Journal of Universal Computer Science, vol. 20, no. 1 (2014), 39-56
submitted: 9/6/13, accepted: 30/9/13, appeared: 1/1/14 © *J.UCS*

Developing a Web-Based Question-Driven Audience Response System Supporting BYOD

Christian Haintz
(Department of Social Learning
Graz University of Technology, Graz, Austria
christian.haintz@cnc.io)

Karin Pichler
(Department of Social Learning
Graz University of Technology, Graz, Austria
karin.pichler@cnc.io)

Martin Ebner
(Department of Social Learning / Information Technology Services
Graz University of Technology, Graz, Austria
martin.ebner@tugraz.at)

Abstract: Question-driven Audience Response Systems (ARSs) are in the focus of research since the 1960s. Since then, the technology has changed and therefore systems have evolved too. This work is about conception and implementation of the web-based ARS RealFeedback which uses the principle of bring your own device (BYOD). A state-of-the-art analysis compares the features of existing web-based ARSs. The most important findings are used for the conception and the implementation of the system. Thinking-aloud tests, and the first usages during lectures confirm that the chosen requirements are very significant and valuable for lecturers.
Key Words: question-driven, audience response system, bring your own device, web-based
Category: L.0.0, L.3.0, L.7.0

1 Introduction

"Audience response systems (ARSs) or clickers, as they are commonly called, offer a management tool for engaging students in the large classroom."
[Caldwell, 2007]. Audience response systems are used to interact with students during the lecture. They consist of input devices for students, and a central server as well as a software, for the communication and the calculation of the results. The lecturer normally asks questions (commonly multiple choice questions) with the ARS and all students can answer them by pressing a button - representing their answer - on the input device. The central server collects all results and presents them immediately to the lecturer [Caldwell, 2007, Liu et al., 2003].

ARSs have engaged researchers since the 1960s [Judson and Sawada, 2002]. The first precursors of ARSs were tested and developed in military scenarios

[Froehlich, 1963] and colleges [Boardman, 1968] at that time. However, since the technology was not as highly developed as nowadays, these ARSs were very complex and expensive systems. To use ARSs, the whole room must have been wired since there was no other way for the input devices to communicate.

One very famous ARS was Classtalk, which was developed in 1996 [Dufresne et al., 1996]. This system offered new ways and possibilities of interacting with students during lectures. At that time, different styles of lecturing with ARSs evolved. To name a few: peer instruction [Mazur, 1997], question cycle [Dufresne et al., 1996] followed by question driven instruction [Beatty et al., 2006] and technology enhanced formative assessment (TEFA) [Beatty and Gerace, 2009].

ARSs have changed, since the rise of wireless technologies [Ebner, 2009]. The input devices were no longer connected via cable but via wireless connections and were called Wireless Internet Learning Devices (WILD) [Roschelle, 2003]. The system became portable and could be used in different lecture halls instead of a single one.

In the last years, the usage of mobile internet devices such as smart phones, tablets, and laptops has increased considerably [Ebner et al., 2012]. Students bring their own devices to lectures - this policy is called Bring Your Own Device - BYOD [Lennon, 2012, Logicalis, 2012, Wong et al., 2011]. Despite the risks of BYOD [Thomson, 2012] it seems to be obvious to use the devices of the students as input devices for ARSs.

ARSs provide students a way of interaction in massive courses. This research publication covers the topic of developing a web-based ARS, which uses mobile Internet devices of students as input devices. A state-of-the-art analysis shows the most crucial facts, which must be kept in mind when developing such a system. Out of these points, different requirements evolved. Afterwards, the implementation is being described. The results of interviews with lecturers who use the system during their lecture are presented afterwards. The conclusion and future work follow.

2 State-of-the-Art Comparison

To get a broad overview of today's ARSs twelve, web-based ARSs are evaluated and compared against each other. The different systems are shown in table 1. They were chosen according to fulfill a number of use cases listed below [Pichler, 2013]:

- creating questions
- opening/starting the questions for voting
- voting for answers
- reviewing results with a purpose-built lecturer interface

System	Website
Socrative	www.socrative.com
TopHatMonocle	www.tophatmonocle.com
SMSPoll	www.smspoll.net
mQlicker	www.mqlicker.com
ClickerSchool	www.clickerschool.com
PollEverywhere	www.polleverywhere.com
understoodit	www.understoodit.com
pinnion	www.pinnion.com
FreeMobilePolls	www.freemobilepolls.com
LectureTools	www.lecturetools.com
Pingo	www.pingo.ubp.de
Mentimeter	www.mentimeter.com

Table 1: The ARSs listed were compared against each other based on [Pichler, 2013]. Each website was visited last on 12th of April 2013.

