, namely **Lazy Start** and **Tight-Placed Errors** were evaluated in more detail at a class in school using shorter learning videos. The results are presented by Section 7.3.1. It was pointed out that the observation regarding the reduced correctness of the first question is also present at the shorter videos. This indicates that the mentioned reasons are plausible in such cases too. Based on the results of this usage, it is possible to state the time of occurrence of the first question where the performance is likely to be the best. So this position should be located at 20%-25% of the length of the video. This statement was confirmed by some evaluations which suggests to place the first question in the zone of high drop-out which is located in this area (see Section 7.2.2).

The observation named **Tight-Placed Errors** could not be observed at videos with a shorter length. A reason for that might be that because of the short length, the gaps between the questions are in general not very long. This means that the effect observed at the longer videos, where some questions could be placed very narrow and some others with a larger gap, is not present at the shorter videos. Because of that, a new hypothesis states that placement of the questions has a bigger effect on the success rates at longer videos in comparison to shorter ones.

## 8.5 Long-Term Learning Effects

In order to evaluate the long-term learning effects of the usage of the web platform presenting interactive components during videos or live-broadcastings to the students, different forms of evaluation were used. This included the following ones as presented by the previous chapter:

- comparing the grades of the lecture using LIVE with the grades of the same lecture before the usage of the interactive videos
- comparing the test results of a group of students using the interactive videos with a group taught in conventional ways

At the evaluation of the usage of LIVE at some lectures at Graz University of Technology, the long-term learning success was studied (see Sections 7.3.3 and 7.3.4). In both cases the grades of the students were compared to the grades of the lecture before the application of the presented didactic concept. It was pointed out that the students performed a grade point average better with the new concept with interactive videos. A reason for that might be that the students were confronted with the questions during the videos which encouraged them to increase their efforts. This is supported by the reported common way of watching videos employed by most of the students. This means that they are trying to answer the questions correct and because of that, they are watching the videos a second time after studying the related content more detailed.

Before and After

In contrast to the evaluation of the grades, a further study analyzed the long-term learning success by comparing the results of a test done by both, students who used interactive videos in a flipped-classroom concept and students who were taught with a classic frontal lecture in school. Section 7.3.1 prints the results which state that the group using interactive videos performed significantly better than the other one.

Different Ways of Teaching

When comparing the two different evaluations (analyzing grades or test results) it is visible that the usage of LIVE is strongly connected to a didactic concept. This means that it seems to be important that the application of interactive videos alone is not enough. There has to be an overall didactic concept including these videos. This statement is motivated by the fact that the web platform at its earlier usages had a decreasing tendency of active users. At these usages the interactive components were not part of the didactic concept. This means that they were only added to simply provide them. Later, with the integration of the interactive components in the concept of the lecture, the number of users as well as the performance

Part of the Didactic Concept

(attention level and question results) stabilized at a higher level through all videos of a lecture. A similar observation was made by Zhang et al. (2006). It stated that the offering of videos alone has no additional positive effect on the learning success of the students.

## 8.6 Attention Profiling

The attention profiling algorithm consists of two parts (see Section 5.5). The first one is responsible for logging the watched timespans of the students at the videos or the live-broadcastings. The second half of the algorithm is the calculation of a value which represents the degree of attention of the students at each watched timespan. This attention level is based on the reaction times to the interactive components, where the value gets worse with the increasing reaction time.

`Parameter Setting` As mentioned earlier, the reaction time is mainly based on the difficulty of the asked questions and their content (see Section 8.3). This suggests the parameters, defining the calculation of the attention level and the difficulty of the question, should be matched to each other. Because of the fact that the parameters for the calculation are defined by the administrator for all questions of an interaction method, the questions must be formulated in a way to match the set difficulty. This is required because the teacher of a lecture is not able to change the parameters for individual questions. It seems to be important to find a valid difficulty because the students become frustrated if the attention level is low even if they tried hard to find an answer.

`Seriousness of Watching` The explained recording of the watched timespans of a video or a live-broadcasting is the base for the profiling of the attention. This means that this part of the algorithm is able to state when which student watched which segment of a video or a live-broadcasting. However, it is clear that such a recording alone is not able to express how serious the corresponding student was while watching. Due to that, the calculation of the attention level can be used to address this issue. This statement seems to be valid because the reaction time based approach leads to a reduced attention level, if it took the students longer to answer the questions. Such an enlarged reaction time could be an indicator for a reduced attention. At least, the value is able to state if the students actually watched the video or just started the playback in the background. If this was the case they will miss most of the questions and because of that the calculated attention level will be very low.

## 8.7 Teacher Workload

On evaluating the workload of the teacher, when using interactive videos provided by LIVE, it was pointed out that under specific circumstances the workload was increased in contrast to a standard lecture. However, it was also shown that it is possible that the workload could be reduced with the usage of the web platform.

