ΕΛΛΗΝΙΚΟ ΑΝΟΙΚΤΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ

Engineering of pervasive computing systems MSc

MASTER THESIS

HOU-CS-PGP-2016-15

Implementation of a distributed mobile based environment to help children learning a foreign language

IOANNIS SALATAS

SUPERVISOR: CHRISTOS GOUMOPULOS

ΠΑΤΡΑ 2016

HELLENIC OPEN UNIVERSITY
SCHOOL OF SCIENCES AND TECHNOLOGY

WWW.EAP.GR

ΕΛΛΗΝΙΚΟ ΑΝΟΙΚΤΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ
HELLENIC OPEN UNIVERSITY
SCHOOL OF SCIENCES AND TECHNOLOGY

Master Thesis HOU-CS- PGP-2016-15

Implementation of a distributed mobile based environment to help children learning a foreign language

Ioannis Salatas

Διπλωματική Εργασία HOU-CS- PGP-2016-15

Υλοποίηση κατανεμημένης εφαρμογής τηλεδιασκέψεων για την υποβοήθηση διδασκαλίας ξένων γλωσσών

Ιωάννης Σαλάτας

Implementation of a distributed mobile based environment to help children learning a foreign language

Ioannis Salatas

Christos Goumopoulos	**Ioannis Zaharakis**	**Achilles Kameas**
Assistant Professor, University of the Aegean	Associate Professor, Technological Educational Institute of Western Greece	Associate Professor, Hellenic Open University

Abstract

This Master Thesis presents the requirement analysis, design and implementation of an e-learning environment for helping children learning a foreign language by promoting communication and language development skills through a virtual collaboration environment in which children are encouraged to interact and communicate with other children from all over the world learning the same language.

The system falls in the field of Mobile Assisted Language Learning (MALL) and is designed around the principles of active learning methodology, in which students are transformed from passive listeners to active participants, by engaging in different types of collaborating activities which exploit their surrounding environment and are not limited in a physical classroom. In the initial survey we performed, most of the responders agreed that both intercultural communicative competence and telecollaboration are important elements in foreign language teaching, a claim which is also confirmed by the existing scientific literature in this field of study.

It is targeting Windows based desktops and tablets, and android based smartphones and tablets and its implementation took place by following a combination of Human Centered Design and the SCRUM agile software engineering methodology. It encompasses different distributed server side components to support different kind of activities, like the BigBlueButton server for synchronous activities (text/video chat, shared boards, etc) and the Moodle server for asynchronous activities (content sharing, forums, wikis, etc), integrated by a middleware component (management server) which is fully implemented in this Thesis.

The system's unique ability to load and integrate external modules developed by third parties, along with the fact that its source code is distributed as free software under an open source license, makes it a great candidate as a testbed for other researchers who need to explore fields related to active learning, collaborative learning activities and online multiplayer educational games, as they could be released from the burden of implementing low-level technical details like security, communications, video and audio streaming, etc. and could exclusively focus on their primary research field.

Keywords: e-learning, foreign languages teaching, web meetings, online multiplayer educational games.

Contents: Text, Screenshots, Source Code in Java, ActionScript, C++, PHP and MySQL.

Υλοποίηση κατανεμημένης εφαρμογής τηλεδιασκέψεων για την υποβοήθηση διδασκαλίας ξένων γλωσσών

Ιωάννης Σαλάτας

Χρήστος Γκουμόπουλος	**Ιωάννης Ζαχαράκης**	**Αχιλλέας Καμέας**
Επίκουρος Καθηγητής, Πανεπιστήμιο Αιγαίου	Αναπληρωτής Καθηγητής, Τεχνολογικό Εκπαιδευτικό Ίδρυμα Δυτικής Ελλάδας	Αναπληρωτής Καθηγητής, Ελληνικό Ανοικτό Πανεπιστήμιο

Περίληψη

Η παρούσα Διπλωματική Εργασία παρουσιάζει την ανάλυση απαιτήσεων, το σχεδιασμό και την υλοποίηση ενός περιβάλλοντος ηλεκτρονικής μάθησης (e-learning) το οποίο αναπτύχθηκε για να βοηθήσει παιδιά που μαθαίνουν μια ξένη γλώσσα. Μέσα από ένα εικονικό περιβάλλον συνεργασίας τα παιδιά ενθαρρύνονται να αλληλεπιδρούν και να επικοινωνούν με άλλα παιδιά από όλο τον κόσμο που μαθαίνουν την ίδια γλώσσα με στόχο την ανάπτυξη δεξιοτήτων εκμάθησης της ξένης γλώσσας. .

Το σύστημα εντάσσεται στο ερευνητικό πεδίο του Mobile Assisted Language Learning (MALL) και έχει σχεδιαστεί γύρω από τις αρχές της ενεργού μεθοδολογίας μάθησης (active learning), σύμφωνα με την οποία οι μαθητές μετατρέπονται από παθητικοί δέκτες σε ενεργοί συμμετέχοντες, με τη συμμετοχή τους σε δραστηριότητες συνεργασίας οι οποίες δεν περιορίζονται στα όρια μιας φυσικής τάξης αλλά αξιοποιούν το καθημερινό περιβάλλον των μαθητών. Στην αρχική έρευνα που πραγματοποιήθηκε, οι περισσότεροι από τους εκπαιδευτικούς ξένων γλωσσών που ανταποκρίθηκαν συμφώνησαν ότι τόσο η διαπολιτισμική επικοινωνιακή ικανότητα και η τηλε-συνεργασία (telecollaboration) αποτελούν σημαντικά στοιχεία στη διδασκαλία ξένων γλωσσών, ένας ισχυρισμός που επιβεβαιώνεται και από την υπάρχουσα επιστημονική βιβλιογραφία σε αυτό το πεδίο της μελέτης.

Το σύστημα μπορεί να χρησιμοποιηθεί σε επιτραπέζιους και φορητούς υπολογιστές καθώς και σε tablets που βασίζονται στο λειτουργικό σύστημα Microsoft Windows, και επίσης σε έξυπνες κινητές συσκευές (smartphones και tablets) που βασίζονται στο λειτουργικό σύστημα Android, και η υλοποίησή του πραγματοποιήθηκε ακολουθώντας ένα συνδυασμό Ανθρωποκεντρικού Σχεδιασμού και τη μεθοδολογία μηχανικής λογισμικού SCRUM. Περιλαμβάνει διάφορα κατανεμημένα υποσυστήματα από την πλευρά του server, όπως το BigBlueButton για σύγχρονες (synchronous) δραστηριότητες (text/video chat, κοινούς πίνακες – shared boards, κλπ.) καθώς και το moodle για ασύγχρονες (asynchronous) δραστηριότητες (διαμοίραση εκπαιδευτικού υλικού, forums, wikis, κλπ), τα οποία ολοκληρώνονται από ένα μεσολογισμικό διαχείρισης (management server), το οποίο αναπτύχθηκε στην παρούσα διπλωματική εργασία.

Η δυνατότητα του συστήματος να φορτώνει και να ενσωματώνει εξωτερικές μονάδες που έχουν αναπτυχθεί από τρίτους, λαμβάνοντας υπόψιν και το γεγονός ότι το σύστημα διατίθεται υπό άδεια ελεύθερου ανοικτού λογισμικού (free open source license), το μετατρέπει σε μια πολύ καλή υποψήφια πλατφόρμα δοκιμών για άλλους ερευνητές που έχουν ως ερευνητικό αντικείμενο τομείς που σχετίζονται με την ενεργό μάθηση, δραστηριότητες συνεργατικής μάθησης και online multiplayer εκπαιδευτικά παιχνίδια, δεδομένου ότι θα μπορούσαν να αποδεσμευτούν από χαμηλού επιπέδου τεχνικές λεπτομέρειες, όπως η ασφάλεια, η

επικοινωνία, η μετάδοση και λήψη βίντεο και ήχου κλπ. και να επικεντρωθούν αποκλειστικά στον πρωτογενή τομέα της έρευνας τους.

Λέξεις-κλειδιά: ηλεκτρονική μάθηση, διδασκαλία ξένων γλωσσών, τηλεδιάσκεψη, online multiplayer εκπαιδευτικά παιχνίδια.

Περιεχόμενο: Κείμενο, Στιγμιότυπα οθονών, πηγαίος κώδικας σε γλώσσες προγραμματισμού Java, ActionScript, C++, PHP και MySQL.

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1. Active Learning Methodologies in Second Language Learning

"*If you tell me, I will listen. If you show me, I will see. But if you let me experience, I will learn*"

This quote is allegedly said by the philosopher Lao-Tse (5^{th} century B.C.) and it could be considered as the basis of many teaching methodologies like "active learning" introduced by the English scholar R.W. Revans [1].

In active learning, students are transformed from passive listeners to active participants. Active learning helps the student understand the subject through inquiry, gathering and analyzing data by solving higher order cognitive problems. Bonwell and Eison [2] suggested learners work collaboratively, discuss materials while role-playing, debate, engage in case study, take part in cooperative learning, or produce short written exercises, etc.

Active learning share many approaches with "collaborative learning", a situation in which two or more people learn or attempt to learn something together [3] and also with "cooperative learning", an educational approach which aims to organize classroom activities into academic and social learning experiences [4].

Finally, "project-based learning" which is another style of active learning, involves a dynamic classroom approach in which students acquire a deeper knowledge through active exploration of real-world challenges and problems [5].

Examples of activities which fall into these educational methodologies include

- A class discussion which may be held in person or in an online environment, can be conducted with any class size and allows for instructor guidance of the learning experience [6].
- A collaborative learning group where students are assigned in groups of 3-6 people and they are given an assignment or task to work on together. This assignment could be either to answer a question to present to the entire class or a project [7].

- Active learning games which is pedagogical technique that uses playful in-class activities designed to actively engage students with key concepts, the faculty and each other [8].

Second language requires that learners take ownership of learning activities through interaction, active participation and the use of the target language in a more authentic context ([9], [10]). The traditional “chalk and talk” method which involves the teacher talking to students and writing notes on the chalkboard results in rote learning, learners’ low level of retention, and passive learning [11].

Having that in mind, one could easily argue about the importance of applying active learning methodologies in second language learning. In today’s connected world, the internet can play a major role in assisting the activities described above, giving students the ability to overcome the limits imposed by the traditional physical classroom and connect with other students, having similar interests, all over the world while participating in collaborative learning activities. In addition, using mobile devices, the boundaries of the traditional classroom can be extended and blended into the students’ everyday activities, further motivating them to actively participate in such learning activities.

1.2. Aim, Objectives and Methodology

Our aim in this thesis, is to implement a mobile based e-learning collaborative environment, in order to help children learning a foreign language by promoting communication and language development skills through a virtual collaboration environment in which children are encouraged to interact and communicate with other children from all over the world learning the same language.

The interaction would involve several kinds of activities, such as multiuser educational games, synchronous communications through video conference or text chatting and finally participation in virtual classrooms. The system should be easy to understand and use even by non-technical users such as teachers and young children. It should be also easily extendable, allowing programmers and instructional designers to develop new functionality without dealing with low level details.

Finally, the system's software should be publicly available, distributed under an open source license, in order to attract more developers to contribute and, of course, more language teachers to adopt it as part of their teaching methodology.

Initially, we spoke with foreign language teachers, who provided us some insight about their needs, the system's characteristics and how such a system could be utilized in their teaching methodology.

In order to validate these initial insights, we created a survey which was distributed in related to the subject mailing lists and social media groups. The survey helped us to better understand the system's requirements and to proceed with its analysis and design.

Lastly, we wanted to evaluate the implemented system in real world conditions involving both foreign language teachers and students, which wasn't feasible for several reasons, which we will try later in this thesis to describe and analyze.

1.3. Contributions

To the best of our knowledge, there is currently no similar system available. Although there are many systems that target foreign language learning, they are supposed to be used in higher education and/or they are mainly targeting desktop PCs.

Furthermore, the ability of the implemented system to load and integrate external modules, in addition to the fact that the system is released as open source software (under the GPL License) as explained in section 4.2. (Architecture, Development Tools and Libraries), can turn it into a research testbed platform in which teachers, developers and instructional designers can utilize in order to explore new teaching methodologies.

Finally, we should also mention our contributions to moodle in which we submitted a patch to handle external course creation based on templates as explained in section 4.2.4.3. (Moodle Server). Also as part of the Thesis, we have implemented the required functionality for using in a mobile device the BigBlueButton's whiteboard in presenter's mode. This isn't submitted yet to BigBlueButton but we plan to do so in the following months, as explain in chapter 7 (Conclusion and Future Research).

1.4. Structure

We start this thesis in Chapter 2, by providing a literature review in the fields of Computer Assisted Language Learning (CALL) and Mobile Assisted Language Learning (MALL) as well as in the fields of e-learning in general and online tele-collaboration environments.

In Chapter 3, we focus on the analysis and design of the implemented system. We first provide a high level description of the system and its functionality, then present the survey we circulated in order to validate our initial description, along with its findings. We also provide a set of UML use cases and sequence diagrams for the core functionality of the system, some additional functional and nonfunctional requirements and, finally, we present some typical usage scenarios, which highlight the main features of the system and how these could be utilized in a foreign language classroom, along with the traditional teaching methodologies.

In Chapter 4, we present the systems' architecture and implementation details, the tools we used and how these tools are integrated and the context information captured and used by the system.

In Chapter 5, we describe the system's extensibility, by providing details about the areas that can be extended, the type of new functionality that can be added and some technical details along with examples on how can this be accomplished.

In Chapter 6, we first try to describe and analyze our inability to evaluate the system in real world conditions, based mainly on the feedback we received from teachers.

Finally, in Chapter 7, we present our conclusion and findings along with proposals on how the implemented system can provide a base framework for further research in teaching methodologies.

2. Computer and Mobile Assisted Language Learning and E-Learning and Tele-Collaboration Systems.

2.1. Computer Assisted Language Learning

In the 1980s, the application of technology in language classrooms included the use of film, radio, television, language labs with audio and videotapes, computers, and interactive video [12]. Various types of computer assisted language learning (CALL) also became commonplace [13]. Although there were some innovative uses of software, as for example in MacLang in [14], the majority of CALL uses were limited in form to drill-and-practice exercises. As the technology advanced, we began to see more interactive uses of CALL as well as an increase in the integration of various media into the computer system [15]. Computer technology became more accessible to both individuals and schools. Moreover, the growing understanding of its potential encouraged a shift in emphasis from computer technology to its applications. That is, finding ways to use computers for enhancing teaching and learning gained prominence in the research. Today, the use of multimedia, the Internet (especially the World Wide Web), and various forms of distance learning are widespread and interest in using computers as tools to support language learning is growing, both from the perspective of a language educator and that of a language learner [16].

The invention of the internet and its widespread adoption, brought new tools into the field of foreign language teaching and learning, such as email, synchronous chat, bulletin boards and digital video [16], which are generally lauded in the literature as an opportunity to expose students to authentic, culture-laden contexts to which they are able to respond by speaking and writing in the target language [17].

Kroonenberg [18] explored the use of email and chat in two classes of multinational students in grades 9-12. Her findings suggest that email was a “very realistic form of communication because it is a real conversation about real, relevant topics with real people”. Also, she suggested that synchronous chat can be used to share and collaborate, and that it helps in developing critical thinking and “cultivates the ability to think and compose spontaneously”.

2.2. Mobile Assisted Language Learning

In recent years, the growth of mobile device market (smartphones and tablets) led the researchers to coin the term MALL (Mobile Assisted Language Learning).

Insights from CALL can be used to inform the learning activities presented through mobile phones [19], as these phones can be considered as miniature form of PCs with additional benefit of portability which surpasses laptop computers. This evolution of technology from desk, to lap, to palm has literally given a potential language learning tool in the hands of the teachers and their students [20].

Gabarre et al. [21] explored the usage of iPads in foreign language classrooms in a Malaysian public university. Among their findings, the authors mentioned that such devices offer a good learning platform, a claim that is also supported by the works of Godwin-Jones [22] and Miangah and Nezarat [23] who explored various mobile learning applications.

Ogata and Yano in their work [24] presented a mobile application that helped beginner Japanese class students to learn vocabulary by associating it with RFID annotated real world objects.

Palalas [25] used iPod Touch devices in order to deliver to second language students either in-house created audio and video casts, or publicly available audiovisual material such as TED's videos. Additionally, students were encouraged to use their devices to communicate through blogs or email, and to record their written reflections. Palalas' work concludes that mobile devices can deliver interesting, engaging and motivating learning activities. Learners liked these affordances of mobile technology where they can connect with their peers to complete learning activities.

Lan et al. [26] studied the effects of mobile technology in collaborative learning on elementary English learning students. They performed two studies, the first one in traditional (non-technology enhanced) collaborative learning setting and the second one in a mobile technology enhanced collaborative setting. They identified weaknesses and limitations in the traditional collaborative setting and they showed that technology enhanced collaborative learning can overcome these limitations and weaknesses while at the same time it seemed to reduce anxiety in students, promoted motivation to learn, and enhanced oral reading confidence.

Finally, in [27] Park, extends the Transactional Distance theory, which presents a definition of distance learning implying the separation of teachers and learners. According to Park, utilizing mobile devices, the instructors can initiate the learners into a learning environment where they provide initial scaffolding and then gradually withdraw the support to allow learners take charge of their learning.

2.3. E-Learning and Tele-Collaboration Systems

Besides the applications in language learning, recent literature describes various other e-learning and tele-collaboration systems used in different learning fields.

Karvounidis et al. in their work [28] developed a new integrated framework, which covers synchronous and asynchronous education for teaching and learning in higher education. This work led to the creation of an integrated suite, the Unisuite which is designed for all the existing operating systems used on desktops and on mobile devices and can operate smoothly in any browser [29].

Furthermore, Belo et al. [30] showed how the challenge of limited academic infrastructure and human resource can be mitigated by deploying a Synchronous Virtual Learning System built on an open source video conference server (BigBlueButton), integrated with a custom designed e-learning portal, which also has the propensity to be integrated with existing learning management system, in order to provide synchronous e-learning solutions. The learning management system and the resource center was integrated with a video conference server to provide e-learning services such as live web conferencing, file sharing, creation and submission of assignment, presentation of portable document format (pdf) and word documents, extended white board capabilities and desktop sharing.

Finally, Wang et al. [31] showed that synchronous technology affords opportunities for synchronous interactions and communications between teachers and students and among students. This great affordance becomes especially valuable in bridging the gaps among learners and teachers, not only geographically but also culturally. When appropriately applied, it improves online learning by enriching synchronous interactions in audio, video, and text formats, encouraging student collaborations, increasing both social and teaching presence of an online course, providing students

with instant feedback from the instructors and peers, and boosting student motivation to learn and self-efficacy in online learning.

2.4. EuroCALL and the TILA Project

Nowadays, the European Union seems to be a major player in research about Computer and Mobile Assisted Language Learning as within the European Union, there are 23 officially recognized languages, more than 60 indigenous regional and minority languages, and many non-indigenous languages spoken by migrant communities [32].

The Commission's first ever Communication on Multilingualism "A New Framework Strategy for Multilingualism", adopted in November 2005 and now superseded by the 2008 Strategy, set out three basic strands to the EU's policy in this area:

- underlining the major role that languages and multilingualism play in the European economy, and finding ways to develop this further
- encouraging all citizens to learn and speak more languages, in order to improve mutual understanding and communication
- ensuring that citizens have access to EU legislation, procedures and information in their own language

This Strategy complements the Commission's Action Plan "Promoting Language Learning and Linguistic Diversity", adopted in 2003, which set out measures aimed at supporting initiatives carried out at local, regional and national levels designed to extend the benefits of language learning to all citizens as a lifelong activity; improve the quality of language teaching at all levels; and build an environment in Europe favorable to languages by embracing linguistic diversity, building language-friendly communities, and making language learning easier [32].

The European Association for Computer-Assisted Language Learning (EuroCALL) [33], started in 1993 as an EU funded project and since its inception has been closely involved with many European projects and is publishing its own journal (ReCALL) [34], which seeks to fulfil the stated aims of EUROCALL as a whole, and more particularly to promote the use of foreign languages within Europe and beyond, providing an international focus for the promulgation of innovative research in the area of computer-assisted language learning and technology-enhanced language learning in education and training.

In our opinion, the most important project is the TILA project [35] which has been funded by the European Commission within the Lifelong Learning Programme (2013-2015) and seeks to implement telecollaborative activities in secondary schools' foreign language curricula in order to make language learning a more authentic, attractive and relevant experience [36].

Dedicated focus studies in the TILA Research Report analyze intercultural communication and learning in relation to telecollaboration in 'tandem' and 'lingua franca' constellations.

Special attention is given to

- task design and blended learning integration,
- teacher and learner perspectives on telecollaboration for intercultural communication and development,
- linguistic self-efficacy and language anxiety in telecollaboration, negotiation of meaning in intercultural communication,
- teacher's roles in telecollaboration.

Compared to this Thesis, the TILA project is also based in the principles of active learning methodology and exploits the same tools (BigBlueButton and moodle) for synchronous and asynchronous learning activities and is targeting desktop PCs.

Furthermore, the TILA projects has integrated the opensim 3D application server in order to create virtual world environments that can be used for synchronous collaboration activities.

On the other hand, the current thesis is primarily targeting mobile devices which gives the ability to blend learning activities with the student's natural environment and thus provide a more student centered learning approach. On the other hand, the advantage of using a virtual world environment is that teacher can configure these virtual environments and thus provide a set of learning activities focused on specific study subjects.

3. Requirement Analysis and Design

3.1. Human Centered Design and Agile Software Engineering Methodologies

The software engineering approach followed by this Thesis, is a combination of Human Centered Design and the SCRUM agile software engineering methodology.

Agile software development has become popular since the late 1990's. It became popular as many projects failed because it took too much time from finalized requirements specification to first tests of the developed system. The approach was consolidated by a manifesto at the beginning of 2000's [37] that characterizes the agile idea by twelve main principles that have to be followed [38].

By far the most popular agile method is Scrum. It was strongly influenced by a 1986 Harvard Business Review article on the practices associated with successful product development groups. It is an iterative, incremental framework for projects and product or application development. It structures development in cycles of work called Sprints. These iterations are no more than one month each, and take place one after the other without pause. The Sprints are timeboxed – they end on a specific date whether the work has been completed or not, and are never extended. At the beginning of each Sprint, a cross-functional team selects items (customer requirements) from a prioritized list. The team commits to complete the items by the end of the Sprint. During the Sprint, the chosen items do not change. Every day the team gathers briefly to inspect its progress, and adjust the next steps needed to complete the work remaining. At the end of the Sprint, the team reviews the Sprint with stakeholders, and demonstrates what it has built. People obtain feedback that can be incorporated in the next Sprint. Scrum emphasizes working product at the end of the Sprint that is really "done"; in the case of software, this means code that is integrated, fully tested and potentially shippable. Key roles, artifacts, and events are summarized in Figure 1 [39].

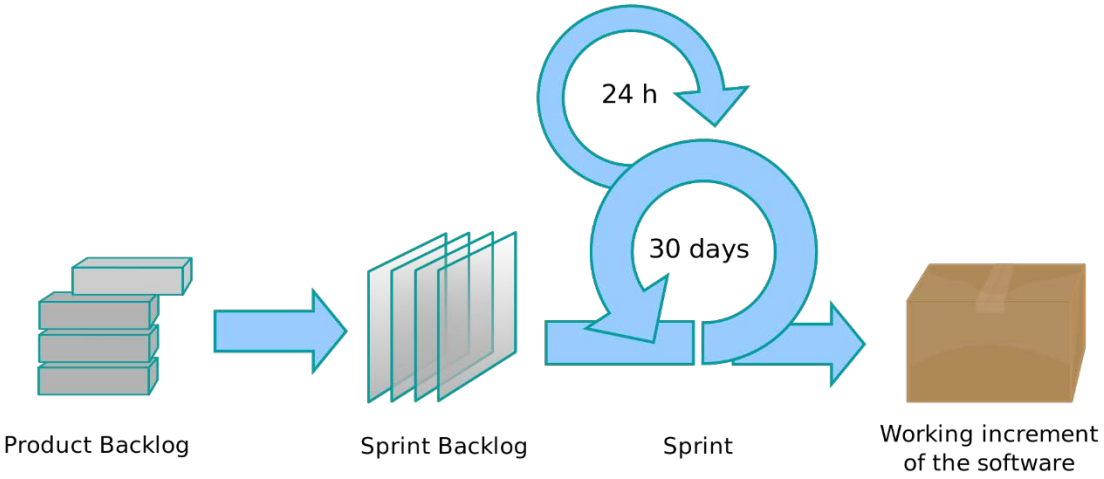


Figure 1: The scrum process
(from https://en.wikipedia.org/wiki/Scrum_(software_development))

A major theme in Scrum is “inspect and adapt”. Since development inevitably involves learning, innovation, and surprises, Scrum emphasizes taking a short step of development, inspecting both the resulting product and the efficacy of current practices, and then adapting the product goals and process practices. Repeat forever [39].

In the same way that SCRUM is popular for software engineering experts, HCD is popular for usability and user experience experts. One of the main reasons for its success is that context of use, the requirements of the users and the evaluation of design solutions play an important role. User requirements are more important than technical features that software engineers might derive. Users get what they really want [39].

The HCD process has been standardized by ISO 9241-210 [40]. Figure 2 gives a visual overview of the corresponding process model [39].

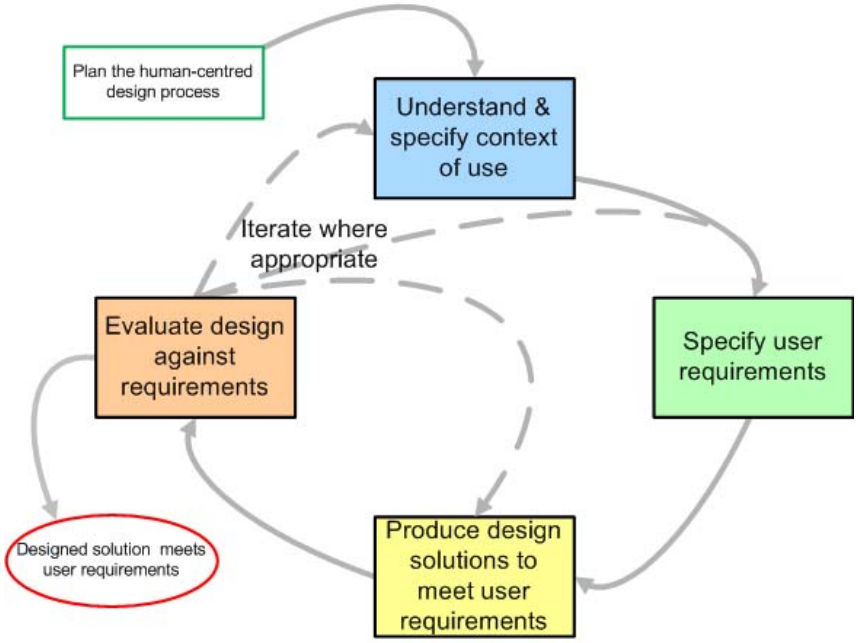


Figure 2: The human-centered design process (ISO 9241-210)

An interesting approach which combines the two methods and would give the chance to combine the advantages of both approaches, is proposed by Paelke et al. [41] who published a process model and called it Agile UCD-Process, which is depicted in Figure 3.

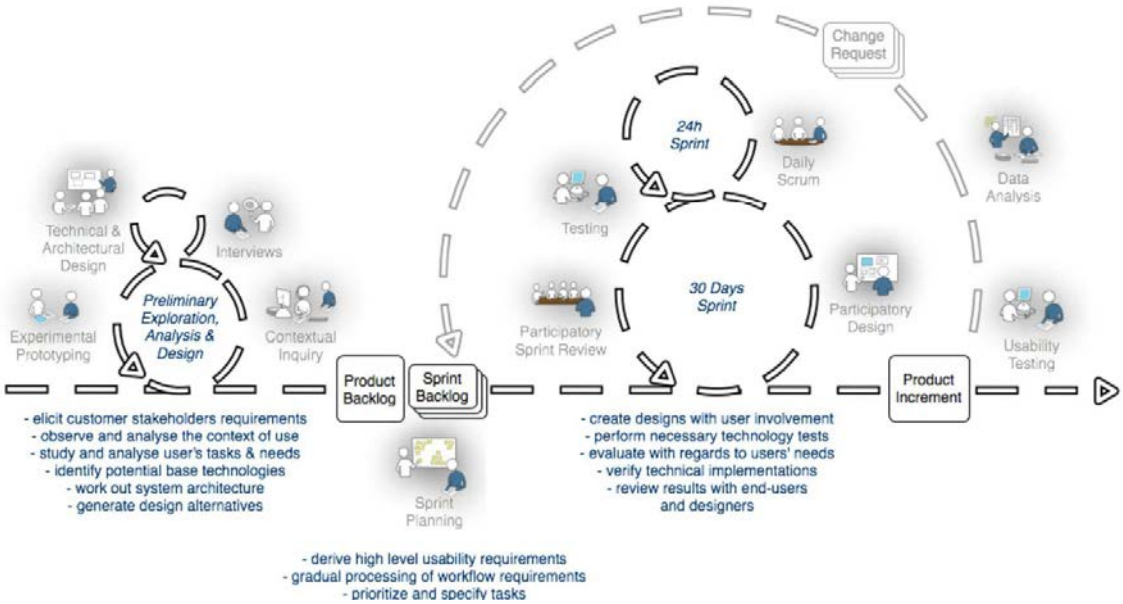


Figure 3: The Agile UCD-Process

In the UCD process model, the requirements elicitation is only loosely coupled with the software development process. A stronger coupling was suggested by Paul et al. [42]

and it is depicted in Figure 4. According to this approach all created software influences the HCD process.

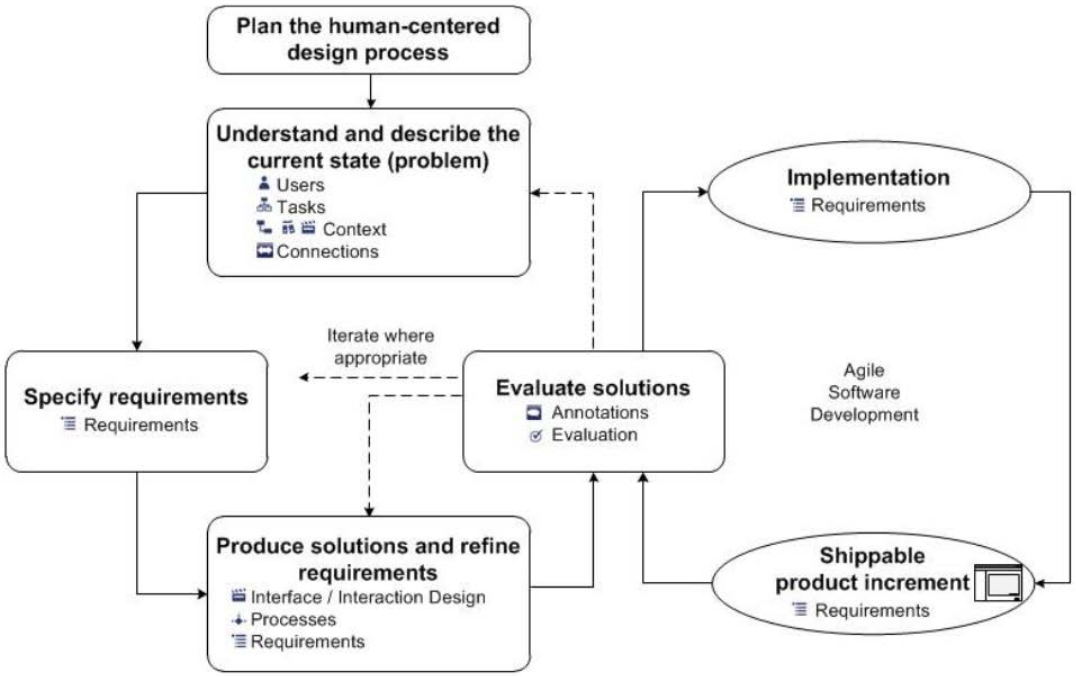


Figure 4: Extended User-Centered Design Process

Having in mind both the requirements and limitations of the current Thesis, we followed a mixed (hybrid) model.

Initially we had one-to-one interviews with a small number of professionals in the field of foreign language teaching. These informal interviews were conducted in a friend-cycle and they provided an intuition about the need for intercultural real time communications in foreign language teaching and how it can be accomplished through the use of technology. They also provided us with a crude set of functional and nonfunctional requirements and helped us envision and provide a high level description of proposed system.

Having this high level description as a starting point, we conducted an online survey which helped us to assess the current use of technology in foreign language teaching classrooms and also to validate the usefulness of the proposed system's characteristics that were captured during the interviews.

Based on the survey's quantitative results, which, as we will see later in this chapter, confirmed the initial findings in the interviews, we compiled a list of functional and nonfunctional requirement and a set of UML Diagrams and proceeded with the

development phase, following the SCRUM methodology, and asking feedback from the users (foreign language teachers) whenever required. For example in cases we missed something during the design process, in cases that we had to choose between different options, etc.

3.2. High Level Description

3.2.1. Architecture Diagram

The architecture diagram, which follows a 3-tier approach, is shown in Figure 5. The concept is that clients (PCs/Laptops, Smart Phones and Tablets) initially connect to the **Management Server** (MS) which a) acts as a coordinator between the clients, the **BigBlueButton Server** (BS) [43] and the **E-learning (Moodle) Server** (ES) [44] and b) offers additional functionality to the clients and middleware.

3.2.2. Synchronous and Asynchronous E-Learning and Collaboration Activities

The system should be able to communicate with either a local or a remote BigBlueButton installation and integrate it, in order to provide synchronous e-learning (online classrooms, educational games, etc.) and collaboration (online meetings, video/text chats, etc.) activities.

Furthermore, the system should be able to communicate with either a local or a remote moodle installation and integrate it, in order to provide asynchronous e-learning (courses, etc.) and collaboration (wikis, forums, etc.) activities.

3.2.3. Additional Functionality

This role of the MS is related to additional functionality offered or needed by the system, such as administrative/logistics related tasks (e.g. user management functionality like authentication, user profile management, classroom and calendar management, etc.). To accomplish this, it uses a **Database Server** (DS) which acts as storage for information saving/retrieval.

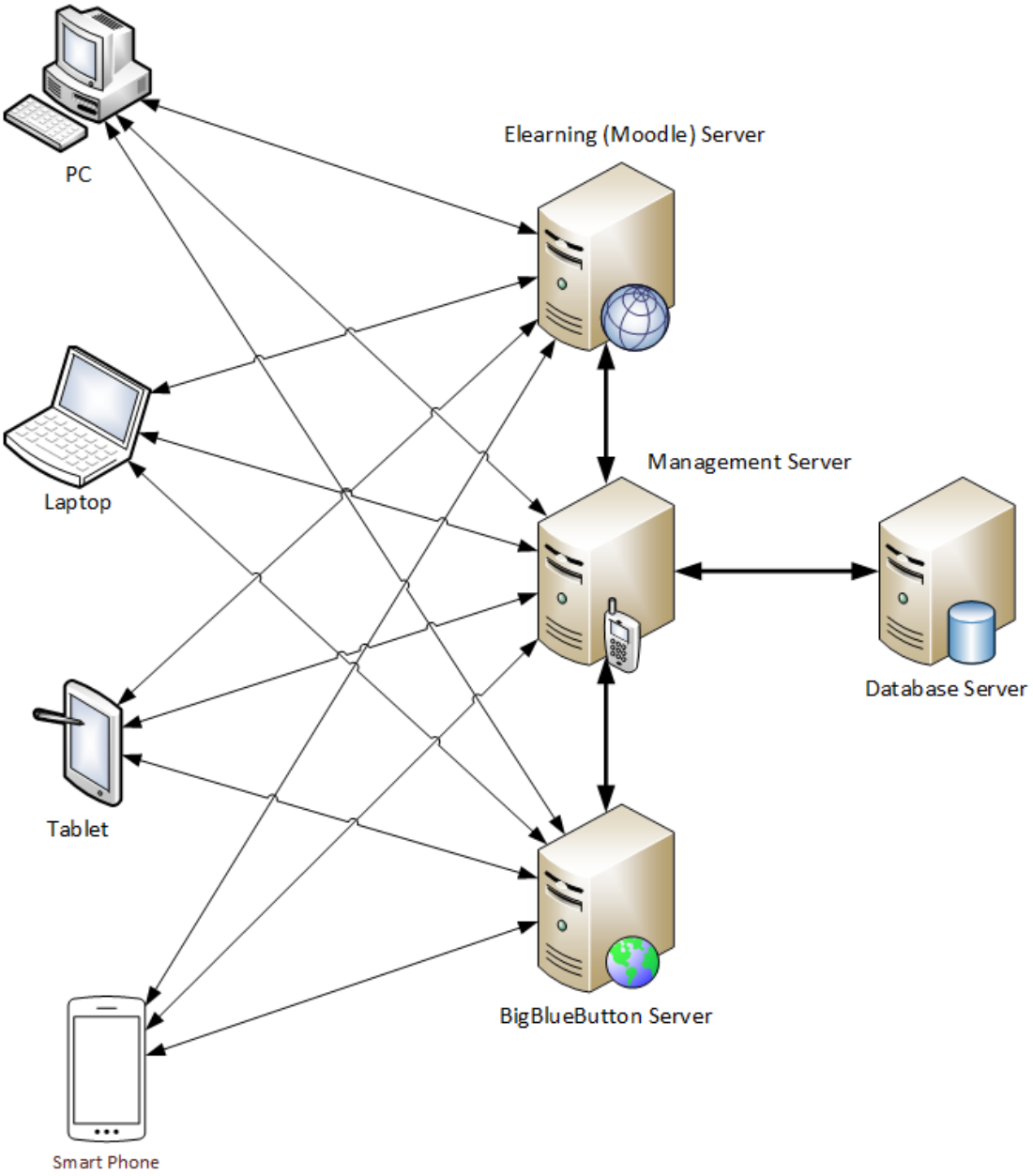


Figure 5: Architecture Diagram

3.2.4. User and Device Context Awareness

Both the desktop and mobile client should be able to adapt to the user and device that is running by exploiting context information such as:

- *User's role*

  There are three different roles used by the application: the student, the teacher and the parent which will be described in more detail in following sections.
- *User's cultural background*

Given that the system will be used by different users all around the world, it should be able to adapt to different cultures. Mainly, that means that the client should support multiple languages and translations which should be modifiable by a teacher according to her student's needs.

