Multiscript – an online student-teacher collaboration platform for classroom lectures

ABM Tariqul Islam, Philipp Jäcks and Clemens H. Cap

Abstract: Online collaboration on lecture contents has gained much popularity, over the few decades, due to its potential to enhance the learning experience. We propose a novel idea of an online collaboration platform, called Multiscript (MS), for the students and the teacher, on classroom lectures. MS combines two online learning approaches into a single collaboration platform. One approach, called outside-of-class-Multiscript (OCMS), is a variant of distance online learning which is applicable for the students who want to discuss the lecture slides outside of class due to certain reasons. Another approach, called inside-of-class-Multiscript (ICMS), allows online collaboration among the students and the teacher while the lecture is going on. In OCMS, the teacher can share the slides – with audio annotations for each slides and/or a single recorded audio for the whole lecture. The students can access the slides and discuss (via text and audio chat) with their peers about the slides and annotate them, post feedback about the slide and ask questions to the teacher directly via MS platform. In ICMS, the students can create annotations for the slides and post feedback about the slides which can be read by the teacher later. MS system can be accessed by using just a web browser on any PCs, tablets, notebooks or mobile devices.

Keywords: Collaborative learning, online learning, open learning, e-learning, distance education.

1 Introduction

During the last few decades, online collaboration on classroom lecture content has been an object of interest to the researchers because of its potential to enhance the overall learning experience. While offline collaboration (face-to-face or F2F), among the teacher and the students during the class, has a certain impact on learning, quite a few researchers state that online collaboration provides an equal voice for all the students to express opinion or ask questions about a lecture [SSH06]. Hence, the students consider online collaboration as more neutral than traditional classroom discussions [St11]. Moreover, through online collaboration, the students get an opportunity to reflect on their peer’s contribution (e.g., comments or questions) about a lecture while they are creating or posting their own comments or questions. It helps the students to foster a certain diligence and deliberation about their peer’s contribution [RS03]. Research (e.g., in [SSH06]) indicates that the way the students, being in an online collaboration, perceive the role of their colleagues, discern the social presence of them and create a certain feeling for the collaborating community, essentially affects the learning process.
Quite a few research (e.g., [SSH06, RS03]) state that online collaboration, among the students and the teacher, indeed plays an important role to enhance the overall learning experience by engaging the students into a comfortable, productive, equal voiced and socially well-perceived environment. Existing research works on online collaborative learning can be categorized into two main types – (i) distance online collaboration (DOC) (e.g., [BWB09]) and (ii) in-class online collaboration (ICOC) (e.g., [An06]). In case of DOC, the participants, being in geographically different locations, form small groups to collaborate on projects or homeworks. Whereas, ICOC mainly takes place in the class, between the students and the teachers, on lecture contents or on group works. Usually DOC or ICOC takes place via a range of devices (e.g., stationary PCs, Tablets or mobile phones). To our knowledge, the existing online collaboration platforms either support DOC or ICOC. We didn’t come across a system which fully supports both types of collaboration in a single platform.

In this paper, we propose a novel online collaboration platform, called Multiscript (MS), where we combine two online learning approaches into a single collaboration platform. Our first approach, outside-of-class-Multiscript (OCMS), is a variant of distance online learning and the second approach, inside-of-class-Multiscript (ICMS), takes place during the class. OCMS is applicable when the students want to discuss the lecture slides at home because – either, after attending the lecture they still have some questions in mind to discuss with their peers, or, they were absent in the class due to medical issue or traveling purpose. Whereas, in ICMS, the collaboration among the students and the teacher takes place in the class while the lecture is going on. To our knowledge, MS is the first platform which supports both of these approaches in a single online collaboration platform. Hence, MS possesses the attributes of a blended learning approach [Po13] where a convergence of F2F meeting and distance learning occurs through available technology. Moreover, MS extends the F2F meeting part of blended learning by including the ICMS support into it.

2 Related Work

Here, we present an overview of existing works pursued on online collaborative learning. We mention in the previous section that existing works in this area can be categorized into two main types – distance online collaboration (DOC) and in-class online collaboration (ICOC). In case of DOC, quite a lot of research (e.g., [St11, SSH06, BWB09]) have been pursued which mostly investigate the effect of collaborative learning, among a group of participants, on a specific project or homework. Furthermore, there are quite a few tools available which support DOC for different purposes for different user groups.