`Increased Workload`

When examining the case with the increased workload, it can be seen that this mainly happened if the interactive videos were additionally provided. This means that the lecture was held in its standard way and the usage of LIVE was a special offering for the students (see Section 7.1.2). It is clear that in this case the teacher is additionally required to create questions, to publish them via the platform and to evaluate them. In the case of a live-broadcasting the teacher has to stop presenting the content, while asking a question in order to enable the students to focus on the process of answering. This not only increases the workload, it also slows down the lecture. A partial solution for these issues is the delegation of the operation of LIVE to an assistant.

`Reduced Workload`

In contrast to that, also a reduced workload was observed during the evaluation in some cases (see Section 7.3.3). This happened at a lecture, where the interactive videos were the integral part of the didactic concept. In this case, the videos were used to evaluate the performance of the students at practical exercises in the lab. Prior to the videos, personnel was required to oversee and assess the performance of the students in the lab. With the interactive videos no additional personnel was required and the teacher simply used the analysis features of LIVE. The time required to create the questions was in no way as large as the amount of time for managing the personnel and to collect the assessment of the students. This suggests that the integration of the web platform in the didactic concept is also a benefit for the teacher and not only for the students (see above).

## 8.8 Attendance Monitoring

There are many positive effects for the students by employing compulsory attendance as explained by Section 2.5. It is clear that a mechanism to monitor the attendance of the students is required to enforce this compulsory attendance. The mentioned section presents some of such mechanisms used in standard classroom situations. A usage of LIVE at a lecture transferred the concept of compulsory attendance to videos (see Section 7.3.2).

Attention Profiling for Attendance Monitoring

To control the attendance of the students at the videos, the attention profiling algorithm of the web platform was used. This means that the teacher received a list from LIVE where the students were listed together with a value indicating how much of the video they watched. In addition, the teacher evaluated the built-in questions for the purpose of monitoring the attendance. This means that long reaction times or mainly wrong answers are considered to be an indicator for absence. To identify the individual students, they were required to use an individual username provided by the teacher.

Online versus Offline

When comparing this approach with the mechanisms typically used in standard classroom situations, it can be seen that the list of the students generated by LIVE, roughly corresponds to the signature list used in a lecture theater. There seems to be the same level of accuracy at both methods because at the offline variant, students are able to sign the list on behalf of others. At the online approach the students are able to share their credentials with other students so one student can watch the videos in the name of many others. The second layer of security employed by the online variant is the evaluation of the multiple-choice questions which could be used as a proof for the attendance. This is a very similar approach like the offline variant using ARS where the answers of the students are a proof for their attendance. However, the problem with shared credentials (online variant) or shared answering handsets (offline variant) is still present.

To address these problems, it is assumed that the implementation of the automatic authentication at the web platform is helpful. This means that with the automatic authentication, the students are required to use the credentials of the official student management system of the University. Because these credentials are providing access to sensitive data, it is valid to assume that they are not shared that easily.

Sharing Credentials

## 8.9 Learning Analytics

The different possibilities of learning analytics are explained by Chapter 6. These possibilities are divided in two groups. The first one consists of the features implemented by the web platform itself. In comparison to that, the second group of such features explains the possibilities using the downloadable data with external methods.

Internal Analyses

The mentioned first group includes different features to evaluate the watched timespans of the students and detailed analyses of the embedded questions. At the different usages presented by Chapter 7 it was possible to observe, that the teachers mainly used the watched timespan analysis (see Section 6.1.1). A reason for the preference of this feature might be that the presented list of the joined students in combination with their watched timespan, is a base for the evaluation of the students. This means that with this list it is possible to perform a very basic monitoring of the attendance. The feature showing details of an individual student (see Section 6.1.3), was used only if a student watched not the full video or live-broadcasting. To explain this behavior of the teachers, it seems to be valid to assume that they are only interested in the overall evaluation and in general not in the details. A further explanation may be that a potential heavy usage of the detailed analysis would increase the workload considerably.

A further analysis feature prints a timeline which provides an overview which part of the video or the live-broadcasting was watched how often (see Section 6.1.2). The teachers used this feature very rarely. It seems that they are not interested in detail in the distribution of the views across a video or a live-broadcasting. Furthermore, it could be that the teachers noticed that the timeline looks very similar most of the time. This means that the number of views is higher at the beginning and decreasing with the length of the video or the live-broadcasting. Based on that, it might be that the usefulness of such a diagram is reduced for the teacher. In contrast to that, researchers used this feature more often to gain the explained overview. This suggests that they are more interested in general evaluations and not in the analyses of individual students.