- *Device specifications*

  Finally, the client should consider the device's specifications and capabilities such as display size and resolution, available sensors (e.g. microphones, web cameras), etc.

3.2.5. Users Roles

As mentioned, there are three different roles used by the application: the student, the teacher and the parent. After a user is logged in to the system, she is provided with a different set of functionality as described below.

3.2.5.1. Student

This is the role that all participating children are assigned to. The functionality offered by the application is shown in Figure 6.

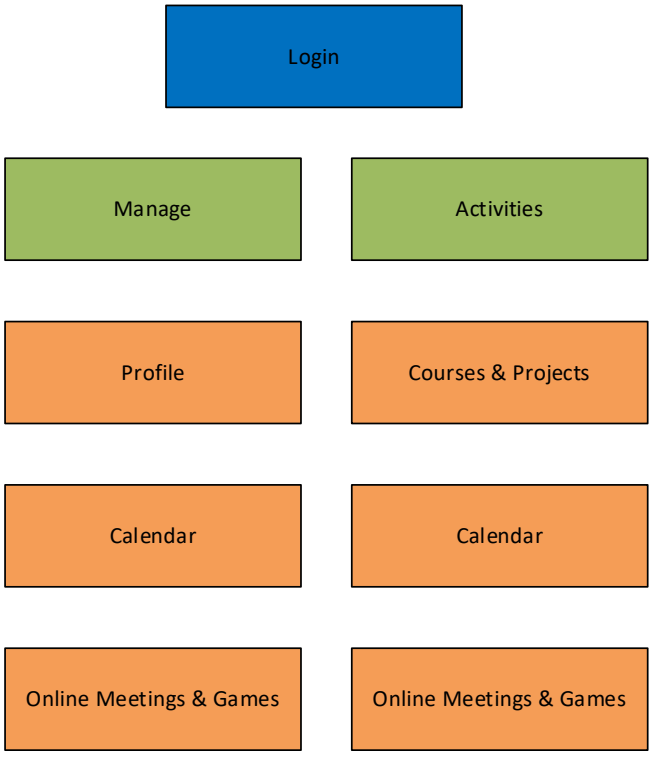


Figure 6: Student's Application Diagram

A student first of all can manage (i) her personal details (Profile) which provide information such as her name, contact information etc., (ii) her calendar (Calendar) which provide information about future tasks (e.g. scheduled online meetings) that were entered either by her, or by one of her teachers. She can manage her unattended online meetings and games (create, modify or delete a meeting) in which she is participating, join a running meeting or game and review a meeting's recording (Online Meetings and Games). Finally, she can visit a moodle course page which she is enrolled through a classroom she is attending or a project she is participating (Courses and Projects).

3.2.5.2. Teacher

As its name implies, this the teacher's role. The functionality offered by the application is shown in Figure 7.

As with the student role, a teacher can also manage various things like (i) her personal details (Profile) which provide information such as her name, contact information etc., (ii) her calendar (Calendar) which provide information about future tasks (e.g. her classroom schedules) and which can be personal or shared with one of her classrooms. She can manage her students' profiles (Students), the online meetings and games in which she or her students are participating, join a running meeting or game and review a meeting's recording (Online Meetings and Games). She can also manage her classrooms (Classroom), the classroom groups related to her classrooms (Classroom Groups), projects that her students are participating (Projects) and also create and edit the contents of moodle courses (Courses) that can be assigned either to a classroom, classroom group or project.

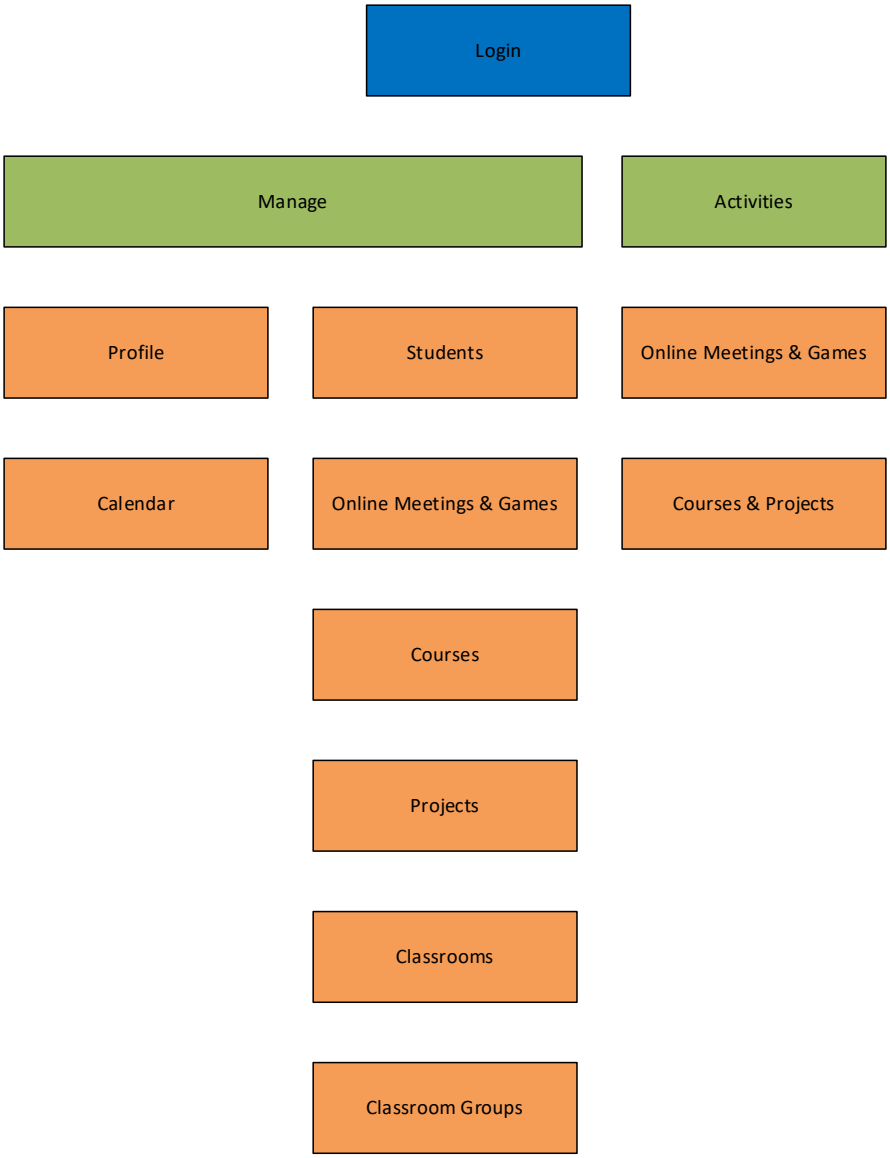


Figure 7: Teacher's Application Diagram

3.2.5.3. Parent

The final role is that of a child's parent. This is the most limited role and its functionality is shown in Figure 8.

The parent's main functionality is related to supervising his child's activities inside the system. So further to her profile (Profile), she can see her children's profiles (Children) and (i) deny them the ability to join unattended meetings or (ii) auto-approve unattended meetings, approve her children's participation to an unattended meeting (Online Meetings and Games) and finally review her children's past and future events (Calendar).

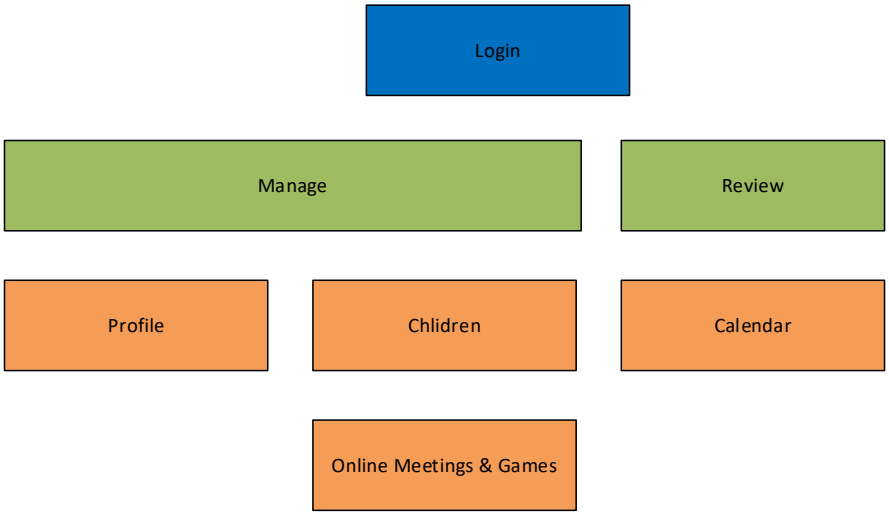


Figure 8: Parent's Application Diagram

3.2.5.4. Administrator

One additional administrator user is predefined in the system. This user, as implied by its name, can manage every aspect of the system. The only exception is that this user cannot interact with other users, by creating and joining meetings with other users, or viewing a meeting's recording, although she can manage these entities if required.

3.3. Survey

Having this high level description, we created an online survey, which was addressed to foreign language teachers through related mailing lists and groups in social media (mainly linkedin and facebook) and took place during September 2014.

3.3.1. Questionnaire

The questionnaire (Appendix A-1) included three sections as follows.

3.3.1.1. Demographics

In the first section we collected demographic data information from the participants. The questions asked where about their country, sex, age, occupation, the language(s) they teach and the total number of years they teach and finally their total number of students.

3.3.1.2. Current usage of ICT in classrooms

Next, we tried to capture the current usage of ICT in foreign language classrooms. We focused on the activities related to the proposed system's characteristics and the technology that support these activities. In particular we wanted to know if foreign language teachers use any technology and the usage frequency (Never, Rarely, Sometimes, Often, Always) for

(a) Multimedia (audio and/or video) playback either from the internet, from local media (CDROM, DVDROM etc.), or through non-computer based equipment (CD/DVD Players, TV, Video Projectors, etc.).

(b) Usage of educational software (games, dictionaries, etc) either from the internet, or from local media (CDROM, DVDROM etc.).

(c) Educational content sharing and distribution with their students through email, personal websites and/or blogs, online file sharing services (dropbox, Microsoft's OneDrive, Google's GDrive, etc.).

(d) Online teaching between students using either common desktop sharing applications (skype, google hangouts, etc.) or more specialized (whiteboard based) telecollaboration software (BigBlueButton, openMeetings, etc.).

(e) Schedule management through online shared calendars (google calendars).

(f) The usage of interactive whiteboards.

(g) Other technologies used.

3.3.1.3. Assessment of the proposed system's characteristics

Finally, we asked the responders to provide their feedback on the proposed system's characteristics. First, we asked their opinion about the importance (Not at all important, Somewhat important, Neutral, Fairly important, Very important) of the following characteristics and optionally any other characteristics they consider important:

(a) Online distance teaching.

(b) Educational content sharing (courses, notes, assignments etc.).

(c) Online tests and quizzes.

(d) Multiplayer educational games.

(e) Online real-time communications between students and/or teachers.

(f) Learning through interaction between students based on their surrounding environment (i.e. their home, their toys and in general their everyday life).

(g) Software adaptation based on the user's profile (age, sex, cultural background, etc.).

(h) Email based user notifications for assignments deadlines, additions of new content, etc.

(i) SMS based user notifications for assignments deadlines, additions of new content, etc.

Then, we asked if intercultural communicative competence and telecollaboration are important elements in foreign language teaching asking them optionally to clarify their answers.

Finally, we asked for their degree of agreement in the following security/privacy related issues and if they have other concerns about the proposed system.

(a) Parents should be able to supervise their children activity in the system.

(b) Parents should be able to impose restrictions on the system's usage by their children.

(c) Students should be able to use the system outside a classroom for socializing with other students.

(d) Students should be able to use the system without a teacher's supervision.

(e) There should be one or more teachers with elevated privileges on the system. For example, teachers who can have access to all the data stored in the system.

(f) The system should automatically record every meeting between children when there are no teachers participating in order to prevent cases of cyberbullying.

3.3.2. Results

3.3.2.1. Demographics

Responders came from six different countries with the majority of them being from USA or Greece (66.04% and 26.42% respectively) (Table 1), with most of them being females (Table 2) and having an age distribution as shown in (Table 3).

Regarding the responders' occupations (Table 4), more than half (58.49%) were private educators and as we can see in Figure 9, there is a strong correlation between private educators and Greece.

Regarding the teaching languages (Table 5), most of the responders are teaching either English or Spanish (49.15% and 32.20% respectively) and 6 out of 53 are teaching more than one language. It is also interesting here to explore the breakdown of languages per country as shown in Figure 10. As we can see in this figure, English is the most common taught language between the responders who came from a non-English speaking country (i.e. non-US), while the most common taught language in the US is Spanish.

Finally, the responders have an average of 14.13 years working as foreign language teachers (standard deviation = 8.26) and each of them has on the average 68.68 students (standard deviation = 69.13). The large standard deviation in this last case implies either a big diversification between the responders or that simply in this case the sample is not large enough.

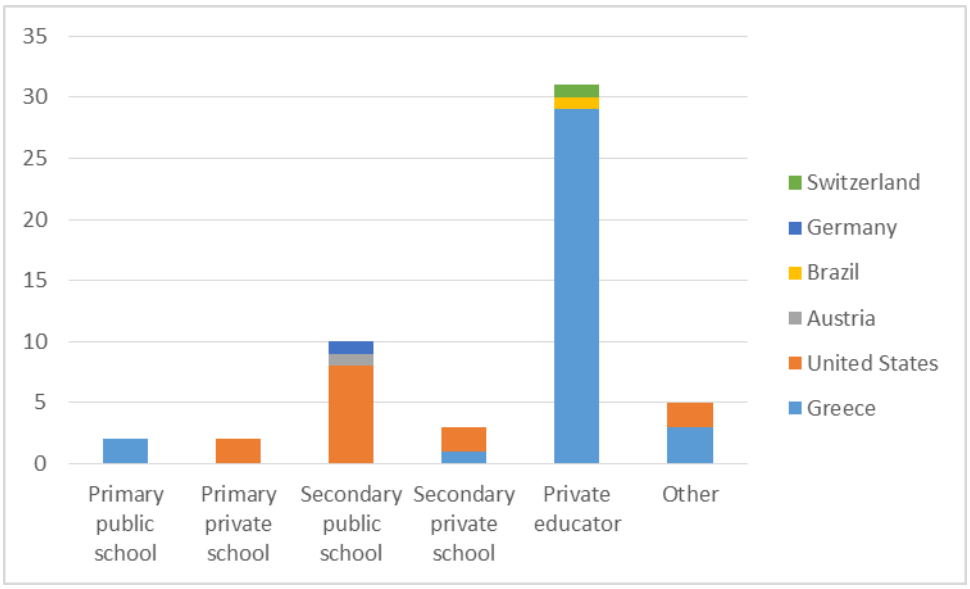


Figure 9: Responders by Occupation and Country

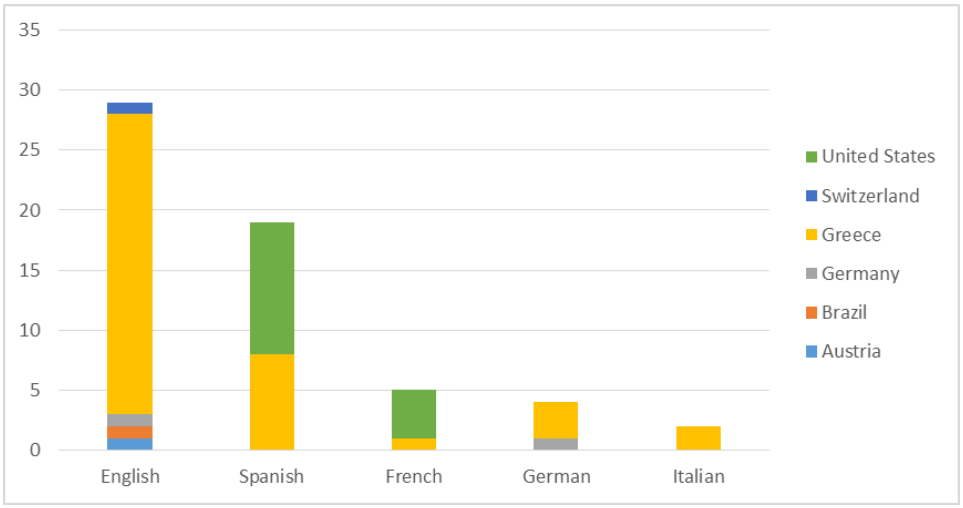


Figure 10: Responders by Teaching Language and Country

Table 1: Responders by Country

Country	Number of Responders	Percent of Responders
Greece	35	66.04%
United States	14	26.42%
Switzerland	1	1.89%
Brazil	1	1.89%
Austria	1	1.89%
Germany	1	1.89%

Table 2: Responders by Sex

Sex	Number of Responders	Percent of Responders
Female	46	86.79%
Male	7	13.21%

Table 3: Responders by Age Group

Age Group	Number of Responders	Percent of Responders
Under 25	1	1.89%
25-29	10	18.87%
30-34	10	18.87%
35-39	10	18.87%
40-44	9	16.98%
45-49	2	3.77%
Over 49	11	20.75%

Table 4: Responders by Occupation

Occupation	Number of Responders	Percent of Responders
Primary public school	2	3.77%
Primary private school	2	3.77%
Secondary public school	10	18.87%
Secondary private school	3	5.66%
Private educator	31	58.49%
Other	5	9.43%

Table 5: Responders by Teaching Language

Teaching Language	Number of Responders	Percent of Responders
English	29	49.15%
Spanish	19	32.20%
French	5	8.47%
German	4	6.78%
Italian	2	3.39%

3.3.2.2. Current usage of ICT in classrooms

Initially, we summarize the activities (Audio Playback, Video Playback, Usage of Educational Software, Content Sharing, Interactive Whiteboard, Schedule Management and Online Teaching) based on the frequency of use and present the percentages of high usage (Often or Always) on any technology for each activity (Table 6) and also the percentages of low usage (Never or Rarely) on all technologies for each activity (Table 7). I.e. in order for a response to be included in Table 6, it must have at least one or more Often or Always answer for an activity, and in order to be included in Table 7 it must have all answers either Never or Rarely for an activity.

The exact responses for this section are summarized in Figure 11.

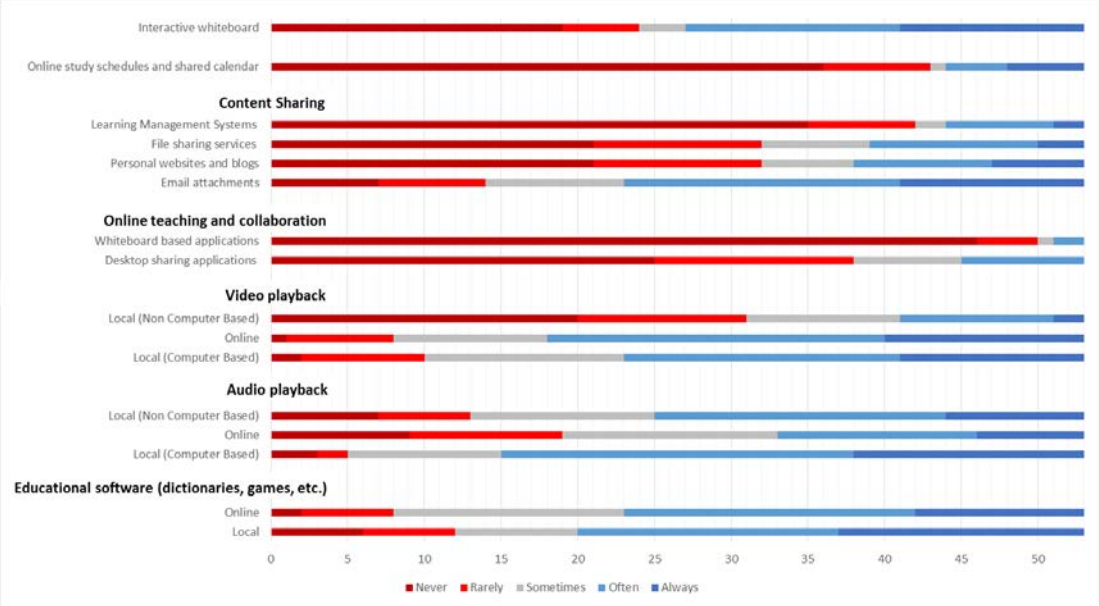


Figure 11: Usage of ICT Technology in Classrooms

Table 6: High Usage for Activities

Activity	Percentage
Audio Playback	92.45%
Video Playback	81.13%
Educational Software	79.25%
Content Sharing	66.04%
Interactive Whiteboard	49.06%
Schedule Management	16.98%
Online Teaching	15.09%

Table 7: Low Usage for Activities

Activity	Percentage
Audio Playback	0.00%
Video Playback	9.43%
Educational Software	7.55%
Content Sharing	22.64%
Interactive Whiteboard	45.28%
Online Teaching	71.70%
Schedule Management	81.13%

As we can see from Tables 6 and 7 Audio Playback is used from almost all teachers (92.45% are using audio playback often or always in their classrooms, compared to 0.00% who are never or rarely using it). Also from Figure 11, we can see that in most cases the audio is delivered offline mostly using computer based equipment although non-computer based equipment is also very common.

Video Playback is used by most teachers (81.13% are using video playback often or always in their classrooms, compared to 9.43% who are never or rarely using it). From Figure 11, we can see that in this case, online video playback is very popular (only 1 out of the 53 responders never uses online videos). Also computer based offline video play back is very common (only 2 out of the 53 responders never use computer based offline videos).

Educational software is used often or always by the 79.25% of the responders and only 7.55% never or rarely uses it. As we can see in Figure 11, local (offline) and online educational software is almost equally common.

ICT for content sharing is used often or always by the 66.04% of the responders, while 22.64% never or rarely uses ICT for content sharing. Email attachments, is the most common way to deliver content to students, while the usage of Learning Management Systems is the less common way (Figure 11).

Interactive whiteboard is used often or always by almost the half of the responders (49.06%), while 45.28% never or rarely uses it.

Schedule management through online shared calendars is not so popular between the responders, as only the 16.98% of them are often or always using shared calendars, comparing with 81.13% who never or rarely uses them.

Finally, online teaching is also not popular. Only 15.09% are often or always using online collaboration applications in contrast with 71.70% who never or rarely uses such applications. From Figure 11, we see that the most common applications used for online teaching are desktop sharing applications (like skype, google hangouts, etc) while the usage of whiteboard based applications (like BigBlueButton) is very limited.

3.3.2.3. Assessment of the proposed system's characteristics

3.3.2.3.1. Importance of characteristics

Figure 12, summarizes the responses for this section. As we can see all of the proposed characteristics are considered either fairly important or very important by more than 50% of the responders.

Software adaptation based on the user's profiles is considered very important by the 50.94% of the responders, with an additional 24.53% of the responders considering it to be fairly important, which leads us to the conclusion that user context awareness is an important feature of the proposed system.

It is also worth to mention here the importance of educational content sharing and online teaching which are considered to be very important or fairly important by the 90.57% and the 52.83% of the responders accordingly. As we saw in the previous section, these two activities are currently not so popular between the responders, and so we conclude that the integration and usage of a Learning Management System (i.e. moodle) for content sharing and also the integration and usage of a whiteboard based telecollaboration software (i.e. BigBlueButton) for online teaching collaboration are also important features of the proposed system.

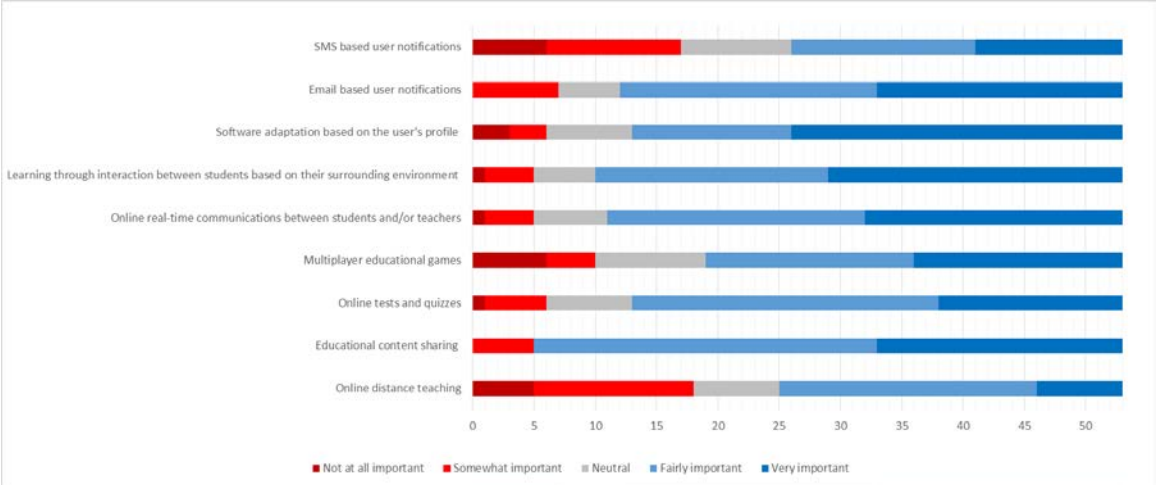


Figure 12: Importance of the proposed system's characteristics

3.3.2.3.2. Importance intercultural communicative competence and telecollaboration

Moving to intercultural communicative competence and telecollaboration we see (Table 8) that responders consider both of these to be important in foreign language teaching (83.02% and 66.04% respectively).

Table 8: Importance of intercultural communicative competence and telecollaboration in foreign language teaching

	Yes		No	
Intercultural communicative competence is important	44	83.02%	9	16.98%
Telecollaboration is important	35	66.04%	18	33.96%

The most important reason for the importance of intercultural communicative competence as clarified by the responders, was that language competence is strongly bound to cultural assimilation, with typical answers being like the followings:

- *"A foreign language not only provides the learner with vocabulary and grammar structures, but also with knowledge about different cultures and their ways of life"*
- *"Language teaching without culture is teaching only theory (swimming without water)"*
- *"A language is more than just what is spoken. Learning a language encompasses culture, history and art"*

Also many consider intercultural communication in foreign language learning to be an important factor against racism:

- *"It can create a bridge between the two cultures and students feel more comfortable with a culture that they know"*
- *"Προωθείται η ειρηνική αλληλοκατανόηση και συνύπαρξη που αποτελεί βασικό στόχο της εκπαίδευσης και εργαλείο επιβίωσης για την κατανόηση του άλλου και την εκρίζωση του ρατσισμού"* ("Promotes mutual understanding and peaceful coexistence which is a key objective of education and survival tool to understand one another and to eradicate racism ")
- *"Είναι σημαντική η επικοινωνία μεταξύ παιδιών από διαφορετικούς πολιτισμούς ώστε να έχουν ανοιχτούς ορίζοντες να είναι δεκτικά και με σεβασμό στην προσωπικότητα του αλλού"* ("Communication between children from different

cultures is important in order to be open-minded, receptive and respectful of different personalities").

Finally, some responders mention in their comments the need of intercultural communication due to the today's world globalization:

- *"Intercultural communication is important because the world is becoming smaller and smaller and English is the lingua franca"*

Regarding the importance of telecollaboration, most of the responders in their comments mention motivation:

- *"Offers a real life situation motive"*
- *"It can motivate and give authentic practice"*
- *"Enhances and enriches instruction and motivation"*

Also some responders see telecollaboration as a learning method through human-to-human interaction:

- *"telecollaboration allows for human to human contact"*
- *"αλληλεπίδραση χωρίς περιορισμούς"* ("[Students'] interaction without restrictions")

And finally many responders agree that technology and online socialization is already part of the students' lives, so it can also be used for learning:

- *"It is the way of the future"*
- *"It is something that actually happens in real world"*
- *"Εφόσον τα νέα παιδιά ασχολούνται καθημερινά με online εφαρμογές και δίκτυα, θα μπορούσε να συνδυαστεί για εκπαιδευτικούς σκοπούς"* ("Since young children are involved daily with online applications and networks, this could be used for educational purposes ")

3.3.2.3.3. Security/privacy related issues

Responders' degree of agreement regarding security and/or privacy issues related to the proposed system are depicted in Figure 13.

First of all we see we can see an agreement about the system's proposed feature to automatically record every online meeting between students where there is no teacher present and also the existence of one or teachers having elevated privileges on the

system (i.e. teachers with admin rights). 77.36% of the responders either agree or strongly disagree with these two features.

For the rest of the characteristics in this section, although there is some degree of agreement from most the responders, we believe that we cannot conclude to a clear result as strong agreement is rather limited, and so the system should give the ability to the teachers to parameterize these feature according to their needs, and making sure they fully understand the effect of each of parameters.

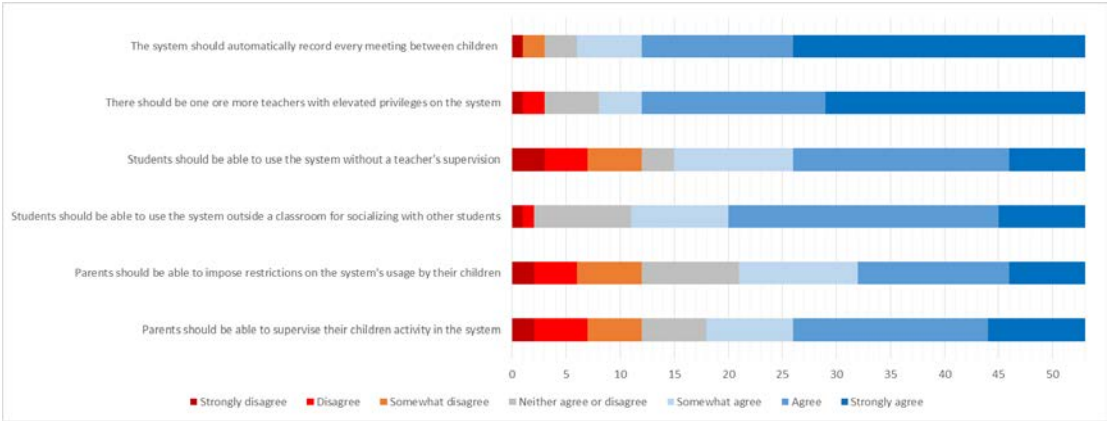


Figure 13: Degree of agreement for security/privacy related issues

3.3.3. Summary

As we saw, most of the responders agree that both intercultural communicative competence and telecollaboration are important elements in foreign language teaching, and thus the responders validate our initial claim about the usefulness of the proposed system.

Furthermore, most of the responders find the proposed system's characteristics important in foreign language teaching. In fact, based on the responders' answers, it seems that some of the proposed characteristics are currently missing from foreign language teaching although considered important, like for example tools for online teaching and telecollaboration. Also, educational content sharing seems to be underused, based on its importance according to the responders' answers.

Finally, we saw that the responders are in general positive regarding the proposed security and privacy features of the system, although there is not strong agreement, which guides us to the conclusion that features which have to do with the

security/privacy of the system should be implemented in a way that they can be customized by each teacher.

3.4. Additional Requirements: User Interactions

Based on the initial interviews, our understanding was that the users' general interactions with the system and between them would follow social media principles. That means that the system would provide both students and teachers the ability to self-organize into (let's say) groups based on common interests and would provide them a more direct way of communications between each other. For example a student while studying, she could seek help from other students or teacher who were available at that time by looking up who was online and directly communicating with one of them.

After some initial prototyping it turned out that this wasn't something that teachers wanted. They wanted a more structured system which they could control. That means that the students couldn't interact directly with each other if a teacher wasn't aware about this. For example, regarding online meetings and games, students can schedule a meeting but that must be approved by a teacher, unless that teacher has authorized the particular users, otherwise. Also students could not create their own groups, but they can only be grouped together by a teacher, in a classroom or in a project.

3.5. Use Case, Domain Model and Sequence Diagrams

As we have seen in a previous section in which we presented the architecture of the system (Figure 5, Section 3.2.1), the server side of the system is composed of three subsystems (a) the Management Server, (b) the BigBlueButton Server and (c) the Moodle (E-learning) Server.

3.5.1. Management Activities

3.5.1.1. Domain Model

The domain model diagram for the main entities of Management Server is depicted in Figure 14.

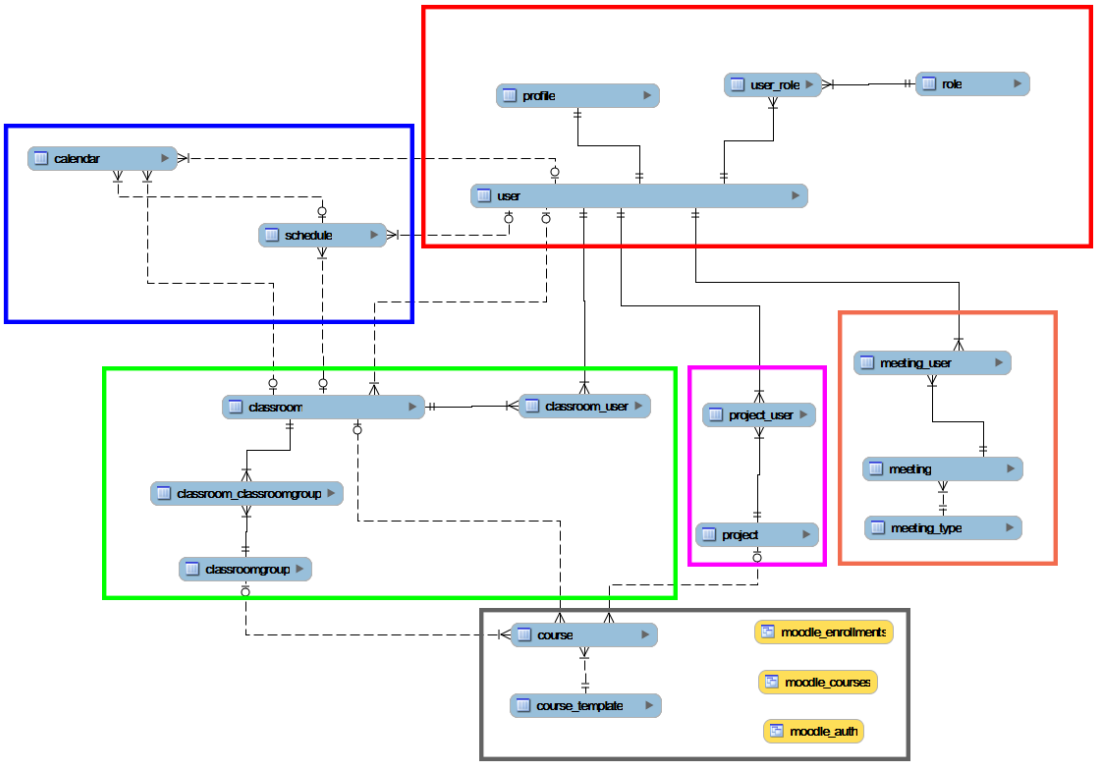


Figure 14: Domain Model

It can be further broken down into 6 main subsystems:

1. User Management Subsystem (Red Color)
2. Classroom Management Subsystem (Green Color)
3. Project Management Subsystem (Magenta Color)
4. Calendar Management Subsystem (Blue Color)
5. Meeting Management Subsystem (Orange Color), which provides the bridge between the Management Server and the BigBlueButton Server.
6. Course Management Subsystem (Grey Color), which provides the bridge between the Management Server and the Moodle (E-learning) Server.

Besides these subsystem, the system contains some additional subsystems, not included in Figure 14, for the shake of clarity:

7. Authentication Subsystem, which includes the required functionality for authenticating users and granting access to resources.
8. Configuration Subsystem, which is responsible for handling installation related configuration.

9. Language and Translation Subsystem, which is responsible for handling user interface languages and the translation of the various messages to the user's language.

3.5.1.2. Use Case: Registration / Authentication

As we need the system to be safe from unauthorized access, the system should employ an authentication/registration mechanism as shown in Figure 15. In its basic form, a new user's registration should be approved by a teacher and the login procedure should be based on the typical username/password combination. The system should also provide the ability for convenient automatic registration approvals and password-less authentication by exploiting context information.

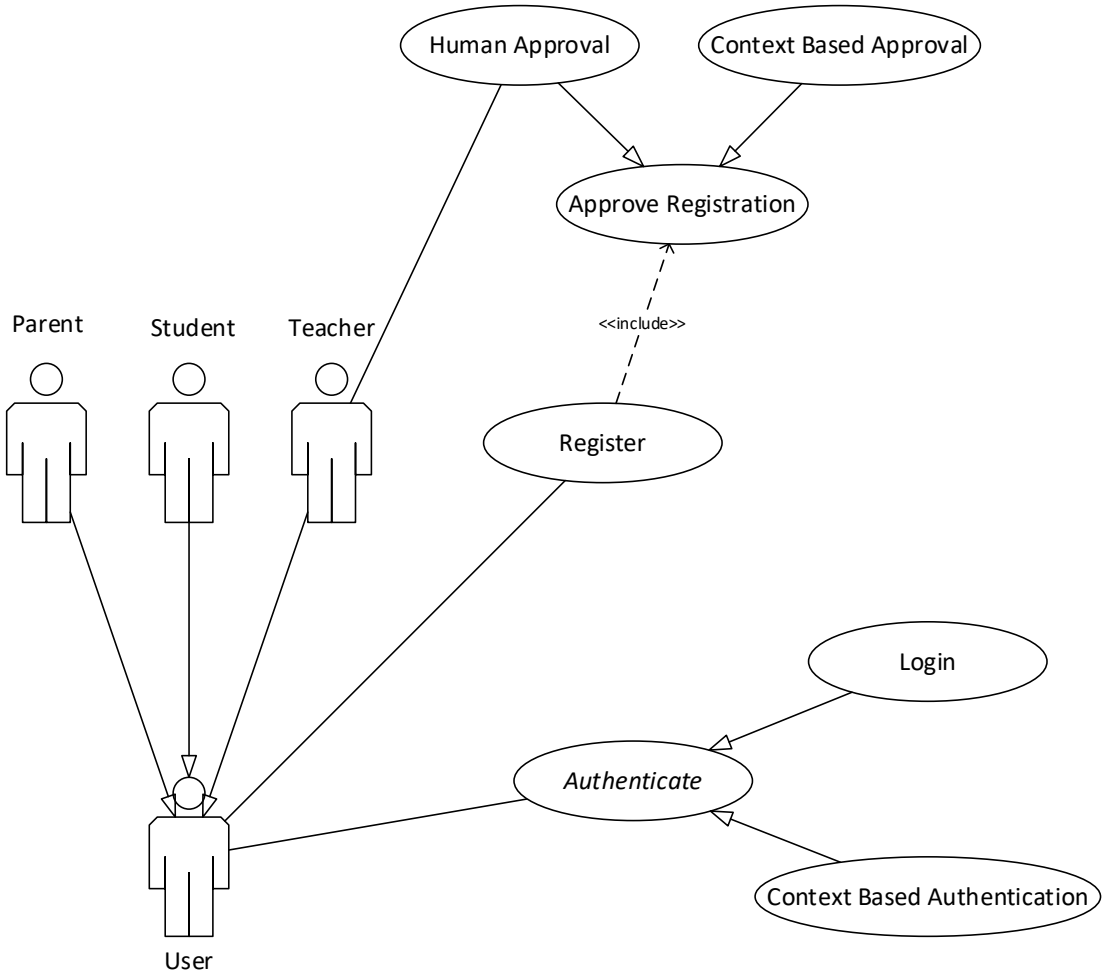


Figure 15: User registration and authentication use case diagram

3.5.1.2.1. Register

Name: Register

Identifier: A1

Actors: User

Description: Describes the User's registration procedure.