2.1 Criteria

Afterwards different groups of measurable criteria are defined. Different ARSs are analyzed to find the essential features and to define the groups. All ARSs listed in table 1 are compared against the groups of criteria to get an overview of the important features of state-of-the-art ARSs. The following groups of criteria are defined:

General features

This group contains general features of an ARS. The different ways of responding to a question are analyzed as well as the functionality for managing questions (sorting questions, grouping questions, etc.). Another important point in this group is the feature of clearing the results of a question without losing them. This feature allows lecturers to take a look at past results and compare them with more recent results. It is also taken into account whether the system provides anonymous voting or not.

Question types

The different questioning types, which are provided by the ARSs are analyzed in this group. Question types can be classified in: multiple/single choice questions, yes/no questions, open questions with free text answer, sorting problems, matching problems, and image quizzes.

Question features

In this group, the different options for questions are analyzed. This group contains features like hiding votes until the voting is finished, showing results in total numbers and percentage, defining the correct answer, and allowing to enter math equations.

Visualization

The different types of visualizations, which are provided by the ARSs, are compared within this group. The visualization for the lecturer as well as for the student is analyzed.

Reporting and statistics

Some of the systems provide very detailed reports for the lecturer whereas others provide no reports or statistics at all. The different approaches are compared within this group.

2.2 Findings

The result of the comparison of the different ARSs against the groups, defined in section 2.1, are listed in table 2 and table 3. Those features, which occur in every ARSs are interpreted as important and valuable feature. These features are listed below.

Finding 1 *Voting for answers via a web site.*

Different ways of responding are available in the examined ARSs (Twitter, SMS, web site, mobile application). The comparison shows that all of the compared ARSs use a web site for responding. It can be assumed that this is the most important way to interact with the users. The reason therefore might be that a web site can be viewed with any Internet connected device.

Finding 2 *Grouping questions.*

Ten out of twelve systems offer a way of grouping questions. Especially in education this is a very valuable feature because the lecturer has the possibility to group questions according to different lectures.

Finding 3 *Sorting questions.*

Sorting questions is another feature, which is provided by eight out of twelve ARSs.

System	Socrative	Top Hat Monocle	SMS Poll	mQlicker	Clicker School	Poll Everywhere	understoodit	Pinnion	Free Mobile Polls	LectureTools	PINGO	Mentimeter
General Features												
Responding via SMS		•	•			•		•		•		
Responding via website	•	•	•	•	•	•	•	•	•	•	•	•
Responding via mobile app	•	•			•			•				
responding via Twitter						•						
Group questions	•	•		•	•	•	•	•	•	•		•
Sort questions	•	•	•		•	•		•		•		•
Copy questions		•				•		•				
Start / stop questions automatically						•						
Clear data of questions without losing previous data	•	•		•	•	•	•					
Bulk start / stop / clear / delete	•	•			•	•		•	•	•		•
Anonymous voting		-	•	•		•	•	•	•		•	•

Table 2: The results of the comparison for the general features are shown in this table [Pichler, 2013]. The groups of criteria are separated by horizontal lines.

Finding 4 *Bulk start/stop/clear/delete.*

Eight out of the compared twelve ARSs provides starting, stopping, clearing, or deleting several questions at once.

Finding 5 *Anonymous voting.*

Literature shows that anonymous voting is important for students [Bruff, 2009, Gauci et al., 2009, Abrahamson, 1999, Roschelle, 2003, Abrahamson, 2006, Collins et al., 2008, Feldman and Capobianco, 2007, Hoffman and Goodwin, 2006], therefore eight out of twelve systems provide anonymous voting for students. When voting is anonymous, students do not have fear of being wrong because nobody sees how each individual responded.