The teachers also used the evaluations of the embedded questions quite regularly. This seems easy to explain, because teachers are interested in the performance of the students at the questions. For that, there are two possible reasons. First, the answers of the questions could be used to validate the accuracy of the attendance monitoring (see above). Second, sometimes teachers are using the results of the questions for assessment. This means that the performance of the students at the questions is part of the final grade.

External Analyses

It can be seen that the built-in possibilities of learning analytics are mainly used by teachers to assess the performance of the students. In contrast to that, the external features (see Section 6.2) are of more use for researchers to evaluate the process of learning. The presented methods are valuable examples of how to use the data generated by LIVE to visualize certain information. For that, it is possible to use different kinds of plots. Section 7.2.2 evaluates the usage of the presented methods at a MOOC. It can be seen that the evaluated issues are of more interest for research purposes than for teaching. This suggests that the implementation of such features by the web platform is not practical because the platform itself is dedicated to the usage by teachers. As pointed out above, the features used by the teachers are mainly related to the watched timespans and the outcome of the questions. In contrast to that, the needs of research are varying, depending on the goal of the research. Due to that, the possibility to download the data as a spreadsheet seems to be the suitable method for researchers because the data could be analyzed with statistical programs much more efficient.

## 8.10 Technical Issues

During the usage of the web platform in a productive way in different versions, some technical problems were observed. This includes issues, regarding the user interface as well as problems with the algorithms.

`Fullscreen`

As reported, the very first prototype provided the live-broadcasting through an external streaming platform. This platform offered the possibility to switch the stream to fullscreen mode and because of that, it could happen that the interactions were not visible because LIVE was not able to control the fullscreen of the external platform. It seems to be obvious that this was a major problem because the missed interactions led to completely wrong results of the attention profiling algorithm. This issue was addressed by implementing the streaming functionalities directly in the platform. Because of that, no external streaming platform was required any longer. Based on that, the fullscreen mode is able to include the interactions as well.

`Finding Positions`

The teachers reported that the creation of questions at planned positions could be difficult and time consuming because they were required to find the position in the timeline of the video. To address this problem, two solutions seemed possible. The first one consisted of the implementation of buttons which allowed the teachers to jump through the video by a specific step length. On examining this solution it was shown that the selected positions are very different at each video and because of that, the jumping by step length was not very useful. A deeper analysis which consisted of speaking with the teachers pointed out that they are watching the videos before creating the questions and due to that, they are knowing the position where they want to place a question very exactly. This was the base for the second solution which is still in productive usage. In this case a text box allows the teachers to enter the position of the question by the means of typing.

`Interaction Dialog`

The interactions which are shown to the students during a video or a live-broadcasting are presented to them, using a dialog which is displayed in front of the playback. The size of this dialog window was part of heavy problems. At the beginning it was set to a height of approximately half of the main window. This worked in many cases, however if a question includes an image, the button to answer the questions vanished from sight and was only reachable by the means of scrolling. Because some students were not able to recognize that it is required to scroll down to the button, they were not able to answer the question in time. It is clear that this influenced the attention profiling algorithm in a negative way. The first solution to this problem set the height to a maximum so that the dialog window always covers the whole height of the main window. This solution looked not very appealing because many questions are of a small height. The final version implements a calculation of the height of the questions and sets it as the height of the dialog window.

`Attention Level Parameters` As mentioned above, it could be problematic if the difficulty of the questions is varying. This means that because of that, the students need different time to answer them. However, the parameters defining the calculation of the attention level (see Section 5.5.2) are set for all questions of a specific interaction method. A technical solution for this issue may be that the teachers could be enabled to change these parameters for single questions. However, there are some related problems which have to be considered. If the teachers are able to change these parameters, they are required to fully understand their meaning in order to get accurate results from the attention level calculation. This understanding consists not only of the basic meaning, it also includes the requirement that the teachers could state for each question how long the students will require to find an answer. Furthermore, the workload of the teachers will be increased if they have to set these parameters manually. Because of these issues, the teachers are not able to set the parameters, influencing the calculation of the attention level. Instead, they are instructed to use simple questions' because the questions main purpose should be the support of the attention of the students and not assessment.

`Scheduling Interactions` The development of the algorithm to schedule interactions during the videos is explained by Section 5.4.4. To reach this final version of the algorithm, several iterations were required. The main reason for these iterations is based on the evaluation which states that students are preferring an even distribution of the interactions. As pointed out above, this even distribution has some benefits for the students too. To analyze the distributions, all of them, gathered at the different usages, were searched for distributions with interactions, placed too narrow or too wide. The algorithm was adapted until no such unwanted distributions existed.