Precondition: No active login exists in the application that is running in the User's device.

Basic Flow:

1. The User chooses the option "Create a new account" on the login screen of the UI.

2. The system presents the registration form.

3. The User enters the required information (username, password and an email account) and presses the "Submit" button.

4. The system validates the data making sure the username is unique, the email address has not been used for another registration and the password meets predefined complexity rules (length etc.).

5. The system creates the new user's account and notifies the User that her registration is complete.

6. *Include Use Case A2 "Approve Registration"*

7. The use case ends in success condition.

Alternate Flow:

4.1. The system fails to validate one or more fields.

4.2. The system notifies the User about the fields failed to validate.

4.3. The use case ends in failure condition.

Post conditions:

Success: The User has a valid account and can now login to the system.

Failure: The User returns to the login screen of the UI.

3.5.1.2.2. New Registration Approval

Name: Approve Registration

Actors: Teacher

Identifier: A2

Description: Describes a user's registration approval.

Precondition: A user has submitted her details in order to create a new user's account.

Basic Flow:

1. The system sets the user's account in disabled state and sends a notification containing an approval link to the Teacher.

2. The Teacher approves the new account by pressing the "Approve Account" link on the notification.

3. The system enables the user's account and notifies the user.

4. The use case ends in success condition.

Alternate Flow:

1.1. The system using context information infers that a Teacher is in proximity and proceeds in step 3.

2.1. The Teacher doesn't approve the new account (i.e. she ignores the notification)

2.2. The use case ends in failure condition.

Post conditions:

Success: The user has a valid account and can now login to the system.

Failure: The user cannot login to the system.

3.5.1.2.3. Login

Name: Login

Actors: User

Identifier: A3

Description: Describes how a User logs in to the system.

Precondition: No active login exists in the application that is running in the User's device.

Basic Flow:

1. The User chooses the "Login" option on the login screen of the UI.

2. The system presents the "Login" form.

3. The User enters her username and password and presses the "Submit" button.

4. The system verifies that the provided information is correct.

5. The system creates a new session for the User and destroys any other active sessions for the same User.

6. The use case ends in success condition.

Alternate Flow:

2.1. The system using context information infers that the User is trying to login from a location that has logged in again in the past and skips to step 5.

4.1. The provided username/password combination is not correct.

4.2. The system notifies the User.

4.3. The use case terminates in Failure condition.

Post conditions:

Success: The user is logged in to the system and she is now in the main screen of the UI.

Failure: The user is not logged in to the system and she is returned to the login screen of the UI.

3.5.1.2.4. Discussion

We deliberately didn't include a typical "Remember me" functionality in order a user to be able to login without providing a username/password, as this could impose a security risk in case of stolen or lost mobile devices. In that case a malicious user could be able to use the system and impersonate the device's owner.

We understand that the fact that a teacher should approve every registration, as described in Use Case A2 ("Approve Registration") and that a user should always

provide a username/password in order to gain access to the system, may be inconvenient in many application's scenarios, so the system should be designed in such way that it can be easily extended with other registration approval and authorization mechanisms.

Such mechanisms, may include context based approaches like proximity based authentication and registration approval where a student can register without approval or can login without providing a username/password combination, if a teacher is in proximity. In this case, the location proximity could be inferred by the system, using knowledge of the shared radio environment in a way similar to the works of Varshavsky et al. [45] and Nguyen et al. [46].

Furthermore, we propose the notion of trusted locations in which a user can login without providing a username/password combination. Such trusted locations, could be for example a user's home or school. The proximity of the actual location of the user to these trusted locations could again be inferred by the system, using knowledge of the current radio environment and comparing it to stored previous locations of the user.

In the next chapter, we will provide more details about the proximity based registration and authentication method we implemented.

3.5.2. Synchronous E-Learning and Collaboration Activities

3.5.2.1. Use Case: Managing and Joining a Meeting

Figure 16 provides the overall use case diagram for the synchronous collaboration activity (meeting). As the system may contain different types of meetings in which users can interact with each other, we don't describe each meeting's use case but a general (abstract) use case.

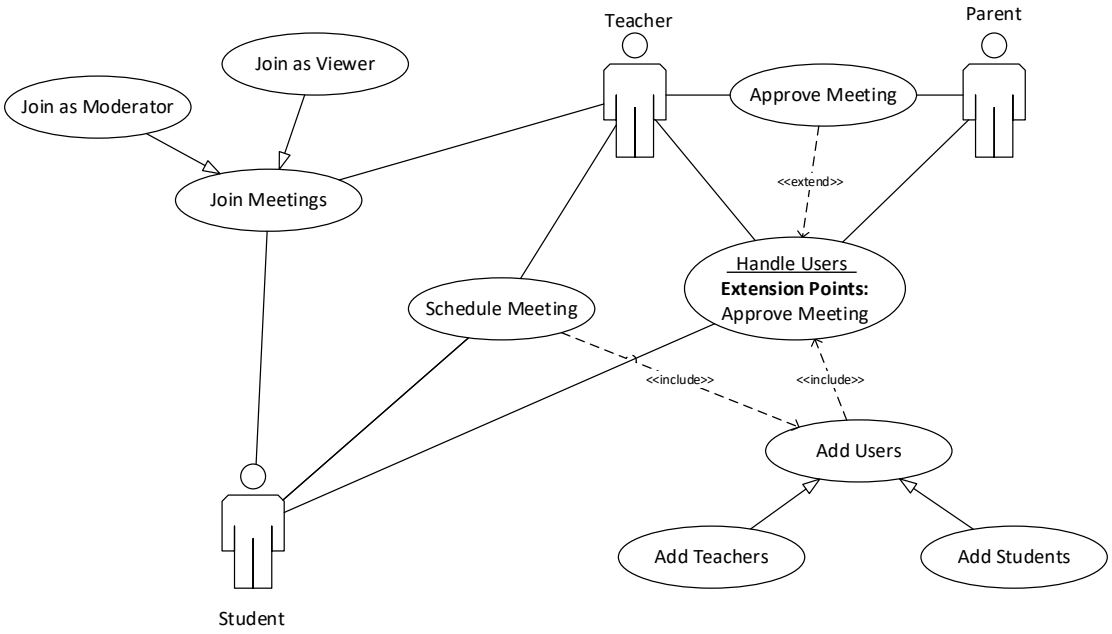


Figure 16: Managing and joining a meeting use case diagram

3.5.2.1.1. Schedule Meeting

Name: Schedule Meeting

Actors: Teacher or Student

Identifier: C1

Description: Describes how an Actor can schedule a meeting for a collaboration activity.

Precondition: The Actor is logged in to the system as either a Student or a Teacher.

Basic Flow:

1. The Actor selects the "Schedule a Meeting" option on the main screen of the UI.

2. The system provides the Actor with a list of available meeting types according to the Actor's role.

3. The Actor selects a meeting type from the available list

4. The system displays a calendar.

5. The User selects a date and time for the schedule, and presses the "Schedule meeting" button on the calendar screen of the UI.

6. The system creates in the Actor's calendar a pending new entry for the scheduled meeting.

7. *Include Use Case C2 "Add Users"*

Post conditions:

Success: A new schedule for a meeting is entered in the participants' calendars.

Failure: Nothing changes.

3.5.2.1.2. Add Users

Name: Add Users

Actors: Teacher or Student

Identifier: C2

Description: Describes how an Actor can add users to a scheduled meeting, she has created.

Precondition: The Actor has scheduled a meeting.

Basic Flow:

1. The Actor chooses the "Add Users" option on the scheduled meeting screen of the UI.

2. The system provides the Actor with a list of users according to the Actor's role.

3. The Actor selects from the list the users she wants to invite and presses the "Add Users" button on the list of users' screen of the UI.

4. The system adds the users to the meeting and creates (if and when required) a notification for their parents and teachers in order to approve the meeting.

5. *For each invited user include Use Case C3 "Handle Users"*

Post conditions:

Success: A new schedule for a meeting is entered in the participants' list of meetings.

Failure: Nothing changes.

3.5.2.1.3. Handle Users

Name: Handle Users

Actors: User

Identifier: C3

Description: Describes how a User handles her addition a scheduled meeting.

Precondition: The User has been added to a scheduled meeting.

Basic Flow:

1. The User opens the scheduled meeting and doesn't perform any action.

2. The use case ends in success condition.

Alternate Flow:

1.1. The User opens the scheduled meeting and removes herself (or her child or student if she is a parent or teacher accordingly) from the meeting.

1.2 The use case ends in failure condition.

Post conditions:

Success: Nothing Changes.

Failure: The User is removed from the scheduled meeting.

Extension points:

2: *Use Case C4: Approve Meeting.*

3.5.2.1.4. Approve a Scheduled Meeting

Name: Approve Meeting

Actors: Teacher or Parent

Identifier: C4

Description: Describes how an Actor can approve a meeting in which only Students are participating.

Precondition: A Student has scheduled a new meeting and only other Students are invited.

Basic Flow:

1. The system creates a notification for the Student's Parent and Teacher, informing them about the scheduled meeting.

2. The Actors approve the Student's participation in the meeting by pressing the "Approve Meeting" link on the notification.

3. The system updates the status of the meeting.

4. The use case ends in success condition.

Alternate Flow:

2.1. The Teacher or the Parent rejects the Student's participation in the meeting or ignores the notification.

2.2 The use case ends in failure condition.

Post conditions:

Success: The scheduled meeting is approved (if the Actor is a teacher), or the specific user is approved for the scheduled meeting (if the Actor is a parent).

Failure: Nothing changes.

3.5.2.1.5. Join Meeting

Name: Join Meeting

Actors: User

Identifier: C5

Description: Describes how a User joins a meeting.

Precondition: A confirmed schedule for a meeting exists in the User's calendar.

Basic Flow:

1. The system displays a reminding notification to the User a predefined amount of time before the meeting's scheduled time.

2. The User presses the "Join Meeting" link on the notification.

3. The system joins the User to the meeting with the appropriate role (either a moderator or viewer) according to the meeting's creator or type.

4. The use case ends in success condition.

Post conditions:

Success: The User is joined in the meeting.

Failure: The User is not joined in the meeting.

3.5.2.1.6. Discussion

Regarding the approval of scheduled meetings, we need to clarify that in case that a teacher is participating in a meeting, it doesn't need a parent's approval. Parent's approvals are required only in case of unattended meetings (if the system's configuration allows for such meetings) and only in case that it isn't already approved by a teacher. In case of meetings that are approved by a teacher, these are just appearing to the parent's notifications for informational purposes only.

3.5.2.2. Sequence Diagram: Joining a Meeting

When joining a meeting, except from the client initiating the request, there are several servers involved for fulfilling it. The coordinator is the Management Server (MS) and the sequence of the exchanged messages for the Use Case C5 ("Join Meeting") is shown in Figure 17.

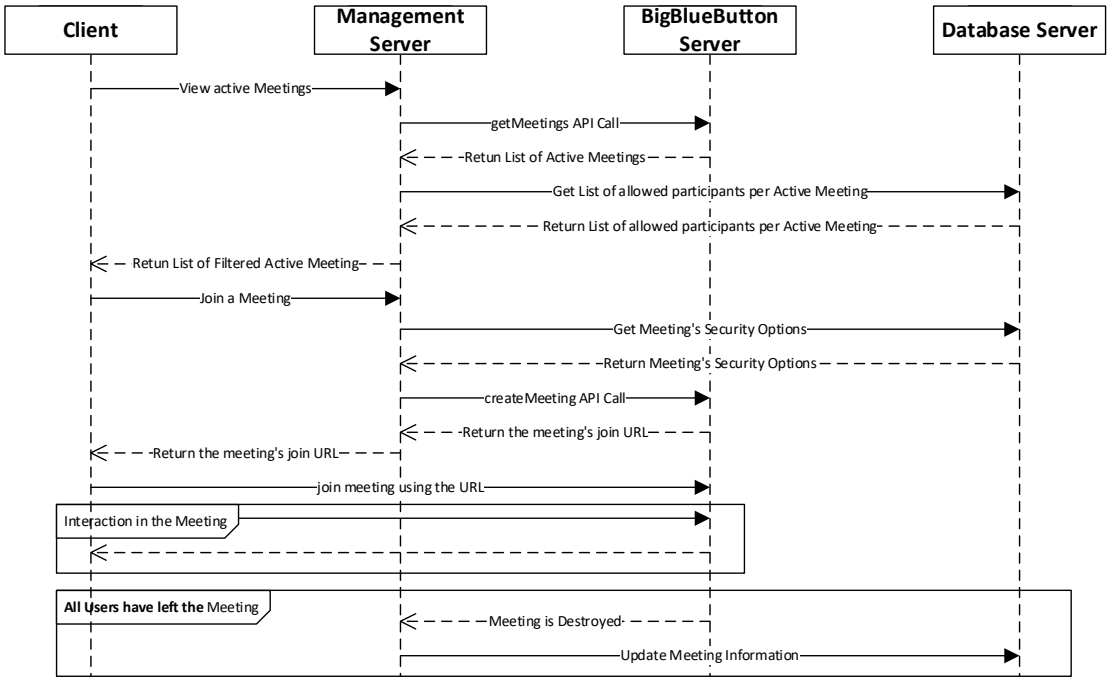


Figure 17: Joining a meeting sequence diagram

3.5.3. Asynchronous E-Learning and Collaboration Activities

The E-learning activities are based in Moodle's courses. A course in Moodle can be parameterized according to the teacher's needs and that functionality will be available in the system as follows.

A course can have different formats such as [47]

1. *Social format* which is based on a single forum for the whole course. It's useful for less formal courses.
2. *Topics format* in which the course is broken down into a number of sections one for each topic. The teacher can add content, forums, quizzes, and other activities to each topic section.
3. *Weekly format* in which the teacher can specify a course start date and the number of weeks the course is to run. A section will be created for each week of the course and the teacher can add content, forums, quizzes, and so on in the section for each week.
4. *Single Resource or Activity*.

Each course can be composed of different types of resources such as [47]:

1. *Text pages* which are simple pages of text. They don't have many formatting options, but they are the simplest tool.
2. *Web pages* which are HTML formatted pages and they can be created using the HTML editor provided by Moodle.
3. *Links to a files, directories or web sites* which as its name implies, are links to external resources such as web sites or to resources uploaded in the Moodle server (files and/or directories).

Finally each course can contain activities for students such as [11]:

1. *Assignments* which is a tool for collecting student work, either uploaded files or assignments created on- and offline.
2. *Forums* which are threaded discussion boards.
3. *Glossaries* which are dictionaries of terms that can be created for each week, topic, or course.
4. *Quizzes* which are web-based quizzes with a variety of question types, such as multiple choice, true/false, short answer, and matching.
5. *Wikis* which are collaboratively edited web pages.

3.5.3.1. Use Case Diagram: Managing and Visiting a Course's Page

Figure 18 depicts the use case diagram of managing a course or visiting a course's page.

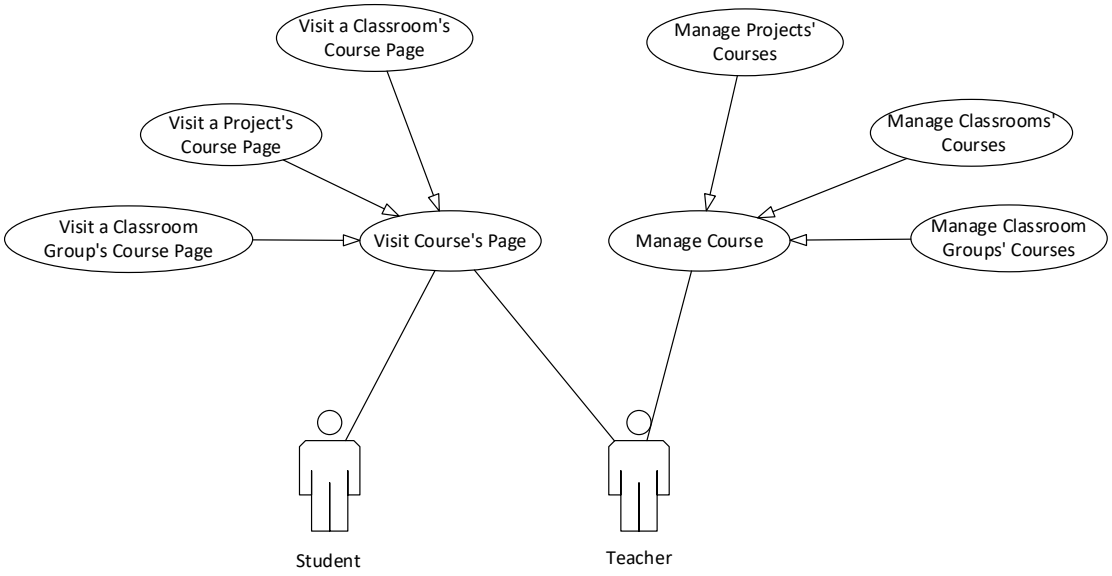


Figure 18: Managing and visiting courses

3.5.3.1.1. Manage Course

Name: Manage Course

Actors: Teacher

Identifier: E1

Description: Describes how a Teacher can manager (create, edit, delete, assign to Students) a course.

Precondition: The Teacher is logged in to the system.

Basic Flow:

1. The Teacher choose the “View Courses” option on the main screen of the UI.

2. The system displays a list of available courses.

3. The Teacher selects a course from the list of available courses.

4. The client application opens a web browser window pointing to the Moodle Server’s course page.

5. The Teacher makes the modifications she needs to the course.

6. The Teacher saves the modifications.

7. The use case ends in success condition.

Alternate Flow:

3.1. The Teacher creates a new course by pressing the “New Course” button in list of available courses screen of the UI.

3.2. The Teacher selects an existing course template and assigns the course to either a classroom, classroom group or project.

5.1. The Teacher doesn’t perform any modifications.

5.2. The use case ends in failure condition.

6.1. The Teacher doesn’t save the modifications.

6.2. The use case ends in failure condition.

Post conditions:

Success: The course is updated according to the Teacher’s modifications.

Failure: The course is not updated.

3.5.3.1.2. Visit Course’s Page

Name: Visit Course’s Page

Actors: Student or Teacher

Identifier: E2

Description: Describes how a Student or Teacher can visit a course’s page in order to interact with it according to her role.

Precondition: The Student or Teacher is logged in to the system.

Basic Flow:

1. The Student or Teacher choose the “View Courses” option on the main screen of the UI.

2. The system displays a list of available courses.

3. The Student or Teacher selects a course from her list of available courses.

4. The client application opens a web browser window pointing to the Moodle Server’s course page.

5. The Student or Teacher interact with the course’s page.

6. The Student or Teacher stop interacting with the course and closes her web browser. 7. The use case ends in success condition.

3.5.3.2. Sequence Diagram: Visiting a Course's Page

As in case of the collaboration activities, in the e-learning activities except from the client initiating the request, there are several involved servers for fulfilling it. Given the web based nature of Moodle, the Management Server's role is limited to creating a onetime password for the current user and provide the client with the URL to the course's page which includes this onetime password. The sequence diagram of the exchanged messages for visiting a course's page (Use Case E2 "Visit Course's Page") is shown in Figure 17.

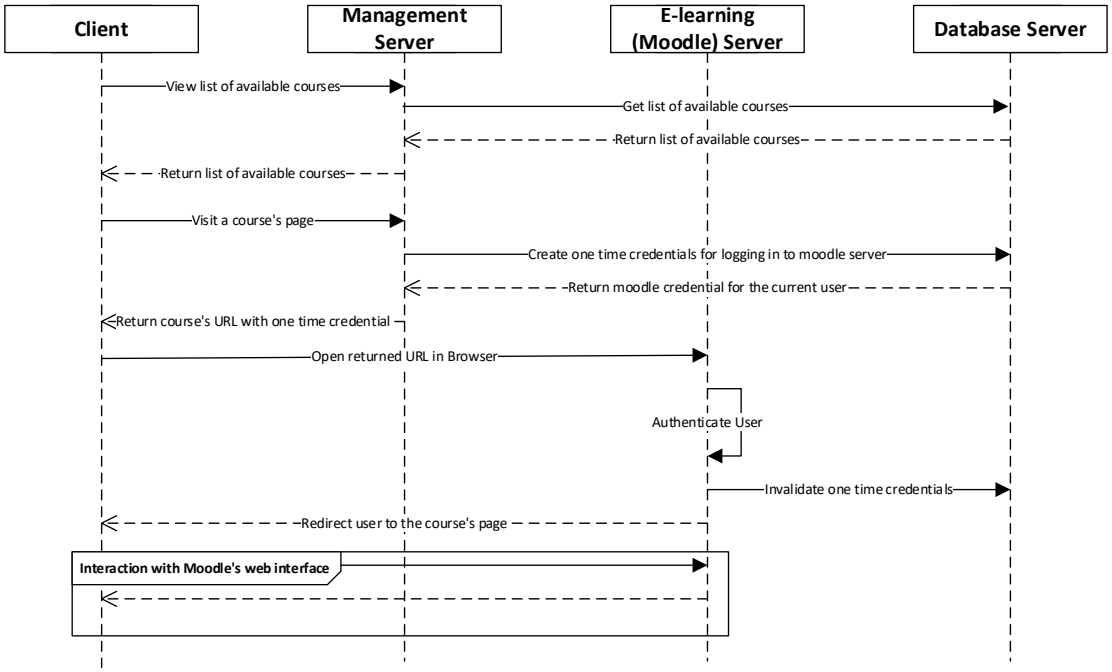


Figure 19: Visit a course's page sequence diagram

3.6. Application Usage Scenarios

As the last section of this chapter, we present a couple of usage scenarios, which offer glimpses about how the application and the tools that it provides can be used and exploited in real world situations, in order to enhance the traditional foreign language teaching methodologies.

3.6.1. Students Introductions

As the academic year was about to end, Letitia, a primary school English teacher, from Spain and Maarten, also a primary school English teacher, from Netherlands, were discussing about the opportunity to have an online meeting with their fifth graders. The application was already used with great excitement by other teachers in their school, and Maarten though that it was a good idea, to get their students of A1 level online in order to get to know each other by introducing themselves. Letitia found the idea great, but she suggested that, in addition to the students' introduction, each student could show off their favorite toys to each other, something that Maarten agreed to.

So, as a first step, after registering as teachers in the system, they created their classes and joined them in a group, so that they could have online meetings together, and instructed their students how to install the application in their mobile phones or tablets.

The two teachers, wanted to make sure that their students, knew the required vocabulary, in order to describe their toys, so they created a group assignment, in which every student could upload up to three photographs of their favorite toys. Having each student's favorite toys photographs, they then created a glossary web page in which they add the uploaded photographs, along with a couple short phrases describing the toys ("This is my ball", "This is my bicycle", etc.) and a sound file with the pronunciation of each of these phrases.

After a couple of offline, traditional classroom, lectures in order to get the students familiar with the vocabulary and its usage, the two teachers, after deciding along with their student the best time of the meeting, created the online meeting and added all the students to it.

During the meeting the teachers instructed the students how to enable their mobile devices' cameras and microphones (which was just a click of a button) and then enabled the students' microphone one a time, having each student introduce himself and then showing the other students his favorite toys. The meeting was a great success and the students were really excited as, until now, they haven't though that they have so many commons with kids living in a different country.

After the meeting, the teachers created a new assignment for their students, asking them to write a few sentences about who they met during the meeting and also what toys they

liked most. The students would have access to the meeting's recording and also to the glossary web page that was created before the meeting.

The two teachers also decided to allow the student's parents to access to the meeting's recording. They thought that it could provide a good marketing material for their schools and they were right. The student's parents were impressed that their kids could participate in a discussion with other kids, after just a year of English teaching.

3.6.2. A virtual visit to museums

Gabriel, a 17 years old boy from France, was for many years interesting in Greek history and especially the classical period and have already visited the Louvre museum several times in order to see the Greek exhibits. In his English classroom (C2 level) he has met, in previous teacher guided online meetings, Nikos, a 17 years old boy from Greece, and Eleni, a 16 years old girl also from Greece, who were also learning English (C2 level) and who both shared the same interest about the Greek classical period with Gabriel.

Gabriel wanted for many years to visit Acropolis and the Acropolis museum in Athens, but so far he didn't have the chance to do so. He thought that it could be a good idea to ask Nikos and Eleni to have a virtual tour. Nikos and Eleni where excited about the idea, as at the same time Gabriel could provide to them a virtual in Louvre museum, so they asked their teacher if they could have an online meeting for this. Their teachers agreed, but in addition to that they asked their students to write a wiki article about the Greek classical period. The three students agreed to that and setup an online meeting. Their teachers after approving the meeting, created a new project in the application, added the three students in it and also setup a wiki page for that project. Later they could move this wiki page to their schools' public wiki.

Gabriel thought that it would be great if Simon, a 17 years old boy from England, who Gabriel met last year in a students' exchange program, could join the meeting, as Simon could provide them a virtual tour to the Parthenon marbles, located in the British museum, and asked his teacher about this. His teacher, after consulting Simon's and the other students' teachers and parents agreed to this, so Simon installed the application in his mobile phone and registered to the system. Gabriel's teacher approved his registration and added him in the project with the rest of the students.

Everything was setup up! At the meeting's time every student was at his place: Gabriel inside Louvre's museum, in front of the famous Aphrodite of Milos statues, taking photos that he could later use in the project's wiki. Simon was already in Room 18, where the Parthenon marbles are kept and also taking photos and short videos with his mobile. Nikos was in the Acropolis, while at the same time Eleni was inside the Acropolis museum. After a short small talk about their day so far, and the introduction of Simon made by Gabriel to Nikos and Eleni, they started by asking Nikos go them through the various buildings in the Acropolis. Simon, has asked one of the museums guides to provide them some additional information about the history of the marbles and with the help of Nikos' camera to show them their original location. Finally, they went through the exhibits in both Louvre and Acropolis museums through Gabriel's and Eleni's cameras respectively. They didn't realize how fast 5 hours have past!

In order to organize their work, the four students asked their teachers to create a new forum for their project, in which they could upload any photos and videos they had from their visits in the museums and Acropolis as well as links to other material found on the internet. Later, they would arrange for additional online meetings in order to discuss the material and decide about the content to include in their project's wiki page.

4. Implementation

4.1. Targeted Client Devices

Our goal was to target the most common devices and operating systems available in today's market. In desktop PCs Microsoft's Windows is the most widespread operating system, having a market share of 83.27% as of May 2016, according to StatCounter [48] as depicted in Figure 20.

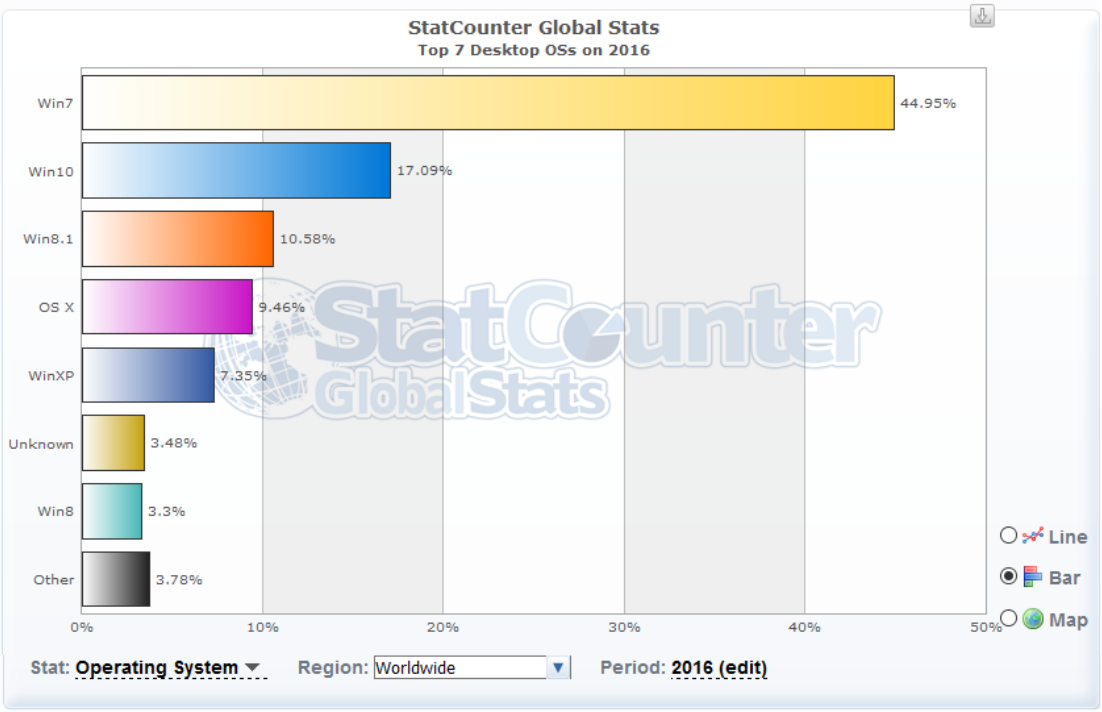


Figure 20: Top 7 Desktop OSs on 2016

Regarding the mobile and tablet devices, according to StatCounter [49] the market is dominated by Google's Android operating system with a total market share of 63.78%, as depicted in Figure 21.

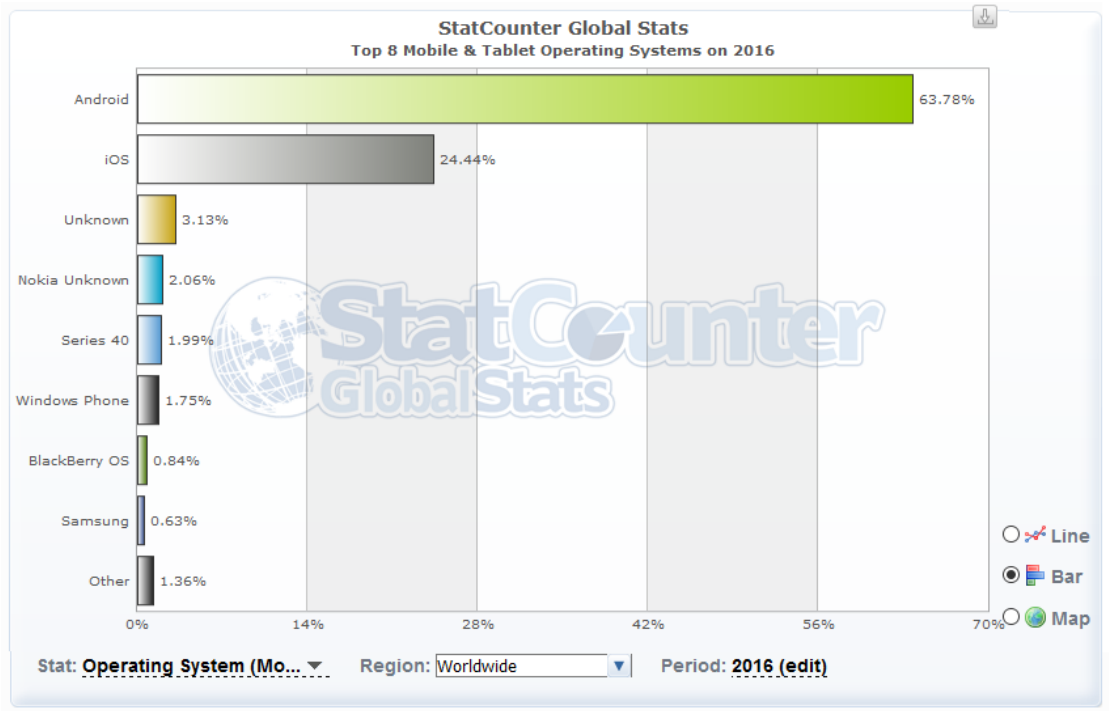


Figure 21: Top 8 Mobile & Tablet OSs on 2016

We should also note that Windows based tablets are starting to gain a significant market share, but these are running the desktop version of Microsoft Windows operating systems, so they are included in the targeted devices for this Thesis. In fact during the implementation phase, we have performed a lot of testing in a Microsoft Surface Pro 3 tablet which is considered one of Microsoft's own flagship devices.

Apart from the obvious requirements of a camera and microphone equipped devices, we assumed that the devices will have a screen resolution of 1920x1080 (Full HD) or more, which seem to be the standard for any modern (3 years old or newer) desktop PC, tablet or smart phone.

Furthermore, in case of tablets and smartphones an active stylus would improve the user experience in some cases (like for example while interacting as a presenter with BigBlueButton's whiteboard module). The most common devices that fall into that category are Microsoft's Surface and Surface Pro devices which are equipped with an N-trig digitizer, or the Samsung's Galaxy Note devices which are equipped with a Wacom digitizer.

Finally, in cases of video/voice meetings, in order to have proper Acoustic Echo Cancelation (AEC), this should be supported by the device's microphone. The system

can detect if it is supported and use that feature automatically, otherwise it informs the user that she needs to use a headset.

4.2. Architecture, Development Tools and Libraries

All of the code described in the following sections, is available in github [50], under the GNU General Public License (GPL) version 3 [51], which is probably one of the most commonly used licenses for open-source projects. The GPL grants and guarantees a wide range of rights to developers who work on open-source projects. Basically, it allows users to legally copy, distribute and modify software and ensures that users of the software have the following four essentials freedoms: [52]

- The freedom to run the program as you wish, for any purpose (freedom 0).
- The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.
- The freedom to redistribute copies so you can help your neighbor (freedom 2).
- The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

The various layers and components in system's architecture is depicted in Figure 22 and will be described in the following sections.

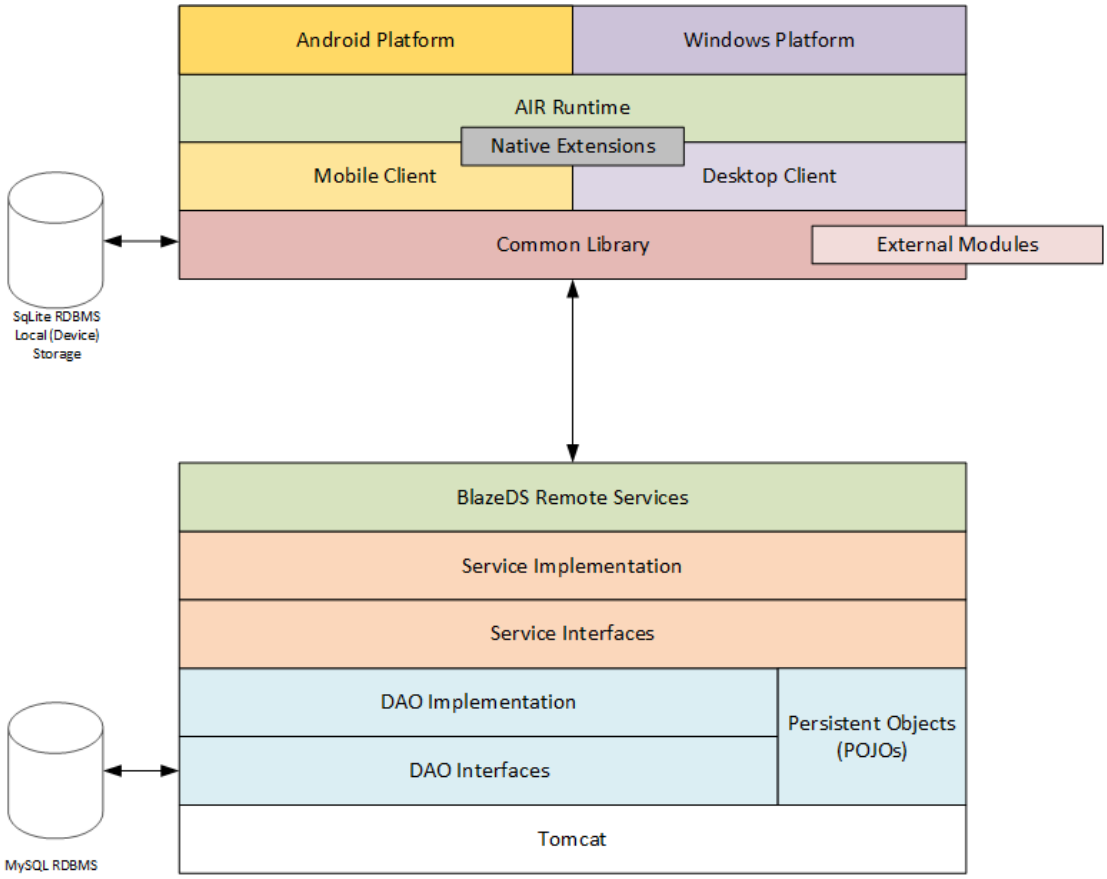


Figure 22: System's architecture

4.2.1. Client Development

4.2.1.1. ActionScript, Flex and AIR

The client side code is developed using ActionScript 3.0 [53], an object-oriented programming language originally developed by Macromedia Inc. (since merged into Adobe Systems), which is used primarily for the development of websites and software targeting the Adobe Flash Player platform, used on Web pages in the form of embedded SWF files.