System	Socrative	Top Hat Monocle	SMS Poll	mQlicker	Clicker School	Poll Everywhere	understoodit	Pinnion	Free Mobile Polls	LectureTools	PINGO	Mentimeter
Question types												
Multiple/Single-choice	•	•	•	•	•	•	•	•	•	•	•	•
Yes/No Question	•						•					
Open Question	•	•		•	•	•	•			•		
Sorting Problem		•								•		
Matching Problem		•										
Image Quiz										•		
Question Features												
Hide votes until the question is closed for voting		-						•	•		•	
View total number of votes / percentage		-	•	•	•	•	•	•	•	•	•	•
View percentage of votes		•	•	•	•	•	•	•			•	
Definition of correct answers	•	•			•					•	•	
Math equations		•			•	•						
Visualization												
Bar chart	•	•	•	•	•	•	•	•	•	•	•	•
Pie chart				•	•							
Visualization for students		-		•	•		•	•	•			
Reporting and statistics												
Downloadable report	•		•	•		•						
Output (filetype)	csv		csv	xls		csv						
Results from previous sessions	•			•		•	•				•	

Table 3: The results of the comparison are shown in this table [Pichler, 2013]. The groups of criteria are separated by horizontal lines.

Finding 6 *Multiple/Single choice questions.*

Different question types are analyzed. Multiple choice questions can be assumed as the most important question type because all twelve compared ARSs support this type.

Finding 7 *Showing total number of votes and percentage.*

The results are presented to the lecturer after voting has finished, and sometimes during the process of voting. Ten out of twelve systems show the number of votes either in percentage, as total number, or both. It can be assumed that this is very valuable information for the lecturer and for the students.

Finding 8 *Showing percentage of overall votes.*

Eight out of twelve systems show the number of all votes compared to all students who are registered at the ARS during the lecture. This information might be an indicator for the lecturer when the question should be closed for voting.

Finding 9 *Bar chart for visualizing the results*

All compared ARSs display the results of single questions as bar chart. It can be assumed that this type of visualization is very well readable and visualizes important information.

The research purpose of this publication is to develop a simple-to-use, BYOD-based system covering all essential features from state-of-the art ARSs.

3 Implementation

This section covers the implementation starting with the requirements and the process definition. The architecture of the system and the used technology are explained in section 3.3. Further, the development process and the realization of the mobile application are described.

3.1 Requirements

The comparison of the different ARSs leads to the assumption that many of these solutions are overengineered and overly complicated. All systems are tested regarding specific use cases. Some of the ARSs are more difficult to use than others. The reason therefore is, that the systems offer too much functionality and therefore become more complicated [Pichler, 2013]. Therefore, the focus of the new system is keeping the system as simple and lean as possible. The user experience is clearly prioritized against feature richness. Therefore, only six findings of the comparison (see section 2.2) are transformed into requirements. The eight most relevant requirements are listed below.

Requirement 1 *Cross-platform capabilities*

Requirement 2 *Prioritize user experience against feature richness*

Requirement 3 *Web-based user interface (Finding 1)*

Requirement 4 *Group questions (Finding 2)*

Requirement 5 *Anonymous voting (Finding 5)*

Requirement 6 *Multiple choice questions (Finding 6)*

Requirement 7 *Show the number and percent of votes for an answer (Finding 7)*

Requirement 8 *Visualize the result as bar chart (Finding 9)*

3.2 Process Definition

Two processes are identified for the main actors in the system, the lecturer process, and the auditor (student) process. These two processes are intentionally kept as simple as possible to fulfill the requirement 2 regarding easy to use, non-distracting, and memorable.

3.2.1 Lecturer Process

In the following the lecturer process of asking questions is described.

Lecturer Step 1 *Create a lecture in the ARS and enter questions into the system.*

Lecturer Step 2 *In the beginning of the lecture, communicate the lecture access code of the ARS to the audience.*

Lecturer Step 3 *During the lecture, ask questions through the ARS by activating them.*

Lecturer Step 4 *To present the result of the audience for a particular question, stop the voting for that question in the ARS. The results are shown as bar chart containing the number of how many students voted for a specific answer and the percentage of votes.*

3.2.2 Auditor Process

This section describes the process of voting for an answer.