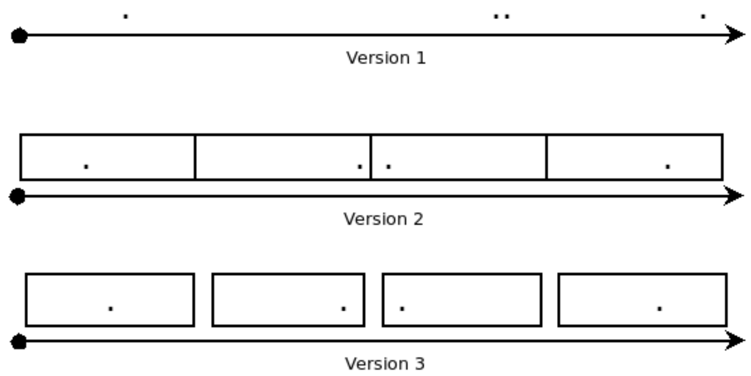



Figure 8.1: The different versions of the scheduling algorithm have different distributions.

Figure 8.1 illustrates the problems of the distributions at the different versions of the scheduling algorithm. Each version is represented by an arrow which resembles the timeline of a video or live-broadcasting and the interactions are rendered as dots. It can be seen that at the first version, completely uneven distributions were possible. This was the phase where students mainly reported the issues regarding the distribution. The reasons for the problem was identified to be the completely random distribution of the interactions. To address this problem, the so-called slot mechanism was introduced. The timeline is divided into slots of a given length and the interactions are placed randomly in a slot. With this improvement, the number of unwanted interactions was considerably reduced and the comments of the students addressed this issue not very often.

For the final version there was one improvement because the second version still had a problem. At the borders of the slots it was still possible that interactions were placed too narrow. For that, a gap between the slots was introduced. It can be seen that at the final version no unwanted distributions are possible.

# 9 Outlook

The evaluation and discussion pointed out that there are some issues which require further investigation. This includes the following ones:

- Are there different impacts of the placement of the questions at different video lengths?
- How strong is the successful usage of the platform connected to the didactic concept?

Question Placement

The first issue which requires more analyses addresses the observations made at different usages. These observations stated that the placement of the questions has an impact on the success rates of the questions. One of these observations pointed out that at longer videos the space between the questions should not be too small because this leads to more wrong answers at these questions. It was not possible to confirm this observation at shorter videos. Due to that, it is recommended to evaluate how the length of the video affects the recommendations for question placement in conjunction with the success rates at the questions.

Didactic Concept

In contrast to that, the second issue addresses the outcome of the evaluation which stated that the successful usage of the platform depends on the integration in the didactic concept. It was observed that the performance of the students as well as their acceptance of the interactive components is considerably higher if the usage of the platform is part of the didactic concept and not used as an additional tool. It is recommended to further evaluate how strong the mentioned connection is. For that, it could be analyzed which didactic concept is the best for the integration of interactive videos. Furthermore, it could be evaluated to which degree the success of the interactive videos is influenced by the didactic concept.

In addition to further research regarding the different didactic concepts, the evaluation of the web platform pointed out the need for more interaction methods. This includes the following ones:

- simple programming tasks
- math exercises

Programming Tasks

The first one should provide the possibility for the teacher to add programming tasks during the video at planned positions. This means that the teacher should be able to create the tasks. Such a task consists of a text-based description and a runnable solution. For automated testing, the teacher should be able to provide some input and the expected output. With these components it should be possible to implement the following workflow:

- The interaction occurs during a video and presents the description of the task.
- The students are able to enter their solution in a text box.
- This solution is run by the platform and fed with the input provided by the teacher.
- The output of this program is compared with the expected output.
- This leads to an evaluation of the correctness of the task.

Such an interaction method should be very usable at lectures which are explaining basic concepts of programming or algorithm design. In these cases, the tasks could be used to provide built-in exercises of the presented content. However, it is required to analyze how the complexity of the tasks affects the usefulness of the videos. This means that it should be evaluated if the programming tasks have a different influence on the learning success with different difficulty.

Math Exercises

The second proposed interaction method should be able to offer planned interactions which are presenting math exercises. For such a method, multiple levels of difficulty are possible. A simple variant could focus on basic math operations like addition, subtraction, division and multiplication. The actual exercises could be obtained from the APIs provided by the corresponding application offered at Graz University of Technology [1]. These applications

[1] https://schule.learninglab.tugraz.at, last accessed January 30, 2019

provide the analysis of the exercises too. A further more complex variant could focus not on actual calculations and their results, it could ask for equations. This means that for instance, the teacher could ask the students to write down the Pythagorean theorem. When using a standardized language for writing the equations (e.g. MATHML[2]), it is possible to write down complex equations in a simple way. It is additionally possible to analyze the answers of the students in an automatic form. Again, it is necessary to evaluate the usage of such interaction methods in combination with the didactic concept of the lecture.