On top of ActionScript we used Apache Flex framework [54] which is a powerful, open source application framework that allows a developer to easily build mobile applications for iOS, Android, and BlackBerry OS devices, as well as traditional applications for browser and desktop using the same programming model, tools, and codebase.

Finally, on top of these, the client application utilizes the Adobe AIR runtime [55] that enables a developer to use Flash/ActionScript/Flex tools in order to build and deploy standalone applications that can run outside a web browser on both desktops and mobile environments.

4.2.1.2. Client-side MVCS Architecture: The Robotlegs Application Framework

The client-side architecture utilizes the Robotlegs [56], an ActionScript Application Framework for Flash and Flex which offers [57]:

- Dependency injection
- Module management
- Command management
- View management
- Plug-and-play extensions

Robotlegs makes use of three object-oriented architectural patterns [58]:

- *Automated Dependency Injection*. Providing objects with their dependencies (other objects they need to use) instead of the objects creating or fetching their own dependencies.
- *The Command Pattern*. Encapsulating individual pieces of application logic inside dedicated objects instead of spreading all of that logic through a single controller class.
- *The Mediator Pattern*. Using a dedicated object as a mailman/woman to facilitate communication between different objects, instead of those objects talking directly to each other.

4.2.1.3. Event Handling: The ActionScript 3.0 Signals Messaging Tools and the SignalCommandMap

The standard message exchanging mechanism in ActionScript is through Events [59]. In order to overcome the ambiguity that accompany Events and their String registry based approach to the Observer pattern, Robert Penner created the AS3-Signals [60] which are light-weight, strongly-typed AS3 messaging tools, based on the following concepts:

- A Signal is essentially a mini-dispatcher specific to one event, with its own array of listeners.
- A Signal gives an event a concrete membership in a class.
- Listeners subscribe to real objects, not to string-based channels.
- Event string constants are no longer needed.
- Signals are inspired by C# events and signals/slots in Qt.

To facilitate the use of Signals within MVCS it is necessary to use SignalCommandMap utility [61], created by Joel Hook, to allow for Signals to be registered as Command triggers.

4.2.1.4. Adobe AIR Native Extensions

Adobe AIR is by design cross-platform and device-independent. This device-independent design, obviously comes with several limitations in using specific device capabilities and APIs. This void is filled through AIR Native Extensions (ANE) [62], which are native code implementations and provide access to device-specific features, enabling you to use platform-specific features, reuse existing native libraries, and achieve native-level speed for performance-sensitive code, features that are not available in the built-in ActionScript classes, and are not possible to implement in application-specific ActionScript classes.

4.2.2. Server Development

4.2.2.1. Apache Tomcat and Java

The server side is developed in Java using Oracle's JDK 7.0 and Apache Tomcat webserver. Although JDK 7.0 has reached its end-of-life at April of 2015, we stuck with it for various reasons that will become clear in the following sections. However, the source code can be compiled and run under more recent versions of JDK like 8 or 9.

4.2.2.2. Spring Framework

The general server architecture is built around the Spring Framework [63] and uses several modules:

- *Inversion of control container*, which provides a consistent means of configuring and managing Java objects (beans). The lifecycle of these managed object is handled by Spring, can be configured by loading XML files or detecting specific Java annotations on configuration classes and can be automatically injected to other classes that need them.
- *Data access framework*, which addresses common difficulties developers face when working with databases in applications. Support is provided for all popular data access frameworks in Java: JDBC, Hibernate, JPA, etc.

  For all of the supported frameworks, Spring provides these features
  - Resource management: Automatically acquiring and releasing database resources.
  - Exception handling: Translating data access related exception to a Spring data access hierarchy.
  - Transaction participation: Transparent participation in ongoing transactions.
  - Resource unwrapping: Retrieving database objects from connection pool wrappers.
  - Abstraction for BLOB and CLOB handling.
- *Integration framework*, which is a framework for Enterprise application integration that provides reusable functions that are essential in messaging, or event-driven architectures.
- *Security framework*, which is a framework that focuses on providing both authentication and authorization to Java applications. Its real power is found on how easily it can be extended to meet custom requirements.

4.2.2.3. Data Access: Hibernate and MySQL

The system uses a MySQL database for storing its data, and utilizes the Hibernate ORM Framework [64] integrated with the Spring Data access framework, for mapping java objects to the relational database.

The Data Access layer can be broken down into a DAO and a Service layer which contain interface and solid implementation classes for accessing the underlying database storage. These interfaces and classes are implemented using Java's generics programming facility and are managed by the Spring framework, in a way that one could create new services and DAO implementations for handling the serialization and

deserialization of Java objects, by just providing to system the related bean configuration, without the need to write any java code. As an example the following Figures (23 and 24) show the bean definitions for the User Management Subsystem's (user, user's profile and user's role) objects.

```
<bean id="userDAO" class="gr.ictpro.mall.dao.GenericDAOImpl">
    <constructor-arg>
        <value>gr.ictpro.mall.model.User</value>
    </constructor-arg>
    <constructor-arg>
        <value>java.lang.Integer</value>
    </constructor-arg>
</bean>
<bean id="profileDAO" class="gr.ictpro.mall.dao.GenericDAOImpl">
    <constructor-arg>
        <value>gr.ictpro.mall.model.Profile</value>
    </constructor-arg>
    <constructor-arg>
        <value>java.lang.Integer</value>
    </constructor-arg>
</bean>
<bean id="roleDAO" class="gr.ictpro.mall.dao.GenericDAOImpl">
    <constructor-arg>
        <value>gr.ictpro.mall.model.Role</value>
    </constructor-arg>
    <constructor-arg>
        <value>java.lang.Integer</value>
    </constructor-arg>
</bean>
```

Figure 23: User Management Subsystem DAO beans

```
<bean id="userService" class="gr.ictpro.mall.service.UserServiceImpl">
    <property name="dao" ref="userDAO"></property>
</bean>
<bean id="roleService" class="gr.ictpro.mall.service.GenericServiceImpl">
    <property name="dao" ref="roleDAO"></property>
</bean>
<bean id="profileService" class="gr.ictpro.mall.service.GenericServiceImpl">
    <property name="dao" ref="profileDAO"></property>
</bean>
```

Figure 24: User Management Subsystem Services beans

4.2.2.4. BlazeDS and Spring Integration

BlazeDS is a server-based Java remoting and web messaging technology that allows the connection to back-end distributed data and push data to Apache Flex and Adobe Integrated Runtime (AIR) applications [65], by providing the following services:

- The Remoting Service allows your Flex application to directly invoke methods of Java objects deployed in your application server.

- The Message Service provides a publish/subscribe infrastructure that enables your Flex application to publish messages and subscribe to a messaging destination, enabling the development of real-time data push and collaborative applications.
- The Proxy Service allows your Flex application to make cross-domain service requests in a secure and controlled manner. In other words, it allows your Flex application to access a service available on a different domain than the domain from where the application was downloaded (without having to deploy a crossdomain xml policy file on the target domain).

BlazeDS uses Action Message Format (AMF) [66], a binary format to serialize object graphs such as ActionScript objects and XML, or send messages between an Adobe Flash or AIR client.

The format is often used in conjunction with Adobe's RTMP to establish connections and control commands for the delivery of streaming media and it has also the ability to convert ActionScript object into Java object and vice-versa (RemoteClass) in a transparent way to the programmer.

Finally, BlazeDS can be integrated with the rest of the Spring Framework, through Spring BlazeDS Integration [67]. In that way BlazeDS configuration can be managed in "Spring's way" and can take advantage many of the Spring's characteristics, like for example the usage of Spring's Security Framework in order to secure BlazeDS destination services (which can be considered the equivalent of Controllers in the MVC design pattern).

4.2.3. Integrated Desktop Environments

4.2.3.1. Adobe FlashBuilder and Eclipse

Eclipse [68] was the basic IDE used for the system's development. It is the most widely used Java IDE and it contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications.

Through its plug-in system it can be extended to provide support for Flex and ActionScript applications. A plugin for that purpose is included in Adobe's

FlashBuilder IDE [69], which is primarily focused in developing applications using Actionscript language and Flex framework.

4.2.3.2. Microsoft Visual Studio

For the AIR Native Extensions we developed, we used Microsoft Visual Studio [70], in order to implement the required native windows libraries (dll) using C++ and the Windows API.

4.2.3.3. Flash Professional

Finally, Adobe's Flash Professional which recently was renamed to Animate CC [71] can be used in order to implement multiplayer games as external modules that can be loaded into the application.

4.2.3.4. Discussion

As a final note, we should mention that Adobe who has initially created many of the ActionScript related tools and technologies mentioned previously, is in a phase of open sourcing them and donating them to the Apache Software Foundation, like in the case of former Adobe Flex which is now Apache Flex.

We suspect that Adobe isn't any more assigning resources to maintain these tools which is obvious in the case of the FlashBuilder IDE, which its plugin is compatible with a really old and outdated version of eclipse (Indigo).

The Indigo version was initially released back in 2011 and it is compatible with JDK versions up to 7 and this is the major reason we chose to go along with JDK 7 through this Thesis.

4.2.4. Integration of Third-party Software

4.2.4.1. BigBlueButton Server

As already mentioned, the synchronous e-learning and collaboration activities are backed by the BigBlueButton Server. During the development phase we used the unmodified BigBlueButton Server version 0.9.1. build 268 which was installed in a dedicated linux server running Ubuntu 14.04 server operating system.

4.2.4.2. Red5 Multimedia Server

A key component in BigBlueButton is the Red5 Media Server [72], an open source media server for live streaming solutions of all kinds. It is designed to be flexible with a simple plugin architecture that allows for customization of virtually any VOD and live streaming scenario.

Apart from its internal use by the BigBlueButton software, we use the same Red5 server in order to implement the synchronization between clients of the Shared Boards in the case of the multiplayer games, as it will be shown in the next chapter were we describe the extensibility of the system.

4.2.4.3. Moodle Server

The Moodle software is responsible for delivering the asynchronous e-learning and activities functionality. It is written in php and it runs through a web server. During the development phase it was installed in a separate virtual machine which was running in the same physical machine in which BigBlueButton was installed.

We use the standard Moodle software (version 3.0) with the addition of a patch we created in order to handle external course creation per our needs. This patch is already submitted to the upstream code of moodle as per the ticket MDL-54792 [73] and it will eventually be included in the official code of Moodle.

Apart of the above patch, we needed to create the backend for handling the authentication of the users using the one-time password as described in Section 3.5.3.2 (Sequence Diagram: Visiting a Course's Page) which is also included in the Thesis' repository in github [50].

4.2.4.4. MConf Client

Lastly, we should mention the work of Roesler et al. [74] who apart from extending the BigBlueButton Server and adding additional functionality and features to it. They created an ActionScript based client for mobile devices, bbb-air-client [75] which is integrated into our client, and is heavily modified for the requirements of our Thesis.

4.3. Context Awareness

The important aspects of context are as follows: where you are, whom you are with, and what resources are nearby. Context to be the constantly changing execution environment and includes the following elements [76]:

- *Computing environment*: available processors, devices accessible for user input and display, network capacity, connectivity, and costs of computing
- *User environment*: location, collection of nearby people, and social situation
- *Physical environment*: lighting and noise level

In this Thesis, we use the first two types of context (Computing Environment and User Environment) as described in the following sections.

4.3.1. Computing Environment

As we needed to have a common code base for all the type of targeted devices (Desktop PCs, Laptops, Tablets and Smart Phones), we needed to introduce several classes and interfaces in order to be able to adapt the client application to each device, in terms of screen sizes, DPI values and type of interaction (mouse vs. touch).

Although, the Flex Framework already includes functionality like dynamic layouts and skins [77], most of this functionality is a compile time option and it cannot be modified in runtime. For example UI control skinning and interaction type (mouse versus touch) is applied in the compile time depending on the type of application that is compiled (desktop versus mobile). Obviously this compile time approach wouldn't work in case of external modules, which need to adapt their skins, dpi and interaction in runtime.

In order to be able to handle these issues in runtime, we created in the Common Library the IDevice interface which contain definitions for getting the correct skins. Each client (desktop and mobile) contain a solid implementation of this interface, which attaches the relevant skins.

The Device class implementation also contain methods for scaling controls and fonts in runtime according the device's type and its screen's DPI. We finally extended all the UI controls we used in the application and override the relevant methods, in order to use the functionality provided by the Device class implementation and scale properly on every device and also in order to set the correct interaction type.

Usage of these extended controls are straightforward, as they behave the same way as the standard controls. When creating a new mxml component through FlashBuilder, one needs just to modify the automatically imported namespace

```
xmlns:s="library://ns.adobe.com/flex/spark"
```

to point to the package that contain our own extended UI controls

```
xmlns:s="gr.ictpro.mall.client.components.*"
```

4.3.2. User Environment

4.3.2.1. User's Location

Regarding the user's environment context, as mentioned in section 3.5.1.2 (Use Case: Registration / Authentication), we wanted to provide a convenient method for student registration and authentication in cases a teacher is close to the student (i.e. they are both located in the same physical classroom), or in cases that a student connects from a known location from which she had logged in again in the past (e.g. the student's school or the student's home).

In order to accomplish this, we followed a procedure similar to the work of Nguyen et al. [46] who used the nearby radio (Wi-Fi) signals received by a user's mobile device, in order to determine if that user is in proximity with other users, without detecting their exact location.

The system first of all gathers information of each SSID detected and its strength (RSSI) and sends them to the server. The server, having these SSID, RSSI sets for two users in question, it creates a set of all SSIDs, and quantizes the RSSI values into 5 distinct values based on their strength:

- Perfect (having a value of 1.0) for RSSI >= -55dB
- Strong (having a value of 0.9) for -55dB > RSSI >= -70dB
- Average (having a value of 0.8) for -70dB > RSSI >= -85dB
- Weak (having a value of 0.7) for -85dB > RSSI >= -100dB
- Out of Reach (having a value of 0.0) for SSIDs that are out of reach for one user but is in reach for the other

Having these transformed sets of SSID, RSSI pairs, which have the same dimensions for both clients, the server calculates the Euclidean distance between the two sets, and normalizes it by dividing it with the total number of dimension. Having that normalized distance, we can tell if the two users are in proximity based on a threshold value (smaller than the threshold means that the user are in proximity).

In order to validate the method, we collected about 100 sets of SSID/RSSI pairs using 3 different devices (a Microsoft Surface Pro 3 windows based tablet, an ASUS TF701T android based tablet and a Samsung Galaxy Note 3 android based smart phone) in two different buildings: (a) a 1 bedroom apartment of about 650 ft^2 (60 m^2) and (b) a large office of about 2700 ft^2 (250 m^2) and also in locations outside these buildings. Our experiment showed that having a threshold value of 0.5, the system can detect that two devices (and thus their users) were located inside the small apartment or outside it in a distance less than 300 ft (100 m). In case of the large office, in most cases, for the same threshold value, the system could detect if the two devices were located in the same room. The cases that it failed to do so was when comparing SSID, RSSI sets that where recorded in adjacent rooms and close to the separating wall.

In order to compensate for this fuzziness in case of large buildings, in the student's registration process, the system asks the student to provide his teacher's name. Having that additional information, the proximity detection can be improved as the system only calculates it for the user and teacher in question and not for all teachers that might be located in adjacent classrooms.

In the case of student registration, having this proximity information, the system automatically enables the student's account and also it assigns her to the correct classroom, based on the teacher's current classroom, actions which otherwise should be performed manually by the teacher for each of her students.

4.3.2.2. User's Language

As mentioned in section 3.2.4 (User and Device Context Awareness), the system should be able to adapt its messages to the user's language. In order to do so, the client application queries the user's device for its language and sets it as the application's language, assuming that a user's device always uses a language that the user can understand. Of course, if desired, this default language can be overridden by the user, or in case of a student, by her current classroom language, if defined by her teacher.

4.3.2.3. Time

Finally, assuming that a teacher has created a schedule for her classrooms, the system can detect the current teacher's classroom (if the time and date falls inside a classrooms schedule) and can propagate that information to all students that are assigned in that classroom. Again, the teacher can override this system's decision and set her current classroom manually, having that information again propagated to the students.

Knowing the current classroom, the client application can adjust the language of the various messages, as described in the previous section. Also this information can be used from any other external module that is loaded into the application.

4.4. Summary and Discussion

To summarize, the implemented system covers all the characteristics that were captured in the analysis phase. That is, it supports online communications between students and teachers (text and video chat), the ability for online teaching (whiteboard in combination with text and video chats) using BigBlueButton and also online multiplayer educational games through shared boards in combination again with BigBlueButton's support for text and video chats. Obviously, using the mobile client, these activities and interactions between students can take place in the student's surrounding environment outside a physical classroom.

Furthermore, moodle's integration provides the ability of educational sharing (courses, notes, assignments etc.) as well as the creation and delivery of online tests and quizzes. In addition to these characteristics, all course types and formats available in moodle can be utilized as needed.

Regarding using user's context information (age, sex, cultural background, etc.) in order to adapt the system, we limited these just translating the UI to the user's language as described in section 4.3.2.2. (User's Language). Requiring more information by the user's (like for example the student's age or sex) we would be able to further adjust the UI to the user (e.g. relying more on visual elements like graphics and color and less in textual representation elements, etc.). We chose to limit the capturing and usage of user

context information due to security and privacy concerns that were raised by the teachers during the analysis phase.

Furthermore, as described in section 3.4. (Additional Requirements: User Interactions), the system was finally designed so that it would be controlled by the teachers, which primarily means that students cannot interact directly with each other if a teacher isn't aware about this. With this controlled/structured design, some characteristics like for example the parents' ability to impose restrictions on the system's usage by their children, become meaningless as we are talking about a system controlled by teachers and not a system that resembles common social networks. Of course parents still have the ability to review their children's calendars (i.e. scheduled and past online meetings), to restrict their children from joining unsupervised meeting etc.

5. Extending the System

A key point in this Thesis is the extensibility of the system from 3rd party developers or instructional designers, in an easy way without the need to compile and redeploy the whole client application.

In the next sections we focus on extending the client's functionality, providing details for extending the server-side only when required, as it is a typical java web application based on the most commonly used frameworks (Spring and Hibernate).

5.1. Creating New Modules

In general one can create a new Flex Project in FlashBuilder and inside this project to define different independent modules. When compiling, FlashBuilder creates a new swf file for each of these modules and these swf can be deployed in the tomcat server running the Management Server.

In the Thesis' source code repository in github [50] we have included an example of a simple "Hello World" module in project named ExternalModuleExample. This module registers a new menu item ("Example"), which when clicked it just opens a simple form having a button. Clicking this button displays a "Hello World" informational message.

5.1.1. Module Initialization

When loading the module in runtime, the system first of all looks for the Initialize class, which should extend the Module class. In order to map any components into robotlegs' context you need first to get a reference of mediator map and injector and then use these references to add any dependencies. In the provided example project we map the ExampleView to ExampleViewMediator and also we register a new menu item during the module initialization as shown in Figure 25.

In order for an external module to be loaded by the system, its swf file should be deployed in tomcat server under the CommonModules path and its url should be added in the configuration file named external-modules.xml located in resources/spring folder.

```
package com.example.mall.external
{
    import gr.ictpro.mall.client.components.Module;
    import gr.ictpro.mall.client.components.TopBarView;

    import org.robotlegs.core.IInjector;
    import org.robotlegs.core.IMediatorMap;

    public class Initialize extends Module
    {
        public function Initialize()
        {
            super();
        }

        [Inject]
        public function set mediatorMap(mediatorMap:IMediatorMap):void {
            mediatorMap.mapView(ExampleView, ExampleViewMediator, TopBarView);
        }

        [Inject]
        public function set injector(injector:IInjector):void {
            var addonMenu:AddonMenu = new AddonMenu();
            injector.injectInto(addonMenu);
            addonMenu.registerAddonMenu();

        }

    }
}
```

Figure 25: Initialize Class for an external Module.

5.1.2. Views and Mediators

There are three types of views and their mediators accordingly, which can be extended, in order to achieve the required functionality.

- TobBarCustomView and TopBarCustomViewMediator can be used to create any kind of form which is not bind to persistent objects.
- TopBarListView and TopBarListViewMediator can be used to provide a list of persistent objects.
- TopBarDetailView and TopBarDetailViewMediator can be used to display the properties of any persistent object.

5.1.3. Creating Persistent Objects

Usually a typical Actionscript persistent object is just a set of name-value properties. The application support either client side (objects that can be stored in a SQLite

database in the client device), or server side (objects that can be stored in the database in the server) persistent objects.

In order to automatically glue CRUD (Create, Retrieve, Update, Delete) operations with the GUI Components (TopBarListView/TopBarListViewMediator and TopBarDetail View/TopBarDetailViewMediator), we need to

- Create the appropriate editor component for mapping the object's properties to GUI controls.
- Create the appropriate model class which extends the AbstractModel and implements either the IServerPersistent (for server-side persistent objects) or IClientPersistent (for client-side persistent objects) interface.
- Create the appropriate list/detail views (based on TopBarList View/TopBarListViewMediator and TopBarDetailView/TopBarDetailView Mediator classes).

And the system will glue these objects together. Usually for the list view/mediator, we don't need to customize any part of the code. On the other hand, for the detail view/mediator, obviously we may need to include any logic related to the objects, like validation before updating or deleting. In order to do so, the TopBarDetailView Mediator class provides a set of methods that can be overridden, like validateSave, beforeSaveHandler and beforeDeleteHandler.

5.1.4. Initiating CRUD Operations and Calling Server-Side Methods

The system provides a set of signals that can be used in order to perform any CRUD operation on either server or client side persistent object or in general to call any server-side method. These signals come in pairs of three (XXXXSignal, XXXXSuccessSignal and XXXXErrorSignal) as follows:

- ListSignal, ListSuccessSignal and ListErrorSignal can be used to retrieve a list of any type of persistent object.
- SaveSignal, SaveSuccessSignal and SaveErrorSignal can be used to save any type of persistent object.
- DeleteSignal, DeleteSuccessSignal and DeleteErrorSignal can be used to delete any type of persistent object.

- GenericCallSignal, GenericCallSuccessSignal and GenericCallErrorSignal can be used to call any server-side method which can return a value or not.

The XXXXSignal is the signal dispatched in order to initiate the call and XXXXSuccessSignal and XXXXErrorSignal signal are dispatched by the system on a successful or failed call accordingly.

Usually the signals related to CRUD operations are dispatched and handled automatically by the system, however, the list related signals should be used in order to get any detail properties (for example in order to provide values for a ComboBox component).

5.2. Registration and Authentication Methods

The system can support multiple authentication and registrations methods, backed by the Spring Security Framework. Besides the standard username/password authentication and registration the system provides also the proximity based authentication and registration methods as described in sections 3.5.1.2. (Use Case: Registration / Authentication) and 4.3.2.1. (User's Location).

The general principle is that the system can use multiple methods in the same time going through each of them and always ending as a fall back to the standard username/password method, in case the previous ones failed. For example having both proximity based and standard authentication methods enabled, the proximity based will be executed first, and it will fall back to the standard method in case it cannot get any contextual information (i.e. when the PC isn't equipped with a Wi-Fi card and is connected to the network through an Ethernet card).

5.2.1. Client Side Module

The client code for these methods are included in the project named CommonModules. The modules' initialization logic is the same as the one described in section 5.1.1. (Module Initialization). There should be an Initialize class extending the base Module class which handles any dependency injection and view/mediators mapping. The component implementing the view part can be any MXML container like the Group container.

The system again provides three pairs signals for authenticating or registering a user, LoginSignal, LoginSuccessSignal, LoginFailedSignal and RegisterSignal, RegisterSuccessSignal, RegisterFailedSignal accordingly. The module should dispatch the LoginSignal or RegisterSignal when it has the required information in order for the system to perform the login or register operation, accordingly, and, if required, it should listen for the other four signals dispatched by the system.

5.2.2. Server Side Implementation

As mentioned, the server side implementation is backed by Spring's Security Framework. First of all, we need to have an authentication token class (like the UsernameLocationAuthenticationToken used in proximity based authentication and registration module) which should extend Spring's AbstractAuthenticationToken, based on the credentials needed by the module.

We need also to provide a solid class implementation for the registration and authentication providers, which should inherit the AbstractRegistrationProvider and AbstractAuthenticationProvider accordingly and which handle the actual registration and authentication logic based on the supplied credentials in a way similar to ProximityRegistrationProvider and ProximityAuthenticationProvider implementations that can be found in the MServer project's source code.

Finally the registration and authentication module swf files should again be deployed in tomcat server under the CommonModules path and there should be an entry in files registration-providers.xml and authentication-providers.xml accordingly, located in resources/spring folder. Apart from their url, we need also to provide the method's priority and the client's class name.

5.3. Shared Boards

For the online meetings, the whiteboard module that is provided by the BigBlueButton can be replaced with other shared boards (boards that their state is shared by all users participating in the meeting), giving the ability to a developer or instructional designer to implement multiplayer games designed in Adobe's Animate CC (former Flash Professional) application. The system's source code, in the project named Potato, provides such an implementation of the KDE's educational game Ktuberling.

The first step for creating a shared board, is to create a normal ActionScript 3.0 project in Adobe's Animate CC application. After the implementation of this standalone application, in order to be able to act as a shared board, we need to add the CommonClientLibrary swc files to the project's library path with its Default linkage set to "Runtime Shared Library RSL" (Figure 26).

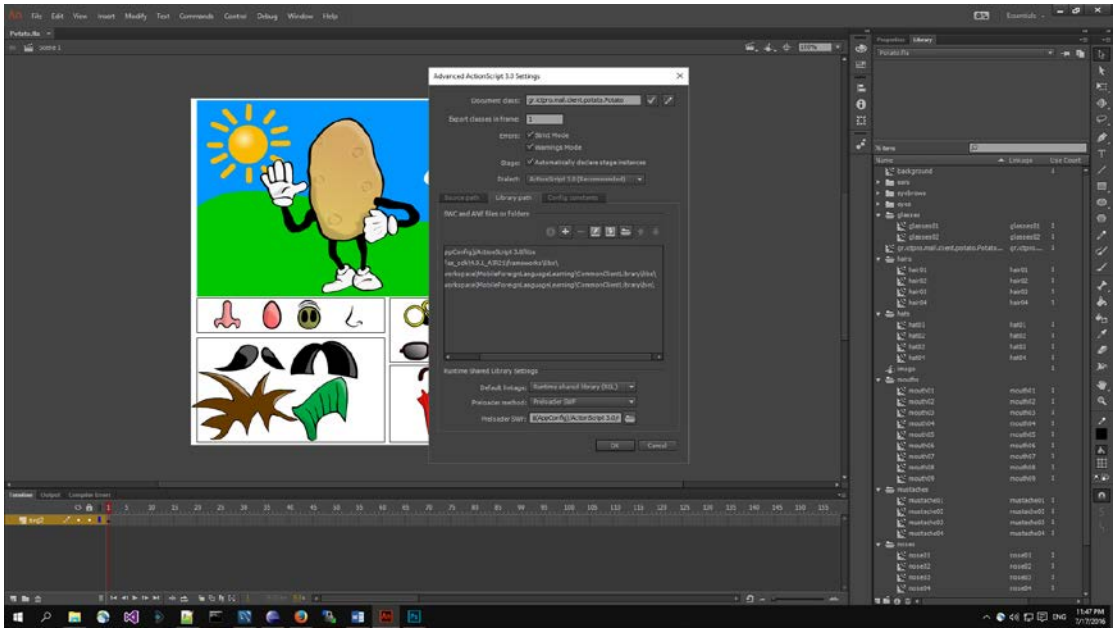


Figure 26: Linking the CommonClientLibrary's swc files in Adobe Animate CC

Objects and actionscript code that are included in the main project's scene should be converted to a single MovieClip symbol and it should be exported for actionscript as shown in Figure 27.

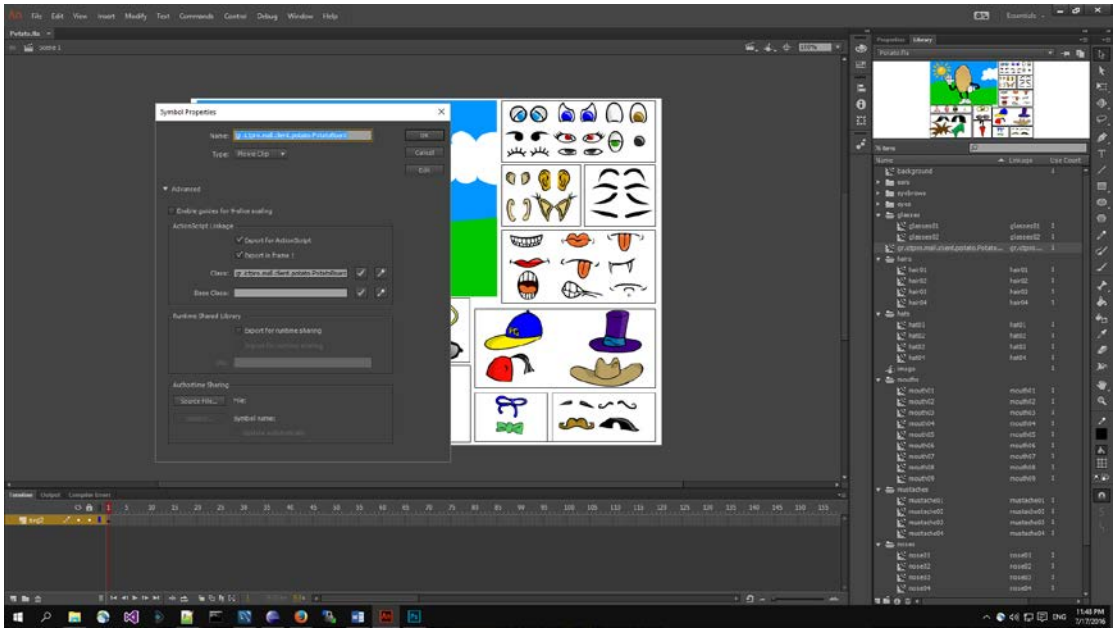


Figure 27: Exporting the Shared Board's MovieClip symbol to actionscript

The MovieClip's symbol class should extend the SharedBoard class and override the boardUpdated function. This function is called by the system each time an object in the board is changed from another user, and it should contain the code required to update the local copy of this object. In order to inform the other users of a local change to an object, we need to call the updateBoard function and provide as a parameter an actionscript object with the changed properties and any other information we need in order to mirror these changes to the other running copies of the shared board. The Potato flash project provides a good example of for propagating object creations, deletions and movements.

The flash project should also an implementation of the main view's mediator which should extend the SharedBoardMediator and typically should not require any code apart from overriding the onRegister function as shown in Figure 28.

```
package gr.ictpro.mall.client.potato {
    import gr.ictpro.mall.client.view.SharedBoardMediator;


    public class PotatoBoardMediator extends SharedBoardMediator {

        override public function onRegister():void
        {
            super.onRegister();
        }

    }

}
```

Figure 28: Shared Board's Mediator

As, in the previous cases, all the pieces of code should be glued together by an Initialize class, which in this case doesn't need to extend any other class (Figure 29).

```
package gr.ictpro.mall.client.potato
{
    import org.robotlegs.core.IInjector;
    import org.robotlegs.core.IMediatorMap;
    import gr.ictpro.mall.client.components.SharedBoard;

    public class Initialize
    {
        public function Initialize()
        {
            super();
        }

        [Inject]
        public function set mediatorMap(mediatorMap:IMediatorMap):void {
            mediatorMap.mapView(PotatoBoard, PotatoBoardMediator, SharedBoard);
        }

    }
}
```

Figure 29: Shared Board's Initialization Class

Finally, after publishing the project in Adobe's Animate CC application, the produced swf file should again be deployed in tomcat server under the CommonModules path and a new record should be inserted in the table named meeting_type Manmagement Server's database, which provide information about the shared board, like its name, its deployed path and the client's class name.

5.4. Discussion and Limitations

Using the above techniques, it is clear that a 3rd party developer or instructions designer can easily extend the system with additional functionality, without the need to redeploy the client, which in our view is an important capability, given the nature of the client application. Especially in cases of android based tablet or mobile devices, a redeploy of the client would require among others a google play developer's account, in order to easily redistribute the application to clients. Also incorporating and keeping up to date different extension modules from different developers in the system, would require a full source code compilation, after merging these extension modules with the rest of code, which would be an unnecessary complex and error prone process.

On the other hand, one could argue that the biggest negative of our approach, is that eventually there is a performance loss in runtime due to two main reasons:

- The time required to download and these external swf files and load them into the main system.
- Having non-optimized by the compiler applications for specific devices (android and windows), as described in section 4.3.1. (Computing Environment).

In our view, the benefits of our approach outweigh this performance loss given the fact that modern desktop and mobile devices, utilize fast multicore CPUs. Also, given the relative small size of the swf files (which typically we wouldn't expect them to exceed few megabytes), a 4G cellular network or an IEEE 802.11-based wireless network should suffice.

Finally, we should note that currently it isn't possible to load external native extensions as these are complex units containing both native and actionscript code which needs to be present during the compile time of the application.

6. Evaluation

In order to perform the evaluation of the implemented system with real users. The ideal evaluation scenario would involve at least two foreign language teachers teaching the same language to students at relatively close ages. Another requirement of course, would be to find a suitable time and date for all teachers and students willing to participate in the evaluation.

As a first step, we created a prescreening survey, which was addressed to teachers and would help us to capture demographic and availability information. The survey, along with the application's usage scenarios presented in section 3.6. (Application Usage Scenarios), was circulated again to related mailing lists and groups in social media (mainly linkedin and facebook) during the last week of May, 2016.

6.1. Questionnaire

The questionnaire included three sections as follows.

6.1.1. Demographics

In the first section we collected demographic data information from the participants. The questions asked where about their country, sex, age, occupation, the language(s) they teach, the total number of years they teach and finally, their students' demographics (number of students broken down per age group and sex).

6.1.2. Available Technology

Next, we tried to capture the number and type of devices owned either by the teacher or by the students, which could be used during the evaluation. In particular we were interested to the following types of devices

(a) Windows based Desktop PCs with both microphone and camera.

(b) Windows based Desktop PCs with microphone only.

(c) Windows based Laptop PCs with microphone and camera.

(d) Windows based Laptop PCs with microphone only.

(e) Windows based Tablets or 2-in-1 (eg Microsoft Surface).

(f) Recent Android Tablets (3 years old or newer).

(g) Recent Android Phones (3 years old or newer).

Based on the numbers of the devices entered in each of the above categories, we optionally asked the responders if they could provide us with the models for the following categories of devices:

(a) Windows based Tablets.

(b) Android based Tablets.

(c) Android based Smartphones.

6.1.3. Date and Time Availability

Finally, in order to capture the date and time availability of the responders, we asked initially asked them to provide their time zone. Next as a multiple choice we asked them to provide the most suitable for them week of either June or July and also, again as multiple choice, the most suitable for them days of the week and time of date. Finally, we asked them optionally to inform us about any comments or additional restrictions that they may have.

6.2. Feedback and Discussion

Based on the web server analytics, both the survey's page and the application's scenarios page, attracted a lot of visitors (about 320 visits for the application's scenarios page and 130 visits for the survey's page including robots).

Unfortunately, we were unable to find volunteer teachers as we got only six incomplete responses: visitors who completed the first pages (demographics and available technology) of the survey and didn't proceed with the rest of the pages.

We received some feedback in the form of email messages from three teachers who expressed their concerns, for participating in the evaluation, related to the security and the privacy of the application, given that it would involve the installation in students' smartphones or tablets. One of them asked in particular if we could avoid the mobile devices and use only his school's lab PCs. Also one teacher commented that the time of the evaluation (June or July) would be convenient for him as it was the end of the academic year.

We believe that the security and privacy and also the bad timing of the evaluation issues raised by these teachers, may be the main reasons that we were unable to find volunteer teachers.

Furthermore, we believe that another key reason would be the complexity of the evaluation procedure, which would require the online real-time cooperation of teachers and students located in different time zones.

7. Conclusion and Future Research

In this Thesis, we presented an e-learning collaborative system, for both mobile and desktop environments. The system can be used in addition to the traditional foreign language teaching methodologies, and extend the physical learning environment outside the classroom, by offering a broad set of both synchronous and asynchronous collaboration activities. These activities can be used by the students in order to interact and communicate with other children from all over the world learning the same language

In our preliminary survey, most of the responders confirmed the usefulness of the system, by agreeing that both intercultural communicative competence and telecollaboration are important elements in foreign language teaching. Also, the responders found the proposed system's characteristics important in foreign language teaching.

Finally, we need to emphasize that, given the system's extensibility and ability to load and integrate external modules, it could serve a testbed for other researchers who need to explore fields related to active learning, collaborative learning activities and online multiplayer educational games. Researchers could be released from the burden of implementing low-level technical details like security, communications, video and audio streaming, etc. and can exclusively focus on their primary research field.

7.1. Future Research and Issues to be Addressed

As noted, a major drawback of our Thesis, is our inability to perform an evaluation of the developed system, in real world conditions and using real users. Such an evaluation would provide us valuable feedback and highlight areas that may need improvement. As trying to recruit volunteer teachers and students online didn't seem to work, we need to follow different tactics. Such tactics would exploit more direct methods for reaching potential volunteers, such as phone calls and in general person-to-person direct communications, compared to the tactic of reaching them through online social media groups or related mailing lists. Of course, financial incentives could also play a key role in recruiting volunteers, although this wasn't considered as an option in the current Thesis.

Besides that, during the development process, we marked some areas that could be improved, as follows:

- Implementation of versioning and local caching of external modules. Such feature could improve the loading time of the application, as it would only need to query the server only for version numbers of any external module, compare this numbers with the local cached copies and download from the server only the updated modules.
- There is a known limitation in Adobe's AIR/Flex Runtime on devices that can change the screen orientation, such as mobile phones and tablets. In such devices a Video object attached to the camera will only show upright video in a landscape-aspect orientation and thus, mobile apps should use a landscape orientation when displaying video and should not auto-rotate.