Auditor Step 1 *Access the ARS through a mobile Internet device and join the lecture session using the lecture access code provided by the lecturer.*

Auditor Step 2 *If a question is open for participation vote anonymously for an answer via the ARS.*

Auditor Step 3 *Review the results of questions closed for voting.*

3.3 Architecture and Technology Stack

The architecture for the new web-based ARS - which is called RealFeedback - is based on a typical cloud-based web application architecture. As shown in figure 1 the user interface is realized as a web application and will run on all modern browsers. This approach fulfills the requirement [1] to provide cross-platform capabilities. BYOD comes along with a heterogeneous device environment policy [Haintz, 2013, p. 100ff], because mobile Internet devices include laptops, smart phones, tablets, etc. To support the different screen resolutions and aspect ratios on these devices, the responsive design approach [Marcotte, 2011] is used. With responsive design, the appearance can be optimized for different devices based on their typical screen resolution. This approach needs less coding effort compared to writing a separate user interface for different devices. Twitter Bootstrap[1] is used as an HTML5 user interface (UI) library, which supports responsive design by default.

The business logic on the client side is implemented using HTML5 and JavaScript. To provide a cleaner and easier to maintain code base, JavaScript frameworks and libraries are extensively used. The model view controller paradigm (MVC) [Krasner and Pope, 1988] is realized with Backbone.js[2]. The jQuery Library[3] makes the JavaScript cross-browser capable. ICanHazJS[4] is used as the JavaScript templating engine. Underscore.js[5] provides functional programming support. The existing functionality of several other JavaScript libraries simplifies the business logic and reduces the maintainable code.

To connect the auditor interface and lecturer interface, a server is implemented. The server is responsible for the persistence of the data, the real-time communication between auditor and lecturer interface, and for securing the system.

[1] http://twitter.github.io/bootstrap (last visited on 3rd of July)
[2] http://backbonejs.org (last visited on 3rd of July)
[3] http://jquery.com (last visited on 3rd of July)
[4] http://icanhazjs.com (last visited on 3rd of July)
[5] http://underscorejs.org (last visited on 3rd of July)

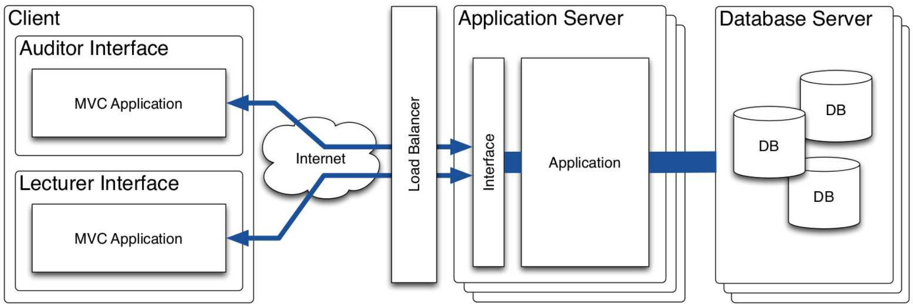

Figure 1: Architecture for the RealFeedback system based on [Haintz, 2013]

The server is implemented in Python[6] programming language. Frameworks and libraries are also used on the server-side. The Pyramid framework[7] is used to implement a fast, secure, and easy to maintain server. The persistence layer is realized using the NoSQL database MongoDB[8]. Both, application server and database server, are capable of horizontal scaling[9] in the cloud.

The client side (auditor and lecturer interface) communicates through a RESTful server interface [Richardson and Ruby, 2008] as shown in figure 1.

Based on requirement [2] regarding user experience, both lecturer, and auditor do not need to register to use the service. The lecturer can optionally register as a user on the RealFeedback system to gain advanced features such as saving the lecture sessions for later use, and exporting the results as CSV[10] file. The auditors cannot register. Auditors should stay anonymous to foster participation.

3.4 Development Process

An agile development process [Beck, 2006] is used. To respect the focus of the user experience, thinking-aloud tests are made during development. The results of the thinking-aloud tests [Sharp et al., 2007] highly influenced the further development. A screenshot of RealFeedback can be seen in figure 2.