Groupboard [Gr16] is one such DOC tool which lets participants to collaborate on a specific topic via text, audio and video chats. The collaboration in Groupboard doesn’t occur directly on lecture slides (slides are not accessible), but occurs on a topic from the lecture. It also has an ICMS part with very limited functionality. Vyew [Vy16] is another DOC tool which is used mostly for collaborating on group projects or homeworks. This tool has the most similarity to our OCMS approach. In Vyew, teachers can upload lecture materials for real-time presentations and then students can meet online and collaborate together in real
time or separately over time. Here, users can create annotation on slides and post suggestions about a lecture content to the teacher via text chat. A couple of other notable DOC tools are Scribbler [Sc16] and Twiddla [Tw16].

Anderson et. al., in [An06], present an ICOC platform, called Classroom Presenter, in which students answer the questions provided by the teacher and submit them to the teacher via wireless connectivity. Teacher evaluates the responses, and randomly selects a few of them to annotate and display simultaneously. Here, the students and the teacher are provided with wirelessly connected stationary Tablet PCs. Hence, it lacks web based connectivity and so, it cannot reach to a wide range of users to interact with the system. It also fails to provide a reliable communication platform. Besides this, in Classroom Presenter, the student’s loss of attention might occur and the specific use of a single device limits the use of technology. Caceffo and Da Rocha, in [CDR11], present another ICOC platform called Ubiquitous Classroom Response System. This approach has the most similarity to our ICMS approach. It works similar to Classroom Presenter, but uses context information to create a dynamic environment where students can use any available mobile device to answer the teacher’s questions. Moreover, since it uses web based connection to support collaboration, it does not have the limitations of Classroom Presenter. Another similar approach to this system is presented by Oura et. al. in [Ou06].

3 Proposed Idea of Multiscript System

3.1 System description

We previously mention that MS brings two online collaboration approaches (outside-of-class-Multiscript (OCMS) and inside-of-class-Multiscript (ICMS)) into a single platform. Here, we describe about the functional requirements of the OCMS and ICMS respectively.

In OCMS mode, we design the MS system for the students to seamlessly collaborate, via web connection, with their peers on the lecture slides uploaded by the teacher. Here, a teacher, after delivering a lecture, can share the lecture slides by uploading it to the MS server. The students can access the lecture slides on any available PCs, tablets, notebooks or mobile devices at any time just by using a web browser. While sharing the lecture slides, a teacher can – attach audio annotations for each of the slides and/or attach a single audio for the whole lecture. So, the students can – listen to the audio annotations for each slide while studying the slides and/or go through all the slides while listening to the entire lecture audio in the background. Figure 1 shows an illustration of OCMS mode. In OCMS mode, the students are offered four different kinds of features, they can –

- perform text chats with their peers
- participate in audio chats with their peers
- annotate the slide for their own purpose
- post feedback about the slides and ask questions to the teacher
For the last case, the feedbacks and the questions are sent to the teacher as a ‘newsletter’ and the teacher, along with providing appropriate response, can also rate the feedbacks/questions. The teacher is also able to provide reason for his/her voting.

In ICMS, the students can collaborate on the lecture content while being in the classroom. Here, the students can create annotations on the slides for themselves and post feedbacks about the slides which are read by the teacher later. The teacher can response to those feedbacks and rate them similar to OCMS mode. This mode is illustrated in Figure 1. While the teacher delivers a lecture in ICMS, the slides are shown in three different modes:

- via the projector in the classroom
- on the browser of a student’s device
- on his own computer’s screen

In the second case, the slide displayed on a student’s device is synchronized with the projector’s current slide. In the third case, the teacher’s screen shows a few additional features along with the slides. S/he can see the currently selected slide, a preview of the other slides and some notes for herself. The teacher can also control, from his/her machine, the number of collaborating options which will be available on the students’ devices.

Apart from the functional features described above, MS system has a few other requirements which are necessary for providing the best possible user experience. The first requirement of MS is: the interaction with MS system should be fast and responsive enough to meet user’s satisfaction. As a second requirement, the users consider a system as comfortable only when the user-interface is easy to use and has the minimalist design. The interface should also have the capability to fit on different screen sizes of different devices (e.g., PCs, Tablets, mobile phones). We design the MS system in such a way that it can be displayed intuitively on various screen size on a range of devices. Besides, while de-
signing the MS system, we have also considered the age of the user’s (specially the age of the teachers) of our system. We kept the design simple enough so that people with minimum knowledge about Internet will be able to familiarize themselves with our system with minimum effort. Moreover, MS system is designed to handle quite a large number of users at a time. In future, it might be used by a large number users, so we have considered the access peak of the system when many users start to use MS at once. The scalability of MS system allows large number of connections simultaneously. Besides this, we keep the identity of the students anonymous so that they can avoid the account creation process, and collaborate autonomously with their peers in the MS system. We have designed the MS to be modular so that, in future, new features can be integrated easily into it.