  We believe that this is an important flaw that needs to be addressed. Addressing it in the client by transforming the video before streaming it to the server doesn't seem to be efficient enough as based on our tests it produced a lot of delays especially in slower devices. An alternative approach that needs to be investigated is the possibility to have the server transform the raw video before streaming it to other clients, based on additional camera and device orientation metadata sent by the client.
- As mentioned, devices equipped with a digitizer, such as Microsoft Surface and Surface Pro series or Samsung's Galaxy Note series, can highly improve the user experience while interacting with BigBlueButton's whiteboard and probably any other shared Board. A problem that could be arise in such cases, is in cases that the user rests her palm in the device's screen while using the stylus, having the application record multiple touch events.

  Currently both Android and Windows operating systems do not offer a solution to this issue via some global API. Although they offer the required APIs for the digitizing device (e.g. Pen API in Microsoft Windows and S-Pen SDK in Samsung Galaxy Note), palm rejection algorithms seem to be implemented in an application based level when required (e.g. Microsoft OneNote or Samsung SNote). Obviously in order to use these APIs we need to implement the related Native Extensions to detect the presence of such digitizers and expose the required functionality to ActionScript.

Furthermore, during this Thesis we implemented the required functionality for using in a mobile device the BigBlueButton's whiteboard in presenter's mode, we find it important to contribute that back to the BigBlueButton project. Although this is a trivial task, we postponed it for the next months. The reason behind this decision to postpone it, is based on the fact that we heavily modified the BigBlueButton's mobile client to fit our needs, so the additional implemented feature (presenter mode in whiteboard) need to backported to the upstream code, a task that mainly requires some amount of work in the presenter's controls. These controls which are mainly buttons, need to be modified so that they can be styled through CSS and bitmap graphics instead of the skinning approach we followed in this Thesis.

Finally, in addition to these improvements, we believe that not targeting iOS (i.e. iphone and ipad) and OSX based devices is an important drawback. Although, Adobe's AIR Runtime Environment is available for these devices and also the FLEX compiler can produce binaries targeting these devices, this possibility wasn't explored in this Thesis as we didn't have access to such devices. In general, porting the client application to iOS and OSX platforms would be a relatively straightforward procedure which would involve the creation of the related project and support classes as described in section 4.3.1. (Computing Environment) and also implement the native part for the AIR Native Extensions used in the application.

8. References

[1] Revans, R. W. (1971). *Developing effective managers: A new approach to business education*. Praeger Publishers.

[2] Bonwell, C. C., & Eison, J. A. (1991). *Active Learning: Creating Excitement in the Classroom. 1991 ASHE-ERIC Higher Education Reports*. ERIC Clearinghouse on Higher Education, The George Washington University, One Dupont Circle, Suite 630, Washington, DC 20036-1183.

[3] Dillenbourg, P. (1999). *Collaborative Learning: Cognitive and Computational Approaches. Advances in Learning and Instruction Series*. Elsevier Science, Inc., PO Box 945, Madison Square Station, New York, NY 10160-0757

[4] Johnson, D. W., Johnson, R. T., & Smith, K. A. (1989). *Cooperative learning*. Interaction Book Company.

[5] Marcos Miguel, N., Fernández Chaves, I., Jara Maroto, C., López Lago, M.Á., Peters, M.J., Díaz Garrido, A.I., Calzas Hernández, M. and Rivas Reyes, M. (2015*). Teachers' perspectives on flipping the language class: More than creating materials*.

[6] McKeachie, W.J., Svinicki,M. (2006). *Teaching Tips: Strategies, Research, and Theory for College and University Teachers*. Wadsworth Publishing, Belmont, CA.

[7] Bens, I. (2005). *Understanding participation. In Facilitating with ease! Core skills for facilitators, team leaders and members, managers, consultants, and trainers* (2nd ed., pp. 69–77). Jossey Bass, San Francisco.

[8] de Weck, O. L., Kim, I. Y., & Hassan, R. (2005, June). Active learning games. *In Proceedings of the 1st Anual CDIO Conference, Ontario, CA, June 7* (Vol. 8).

[9] Lantolf, J. P. (1994). Sociocultural theory and second language learning: Introduction to the special issue. *The Modern Language Journal*, 78(4), 418-420.

[10] Tabbers, H. K., & de Koeijer, B. (2010). Learner control in animated multimedia instructions. *Instructional Science, 38(5)*, 441-453.

[11] Agbatogun, A. O. (2014). Developing learners' second language communicative competence through active learning: clickers or communicative approach?. *Educational Technology & Society, 17(2)*, 257-269.

[12] Cunningham, D. (1998). 25 years of technology in language teaching: A personal experience. *Babel: journal of the Australian Federation of Modern Language Teacher/Associations, 33*(1), 4-7, 35.

[13] Iandoli, L. J. (1990). CALL and the profession: The current state. *French Review, 64*(2), 261-272.

[14] Frommer, J. (1989). MacLang [Computer software]. Cambridge, MA: Author.

[15] Pusack, J.P., Otto, S. K. (1990). Applying instructional technologies. *Foreign Language Annals, 23*(5), 409-417.

[16] Liu, M., Moore, Z., Graham, L., Lee S. (2002) A Look at the Research on Computer-Based Technology Use in Second Language Learning, *Journal of Research on Technology in Education*, *34:3*, 250-273

[17] Hellebrandt, J. (1999). Virtual collaborations in the Spanish class: From email to Web design and CD-ROM development. *Journal of Educational Computing Research, 20*(1), 59-70.

[18] Kroonenberg, N. (1994-1995). Developing communicative and thinking skills via electronic mail. *TESOL Journal, 4*(2), 24-27.

[19] Kukulska-Hulme, A. (2005). *Mobile learning: A handbook for educators and trainers*. New York, NY 10016: Routledge.

[20] Kukulska-Hulme, A. (2009). Will mobile learning change language learning? *ReCALL, 21*(02), 157-165.

[21] Gabarre, C., Gabarre, S., Din, R., Shah, P. M., & Karim, A. A. (2014). iPads in the foreign language classroom: A learner's perspective. *Southeast Asian Journal of English Language Studies, 20*(1), 115-127.

[22] Godwin-Jones, R. (2011). Emerging technologies: Mobile apps for language learning. *Language Learning & Technology, 15*(2), 2-11.

[23] Miangah, T. M., & Nezarat, A. (2012). Mobile-assisted language learning. *International Journal of Distributed and Parallel Systems, 3*(1), 309-319.

[24] Ogata, H., & Yano, Y. (2003). How ubiquitous computing can support language learning. *Proceedings of KEST*, 1-6.

[25] Palalas, A. (2011). Mobile-assisted language learning: Designing for your students. In S. Thouësny & L. Bradley (Eds.), Second language teaching and learning with technology: Views of emergent researchers. Dublin, Ireland: Research-publishing.net.

[26] Lan, Y.-J., Sung, Y.-T., Chang, K.-E. (2007). A mobile-device-supported peer-assisted learning system for collaborative early EFL reading. *Language Learning & Technology*, *11*(3), 130-151.

[27] Park, Y. (2011). A pedagogical framework for mobile learning: Categorizing educational applications of mobile technologies into four types. *The International Review of Research in Open and Distributed Learning*, *12*(2), 78-102.

[28] Karvounidis, T., Chimos, K., Bersimis, S., Douligeris, C. (2012, April). An integrated self-evaluated framework for embedding Web 2.0 technologies in the educational process. *In Global Engineering Education Conference (EDUCON), 2012 IEEE* (pp. 1-7). IEEE.

[29] Chimos, K., Douligeris, C., Karvounidis, T., Basios, M., Bersimis, S. (2013, March). Unisuite: An innovative integrated suite for delivering synchronous and asynchronous online education. *In Global Engineering Education Conference (EDUCON), 2013 IEEE* (pp. 400-404). IEEE.

[30] Bello, O. W., Ehira, D. N., Balogun, N. A., Ayeni, J. K., Faruk, N. (2015). Synchronous Virtual Learning System: An Implementation. *4th iSTEAMS Research Nexus 2015 Vol. 4*. Ilorin, Nigeria.

[31] Wang, C. X., Jaeger, D., Liu, J., Guo, X., Xie, N. (2013). Using synchronous technology to enrich student learning. *TechTrends*, *57*(1), 20-25.

[32] Eurobarometer, S. 386. 2012. Europeans and their languages. *European Commission*.

[33] EuroCALL: European Association for Computer-Assisted Language Learning, http://www.eurocall-languages.org/, Retrieved May 30, 2016.

[34] ReCALL: The Journal of the European Association for Computer Assisted Language Learning. http://journals.cambridge.org/action/displayJournal?jid=REC, Retrieved May 30, 2016.

[35] TILA Project: Telecollaboration for intercultural language acquisition. http://tilaproject.eu/, Retrieved May 30, 2016.

[36] Jauregi, K., Melchor-Couto, S., Vilar Beltrán, E. (2013). The European Project TILA. L. *Bradley & S. Thouësny (Eds.)*, 20, 123-128.

[37] Agile Manifesto, http://agilemanifesto.org/, Retrieved June 7, 2016.

[38] Forbrig, P., & Herczeg, M. (2015, September). Managing the Agile Process of Human-Centred Design and Software Development. In *INTERACT* (pp. 223-232).

[39] Deemer, P., Benefield, G., Larman, C., Vodde, B. (2010). The scrum primer. Scrum Primer is an in-depth introduction to the theory and practice of Scrum, albeit primarily from a software development perspective, available at: http://assets.scrumtraininginstitute.com/downloads/1/scrumprimer121.pdf, 1285931497, 15.

[40] DIS, I. (2009). 9241-210: 2010. Ergonomics of human system interaction-Part 210: Human-centred design for interactive systems. *International Standardization Organization (ISO)*. Switzerland.

[41] Paelke, V., Nebe, K. (2008, February). Integrating agile methods for mixed reality design space exploration. In *Proceedings of the 7th ACM conference on Designing interactive systems* (pp. 240-249). ACM.

[42] Paul, M. (2015). *Systemgestützte Integration des Usability-Engineerings in den Software-Entwicklungsprozess* (Doctoral dissertation, PhD Thesis, Univerity of Lübeck).

[43] BigBlueButton. http://bigbluebutton.org/ , Retrieved June 7, 2016.

[44] Moodle - Open-source learning platform. https://moodle.org/, Retrieved June 7, 2016.

[45] Varshavsky, A., Scannell, A., LaMarca, A., & De Lara, E. (2007). *Amigo: Proximity-based authentication of mobile devices* (pp. 253-270). Springer Berlin Heidelberg.

[46] Nguyen, K. A., Luo, Z., & Watkins, C. (2015). On the Feasibility of Using Two Mobile Phones and WLAN Signal to Detect Co-Location of Two Users for Epidemic Prediction. In *Progress in Location-Based Services 2014* (pp. 63-78). Springer International Publishing.

[47] Cole, J., & Foster, H. (2007). *Using Moodle: Teaching with the popular open source course management system*. " O'Reilly Media, Inc.".

[48] StatCounter, Global Stats. (2016). Top 7 Desktop OSs on 2016. http://gs.statcounter.com/#desktop-os-ww-yearly-2016-2016-bar, Retrieved June 12, 2016.

[49] StatCounter, Global Stats. (2016). Top 8 Mobile & Tablet OSs on 2016. http://gs.statcounter.com/#mobile+tablet-os-ww-yearly-2016-2016-bar, Retrieved June 12, 2016.

[50] Salatas J. (2016). Mobile Foreign Language Learning repository. Github Repository. https://github.com/jsalatas/MobileForeignLanguageLearning, Retrieved June 15, 2016.

[51] License, G. G. P. (2007). version 3. *Free Software Foundation*. http://www.gnu.org/copyleft/gpl.html, Retrieved June 15, 2016.

[52] GNU Project. What is free software? *Free Software Foundation*. https://www.gnu.org/philosophy/free-sw.en.html, Retrieved June 15, 2016.

[53] Adobe, Systems Incorporated. ActionScript 3.0 Reference for the Adobe Flash Platform. http://help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/index.html, Retrieved June 15, 2016.

[54] Apache, Software Foundation. About Apache Flex. http://flex.apache.org/about-whatis.html, Retrieved June 15, 2016.

[55] Adobe, Systems Incorporated. Adobe AIR Developer Center. http://www.adobe.com/devnet/air.html, Retrieved June 15, 2016.

[56] Robotlegs AS3 Micro-Architecture. http://www.robotlegs.org/, Retrieved June 17, 2016.

[57] Robotolegs repository. Github Repository. https://github.com/robotlegs/robotlegs-framework, Retrieved June 17, 2016.

[58] Hooks, J., & Fallow, L. (2011). ActionScript Developer's Guide to Robotlegs. O'Reilly Media, Inc.

[59] McCauley T. (2008). Introduction to event handling in ActionScript 3.0. http://www.adobe.com/devnet/actionscript/articles/event_handling_as3.html, Retrieved June 17, 2016.

[60] Penner R. (2016). Signals: Think Outside the Event. Github Repository. https://github.com/robertpenner/as3-signals, Retrieved June 17, 2016.

[61] Hooks J. (2016). The SignalCommandMap. Github Repository. https://github.com/joelhooks/signals-extensions-CommandSignal, Retrieved June 17, 2016.

[62] Adobe, Systems Incorporated. (2015). Developing Native Extensions for ADOBE AIR. http://help.adobe.com/en_US/air/extensions/air_extensions.pdf, Retrieved June 17, 2016.

[63] Spring Application Framework. http://spring.io/, Retrieved June 20, 2016.

[64] JBoss. Hibernate ORM. http://hibernate.org/orm/, Retrieved June 20, 2016.

[65] Adobe, Systems Incorporated. BlazeDS. https://sourceforge.net/adobe/blazeds/wiki/Home/, Retrieved June 20, 2016.

[66] Adobe, Systems Incorporated. (2013). Action Message Format: AMF 3. http://wwwimages.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/amf/pdf/amf-file-format-spec.pdf, Retrieved June 20, 2016.

[67] Spring BlazeDS Integration. http://docs.spring.io/spring-flex/docs/1.5.2.RELEASE/reference/html/, Retrieved June 20, 2016.

[68] Eclipse IDE. https://eclipse.org/ide/, Retrieved June 20, 2016.

[69] Adobe FlashBuilder 4.7 Premium. http://www.adobe.com/products/flash-builder.html, Retrieved June 20, 2016.

[70] Microsoft Visual Studio. https://www.visualstudio.com/en-us/visual-studio-homepage-vs.aspx, Retrieved June 20, 2016.

[71] Adobe Flash Professional. http://www.adobe.com/products/animate.html, Retrieved June 20, 2016.

[72] Red5 Media Server. http://red5.org/, Retrieved June 20, 2016.

[73] Moodle Traker. [MDL-54792] External database enrolment: Creation of new courses using different templates. https://tracker.moodle.org/browse/MDL-54792, Retrieved June 20, 2016.

[74] Roesler, V., Cecagno, F., Dixon, F., & Daronco, L. C. (2012). *Mconf: An open source multiconference system for web and mobile devices*. INTECH Open Access Publisher.

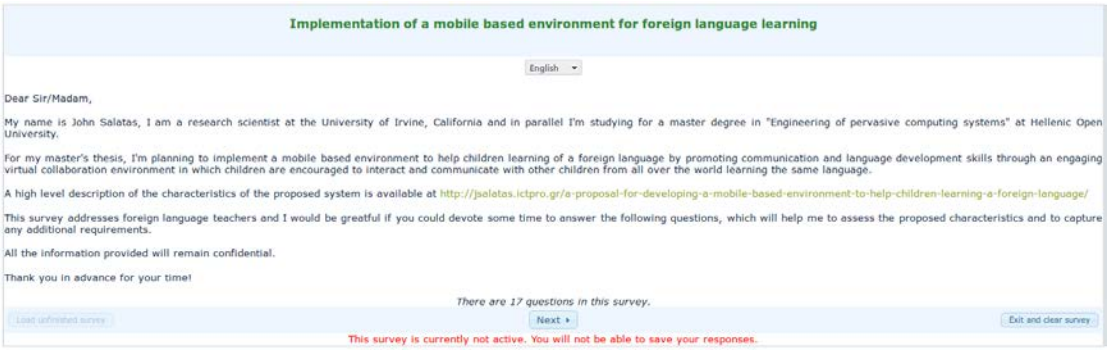
[75] MConf. Mobile client for BigBlueButton and Mconf based on Adobe Air. . Github Repository. https://github.com/mconf/bbb-air-client, Retrieved June 20, 2016.

[76] Dey, A. K. (2010). Context-aware computing. *Ubiquitous Computing Fundamentals*, 321-352.

[77] Adobe, Systems Incorporated. Support multiple screen sizes and DPI values in a mobile application. http://help.adobe.com/en_US/flex/mobileapps/WS19f279b149e7481c682e5a9412cf5976c17-8000.html, Retrieved June 21, 2016.

Appendix A: Surveys

A-1: Requirement Analysis Survey



Implementation of a mobile based environment for foreign language learning

English

Dear Sir/Madam,

My name is John Salatas, I am a research scientist at the University of Irvine, California and in parallel I'm studying for a master degree in "Engineering of pervasive computing systems" at Hellenic Open University.

For my master's thesis, I'm planning to implement a mobile based environment to help children learning of a foreign language by promoting communication and language development skills through an engaging virtual collaboration environment in which children are encouraged to interact and communicate with other children from all over the world learning the same language.

A high level description of the characteristics of the proposed system is available at http://jsalatas.ictpro.gr/a-proposal-for-developing-a-mobile-based-environment-to-help-children-learning-a-foreign-language/

This survey addresses foreign language teachers and I would be greatful if you could devote some time to answer the following questions, which will help me to assess the proposed characteristics and to capture any additional requirements.

All the information provided will remain confidential.

Thank you in advance for your time!

There are 17 questions in this survey.

Load unfinished survey | Next › | Exit and clear survey

This survey is currently not active. You will not be able to save your responses.

Implementation of a mobile based environment for foreign language learning

0% 100%

English

Demographic Data

* **Your sex**
- Male
- Female

* **Your age**
- 25-29
- 30-34
- 35-39
- 40-44
- 45-49
- 50 and greater

* **Your occupation**
- Primary public school
- Secondary public school
- Primary private school
- Secondary private school
- Private educator
- Other:

* **Languages you teach**
- English
- French
- German
- Spanish
- Other:

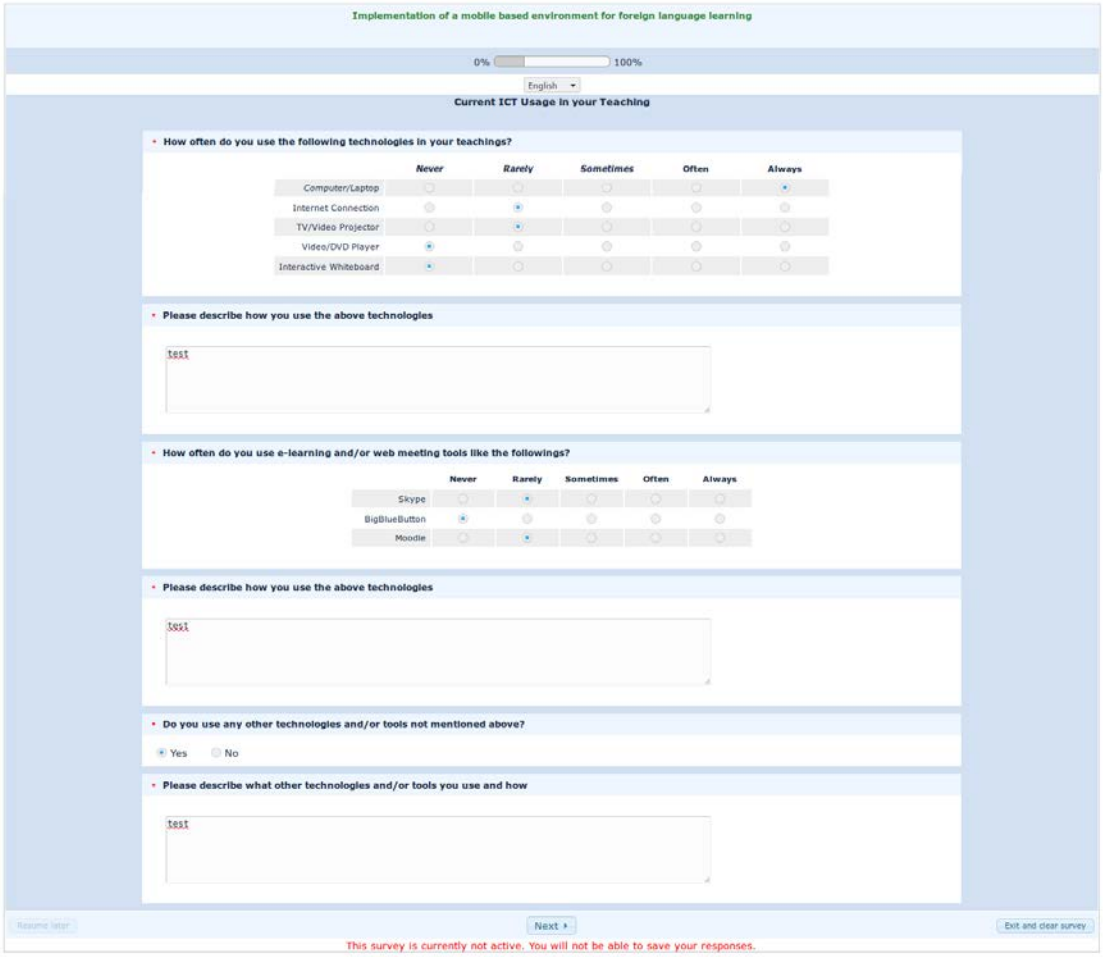
* **Total number of your students**
Only numbers may be entered in this field.

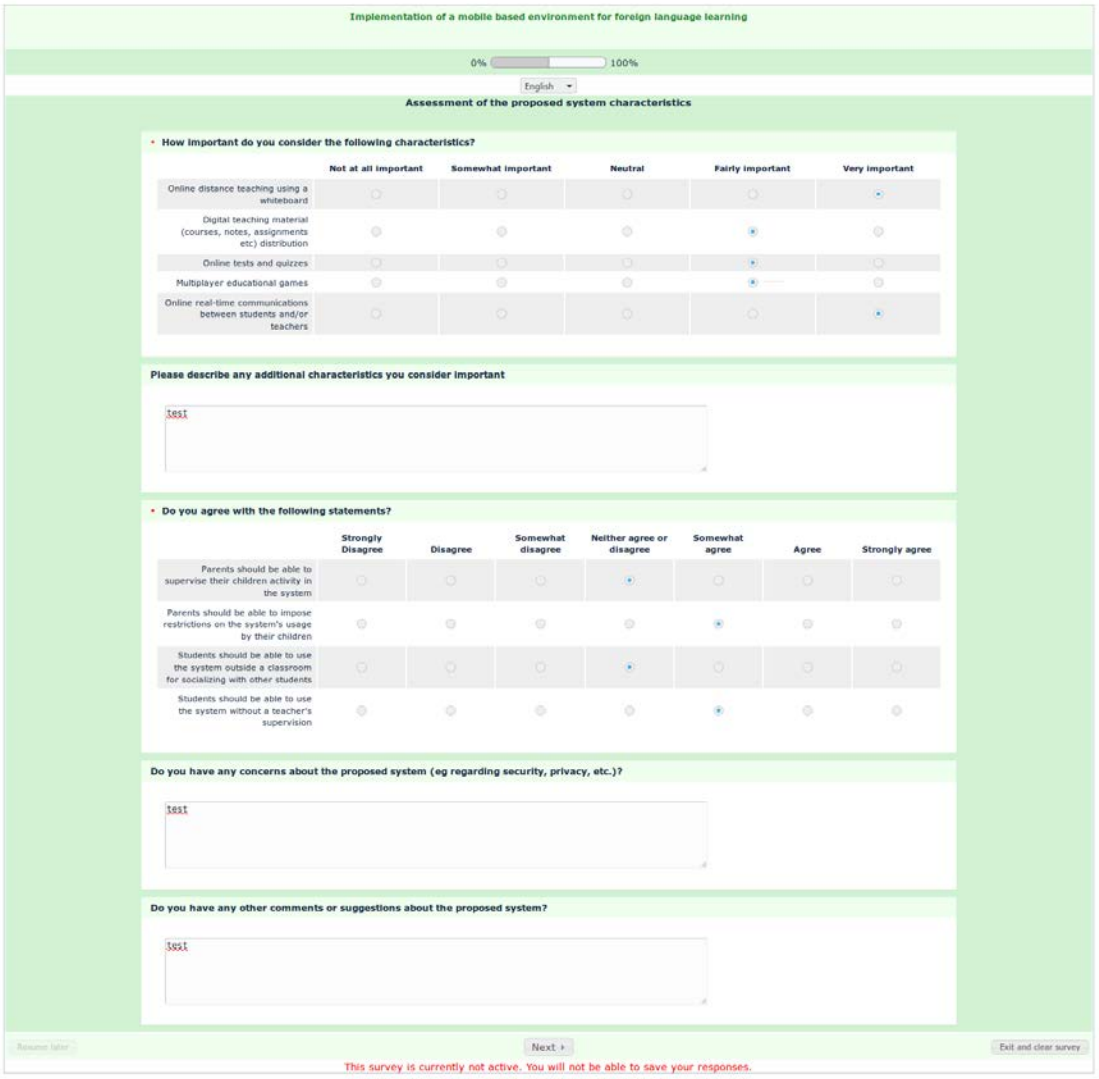
12

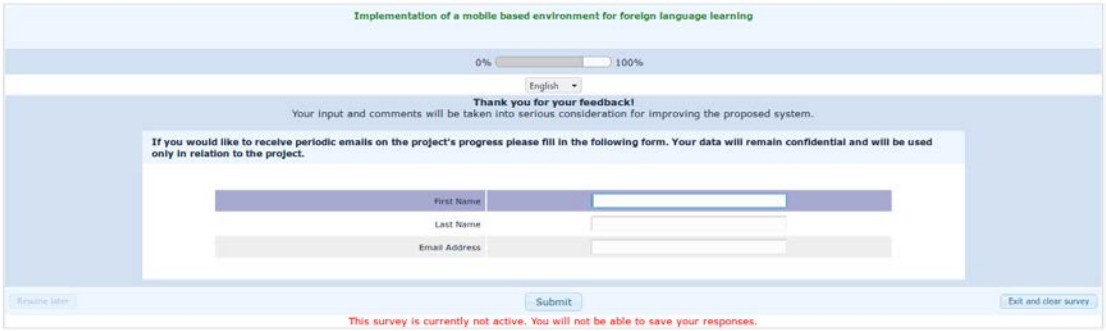
Please enter the total number of your students you are teaching in all of your classes

Resume later | Next › | Exit and clear survey

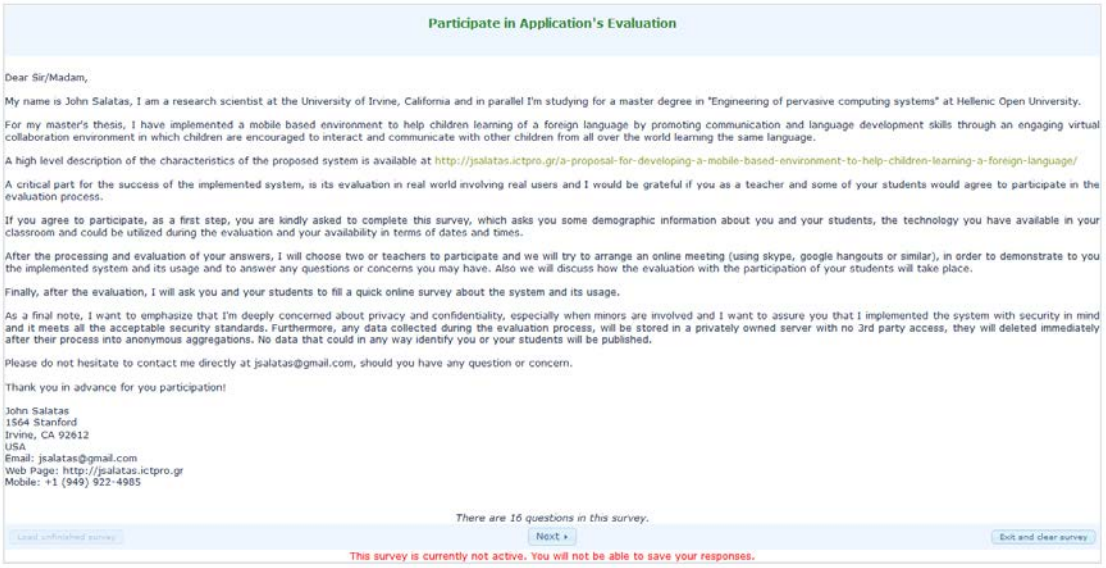
This survey is currently not active. You will not be able to save your responses.


Implementation of a mobile based environment for foreign language learning
0%
100%
English
Current ICT Usage in your Teaching
How often do you use the following technologies in your teachings?
Never
Rarely
Sometimes
Often
Always
Computer/Laptop
Internet Connection
TV/Video Projector
Video/DVD Player
Interactive Whiteboard
Please describe how you use the above technologies
test
How often do you use e-learning and/or web meeting tools like the followings?
Never
Rarely
Sometimes
Often
Always
Skype
BigBlueButton
Moodle
Please describe how you use the above technologies
test
Do you use any other technologies and/or tools not mentioned above?
Yes
No
Please describe what other technologies and/or tools you use and how
test
Resume later
Next
Exit and clear survey
This survey is currently not active. You will not be able to save your responses.


Implementation of a mobile based environment for foreign language learning
0%
100%
English
Assessment of the proposed system characteristics
How important do you consider the following characteristics?
Not at all important
Somewhat important
Neutral
Fairly important
Very important
Online distance teaching using a whiteboard
Digital teaching material (courses, notes, assignments etc) distribution
Online tests and quizzes
Multiplayer educational games
Online real-time communications between students and/or teachers
Please describe any additional characteristics you consider important
test
Do you agree with the following statements?
Strongly Disagree
Disagree
Somewhat disagree
Neither agree or disagree
Somewhat agree
Agree
Strongly agree
Parents should be able to supervise their children activity in the system
Parents should be able to impose restrictions on the system's usage by their children
Students should be able to use the system outside a classroom for socializing with other students
Students should be able to use the system without a teacher's supervision
Do you have any concerns about the proposed system (eg regarding security, privacy, etc.)?
test
Do you have any other comments or suggestions about the proposed system?
test
Next
Exit and clear survey
This survey is currently not active. You will not be able to save your responses.


Implementation of a mobile based environment for foreign language learning
0%
100%
English
Thank you for your feedback!
Your input and comments will be taken into serious consideration for improving the proposed system.
If you would like to receive periodic emails on the project's progress please fill in the following form. Your data will remain confidential and will be used only in relation to the project.
First Name
Last Name
Email Address
Submit
Exit and clear survey
This survey is currently not active. You will not be able to save your responses.

A-2: Evaluation's Preliminary Survey



Participate in Application's Evaluation

Dear Sir/Madam,

My name is John Salatas, I am a research scientist at the University of Irvine, California and in parallel I'm studying for a master degree in "Engineering of pervasive computing systems" at Hellenic Open University.

For my master's thesis, I have implemented a mobile based environment to help children learning of a foreign language by promoting communication and language development skills through an engaging virtual collaboration environment in which children are encouraged to interact and communicate with other children from all over the world learning the same language.

A high level description of the characteristics of the proposed system is available at http://jsalatas.ictpro.gr/a-proposal-for-developing-a-mobile-based-environment-to-help-children-learning-a-foreign-language/

A critical part for the success of the implemented system, is its evaluation in real world involving real users and I would be grateful if you as a teacher and some of your students would agree to participate in the evaluation process.

If you agree to participate, as a first step, you are kindly asked to complete this survey, which asks you some demographic information about you and your students, the technology you have available in your classroom and could be utilized during the evaluation and your availability in terms of dates and times.

After the processing and evaluation of your answers, I will choose two or teachers to participate and we will try to arrange an online meeting (using skype, google hangouts or similar), in order to demonstrate to you the implemented system and its usage and to answer any questions or concerns you may have. Also we will discuss how the evaluation with the participation of your students will take place.

Finally, after the evaluation, I will ask you and your students to fill a quick online survey about the system and its usage.

As a final note, I want to emphasize that I'm deeply concerned about privacy and confidentiality, especially when minors are involved and I want to assure you that I implemented the system with security in mind and it meets all the acceptable security standards. Furthermore, any data collected during the evaluation process, will be stored in a privately owned server with no 3rd party access, they will deleted immediately after their process into anonymous aggregations. No data that could in any way identify you or your students will be published.

Please do not hesitate to contact me directly at jsalatas@gmail.com, should you have any question or concern.

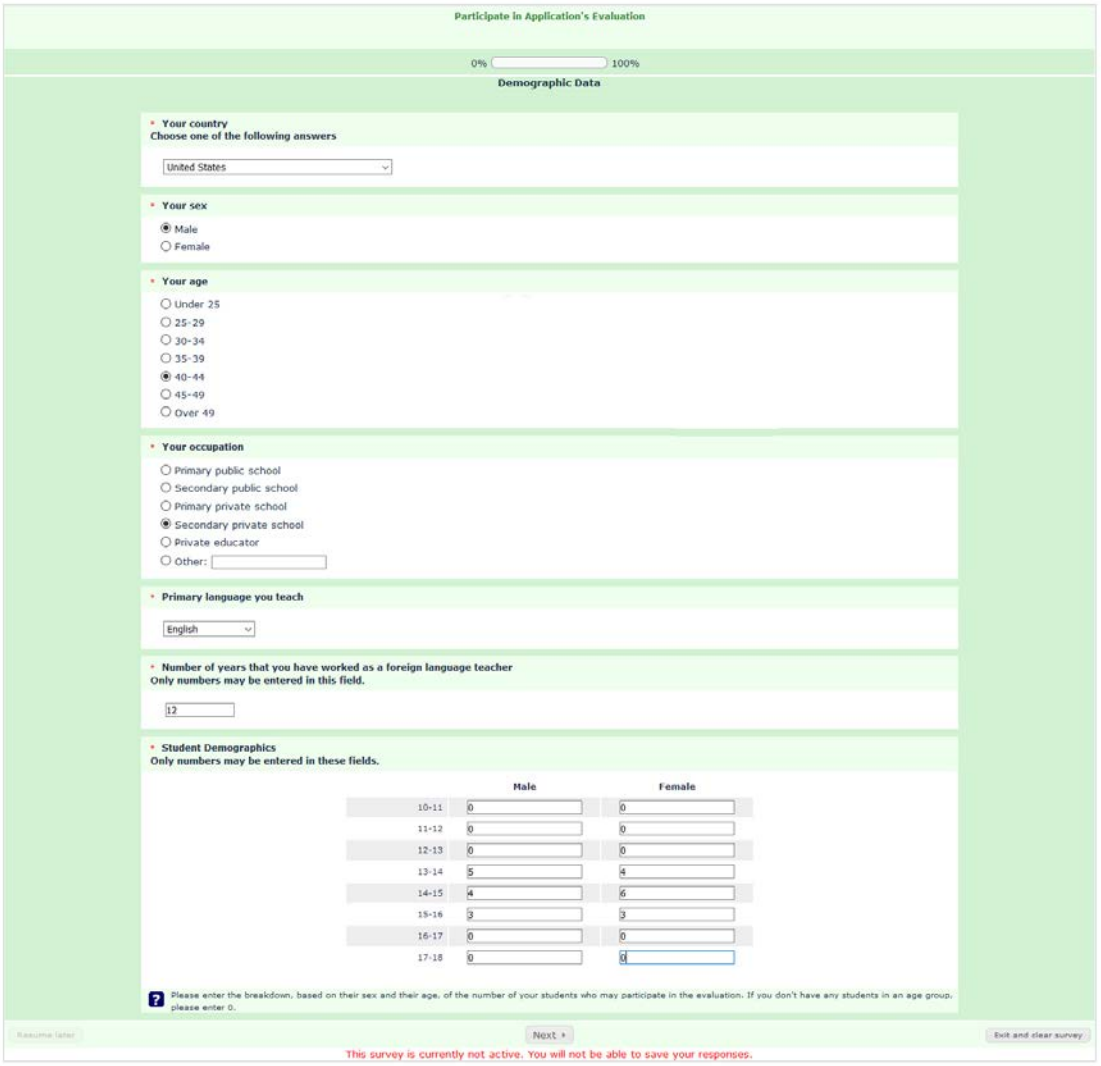
Thank you in advance for you participation!

John Salatas
1564 Stanford
Irvine, CA 92612
USA
Email: jsalatas@gmail.com
Web Page: http://jsalatas.ictpro.gr
Mobile: +1 (949) 922-4985

There are 16 questions in this survey.

Load unfinished survey | Next ▸ | Exit and clear survey

This survey is currently not active. You will not be able to save your responses.



Participate in Application's Evaluation

0% 100%

Demographic Data

* **Your country**
Choose one of the following answers

United States

* **Your sex**

- ◉ Male
- ○ Female

* **Your age**

- ○ Under 25
- ○ 25-29
- ○ 30-34
- ○ 35-39
- ◉ 40-44
- ○ 45-49
- ○ Over 49

* **Your occupation**

- ○ Primary public school
- ○ Secondary public school
- ○ Primary private school
- ◉ Secondary private school
- ○ Private educator
- ○ Other:

* **Primary language you teach**

English

* **Number of years that you have worked as a foreign language teacher**
Only numbers may be entered in this field.