[2] https://www.w3.org/Math/, last accessed January 30, 2019

# 10 Conclusion

Interaction and communication are considered to be very important factors of the process of learning, mainly at online courses (Carr-Chellman & Duchastel, 2000). This statement is motivated by the assumption that they are valuable tools to manage the attention of the students. Because online videos are currently used again for learning purposes, it seems to be from high importance to provide such features at videos and live-broadcastings. The increased need for supportive tools in videos and live-broadcastings is additionally encouraged by the fact that such media were not considered to be very useful for learning.

`Interaction and Communication`

To address these reasons for providing interactive components for online videos and live-broadcastings, this work is based on the research questions presented by Section 1.2. To answer the main question, it is divided into five sub-questions which are tried to be answered by this work. The foundations of this work were laid by the author's master thesis (see Chapter 4). For that, a very simple prototype was developed and evaluated. This prototype was able to provide interactive questions in an automatic or manual way in real time at live-broadcastings. Because of the limitations of this prototype, a new web platform has been developed which is based on the concepts of the first prototype only.

`Research Questions`

The implementation and the functionalities of the web platform, called LIVE, are presented by Chapter 5. The platform is able to support on-demand videos and live-broadcastings with different interactive components. For that, the interactive components are displayed during the video or the live-broadcasting.

`Web Platform`

It is clear that the platform requires registered and authenticated users in order to identify the students at the analysis done by the teacher. For that, different groups of users (students, teachers, researchers and administrators)

`Users and their Tasks`

are available. The administrators are able to access the special panel, which enables them to change the configuration of the platform and to assign the groups to the users. The general workflow of the other groups could be summarized to the following tasks:

- At first, a teacher creates a video or a live-broadcasting and the offered interactive components.
- After that, the students are able to watch them and to participate at the interactions.
- Now, the teacher can analyze the performance of the students by viewing different offered analysis features.
- The researchers can download all of the data generated by the platform in the form of spread sheets to analyze the learning behavior of the students.

Interaction Types The different interaction methods are implemented as independent plugins. Each plugin has to be of a given type. By setting a type, the way of invoking the interactions is defined. The first type declares that the interactions are occurring in an automatic and random way. Interaction methods of this type are typically presenting questions which are not related to the content of the video or the live-broadcasting. The second type states that its interactions have to be invoked by a manual action of the students. To perform such an action, the required control elements are placed on the right side of the video or the live-broadcasting. The third type is the opposite version of the second one. This means that the teacher has to trigger the interactions manually using offered control elements. It seems to be obvious that this type is only available for live-broadcastings because in this case, the teacher is present to invoke the interactions. Finally, the fourth type represents interactions which have to be planned by the teacher before the video or the live-broadcastings is released. This planning includes the creation of the interactions at specific positions in the timeline of the video or the live-broadcasting.

Interaction Methods Currently, there are two interaction methods of the first type. One method presents simple and general questions to the students randomly and automatically. The other one does the same with a CAPTCHA instead of the questions. Interaction methods, which could be invoked by the students

(Type 2), are offering the following features. The first one enables the students to ask a text-based question to the teacher. At a live-broadcasting, the question is sent to the teacher in real time with the possibility to submit an answer. In contrast to that, at a video the question is listed at the backend of the platform for the teacher. A further interaction method offers a slider to the students which enables them to set their current level of attention. The final interaction method of the second type provides the possibility to report a technical problem to the teacher.

Of the third type only one interaction method exists. It implements the opposite version of the first interaction method of the second type. This means that it enables the teacher to ask text-based questions to the students in real time at live-broadcastings. She/he is additionally able to view the answers of the students instantly.

Interaction methods of the fourth type (occurring at planned positions) are most useful at videos. Currently, two interaction methods implement this type. The first one offers multiple-choice questions and the second one provides text-based questions. In both cases, the teacher is required to plan them before releasing the video or the live-broadcasting according to their type. The answers of the students are shown to the teacher at the backend in a systematic way.

`Scheduling Algorithm`

The different types of interactions, which are shown to the students, are requiring an algorithm to schedule them. This algorithm (see Section 5.4.4) uses the concepts of a state machine. This means that each interaction is such a machine and depending on the current state, the interaction is shown or not.

`Attention Profiling Algorithm`

In addition to the analysis features of the presented interaction methods which provide a detailed analysis of the understanding of the students of the content of the videos or the live-broadcastings, there is the so-called attention profiling algorithm. It consists of two parts. The first one records the watched timespans of the students in absolute and relative values. With these values it is possible to state when which student watched which part of the video or the live-broadcasting. The second part calculates a level of attention for each watched timespan. This attention level tries to indicate how attentive the student was during the timespan.

