

ASQ: Interactive Web Presentations for Hybrid MOOCs

Vasileios Triglianos
University of Lugano, Switzerland
vasileios.triglianos@usi.ch
<http://asq.inf.usi.ch>

Cesare Pautasso
University of Lugano, Switzerland
c.pautasso@ieee.org
<http://www.pautasso.info>

ABSTRACT

ASQ is a Web application for creating and delivering interactive HTML5 presentations. It is designed to support teachers that need to gather real-time feedback from the students while delivering their lectures. Presentation slides are delivered to viewers that can answer the questions embedded in the slides. The objective is to maximize the efficiency of bi-directional communication between the lecturer and a large audience. More specifically, in the context of a hybrid MOOC classroom, a teacher can use ASQ to get feedback in real time about the level of comprehension of the presented material while reducing the time for gathering survey data, monitoring attendance and assessing solutions.

Categories and Subject Descriptors

K.3.1 [Computers and Education]: Computer and Information Science Education—*Collaborative learning, Computer-assisted instruction (CAI)*

Keywords

HTML5, Software clicker, impress.js

1. INTRODUCTION

The Web is disrupting the education sector by widely broadening and flattening the field for higher education. While most of the ongoing efforts in Massive Open Online Courses (MOOCs [4]) are dedicated to scaling teaching to online classes of hundreds of thousands of students, in our work we target so-called hybrid MOOCs [3] which are intended to improve the traditional offline teaching experience with the support of modern Web technologies. In this poster we demonstrate a novel tool which pushes the limits of current Web technologies to support the delivery of highly interactive presentations and enhance the classroom learning experience. The tool name includes the concepts of "Slides, Questions and Answers" summarising its main features: broadcasting presentation slides to the audience with explicit support for asking and answering questions – both from the teacher to the students as well as from the students back to the teacher. The Web-based nature of the tool lowers to a great extent the barriers to set up and use such a system.

Copyright is held by the author/owner(s).
WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2038-2/13/05.

2. FEATURES

The notion of a presentation in ASQ assumes the existence of two roles: a teacher and multiple 'audience' participants. The teacher can insert questions within an HTML5 slideshow powered by impress.js in order to gather information from the audience through background knowledge surveys, answers to comprehension question regarding the presented material, brainstorming and polling questionnaires, as well as small programming assignments. Teachers can manage a set of presentations that have been uploaded on the ASQ server, they can edit them and start the broadcast. At presentation time, clients connect to the ASQ server using a standard Web browser, receive a copy of the slideshow and follow the presentation synchronously with the teacher. If the students are co-located with the teacher, the slideshow is also projected with a beamer. By design, the participants have no control over the flow of the presentation shown on their Web browsers. To track and provide general awareness of the status of the connected users, a map of the audience is shown on the teacher's tablet. When questions from the teacher pop up, the audience members can answer them. The teacher is in control of the process as the clients send real-time notification about their progress and status. After all answers have been gathered, ASQ logs them together with timing meta-data and displays a statistical analysis about the feedback. This can be shown and discussed with the audience before moving on with the presentation. If participants have a question, they can enter it anytime through ASQ and associate it with the slide that provoked it. The teacher is notified and can decide to go back to the question either immediately or at the end of the lecture. With large classes this mechanism is expected to enable "shy students" to provide insightful feedback, while also enabling the teacher (or more likely the teaching assistants) to filter irrelevant questions or comments.

3. ARCHITECTURE & USER INTERFACE

ASQ is a Web-based, distributed User Interface application, which provides several views that are tailored to the requirements of specific user roles (the presenter and the audience participants and in some cases also the presenter assistants) and take advantage of specific capabilities of client devices (i.e., desktops, tablets and smartphones). ASQ is built with HTML5 on top of the express.js framework for node.js. It leverages WebSockets to exchange events in real-time between connected Web browsers and the ASQ server and extends the impress.js HTML5 presentation framework to broadcast interactive presentations over the Web.