12

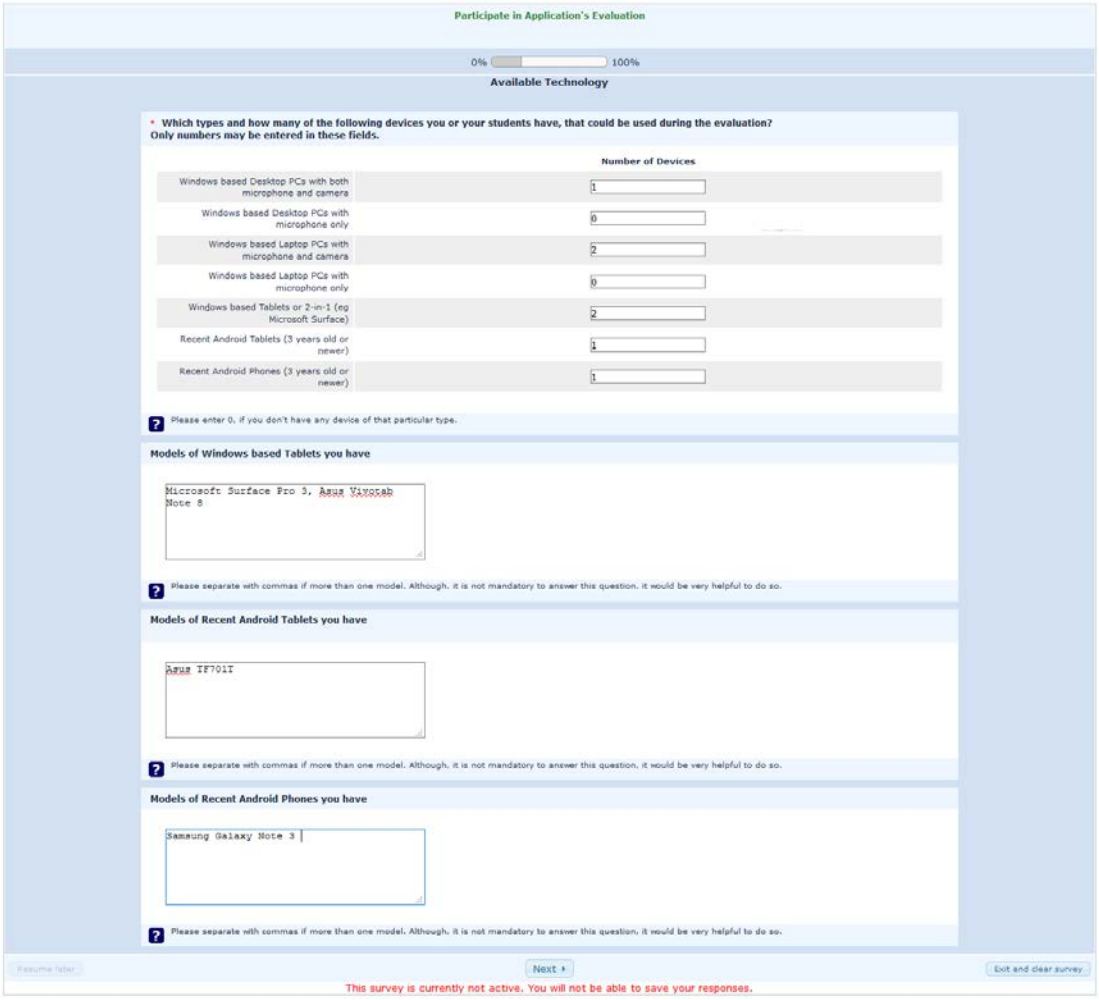
* **Student Demographics**
Only numbers may be entered in these fields.

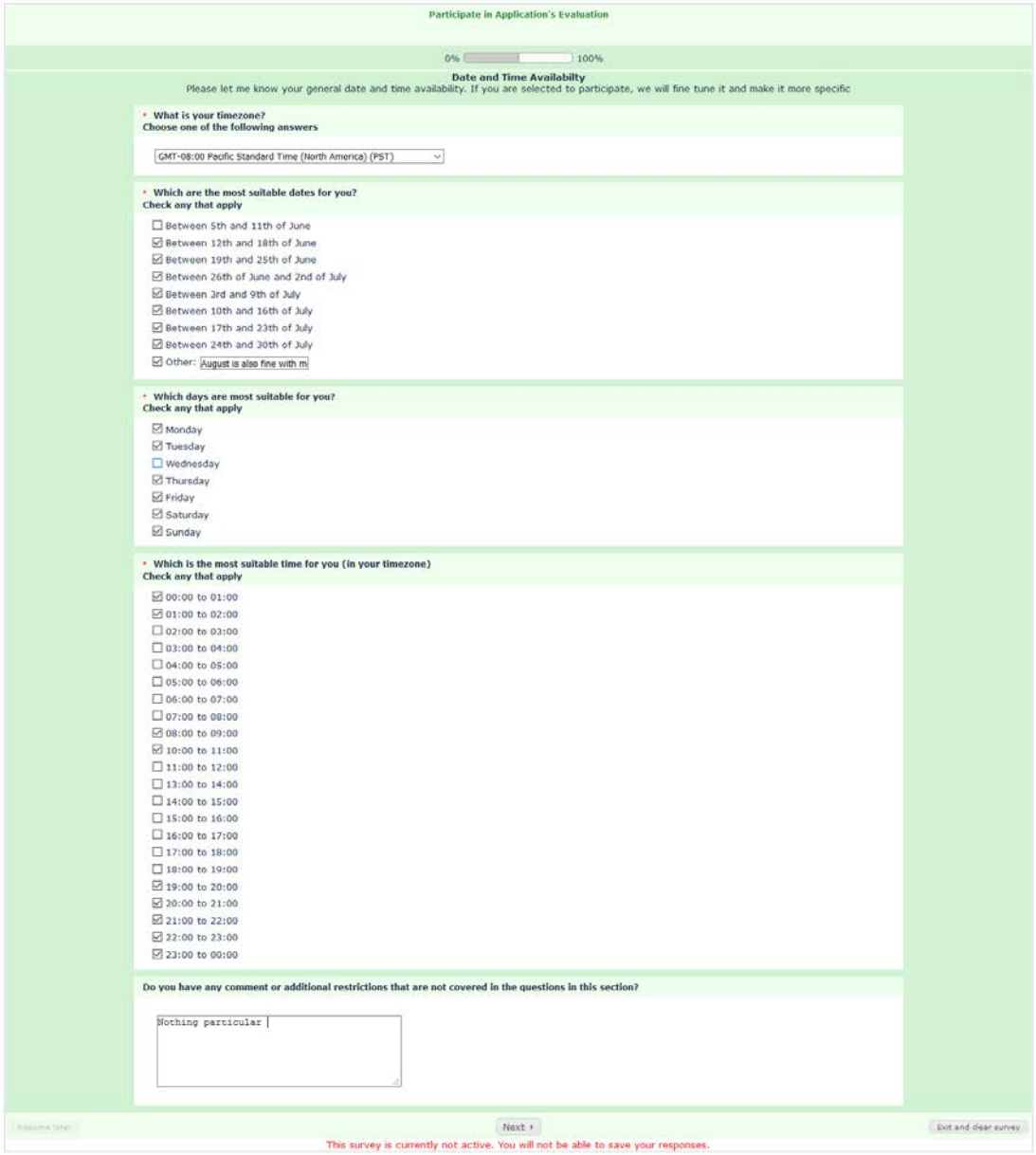
	Male	Female
10-11	0	0
11-12	0	0
12-13	0	0
13-14	5	4
14-15	4	6
15-16	3	3
16-17	0	0
17-18	0	0

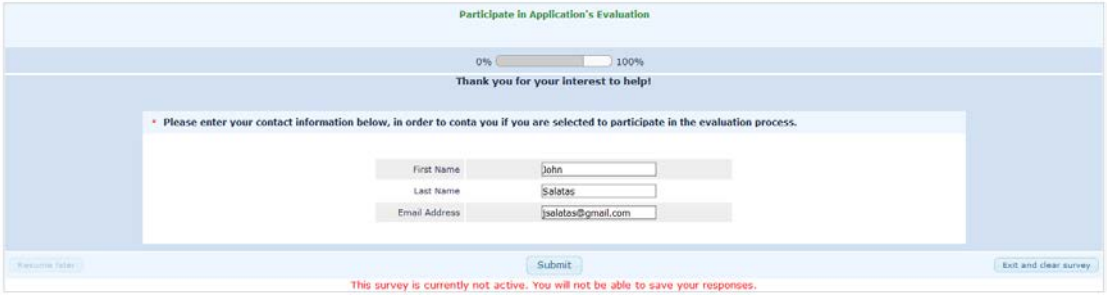
? Please enter the breakdown, based on their sex and their age, of the number of your students who may participate in the evaluation. If you don't have any students in an age group, please enter 0.

Resume later | Next ▸ | Exit and clear survey

This survey is currently not active. You will not be able to save your responses.


Participate in Application's Evaluation
0%
100%
Available Technology
* Which types and how many of the following devices you or your students have, that could be used during the evaluation?
Only numbers may be entered in these fields.
Number of Devices
Windows based Desktop PCs with both microphone and camera
1
Windows based Desktop PCs with microphone only
0
Windows based Laptop PCs with microphone and camera
2
Windows based Laptop PCs with microphone only
0
Windows based Tablets or 2-in-1 (eg Microsoft Surface)
2
Recent Android Tablets (3 years old or newer)
1
Recent Android Phones (3 years old or newer)
1
Please enter 0, if you don't have any device of that particular type.
Models of Windows based Tablets you have
Microsoft Surface Pro 3, Asus Vivotab Note 8
Please separate with commas if more than one model. Although, it is not mandatory to answer this question, it would be very helpful to do so.
Models of Recent Android Tablets you have
Asus TF701T
Please separate with commas if more than one model. Although, it is not mandatory to answer this question, it would be very helpful to do so.
Models of Recent Android Phones you have
Samsung Galaxy Note 3
Please separate with commas if more than one model. Although, it is not mandatory to answer this question, it would be very helpful to do so.
Resume later
Next
Exit and clear survey
This survey is currently not active. You will not be able to save your responses.


Participate in Application's Evaluation
0% 100%
Date and Time Availabilty
Please let me know your general date and time availability. If you are selected to participate, we will fine tune it and make it more specific
* What is your timezone?
Choose one of the following answers
GMT-08:00 Pacific Standard Time (North America) (PST)
* Which are the most suitable dates for you?
Check any that apply
Between 5th and 11th of June
Between 12th and 18th of June
Between 19th and 25th of June
Between 26th of June and 2nd of July
Between 3rd and 9th of July
Between 10th and 16th of July
Between 17th and 23th of July
Between 24th and 30th of July
Other: August is also fine with m
* Which days are most suitable for you?
Check any that apply
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
* Which is the most suitable time for you (in your timezone)
Check any that apply
00:00 to 01:00
01:00 to 02:00
02:00 to 03:00
03:00 to 04:00
04:00 to 05:00
05:00 to 06:00
06:00 to 07:00
07:00 to 08:00
08:00 to 09:00
10:00 to 11:00
11:00 to 12:00
13:00 to 14:00
14:00 to 15:00
15:00 to 16:00
16:00 to 17:00
17:00 to 18:00
18:00 to 19:00
19:00 to 20:00
20:00 to 21:00
21:00 to 22:00
22:00 to 23:00
23:00 to 00:00
Do you have any comment or additional restrictions that are not covered in the questions in this section?
Nothing particular
Next
Exit and clear survey
This survey is currently not active. You will not be able to save your responses.


Participate in Application's Evaluation
0% 100%
Thank you for your interest to help!
* Please enter your contact information below, in order to conta you if you are selected to participate in the evaluation process.
First Name
John
Last Name
Salatas
Email Address
jsalatas@gmail.com
Submit
Exit and clear survey
This survey is currently not active. You will not be able to save your responses.

Appendix B: Application’s Screenshots

The following figures depict indicative screenshots of the application running on A. Windows 10 based PC with a screen resolution of 1920x1080 pixels (96 dpi).

B. Android 5.0 based smartphone with a screen resolution of 1080x1920 pixels (480 dpi).

The views shown are as follows

1. Initial configuration

2. Login

3. Registration

4. Home screen

5. Home screen with menu

5.1. Admin

5.2. Teacher

5.3. Student

5.4. Parent

6. List/Detail View

6.1. Classroom List

6.2. Classroom View

7. Online Meetings

7.1. Settings View

7.2. Chat View

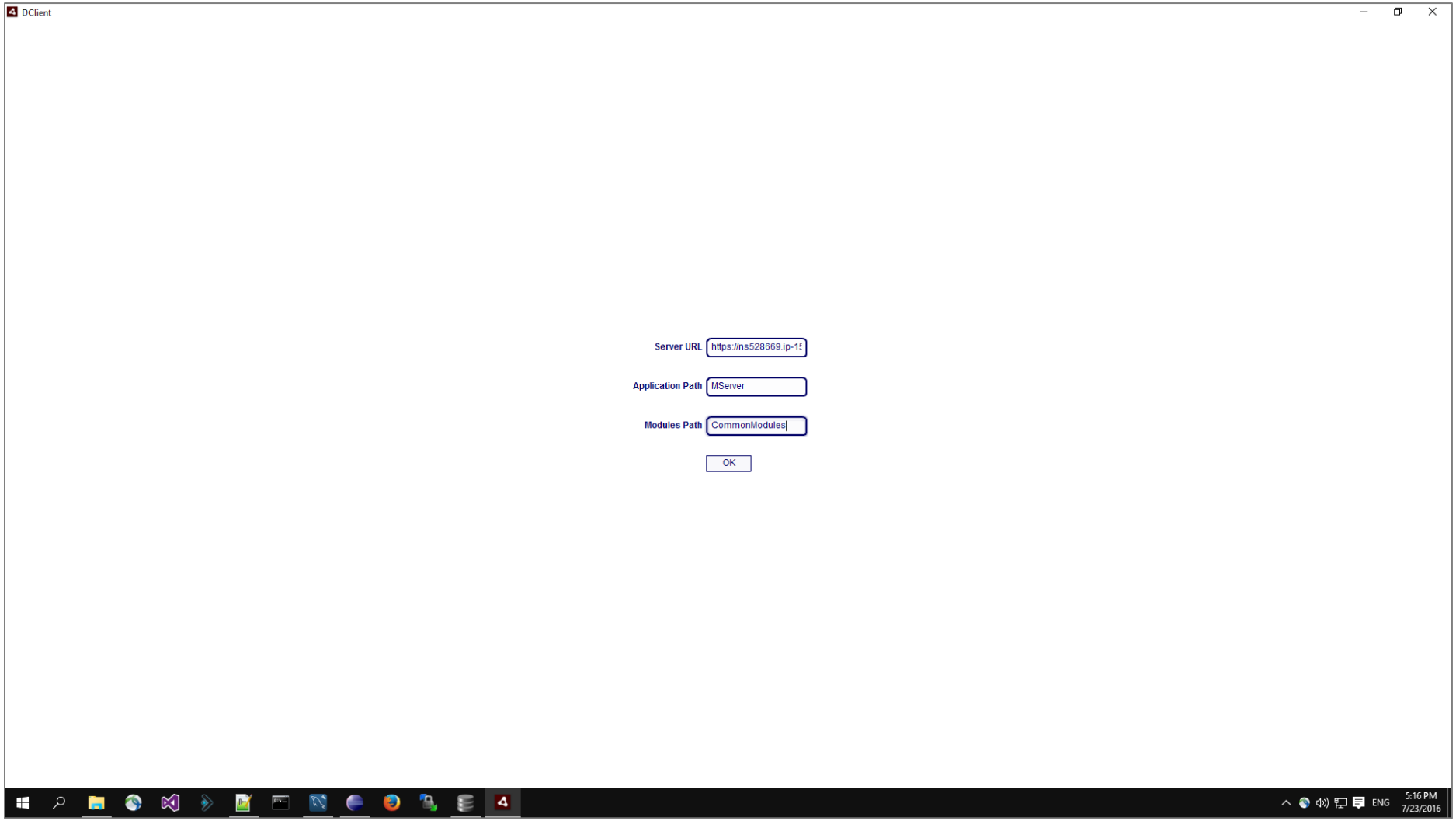
7.3. Camera View

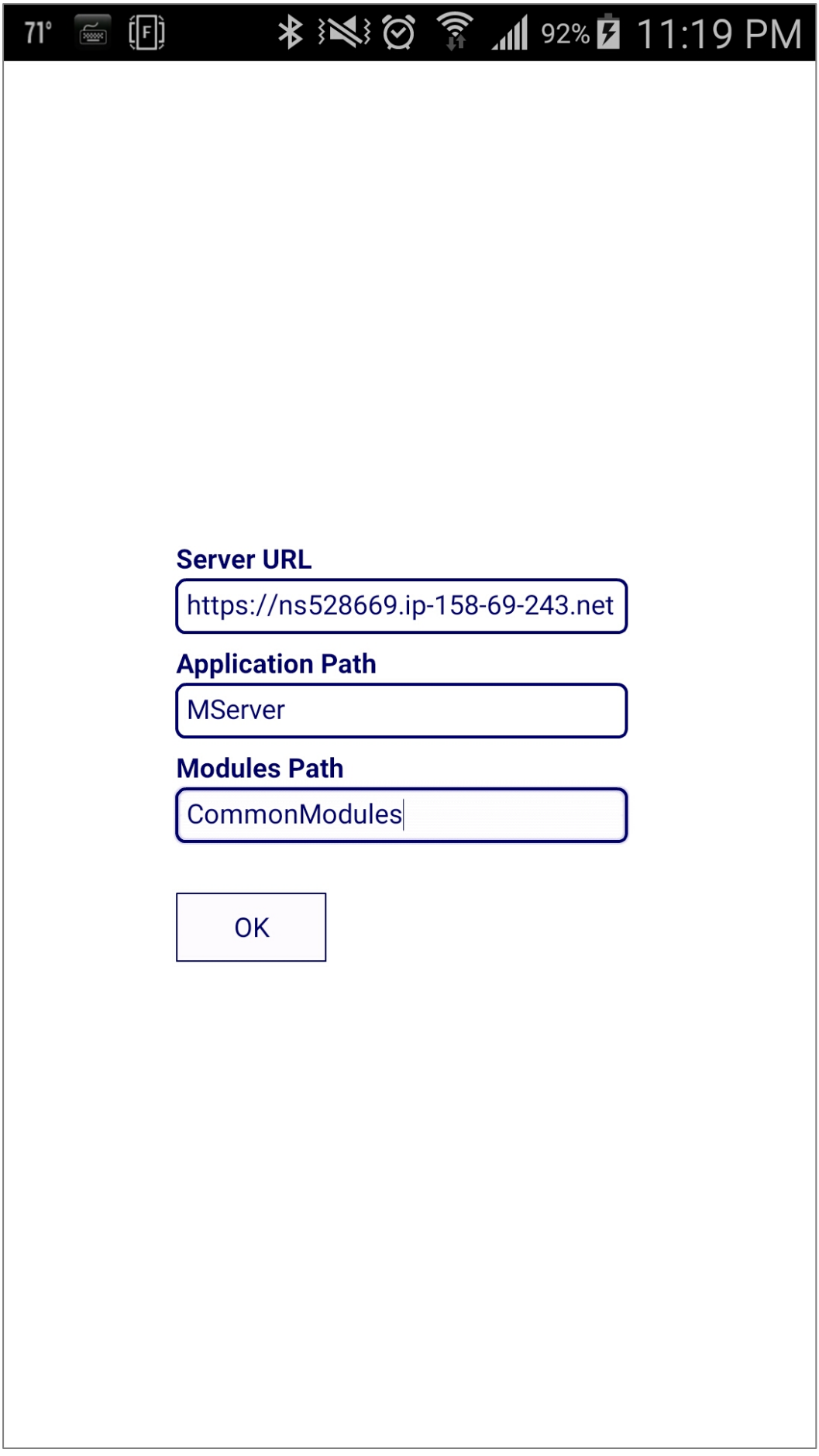
7.4. Participants View

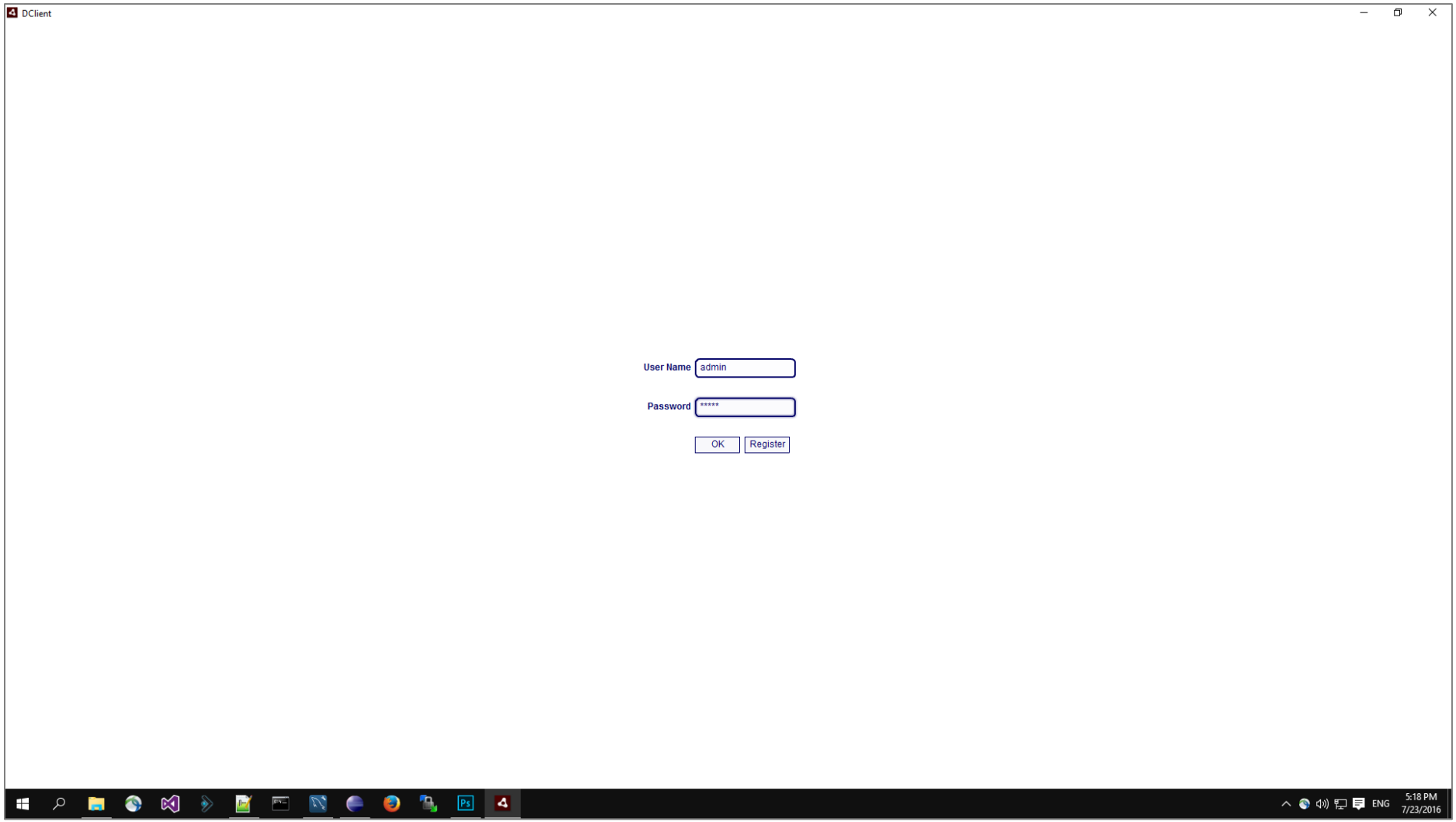
7.5. Board View

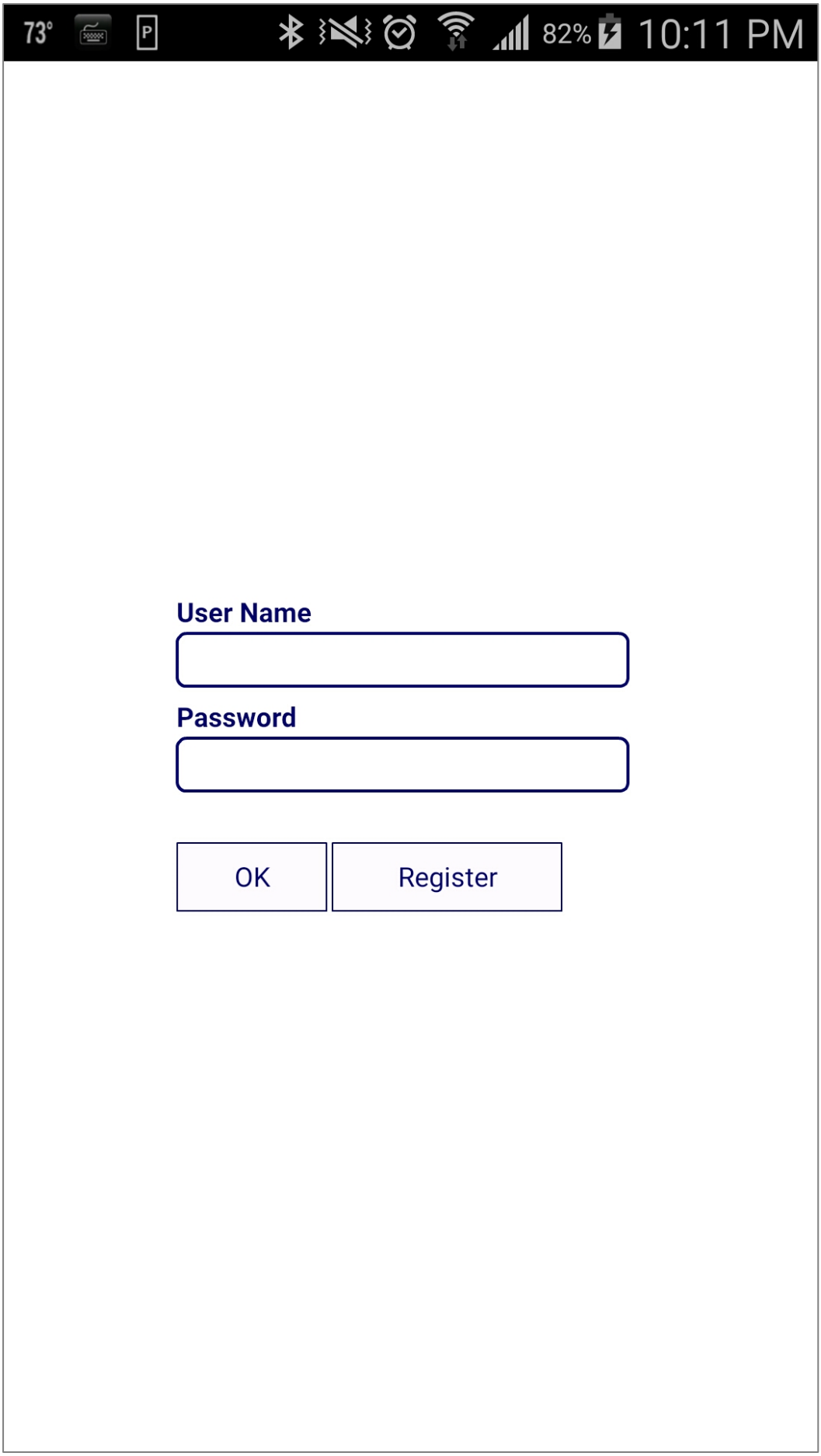
7.5.1. WhiteBoard

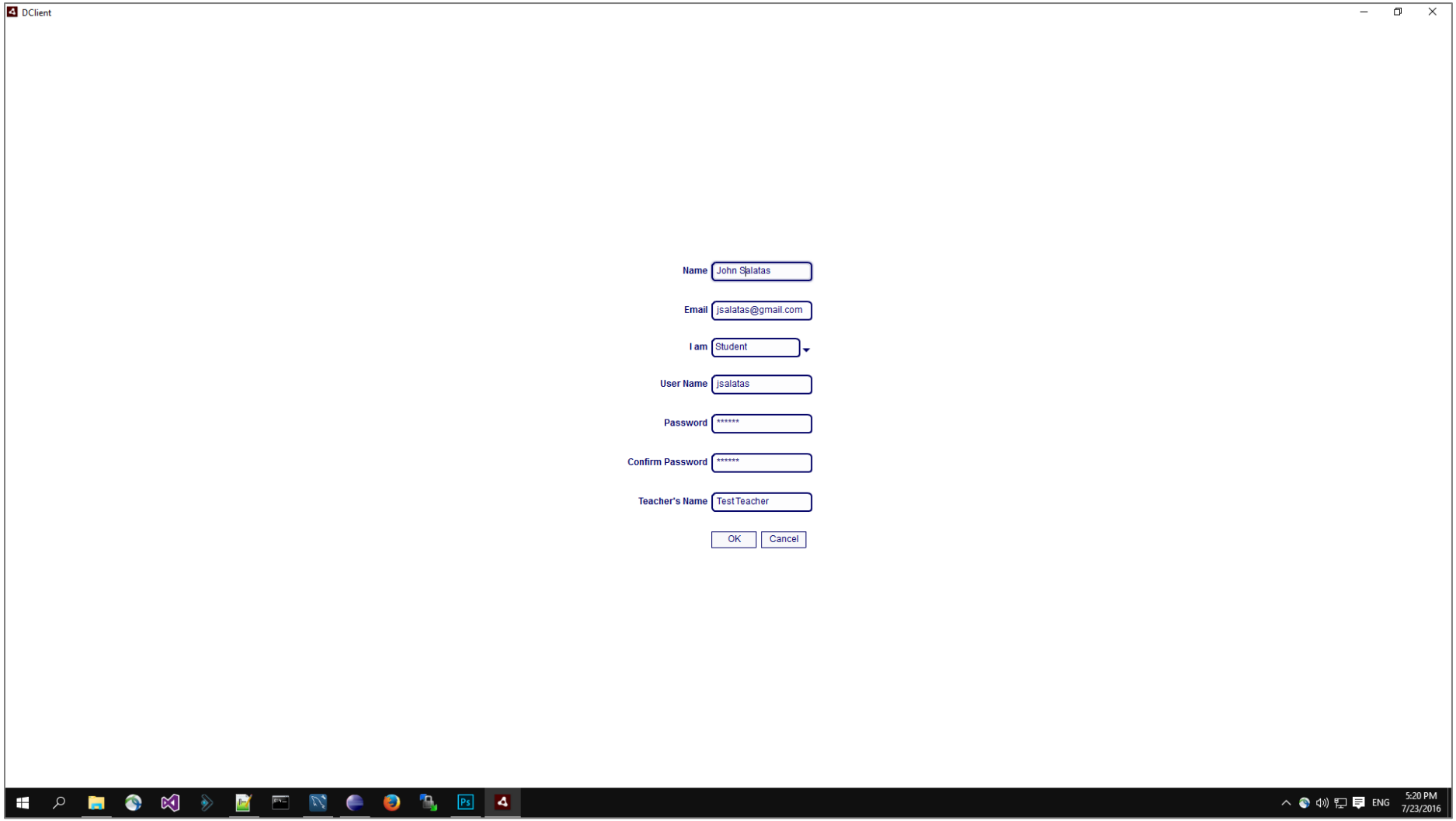
7.5.2. Potato Guy

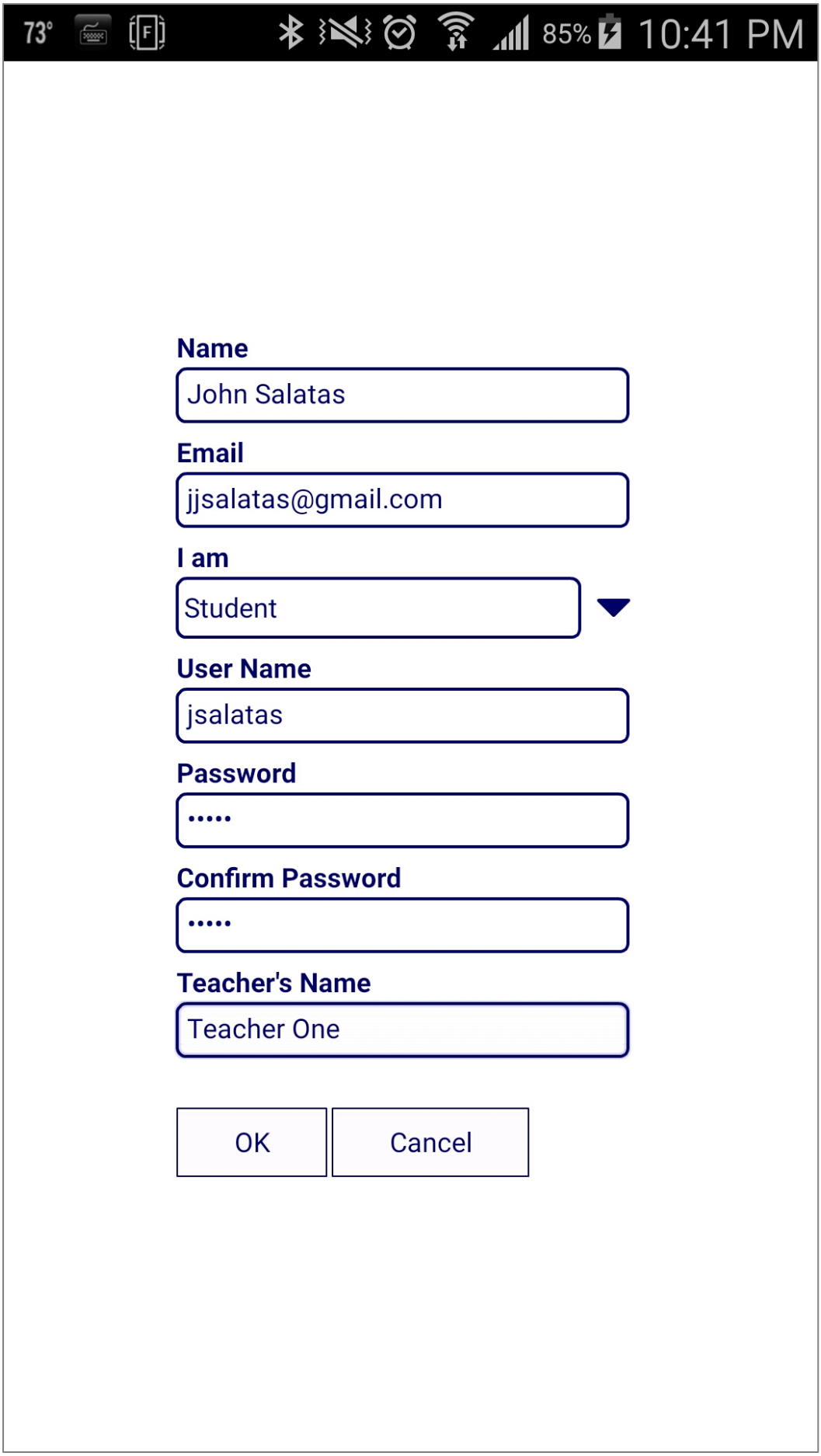

DClient
Server URL
https://ns528669.ip-15
Application Path
MServer
Modules Path
CommonModules
OK
ENG
5:16 PM
7/23/2016

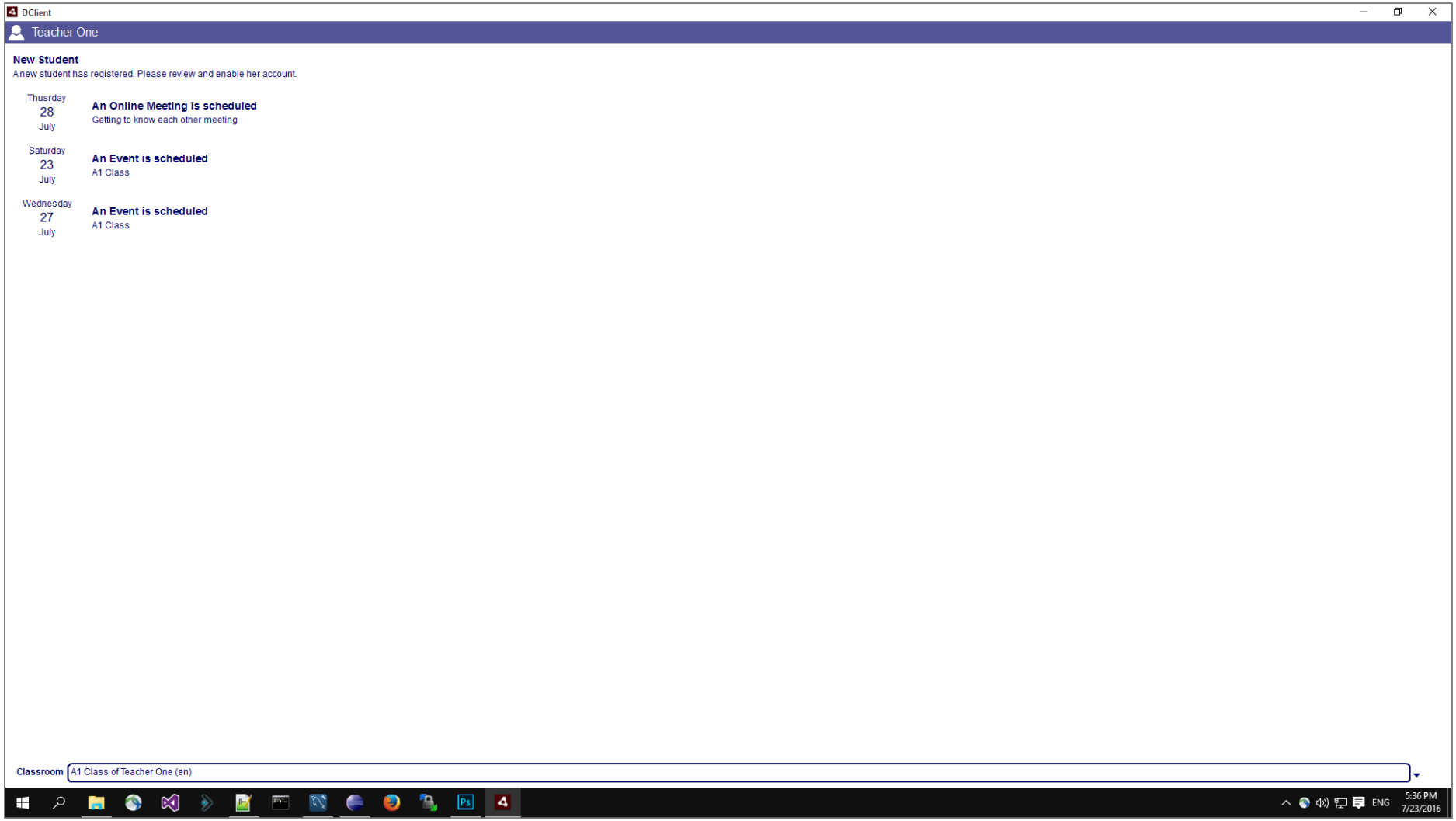

71°
92%
11:19 PM
Server URL
https://ns528669.ip-158-69-243.net
Application Path
MServer
Modules Path
CommonModules
OK