Evaluation The evaluation of the platform was performed at different usages (see Chapter 7). These usages consisted of live-broadcastings or recordings of lectures, massive open online courses, videos used for the flipped classroom concept in a school and for the presentation of lab experiments. It was pointed out, that the benefits for the students and for the teacher are increased if the interactive components during the videos or live-broadcastings are part of the didactic concept of the lecture. This means that at earlier usages the functionalities of the platform were offered only additionally to the typical lecture. In these cases, the number of students as well as the attention level decreased. Furthermore, the satisfaction of the students with the interactive components and the calculated attention level was lower. In contrast to that, these values and the feedback returned much better results at later usages, where the videos and the interactive questions were integral parts of the didactic concept. In such scenarios it was additionally possible to measure an increased learning success of the students and a more efficient workload of the teacher.

Attendance Monitoring According to Rodgers (2002), there are many benefits of employing compulsory attendance. To implement compulsory attendance also at videos and live-broadcastings, a mechanism to monitor the attendance of the students is required. For that, the attention profiling algorithm can be used. It was shown, that it is possible to reach a similar level of accuracy as with methods used in standard classroom situations to monitor the attendance.

Learning Analytics All of the analysis features are offering a detailed evaluation of the performance of the students to the teacher. In addition to that, the possibility to download all the data generated by LIVE as spread sheets enables researchers to perform learning analytics in many different forms. Some possibilities are shown by Section 6.2 and practically used as shown by Section 7.2.2. The examples are able to provide a detailed insight in the behavior of the students.

It can be seen that with the implementation and the evaluation of the platform, the five sub questions of the research question are answered. With these answers it is possible to state, according to the research question, students and teachers are able to benefit from interactivity used in different learning environments, if the results of this work are taken into account.

# Appendix

# Bibliography


Andrews, K. (2012). Information architecture and web usability lecture notes. Retrieved October 18, 2012, from http://courses.iicm.tugraz.at/iaweb/iaweb.pdf

apa. (2009). Anwesenheitskontrolle mit dem iPhone.

Bai, Y. & Chang, T.-S. (2016). Effects of class size and attendance policy on university classroom interaction in taiwan. *Innovations in Education and Teaching International*, *53*(3), 316–328.

Bailie, J. L. & Jortberg, M. A. (2009). Online learner authentication: Verifying the identity of online users. *MERLOT Journal of Online Learning and Teaching*, *5*(2).

Beck, K. (2003). *Test-driven development: By example*. Addison-Wesley Professional.

Bootstrap Team. (2018). Bootstrap documentation. Retrieved October 9, 2018, from https://getbootstrap.com/docs/

Caplan, R. (2018). Thinking about design is hard, but not thinking about it can be disastrous. Retrieved October 9, 2018, from http://www.goodreads.com/quotes/540494-thinking-about-design-is-hard-but-not-thinking-about-it

Carr-Chellman, A. & Duchastel, P. (2000). The ideal online course. *British Journal of Educational Technology*, *31*(3), 229–241. doi:10.1111/1467-8535.00154

Cummins, S., Beresford, A., & Rice, A. (2016). Investigating engagement with in-video quiz questions in a programming course. *IEEE Transactions on Learning Technologies*, (1), 1–1.

Cutrim, E. S. (2008). Using a voting system in conjunction with interactive whiteboard technology to enhance learning in the english language classroom. *Comput. Educ. 50*, 338–356.

Devadoss, S. & Foltz, J. (1996). Evaluation of factors influencing student class attendance and performance. *American Journal of Agricultural Economics*, *78*(3), 499–507.

Django Team. (2018a). Django documentation. Retrieved October 9, 2018, from https://docs.djangoproject.com/en/1.11/

Django Team. (2018b). Django: Design philosophies. Retrieved October 9, 2018, from https://docs.djangoproject.com/en/1.11/misc/design-philosophies/

dpa. (2015). Webvideopreise für "Tubeclash"und Kelly Misses Vlog. Retrieved February 22, 2016, from http://futurezone.at/digital-life/webvideopreise-fuer-tubeclash-und-kelly-misses-vlog/135.990.258

Ebner, M., Stickel, C., & Kolbitsch, J. (2010). Iphone/ipad human interface design. In *Symposium of the austrian hci and usability engineering group* (pp. 489–492). Springer.

Ebner, M., Wachtler, J., & Holzinger, A. (2013). Introducing an information system for successful support of selective attention in online courses. In *Universal access in human-computer interaction. applications and services for quality of life* (pp. 153–162). Springer.