[6] http://www.python.org (last visited on 3rd of July)
[7] http://www.pylonsproject.org (last visited on 3rd of July)
[8] http://www.mongodb.org (last visited on 3rd of July)
[9] Horizontal scaling: performance increase of the system by adding additional computing nodes. Vertical scaling: performance of each node is increased.
[10] Comma Separated Values

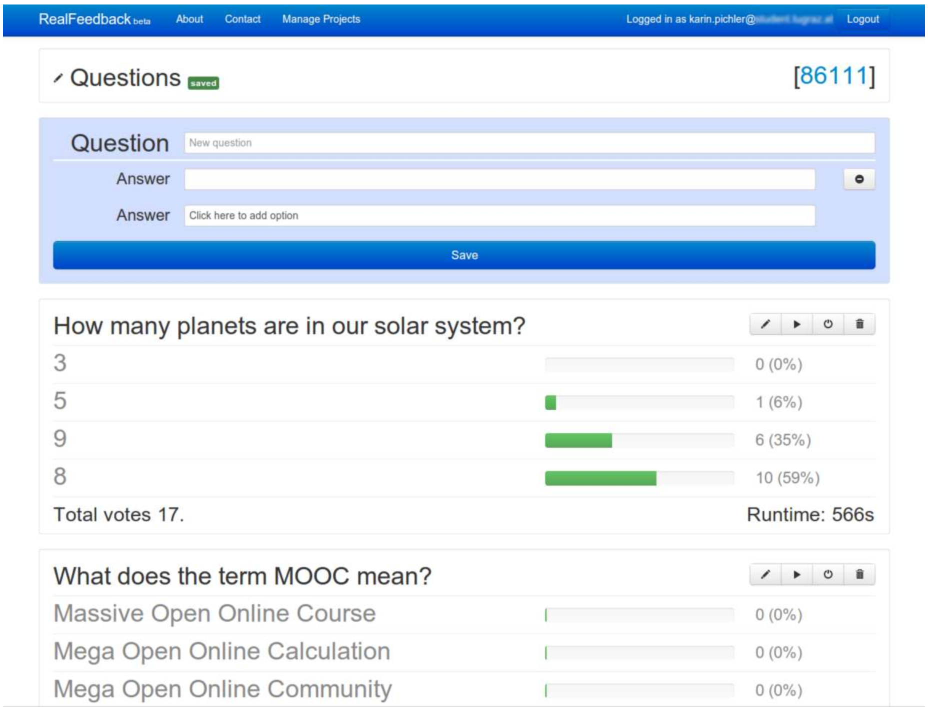

Figure 2: Screenshot of RealFeedback (http://realfeedback.tugraz.at)

3.5 Mobile App

To make the usage of the RealFeedback client even easier, or at least to give the auditors an alternative to the web application, different mobile applications are developed. One mobile application is based on the web application of the client interface. Most of the code of the web application was reused. Only minor modifications were needed to support the PhoneGap[11] framework. The PhoneGap framework encapsulates a web application in a native mobile application, which can be deployed through the respective app stores of the different mobile platforms such as Android, iOS, Windows Phone, Blackberry, WebOS, and Symbian. It follows the principle *write once run everywhere*. Because the RealFeedback PhoneGap application is in the early phase, it is yet not available in the appropriate app stores. To evaluate the user experiences of different app development approaches in future work, also conventional native apps are developed. At the time of writing there is already a fully functional native iOS

[11] http://phonegap.com (last visited on 3rd of July)

Lecturer	Experience in teaching	Field of expertise	Part of research team	Knows other ARSs
A	lecturing since 2001	high voltage technology	No	No
B	first lecture in 2013	software development	No	No
C	internal training courses since 2011	pedagogy, natural sciences	No	No
D	lecturing since 2000	eLearning	Yes	Yes

Table 4: Four lecturers who use RealFeedback were interviewed. Some general information about the lecturers is shown.

app of RealFeedback[12] in the Apple App Store. A native Android app is also developed, which will be available in the Google Play Store soon. Although all mobile apps use the same simplistic auditor process, future research will show if the PhoneGap approach can keep up with the conventional native apps in terms of user experience.

4 Evaluation

RealFeedback is released as a free public beta version, and several lecturers have already used it. At the time of publication RealFeedback has 257 active projects, 460 questions, 1,486 answers. The audience has submitted 1,550 votes in total. The maximum measured load in the system are 82 votes for one question. Students voted with their laptops (Mac OS X, Windows, Linux), tablets (iOS, Android) and smart phones (iOS, Android, Windows Phone). Up to now no performance problems are reported. Further performance tests will be done in future work.