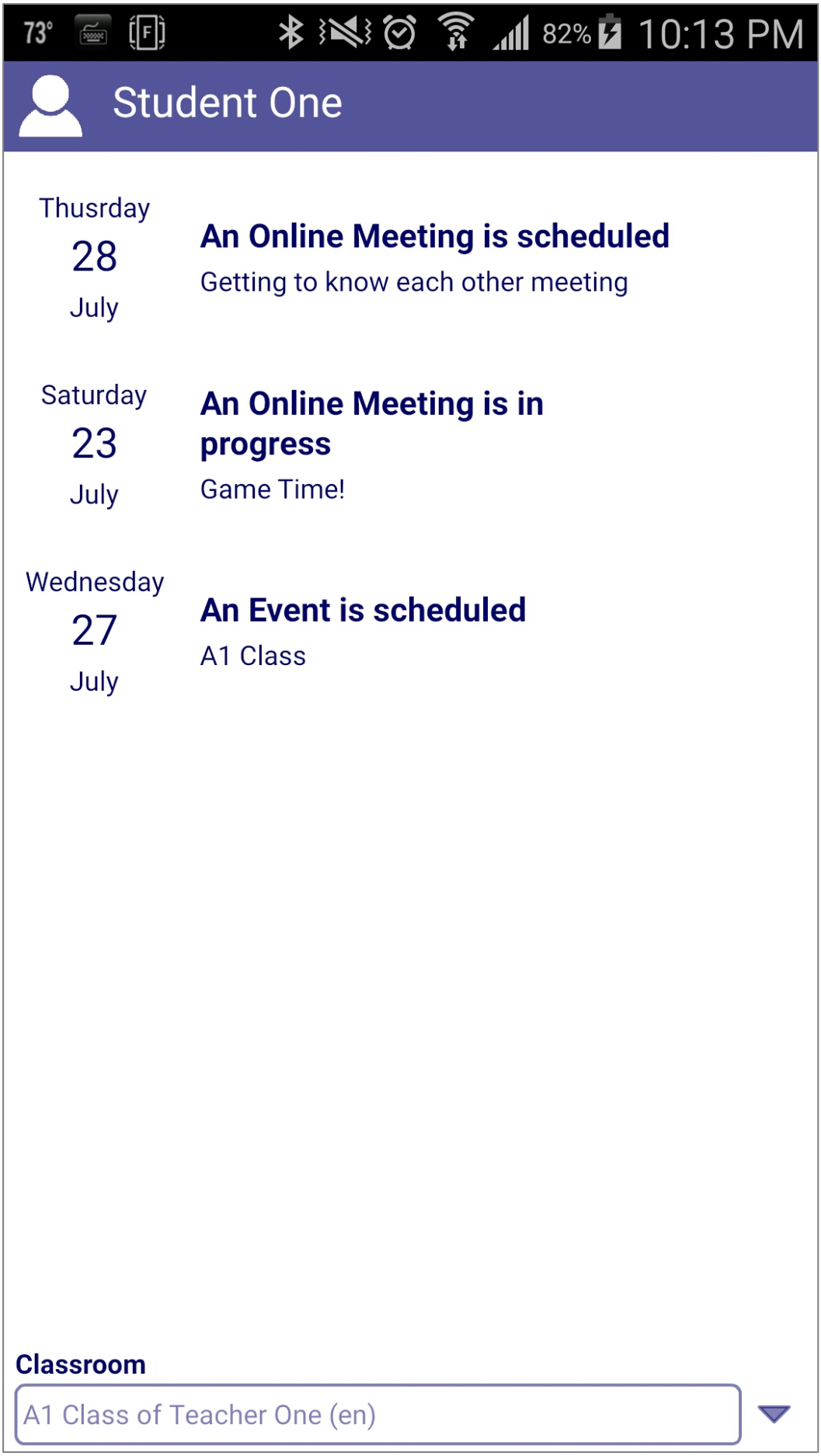

DClient
User Name
admin
Password
*****
OK
Register
ENG
5:18 PM
7/23/2016

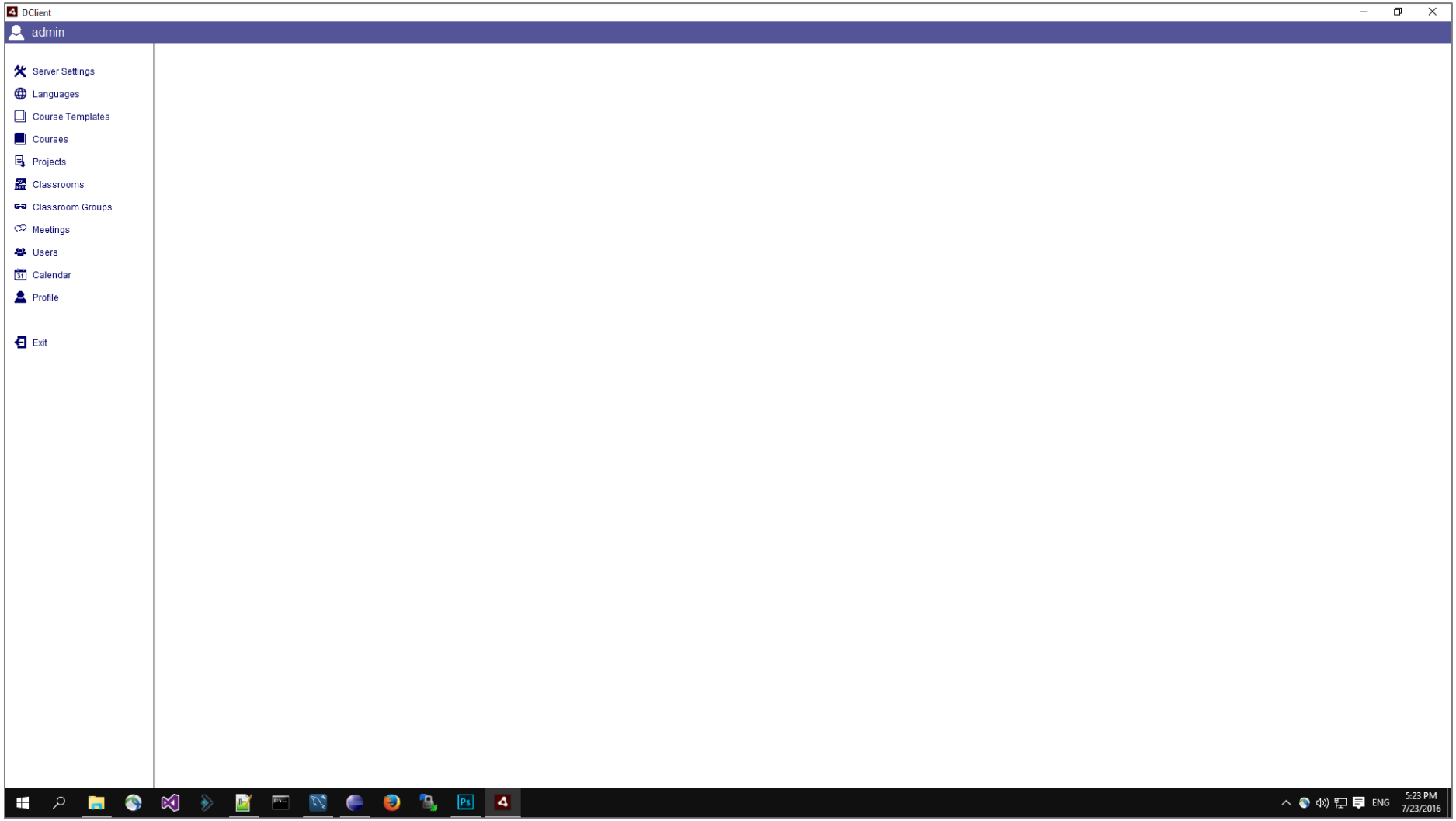

73°
82%
10:11 PM
User Name
Password
OK
Register

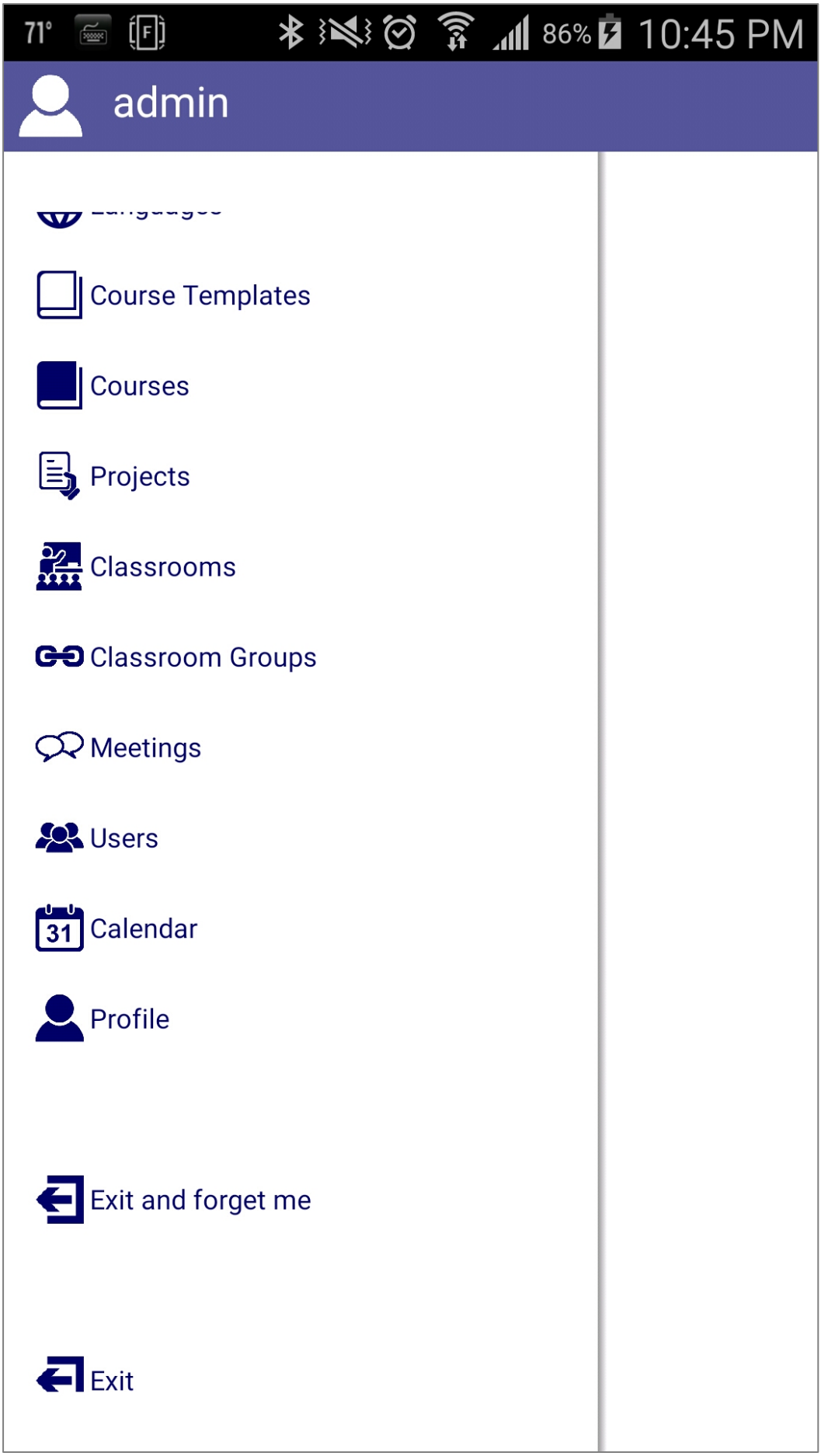

DClient
Name
John Salatas
Email
jsalatas@gmail.com
I am
Student
User Name
jsalatas
Password
******
Confirm Password
******
Teacher's Name
Test Teacher
OK
Cancel
ENG
5:20 PM
7/23/2016

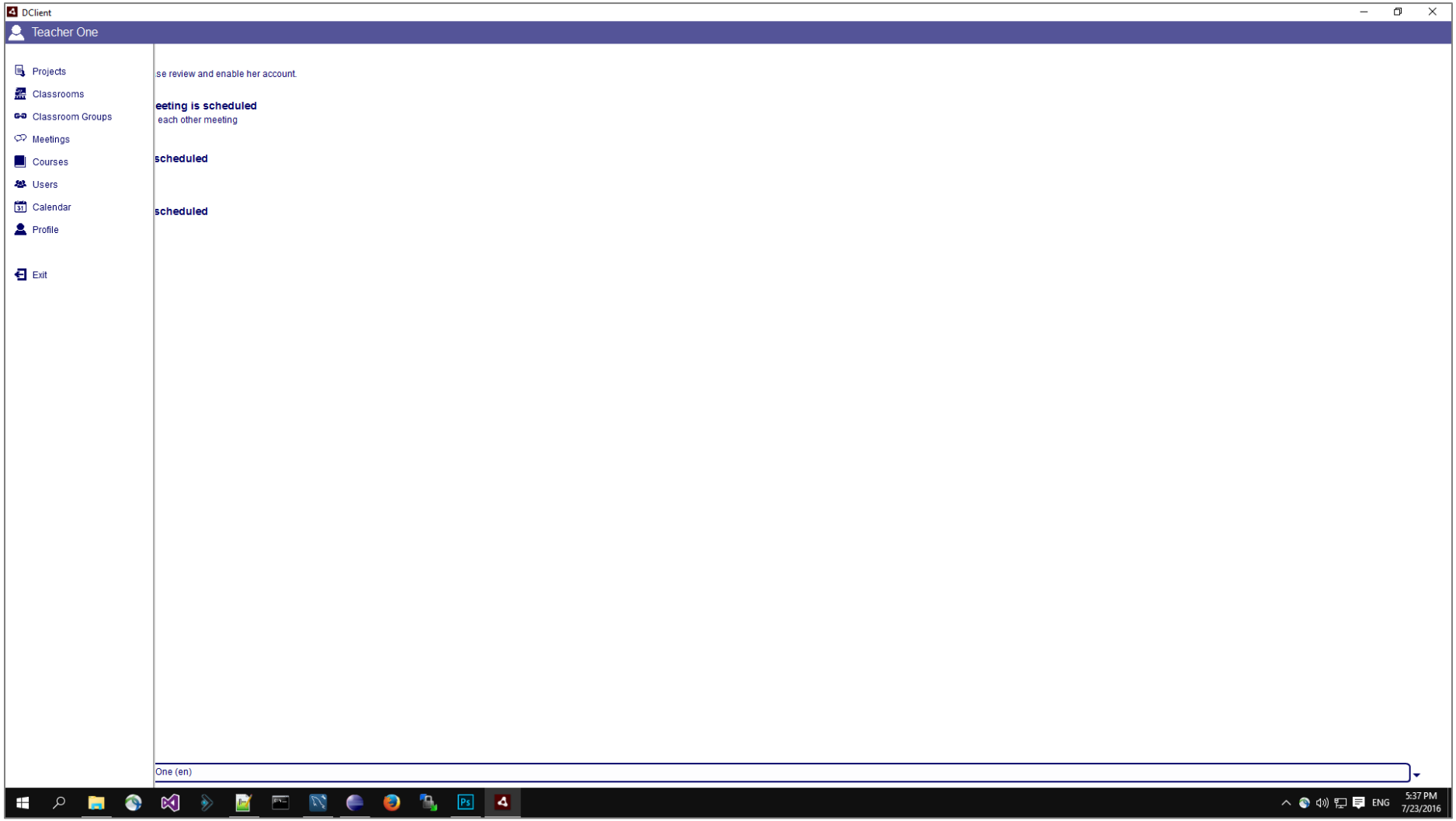

73°
85%
10:41 PM
Name
John Salatas
Email
jjsalatas@gmail.com
I am
Student
User Name
jsalatas
Password
•••••
Confirm Password
•••••
Teacher's Name
Teacher One
OK
Cancel

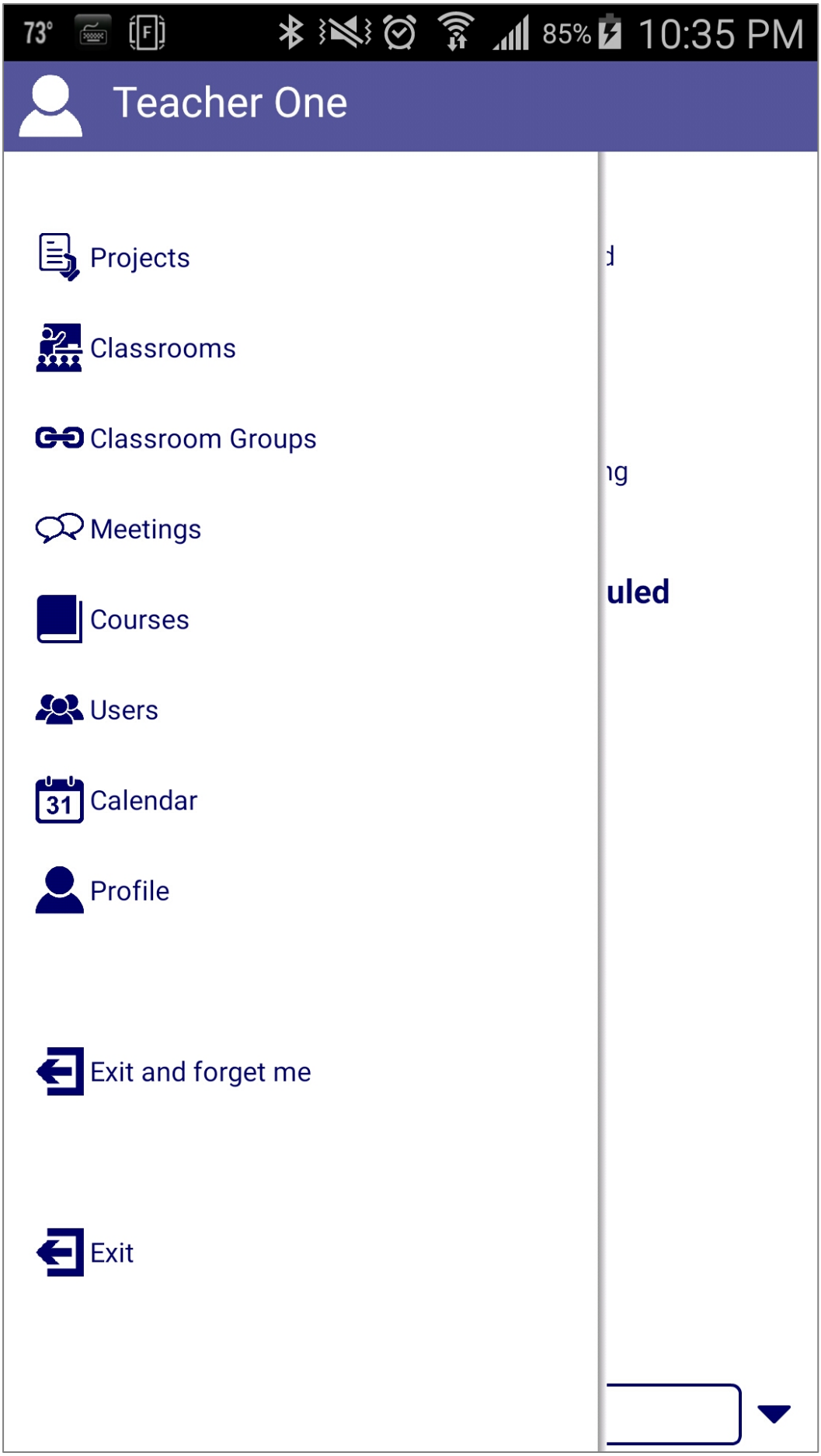

DClient
Teacher One
New Student
A new student has registered. Please review and enable her account.
Thusrday
28
July
An Online Meeting is scheduled
Getting to know each other meeting
Saturday
23
July
An Event is scheduled
A1 Class
Wednesday
27
July
An Event is scheduled
A1 Class
Classroom
A1 Class of Teacher One (en)
ENG
5:36 PM
7/23/2016

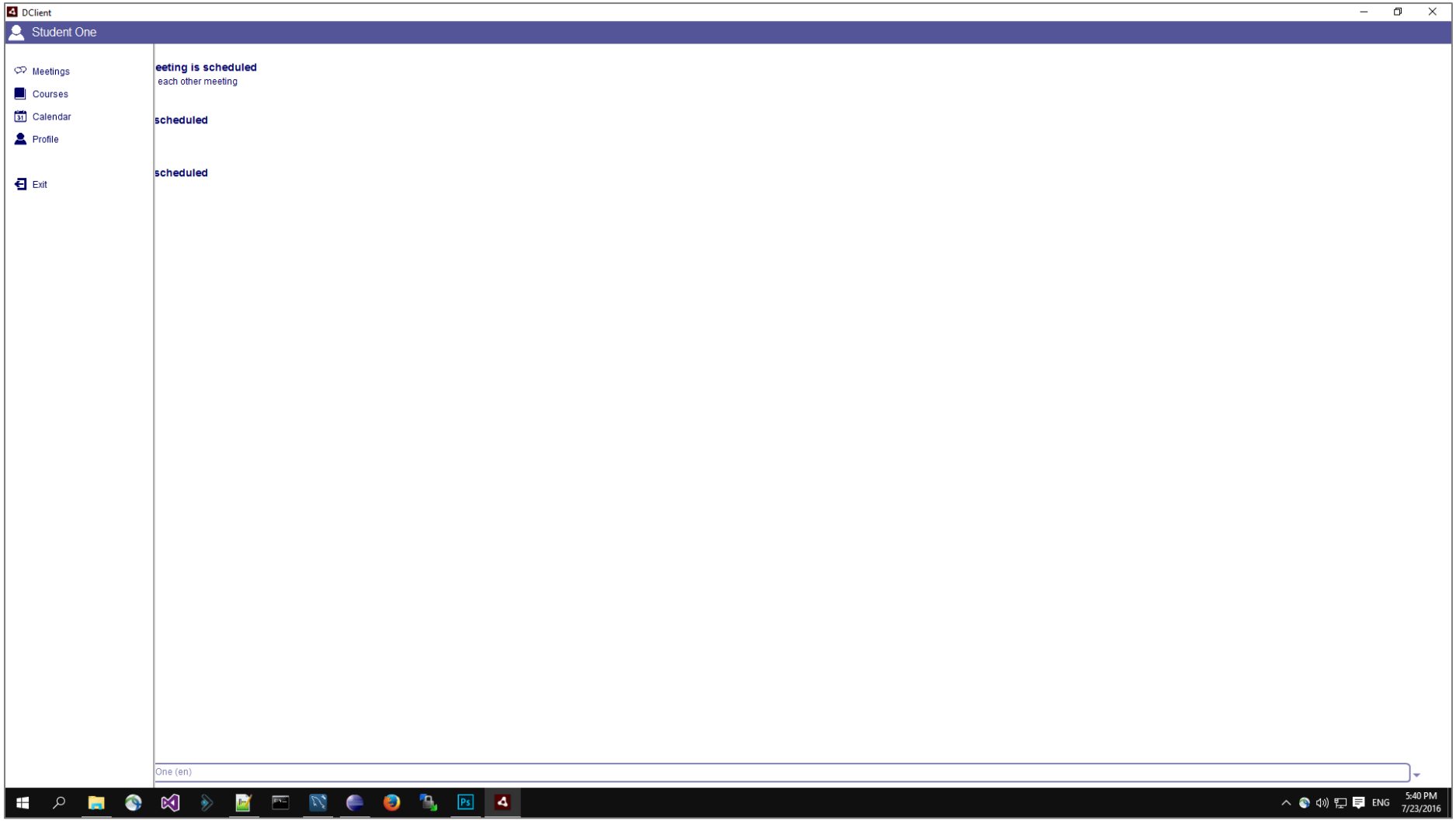

73°
82%
10:13 PM
Student One
Thusrday
28
July
An Online Meeting is scheduled
Getting to know each other meeting
Saturday
23
July
An Online Meeting is in progress
Game Time!
Wednesday
27
July
An Event is scheduled
A1 Class
Classroom
A1 Class of Teacher One (en)

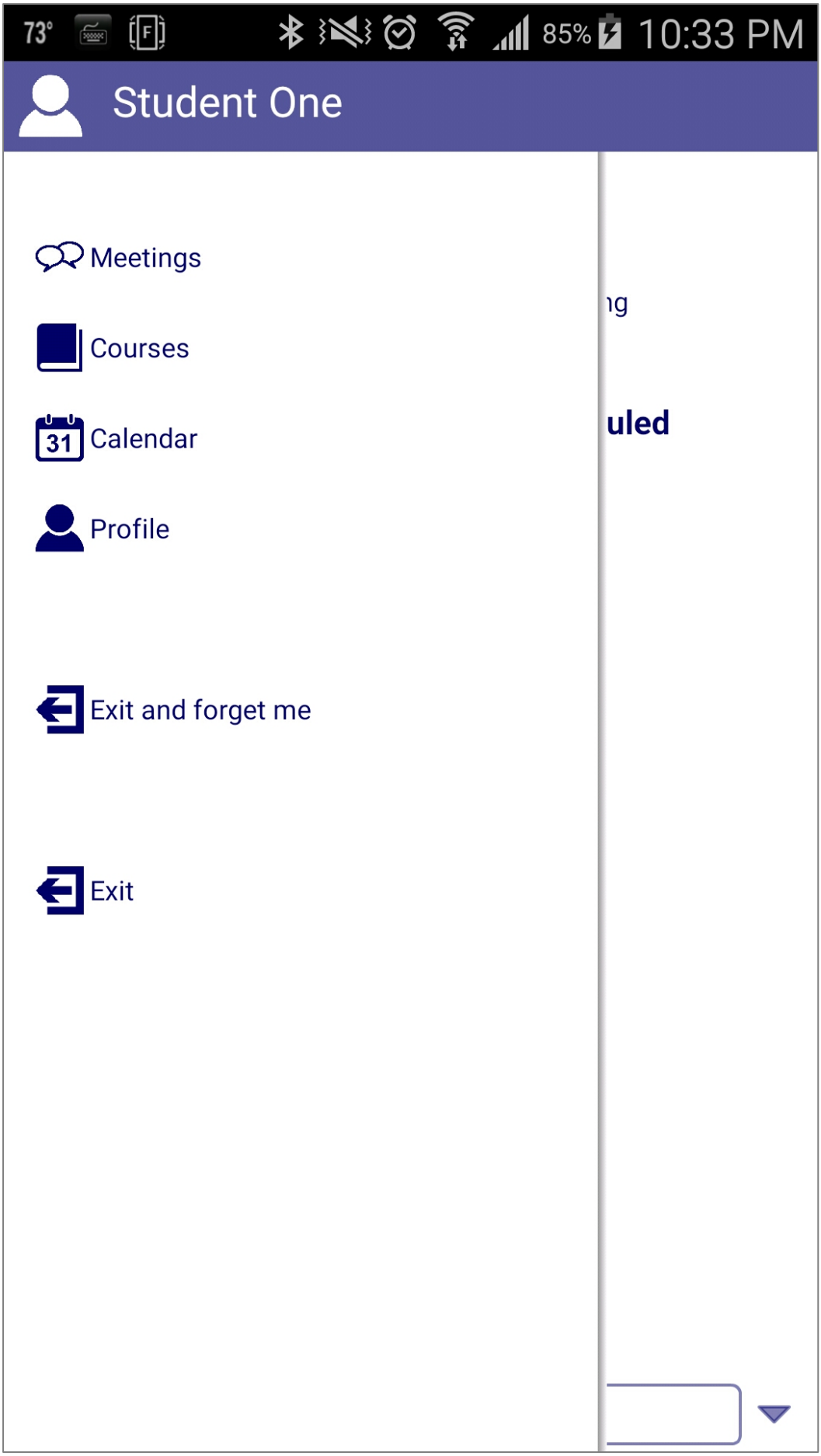

DClient
admin
Server Settings
Languages
Course Templates
Courses
Projects
Classrooms
Classroom Groups
Meetings
Users
Calendar
Profile
Exit
ENG
5:23 PM
7/23/2016

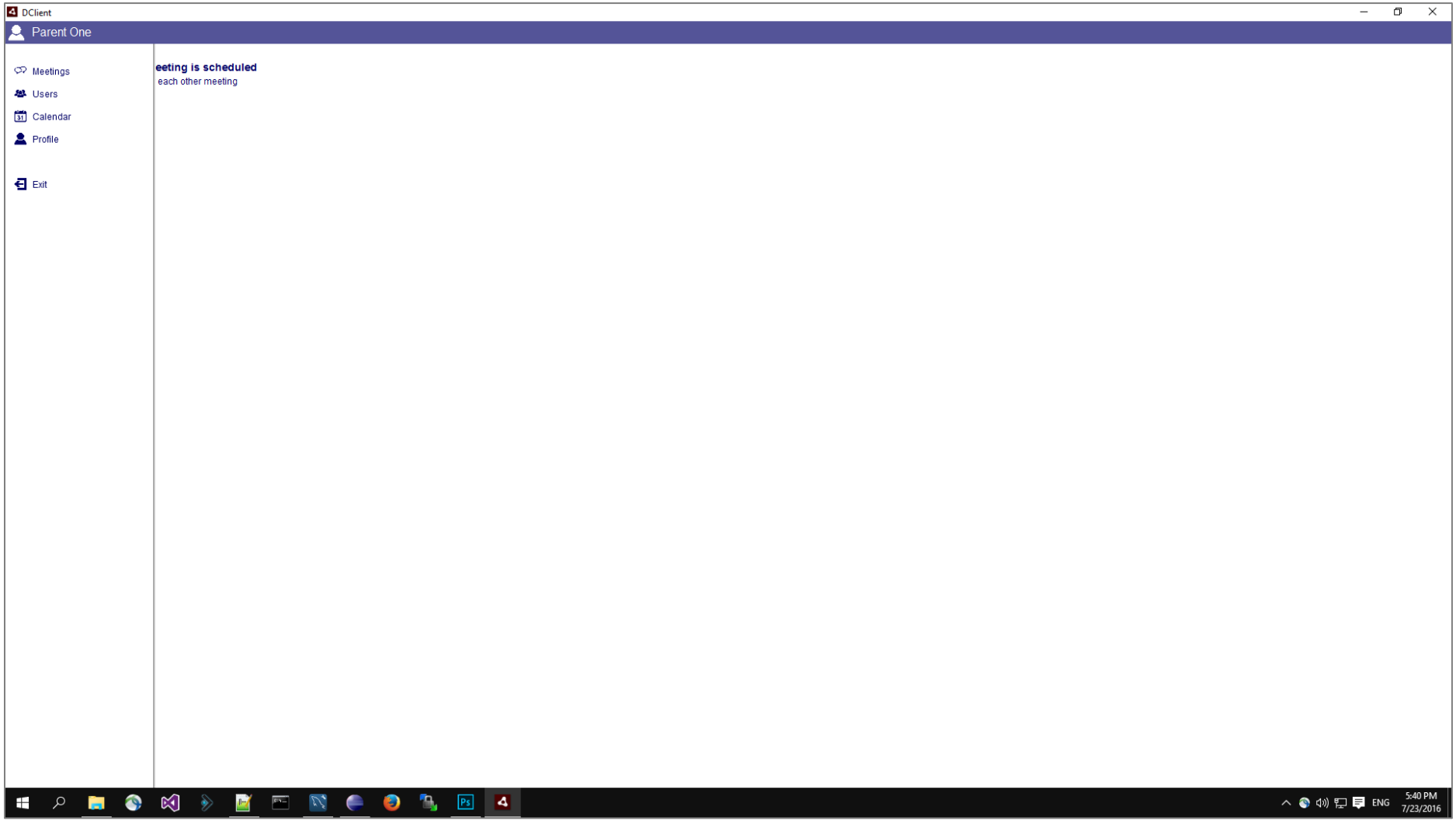

71°
86%
10:45 PM
admin
Course Templates
Courses
Projects
Classrooms
Classroom Groups
Meetings
Users
Calendar
Profile
Exit and forget me
Exit

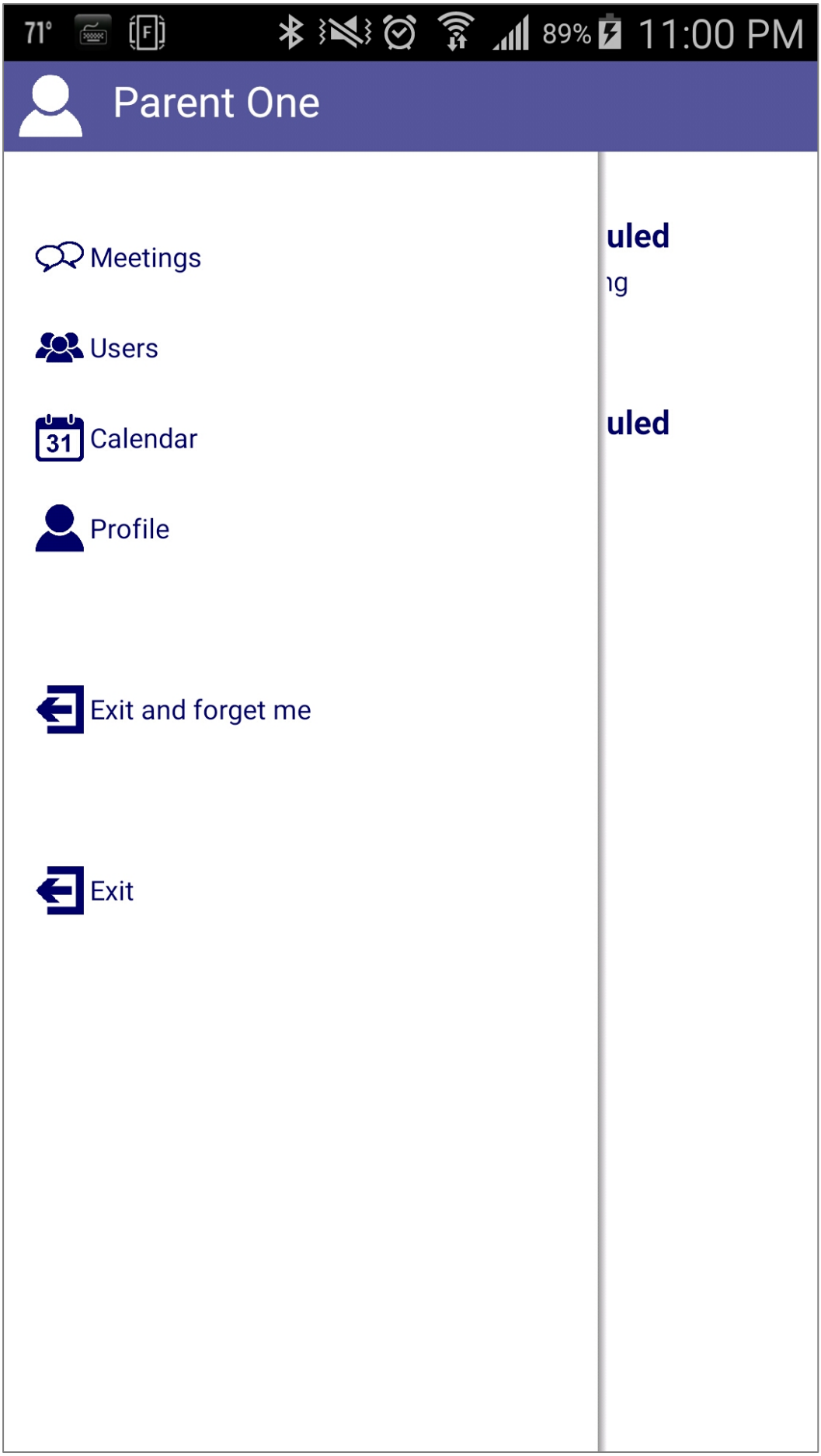

DClient
Teacher One
Projects
Classrooms
Classroom Groups
Meetings
Courses
Users
Calendar
Profile
Exit
se review and enable her account.
eeting is scheduled
each other meeting
scheduled
scheduled
One (en)
ENG
5:37 PM
7/23/2016