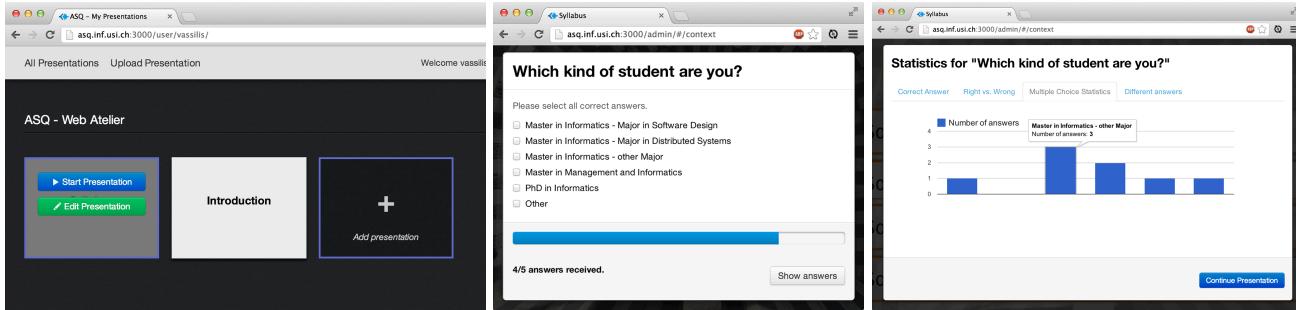


Figure 1: ASQ Screenshots: 1) Presenter Control Panel; 2) Question; 3) Answer Statistics

3.1 Presenter and Audience Views

Whenever a teacher logs in to his/her account on the server that is hosting ASQ, a control panel view with the uploaded presentations is rendered (Fig. 1). Presentations can be added by uploading them and directly edited from the control panel. Clicking on "start presentation" begins the broadcast: a popup advertising the URI to be shared with the audience is displayed. The audience may copy this invitation URI and connect to the server to watch the presentation. It is also possible to connect to the server and retrieve a list of the currently ongoing public presentation sessions. While preparing a presentation the presenter may inject questions at specific points of the presentation. To do so, questions may be defined using a simple data model which can either be embedded in the presentation HTML as a custom microformat or through a simple editor available from within ASQ. Questions and the corresponding answers statistics can also be uploaded and downloaded using a simple JSON serialization. For each question, the author needs to specify the question text and the type of question. For multiple choice answers, it is possible to enumerate the possible answers and mark which of them are correct so that ASQ can automatically grade the answers.

After the participants in the class point their Web browsers to the presentation URI, they are presented with the current slide as shown by the teacher. When the presentation reaches a question, each member of the audience is presented with a popup that allows them to review the question, provide an answer for it and explicitly indicate that they are done answering the question. At the same time, the teacher's view shows a progress bar indicating the number of connected clients that have already answered the question. Each audience member can change the given answer until the teacher closes the question. ASQ can do so automatically when the progress bar has reached its completion or when the preset time for the question is up, at which point the question popup disappears and is replaced with a statistical analysis of the answers, which can lead to further class discussions.

3.2 Assessment modes

Depending on the type of question, ASQ supports different mechanisms to check the quality of the feedback. For open questions, a simple filtering step is inserted between the stream of feedback coming from the audience and the presentation view summarizing it. For multiple choice questions, ASQ supports their automatic assessment, assuming that the teacher has configured it with the list of correct

answers. The same holds for programming questions, for which ASQ can run the solutions against the corresponding test suites or oracle functions. Other types of questions may require more than a simple binary feedback. Thus it is possible to configure ASQ for staff-assessment, peer-assessment or self-assessment. In the first case, answers are graded and corrected by a pool of teaching assistants, who also collect a number of exemplary solutions to be discussed in plenary. In peer-assessment mode, students that have already answered to a question can spend their time checking the answers of their peers. This process has the double advantage of minimizing idle time, thus keeping students engaged, as well as allowing students to see the different ways their classmates approach the same problem. Finally, self-assessment works by providing students with a well-defined rubric [1].

4. RELATED WORK & CONCLUSION

ASQ is inspired by Informa [2], a Java-based software clicker that supports a rich set of problem types and promotes peer-assessment during the idle time between answering and waiting for the whole class to finish. ASQ adds support for integrating bi-directional questions within slideshow presentations. In our experience, thanks to the use of Web technologies, ASQ is much more lightweight to introduce in the class, since only a Web browser is required. We have started to use ASQ at the University of Lugano with promising initial results. The tool has shown good acceptance with its target users and good scalability in large classrooms. We are closely following the evolution of the HTML5 platform to incorporate additional features, such as WebRTC support for live video streaming in the browser, which would enable to apply ASQ within distributed teaching scenarios.

Acknowledgements: We are grateful to Jacques Dafflon, Margarita Grinvald, Max von Bülow and Georgios Kokosioulis for their hard work in creating the ASQ prototype.

5. REFERENCES

- [1] H. Andrade. The effects of instructional rubrics on learning to write. *Current Issues in Education*, 4(4), April 2001.
- [2] M. Hauswirth and A. Adamoli. Solve and evaluate with informa: a java-based classroom response system for teaching java. In *Proc. of PPPJ '09*, pages 1–10, 2009.
- [3] F. G. Martin. Will massive open online courses change how we teach? *CACM*, 55(8):26–28, August 2012.
- [4] A. McAuley, B. Stewart, G. Siemens, and D. Cormier. The MOOC model for digital practice. <http://www.elearnspace.org/>, 2007.