To get a deeper understanding on how the lecturers use RealFeedback, qualitative interviews were carried out with four active lecturers. The intention is to get a deeper knowledge on how RealFeedback is used during lectures, what features are important to the lecturer and what features are missing. All interviewed lecturers have a technical background and are employed at Graz University of Technology. RealFeedback was introduced to the students by every lecturer at the beginning of the lecture. One of the interview-partners is part of the research team. Some more information can be seen in table 4 [Pichler, 2013].

[12] http://itunes.apple.com/WebObjects/MZStore.woa/wa/viewSoftware?id=661322020 (last visited on 3rd of July)

4.1 Interview

To get the information how lecturers use RealFeedback and what features are important to them, a qualitative interview is developed [Turner, 2010]. Each interview partner was interviewed for about 30 minutes. The interviews are recorded and transliterated. To find the most important points of the interviews, all interviews are evaluated and compared against each other. At the time the interviews were held, the mobile applications were not yet developed. The interview is structured into six groups [Pichler, 2013]:

General questions about the lecturer

This group contains several questions regarding the lecturer, for example, what type of media is used during lecture if he/she has ever used an ARS before and how long he/she is lecturing at universities. The aim of these questions is getting a deeper knowledge of the lecturer and the lecturing method.

General questions about RealFeedback

In this group, the usage of RealFeedback is covered. It is asked how often RealFeedback is used during lectures, what are the goals of the lecturers and how do the students like the system.

Generating questions in RealFeedback

This group covers the question generation of RealFeedback. It is asked how long does it take the lecturer to enter questions into the system and if the lecturer can think of generating the questions during lectures instantly.

Usability of RealFeedback

Questions regarding the usability of RealFeedback are covered in this group. This includes all phases of interacting with the system, starting with the question generation, asking questions and taking a look at the reports afterwards.

Statistics and reports

In this group the statistical part and the reports are covered.

Other

This group covers topics, which are not included in the above mentioned groups. The aim of this group is to get some more feedback from the lecturers.

4.2 Findings

This section covers the findings of the interviews. The findings can be separated into two groups: statements about the status quo of RealFeedback and missing features.

4.2.1 Status Quo

The statements about the status quo of RealFeedback are listed in table 5. Much effort is put on the clear and easy to use interface. It can be seen that the lecturers value this feature very much. They mentioned that there is no problem of using the system during the lecture because it is not complicated to use.

Every lecturer mentions that students like to use RealFeedback because they get more engaged. One lecturer mentioned that the awareness and concentration at this point are particularly high. This leads to a better learning effect for the students [Yerkes and Dodson, 1908].

The questions, which are asked during lectures, are prepared in general before the lecture starts. Three out of four interview partners mentioned that they could not imagine asking questions instantly during lectures. The reason is that they have to think about good wrong answers in a very short time.

4.2.2 Missing Features

Table 6 lists the features which are important to the lecturer but are not implemented in RealFeedback at the moment. A very important point for all interview partners is that they need to see how many students of all registered students have already voted for a specific question. According to the interview partners, this would help them to decide when to stop the question for voting.

Another missing feature is the possibility of marking answers of questions as correct. This information is useful for the lecturer because then he/she can see in a very quick way if the students responded correctly. However, it was also mentioned as useful for the students because they can use the system after the lecture has finished and they can see the right answer during their learning process.

Three of the interview partners would like to have the possibility of comparing previous results with present results. This feature would offer a way for the lecturers to reflect on their lectures.

5 Conclusion and Future Work

Although question-driven ARSs are a well-researched topic since years, a new system called RealFeedback was developed to adapt ARSs to the state-of-the-art

Important features to the lecturer	Mentioned by interviewees
RealFeedback provides a clear interface and is easy to use	4
Asking questions during lectures is simple	4
Students like to use the system in lectures	4
RealFeedback is used to gather information about the knowledge of students	3
Asking live questions cannot be done easily	3
Questions are prepared before lecture starts	3
The reports, which are provided by RealFeedback, are valuable	3

Table 5: The statements about the current status of RealFeedback mentioned by the lecturers during interviews.