Faulkner, S., Eichholz, A., Leithead, T., Danilo, A., & Moon, S. (2017). Html5 w3c recommendation. Retrieved October 9, 2018, from https://www.w3.org/TR/html5/

Gaytan, J. & McEwen, B. C. (2007). Effective online instructional and assessment strategies. *American Journal of Distance Education*, *21*(3), 117–132. Retrieved from http://www.tandfonline.com/doi/abs/10.1080/08923640701341653

Haintz, C., Pichler, K., & Ebner, M. (2014). Developing a web-based question-driven audience response system supporting byod. *J. UCS*, *20*(1), 39–56.

Heinze, H. J., Mangun, G. R., Burchert, W., Hinrichs, H., Scholz, M., Münte, T. F., ... Hillyard, S. A. (1994). Combined spatial and temporal imaging of brain activity during visual selective attention in humans. *Nature*, *372*, 543–546.

Helmerich, J. & Scherer, J. (2007). Interaktion zwischen Lehrenden und Lernenden in Medien unterstützten Veranstaltungen. In M. H. Breitner, B. Bruns, & F. Lehner (Eds.), *Neue trends im e-learning* (pp. 197–210). Physica-Verlag HD.

Hernandez, J., Ortiz, A., Andaverde, J., & Burlak, G. (2008). Biometrics in online assessments: A study case in high school students. In *18th international conference on electronics, communications and computers conielecomp 2008,* (pp. 111–116).

Holovaty, A. & Kaplan-Moss, J. (2012). The django book. Retrieved October 9, 2018, from http://www.djangobook.com/

Jacobs, P. F. (1992). *Rapid prototyping & manufacturing: Fundamentals of stereolithography*. Society of Manufacturing Engineers.

Janzen, D. & Saiedian, H. (2005). Test-driven development concepts, taxonomy, and future direction. *Computer*, *38*(9), 43–50.

jQuery Team. (2018). Jquery documentation. Retrieved October 9, 2018, from https://jquery.com/

Khalil, H. & Ebner, M. (2013). Interaction possibilities in moocs–how do they actually happen. In *International conference on higher education development* (pp. 1–24).

Kim, J., Guo, P. J., Seaton, D. T., Mitros, P., Gajos, K. Z., & Miller, R. C. (2014). Understanding in-video dropouts and interaction peaks inonline lecture videos. In *Proceedings of the first acm conference on learning@ scale conference* (pp. 31–40). ACM.

Kovacs, G. (2015). Quizcram: A question-driven video studying interface. In *Proceedings of the 33rd annual acm conference extended abstracts on human factors in computing systems* (pp. 133–138). ACM.

Kovacs, G. (2016). Effects of in-video quizzes on mooc lecture viewing. In *Proceedings of the third (2016) acm conference on learning@ scale* (pp. 31–40). ACM.

Latessa, R. & Mouw, D. (2005). Use of an audience response system to augment interactive learning. *Family Medicine*, *37*(1), 12–14.

Lehner, F. (2014). Interaktive Videos als neues Medium für das eLearning. *HMD Praxis der Wirtschaftsinformatik*, *48*(1), 51–62. doi:10.1007/BF03340549

Mason, J. (2002). Linking qualitative and quantitative data analysis. In *Analyzing qualitative data* (pp. 103–124). Routledge.

Moran, J. & Desimone, R. (1985). Selective attention gates visual processing in the extrastriate cortex. *Science*, *229*, 782–784.

Neuman, W. L. (2013). *Social research methods: Qualitative and quantitative approaches*. Pearson education.

Park, K. H. & Kerr, P. M. (1990). Determinants of academic performance: A multinomial logit approach. *The Journal of Economic Education*, *21*(2), 101–111.

Pekrun, R., Goetz, T., Titz, W., & Perry, R. P. (2002). Academic emotions in students' self-regulated learning and achievement: A program of qualitative and quantitative research. *Educational Psychologist*, *37*(2), 91–105.

Probst, S. (2012). Django model-template-view(mtv) - cc-by-sa 3.0/at. Retrieved November 18, 2012, from http://de.wikipedia.org/w/index.php?title=Datei:MTV_Django.svg&page=1&filetimestamp=20120208222359

Ricketts, C. & Wilks, S. J. (2002). Improving student performance through computer-based assessment: Insights from recent research. *Assessment & Evaluation in Higher Education*, *27*(5), 475–479. Retrieved from http://www.tandfonline.com/doi/abs/10.1080/0260293022000009348

Rodgers, J. R. (2002). Encouraging tutorial attendance at university did not improve performance. *Australian Economic Papers*, *41*(3), 255–266.

Romer, D. (1993). Do students go to class? Should they? *Journal of Economic Perspectives*, *7*(3), 167–174.

Salomon, G. (1984). Television is easy and print is tough: The differential investment of mental effort in learning as a function of perceptions and attributions. *Journal of educational psychology*, *76*(4), 647.