3.2 System architecture

Since we plan to support a wide range of devices (with varying screen size and operating system), rather than developing individual app for each cases, we develop a web application, which serves HTML, for the MS system. Since most of the devices (stationary or mobile) possess the capability to render HTML5, Javascript and CSS, we develop one browser-based client interface which intuitively fits on these devices. This removes the necessity to maintain different repositories for different versions (mobile phone, desktop, iOS, Android etc.) of the system. Moreover, we also consider an easy and minimal design for the user interface. Hence, to achieve a unique development platform and simplistic interface design, we choose to use established frameworks for the web-based user interface. We use Express and SocketIO framework, Node.js, Jade and AngularJS for building the web application of MS. Figure 2 outlines the technology stack of MS. In MS, we allow anonymous participation of the students so that the peers and the teacher cannot trace back to the user. To achieve this, we remove the necessity to register an individual with a university account, rather we allow everyone. Nevertheless, we define certain thresholds to prevent any misuse of the MS system with an anonymous account.
Performance, scalability and responsiveness are the factors which matter most for a good web application, and yet, these are the requirements which are quite difficult to achieve. To develop a stable and fast web application for MS, we looked into numerous solutions for asynchronous web servers which would work with two-way communication channel in tandem. We decide to use the Node.js as our web server. It uses V8, the Google Chromes JavaScript engine. It is event-driven, stable and has the capability to perform thousands of concurrent connections. For MS, a two-way communication channel is necessary, because both, the server and the client, sends messages to each other. In MS, the client is the browser of both the students and the teacher. We use the SocketIO library as the two-way communication layer which is well implemented for Javascript and for the Node.js web server. To support audio chat in MS, we use WebRTC. We choose to use MongoDB, a non-relational database which possesses flexible scheme, to store all permanent data.

### 3.3 Data flow

In MS, the users are assigned hash values as UserID and a lecture content is assigned a unique LessonID. The data flow in MS are initiated – either by the teacher or by the students. Figure 3 shows the data flow diagram of MS system. When the teacher uploads the lecture content (called a Slideset) with audio annotations, an upload command is executed within MS. As a result, a corresponding LessonID is generated for the Slideset, and subsequently, the Slideset with audio annotations and LessonID are sent to the MS server to be stored. When a student selects a Slideset from the list of displayed Slidesets, a request is sent to the MS server to fetch the Slideset. Upon getting the request, the MS server fetch the Slideset and associated annotations (if already existed) for that particular student and send them to the students browser. When the students create new annotations and/or feedback for a particular slide, those are sent (along with UserID and LessonID) to the MS server to be stored.
3.4 Development status

Currently, as of May 2016, we have completed the development of most of the features for both OCMS and ICMS mode, except the feedback feature (currently under development) for the students. An overview of the OCMS and the ICMS modes is depicted in Figure 4 and Figure 5 respectively. Figure 4 depicts that the students can look into the slide contents while listening to the audio annotations, collaborate anonymously with their peers via audio and text chat, and annotate the slides for themselves. Figure 5 shows that the teacher can start the presentation on the projector from his/her device while his/her screen shows different options (e.g., preview of current and other slides, private mode). S/he, while in private mode, can press the eye icon to display a particular slide on the projector and simultaneously, on the devices of the students. Currently, we are also developing a better
user interface by collaborating with professional interface designers which is expected to be deployed soon.

4 Conclusion

Online collaboration on lecture content has gained much popularity due to its potential in enhancing the learning process. We propose a novel idea of an online collaboration platform, called Multiscript (MS). MS offers a unique collaboration platform in which the participants can collaborate, using just a web browser, both while being inside of class and outside of class. MS is optimized to be used on different devices with varying screen size. We design the overall MS system to be fast, responsive and scalable, and keep the user interface easy and simple to meet user demand and satisfaction. We allow anonymous access for the students to persuade an autonomous participation during collaborating with peers or the teacher. We plan to test the MS system for computer science lecturers soon.

References