Missing features	Mentioned by interviewees
A visualization of how many students have already voted for the question is missing	4
Defining the correct answer is missing	3
Comparison of the results over different sessions is missing	3
A mobile application is missing	1
A function for taking notes for a question is missing	1
Visualizing how fast the students responded to a question is missing	1

Table 6: The missing features, which are mentioned by the interview-partners are listed in this table.

technology and sociotechnological changes. The new system supports the growing trend of BYOD and uses state-of-the-art web technology to provide the functionality. It is assumed that the long history of ARS has led to overengineered solutions, which are overly complicated and hard to use in real-life situations. Therefore, a user-centric approach is used for the new system. The user processes are designed with simplicity in mind and are validated during the development process through thinking-aloud tests. The features are reduced to a minimum. This is done by comparing existing ARSs. RealFeedback provides only features, which most other ARSs have in common. After the first development phase, interviews are made with lecturers to gather feedback. The feedback shows that simplicity and user experience are important to lecturers. Although RealFeedback supports only a small subset of features other ARS solutions provide, the interviewed lecturers are not missing most of them. The missing features, which were mentioned, are only improvements to the existing system in terms of better user experience. RealFeedback[13] is currently free to use for everyone without limitations. It is shown that the simplicity of the system and the usage of the BYOD policy led to a high acceptance and a high participation from lecturers and students. Lecturers mentioned the higher attention and engagement of the students during lecture.

For future work, the mentioned missing features will be integrated in the system. Further feedback of real-life application needs to be considered to improve the system to support the users while not overengineering the system. During the development of this question-driven ARS the idea of a simple to use web-based quantitative backchannel tool came up. Such an ARS in contrast to a frontchannel question-driven ARS like RealFeedback can provide continuous real-time feedback of the audience perception in large lectures without disruption of the lecture. Both systems RealFeedback and the Backchannel System[14] [Haintz, 2013] are fostering interactivity in large lectures. Future work will investigate if it makes sense to combine both systems and how learning can be improved through interaction in large lecture halls.

References

[Abrahamson, 1999] Abrahamson, L. (1999). Teaching with a Classroom Communication System - What it Involves and Why it Works.

[Abrahamson, 2006] Abrahamson, L. (2006). A Brief History of Networked Classrooms: Effects, Cases, Pedagogy, and Implications. In Banks, D., editor, *Audience Response Systems in Higher Education*. IGI Global.

[Beatty and Gerace, 2009] Beatty, I. D. and Gerace, W. J. (2009). Technology-Enhanced Formative Assessment: A Research-Based Pedagogy for Teaching Science with Classroom Response Technology. *Journal of Science Education and Technology*, 18(2):146–162.

[Beatty et al., 2006] Beatty, I. D., Leonard, W. J., Gerace, W. J., and Dufresne, R. J. (2006). Question Driven Instruction: Teaching science (well) with an audience response system. In Banks, D. A., editor, *Audience Response Systems in Higher Education Applications and Cases*, volume 0, chapter 7, pages 96–115. Idea Group Inc.

[Beck, 2006] Beck, K. (2006). Extreme Programming Explained. *Embrace change*.

[Boardman, 1968] Boardman, D. E. (1968). *The use of immediate response systems in junior college*. PhD thesis, University of California.

[Bruff, 2009] Bruff, D. (2009). *Teaching with Classroom Response Systems: Creating Active Learning Environments*, volume 33 of *Jossey-Bass higher and adult education series*. Jossey-Bass.

[Caldwell, 2007] Caldwell, J. E. (2007). Clickers in the Large Classroom : Current Research and Best-Practice Tips. In *CBE Life Sci Educ. 2007*, pages 9–20.

[13] https://realfeedback.tugraz.at (last visited on 3rd of July)

[14] http://understand-your-audience.com (last visited on 3rd of July)

[Collins et al., 2008] Collins, B. L., Tedford, R., and Womack, H. D. (2008). Debating the merits of clickers. *North Carolina Libraries (Online)*, 66(1/2):20.

[Dufresne et al., 1996] Dufresne, R. J., Gerace, W. J., Leonard, W. J., Mestre, J. P., and Wenk, L. (1996). Classtalk: A classroom communication system for active learning. *Journal of Computing in Higher Education*, 7(2):3–47.