Sandelowski, M. (2000). Combining qualitative and quantitative sampling, data collection, and analysis techniques in mixed-method studies. *Research in nursing & health*, *23*(3), 246–255.

Schwan, S. & Riempp, R. (2004). The cognitive benefits of interactive videos: Learning to tie nautical knots. *Learning and instruction*, *14*(3), 293–305.

Shiffrin, R. M. & Gardner, G. T. (1972). Visual processing capacity and attentional control. *Journal of Experimental Psychology*, *93*(1), 72–82.

Silliman, S., Abbott, K., Clark, G., & McWilliams, L. (2004). Observations on benefits/limitations of an audience response system. In *Proceedings of the american society for engineering education annual conference*. Salt Lake City, Utah.

Sim, G., Holifield, P., & Brown, M. (2004). Implementation of computer assisted assessment: Lessons from the literature. *Research in Learning Technology*, *12*(3).

Spitzer, H., Desimone, R., & Moran, J. (1988). Increased attention enhances both behavioral and neuronal performance. *Science*, *240*, 338–340.
Stowell, J. R. & Nelson, J. M. (2007). Benefits of electronic audience response systems on student participation, learning, and emotion. *Teaching of Psychology*, *34*(4), 253–258. Retrieved from http://www.tandfonline.com/doi/abs/10.1080/00986280701700391
Tembrink, C., Szoltysek, M., & Unger, H. (2013). *Das Buch zum erfolgreichen Online-Marketing mit YouTube*. O'Reilly Verlag. Retrieved from https://books.google.at/books?id=eSCwAgAAQBAJ
Tobin, B. (2005, November 7). Audience response systems. Retrieved October 9, 2012, from http://med.stanford.edu/irt/edtech/contacts/documents/2005-11_AAMC_tobin_audience_response_systems.pdf
Vlissides, J., Helm, R., Johnson, R., & Gamma, E. (1995). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.
Wachtler, J. (2012). *Echtzeit-Interaktionsformen für Web-Inhalte*. Graz Universtiy of Technology.
Wachtler, J. & Ebner, M. (2014a). Attention profiling algorithm for video-based lectures. In *Learning and collaboration technologies. designing and developing novel learning experiences* (pp. 358–367). Springer.
Wachtler, J. & Ebner, M. (2014b). Support of video-based lectures with interactions-implementation of a first prototype. In *World conference on educational multimedia, hypermedia and telecommunications* (Vol. 2014, *1*, pp. 582–591).
Wachtler, J. & Ebner, M. (2015). Impacts of interactions in learning-videos: A subjective and objective analysis. In *Edmedia: World conference on educational media and technology* (Vol. 2015, *1*, pp. 1642–1650).
Wachtler, J. & Ebner, M. (2017). On using interactivity to monitor the attendance of students at learning-videos. In *Edmedia: World conference on educational media and technology* (pp. 356–366). Association for the Advancement of Computing in Education (AACE).
Wachtler, J. & Ebner, M. (2019). Scheduling interactions in learning videos: A state machine based algorithm. *International Journal of Learning Analytics and Artificial Intelligence for Education*, *1*(1), 58–76.
Wachtler, J., Geier, G., & Ebner, M. (2015). Design and evaluation of a user interface for an interaction supported video platform.

Wachtler, J., Hubmann, M., Zöhrer, H., & Ebner, M. (2016). An analysis of the use and effect of questions in interactive learning-videos. *Smart Learning Environments*, *3*(1), 13.

Wachtler, J., Khalil, M., Taraghi, B., & Ebner, M. (2016). On using learning analytics to track the activity of interactive mooc videos. In *Se@ vbl@ lak* (pp. 8–17).

Wachtler, J., Scherz, M., & Ebner, M. (2018). Increasing learning efficiency and quality of students homework by attendance monitoring and polls at interactive learning videos. In *Edmedia+ innovate learning* (pp. 1357–1367). Association for the Advancement of Computing in Education (AACE).

Wachtler, J., Scherz, M., & Ebner, M. (2019). Automatic authentication of students at an interactive learning-video platform. In *Proceedings of edmedia + innovate learning 2019* (pp. 715–728). Amsterdam, Netherlands: Association for the Advancement of Computing in Education (AACE). Retrieved from https://www.learntechlib.org/p/210069

Yan, X. & Gu, P. (1996). A review of rapid prototyping technologies and systems. *Computer-Aided Design*, *28*(4), 307–318.

Zhang, D., Zhou, L., Briggs, R. O., & Nunamaker Jr, J. F. (2006). Instructional video in e-learning: Assessing the impact of interactive video on learning effectiveness. *Information & management*, *43*(1), 15–27.