[Ebner, 2009] Ebner, M. (2009). Introducing live microblogging: how single presentations can be enhanced by the mass. *Journal of research in innovative teaching*, 2(1):91–101.

[Ebner et al., 2012] Ebner, M., Nagler, W., and Schön, M. (2012). Have They Changed? Five Years of Survey on Academic Net-Generation. *World Conference on Educational Media and Technology*, 8:21–31.

[Feldman and Capobianco, 2007] Feldman, A. and Capobianco, B. M. (2007). Teacher Learning of Technology Enhanced Formative Assessment. *Journal of Science Education and Technology*, 17(1):82–99.

[Froehlich, 1963] Froehlich, H. P. (1963). What about classroom Communicators? *Audiovisual communication review*, 11(3):19–26.

[Gauci et al., 2009] Gauci, S., Dantas, A. M., Williams, D., and Kemm, R. E. (2009). Promoting student-centered active learning in lectures with a personal response system. *Advances in physiology education*, 33(1):60–71.

[Haintz, 2013] Haintz, C. (2013). *Quantitative Digital Backchannel: Developing a Web-Based Audience Response System for Measuring Audience Perception in Large Lectures*. PhD thesis, Graz University of Technology.

[Hoffman and Goodwin, 2006] Hoffman, C. and Goodwin, S. (2006). A clicker for your thoughts: technology for active learning. *New Library World*, 107(9/10):422–433.

[Judson and Sawada, 2002] Judson, E. and Sawada, D. (2002). Learning from Past and Present: Electronic Response Systems in College Lecture Halls. *Journal of Computers in Mathematics and Science Teaching*, 21(2):167–181.

[Krasner and Pope, 1988] Krasner, G. E. and Pope, S. T. (1988). A description of the model-view-controller user interface paradigm in the smalltalk-80 system. *Journal of Object Oriented Programming*, 1.

[Lennon, 2012] Lennon, R. (2012). Bring Your Own Device (BYOD) with Cloud 4 Education. In *SPLASH '12*, pages 171–179.

[Liu et al., 2003] Liu, T.-c., Liang, J.-k., Wang, H.-y., and Chan, T.-w. (2003). The Features and Potential of Interactive Response System. In *Proceedings of International Conference on Computers in Education (ICCE)*, pages 315–322, Hong Kong.

[Logicalis, 2012] Logicalis (2012). BYOD : an emerging market trend in more ways than one. Technical report.

[Marcotte, 2011] Marcotte, E. (2011). *Responsive web design*.

[Mazur, 1997] Mazur, E. (1997). Peer Instruction. In *Peer Instruction*, volume 30, chapter 2, page 278. Prentice Hall.

[Pichler, 2013] Pichler, K. M. (2013). *Relevant Information and Information Visualizations for Lecturers in Web-Based Audience Response Systems*. Master's thesis, Graz University of Technology.

[Richardson and Ruby, 2008] Richardson, L. and Ruby, S. (2008). *RESTful Web Services*. O'Reilly Media, Inc.

[Roschelle, 2003] Roschelle, J. (2003). Unlocking the learning value of wireless mobile devices. *Journal of Computer Assisted Learning*, 19(May):260–272.

[Sharp et al., 2007] Sharp, H., Rogers, Y., and Preece, J. (2007). *Interaction Design: Beyond Human-Computer Interaction*. Wiley.

[Thomson, 2012] Thomson, G. (2012). FEATURE BYOD : enabling the chaos. *Network Security*, 2012(2):5 – 8.

[Turner, 2010] Turner, D. W. (2010). Qualitative Interview Design : A Practical Guide for Novice Investigators. *The Qualitative Report*, 15(3):754–760.

[Wong et al., 2011] Wong, K., Mohan, J., and Lam, P. (2011). Classroom communication on mobile phones first experiences with web-based clicker system. In *ascilite Conference*, number 1993, pages 763–777.

[Yerkes and Dodson, 1908] Yerkes, R. M. and Dodson, J. D. (1908). The relation of strength of stimulus to rapidity of habit-formation. *Journal of Comparative Neurology and Psychology*, 18(5):459–482.