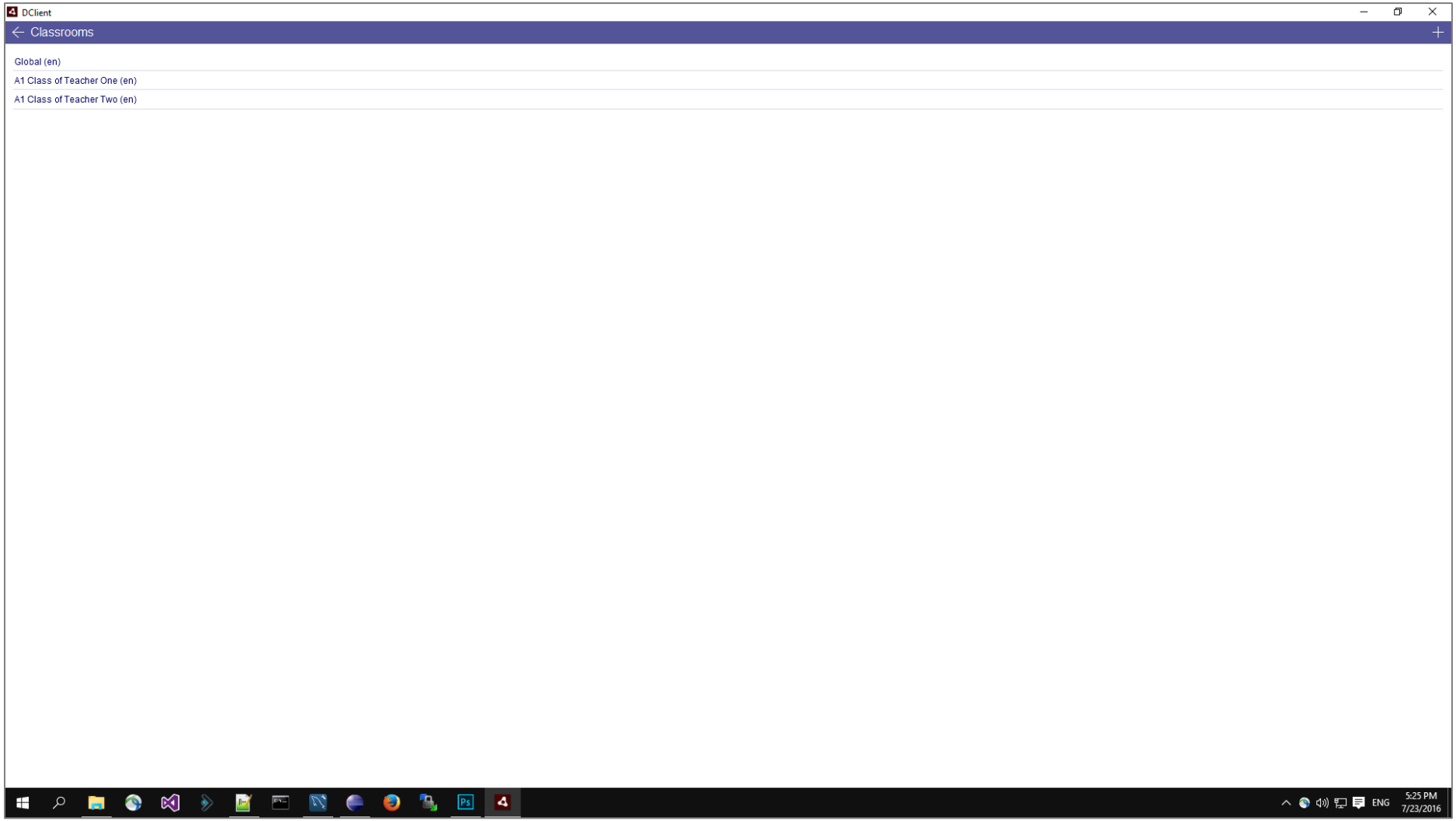

73°
85%
10:35 PM
Teacher One
Projects
Classrooms
Classroom Groups
Meetings
Courses
Users
Calendar
Profile
Exit and forget me
Exit

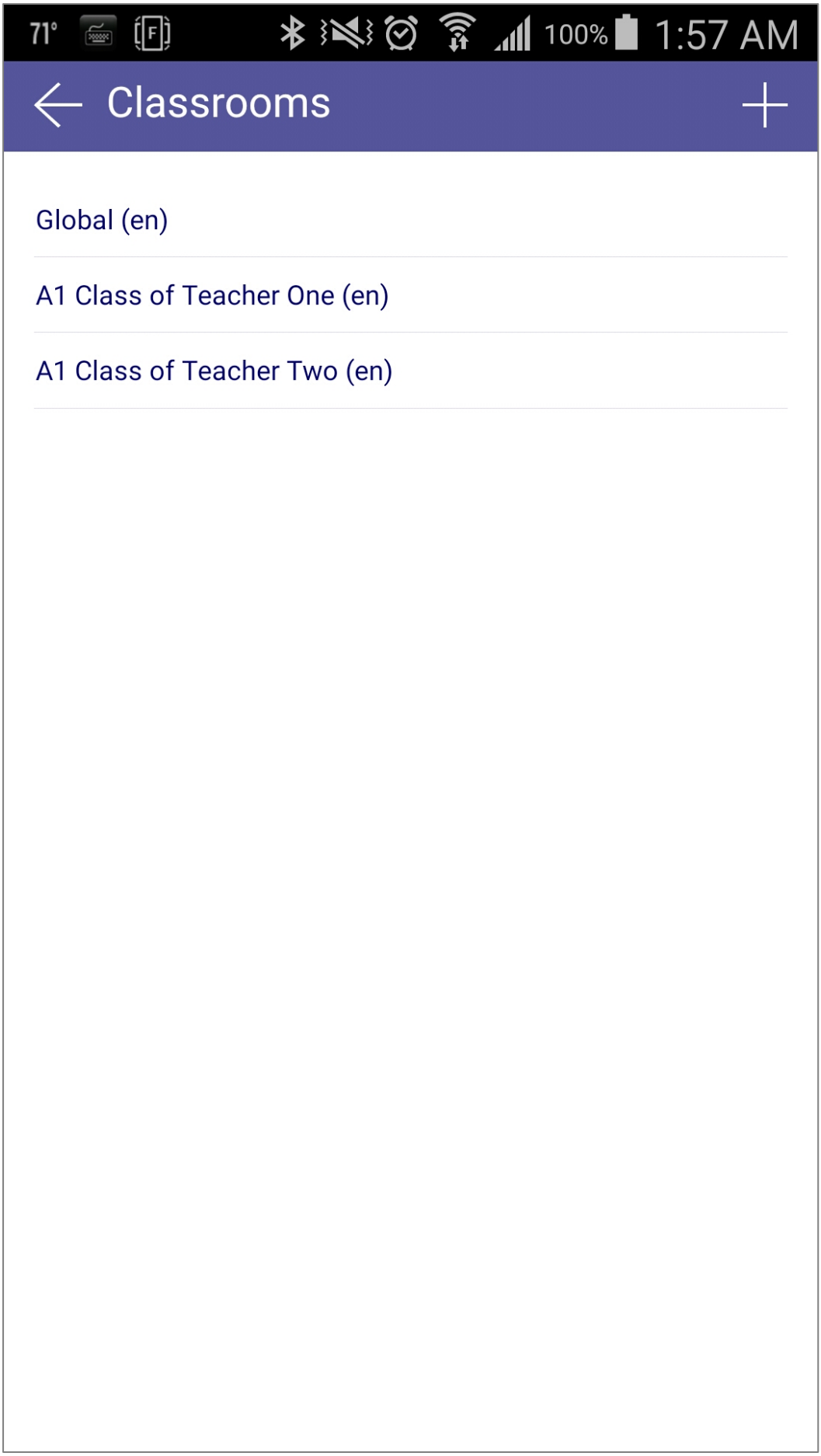

DClient
Student One
Meetings
Courses
Calendar
Profile
Exit
eeting is scheduled
each other meeting
scheduled
scheduled
One (en)
ENG
5:40 PM
7/23/2016

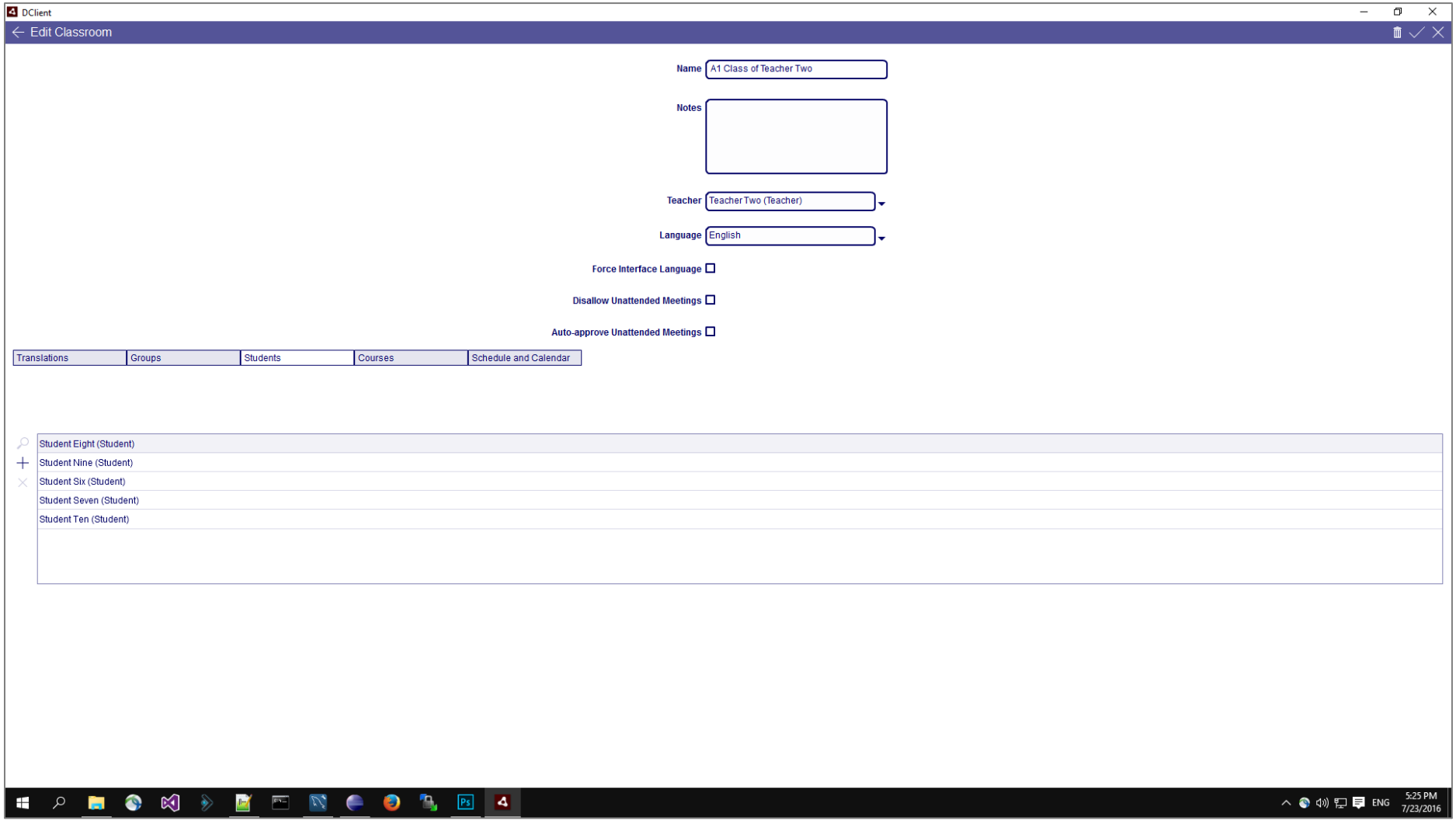

73°
85%
10:33 PM
Student One
Meetings
Courses
Calendar
31
Profile
Exit and forget me
Exit
ıg
uled

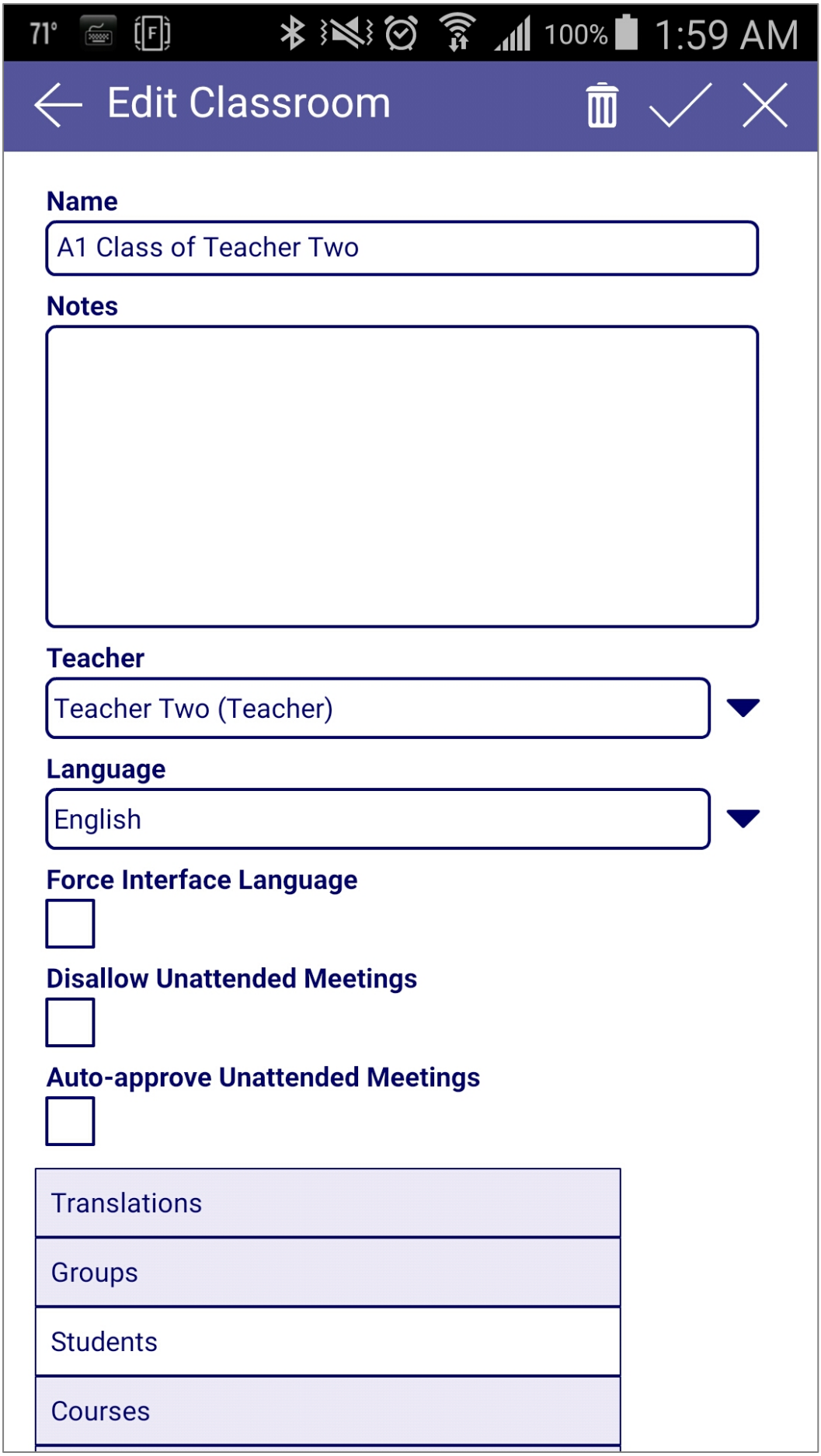

DClient
Parent One
Meetings
Users
Calendar
Profile
Exit
eeting is scheduled
each other meeting
ENG
5:40 PM
7/23/2016

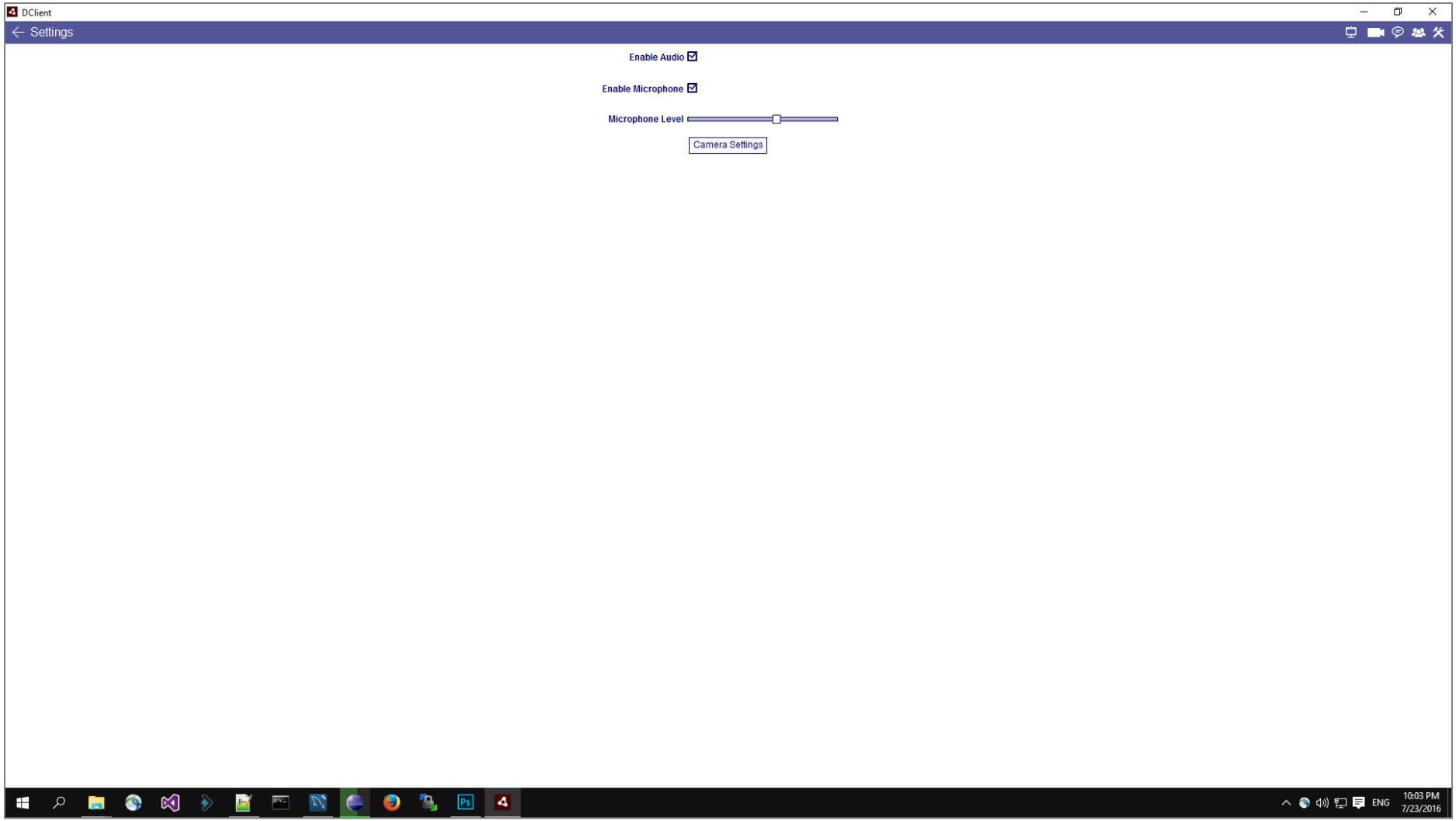

71°
89%
11:00 PM
Parent One
Meetings
Users
Calendar
Profile
Exit and forget me
Exit
uled
ıg
uled

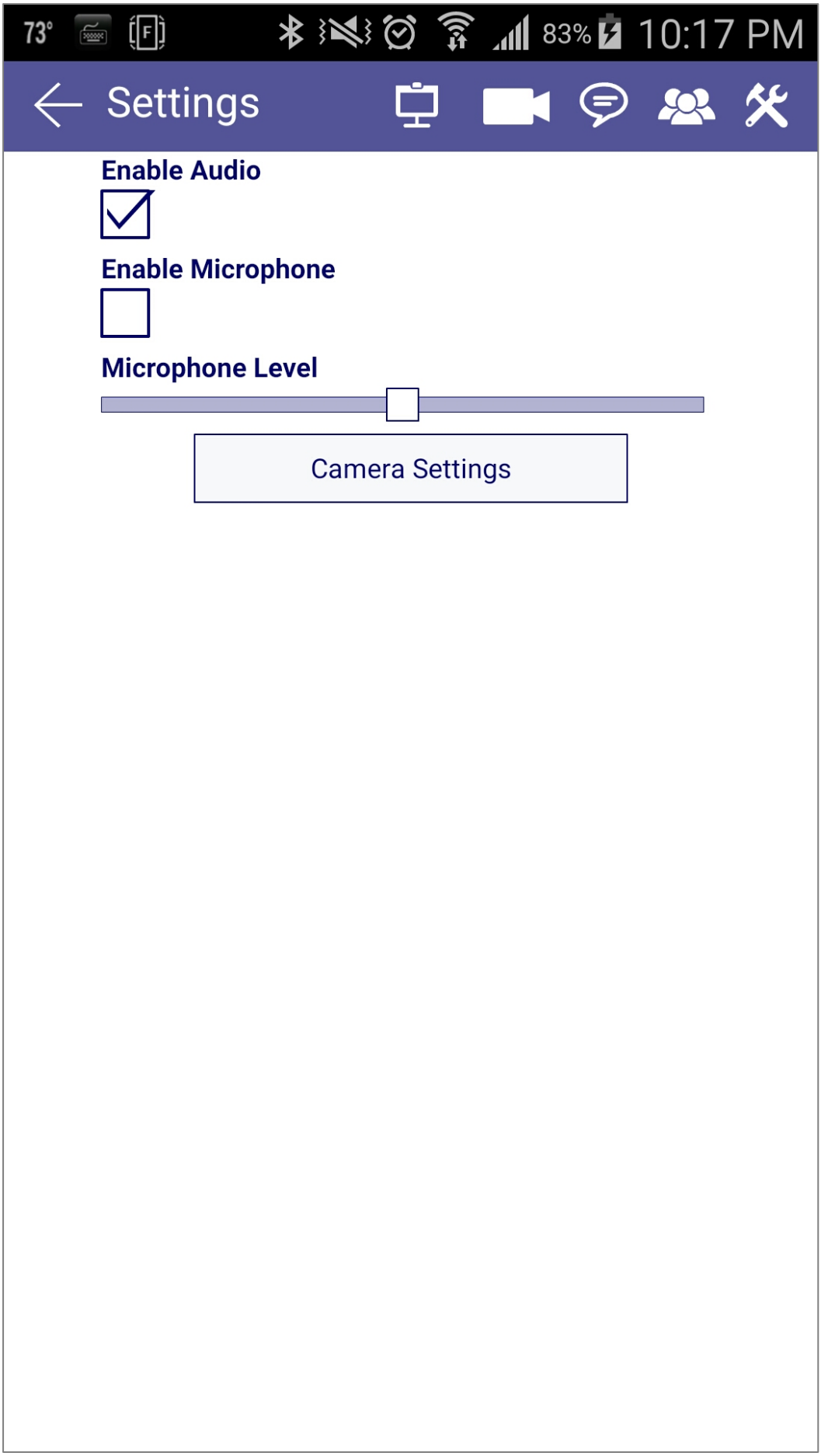

DClient
Classrooms
Global (en)
A1 Class of Teacher One (en)
A1 Class of Teacher Two (en)
ENG
5:25 PM
7/23/2016

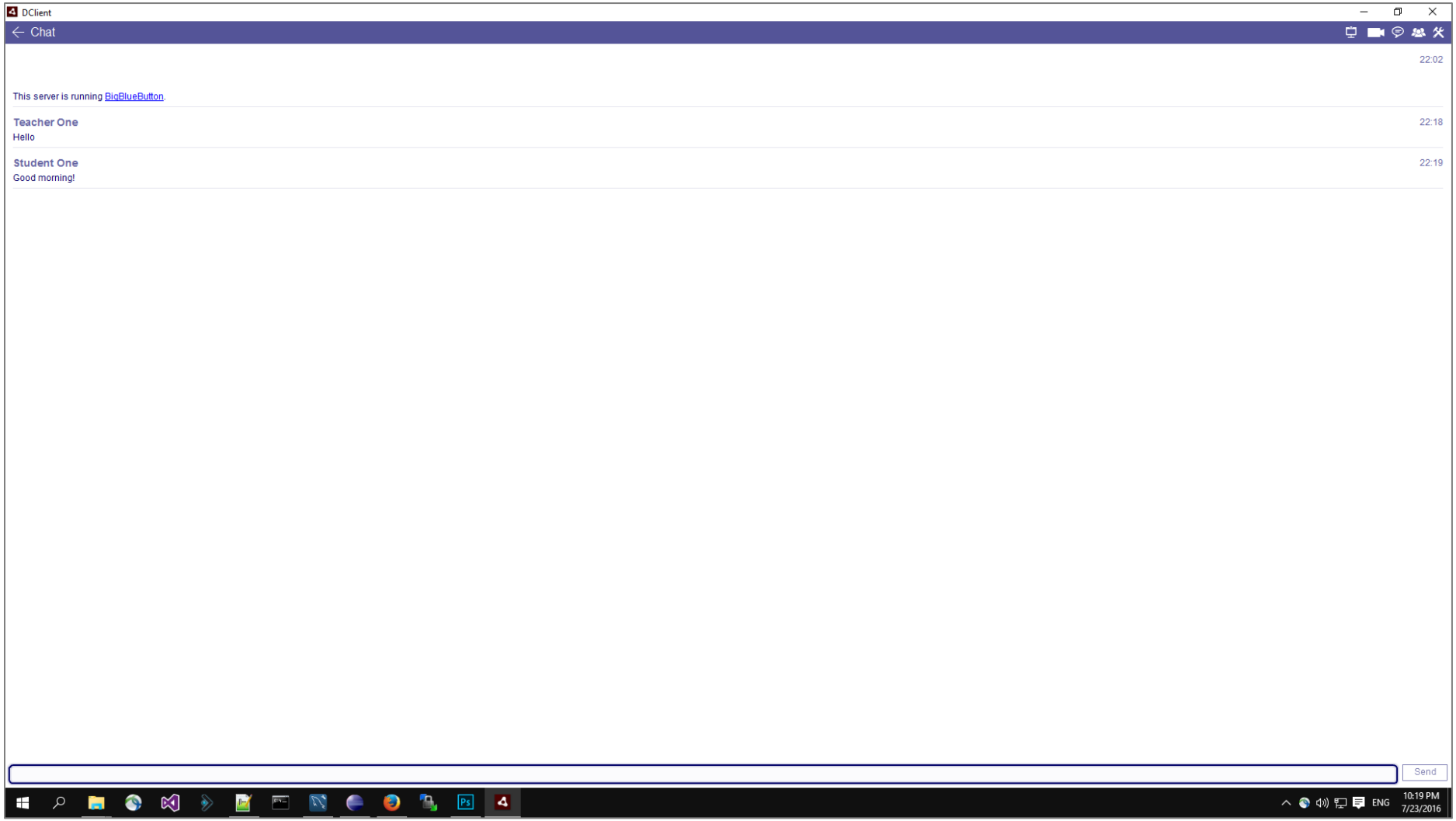

71°
100%
1:57 AM
Classrooms
Global (en)
A1 Class of Teacher One (en)
A1 Class of Teacher Two (en)

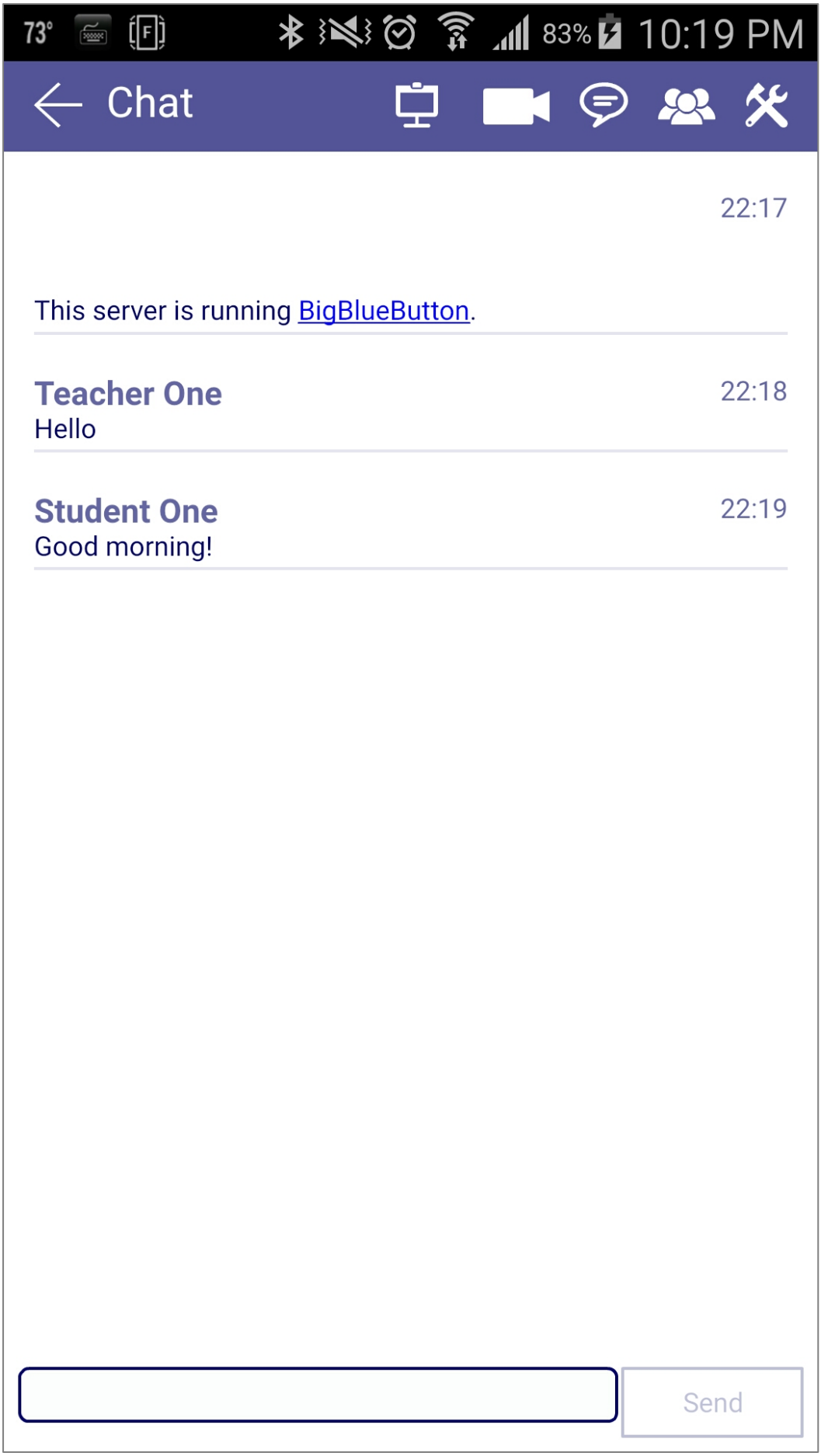

DClient
Edit Classroom
Name
A1 Class of Teacher Two
Notes
Teacher
Teacher Two (Teacher)
Language
English
Force Interface Language
Disallow Unattended Meetings
Auto-approve Unattended Meetings
Translations
Groups
Students
Courses
Schedule and Calendar
Student Eight (Student)
Student Nine (Student)
Student Six (Student)
Student Seven (Student)
Student Ten (Student)
ENG
5:25 PM
7/23/2016

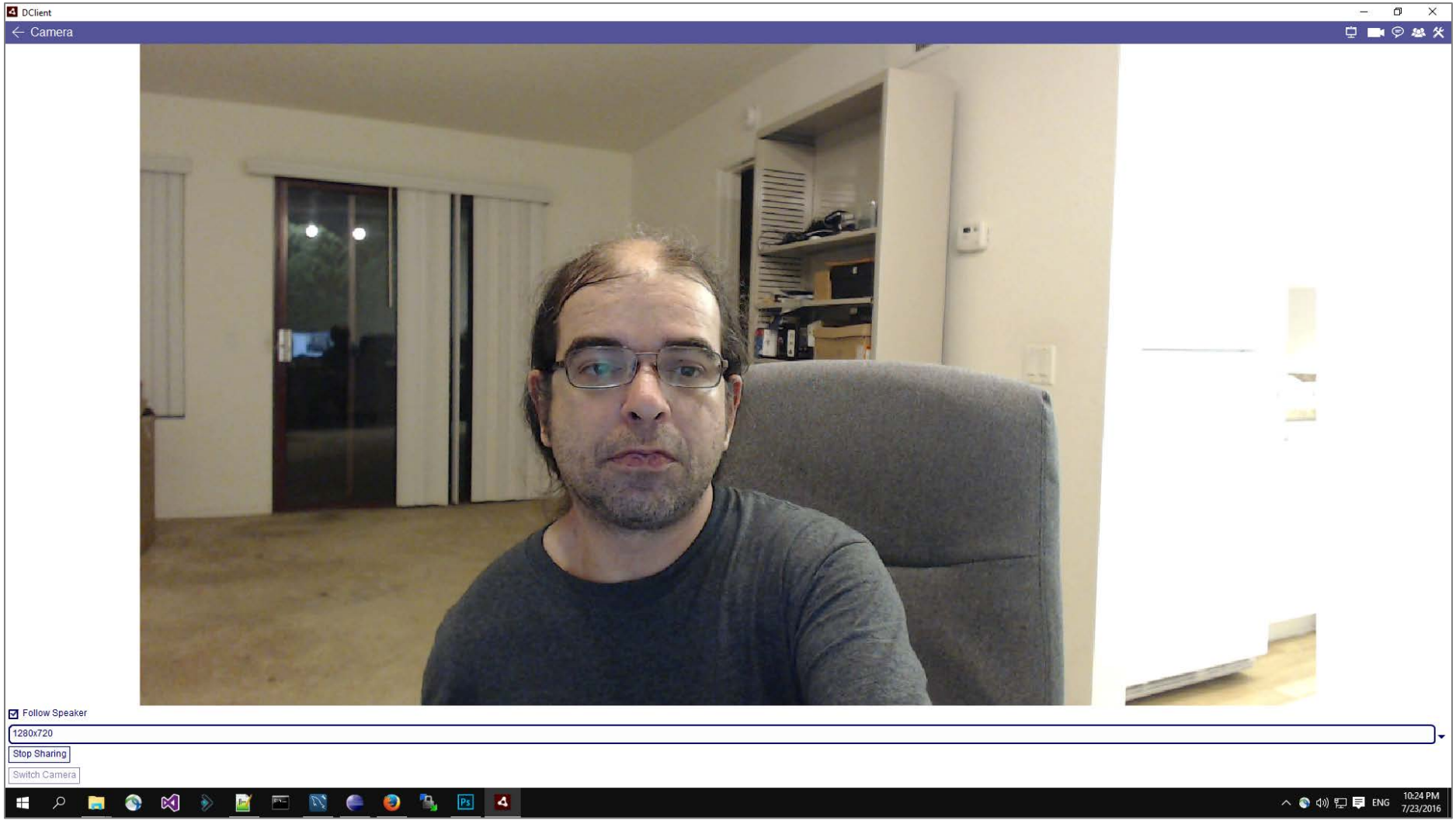

71°
100%
1:59 AM
Edit Classroom
Name
A1 Class of Teacher Two
Notes
Teacher
Teacher Two (Teacher)
Language
English
Force Interface Language
Disallow Unattended Meetings
Auto-approve Unattended Meetings
Translations
Groups
Students
Courses

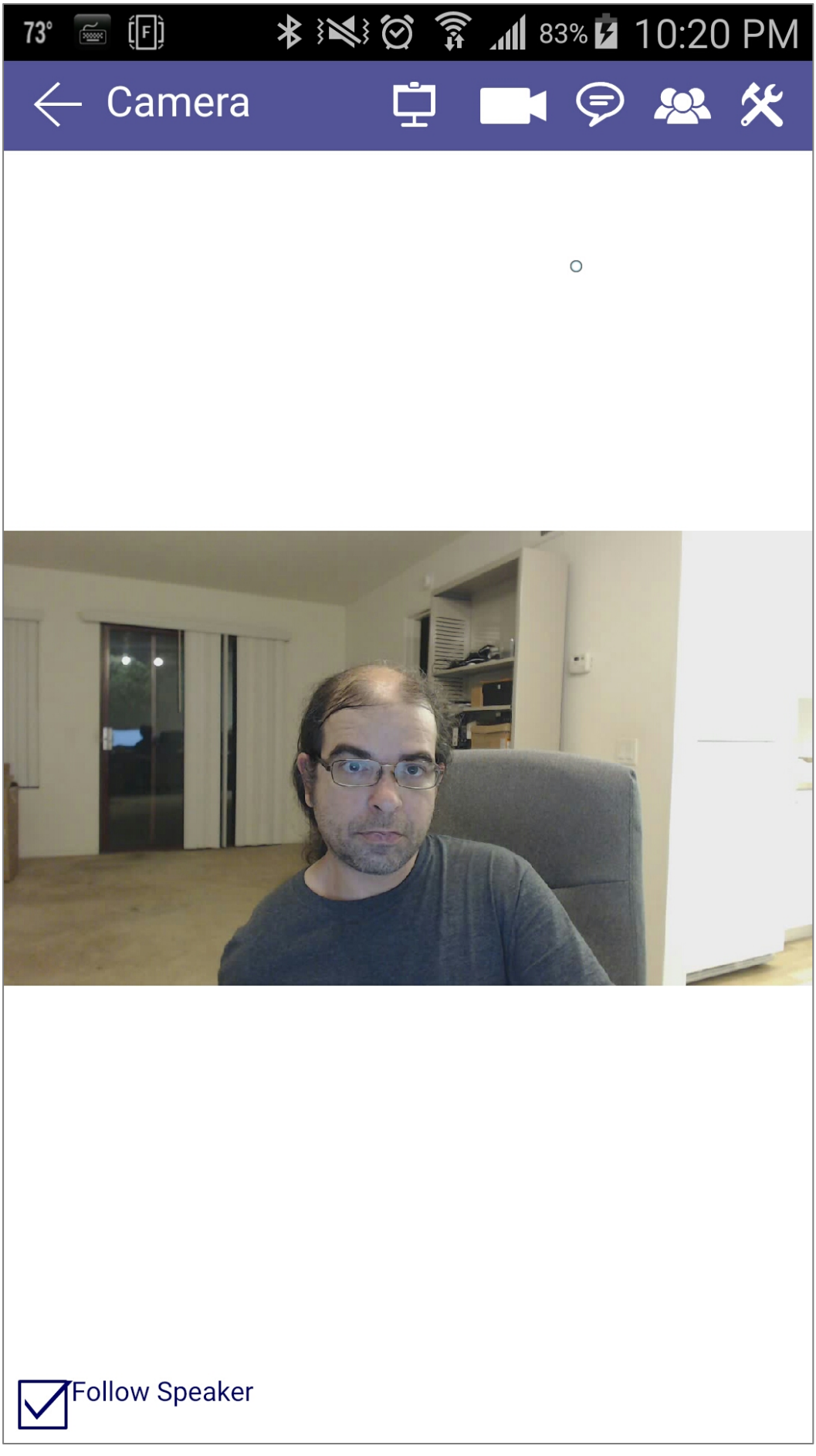

DClient
Settings
Enable Audio
Enable Microphone
Microphone Level
Camera Settings
ENG
10:03 PM
7/23/2016

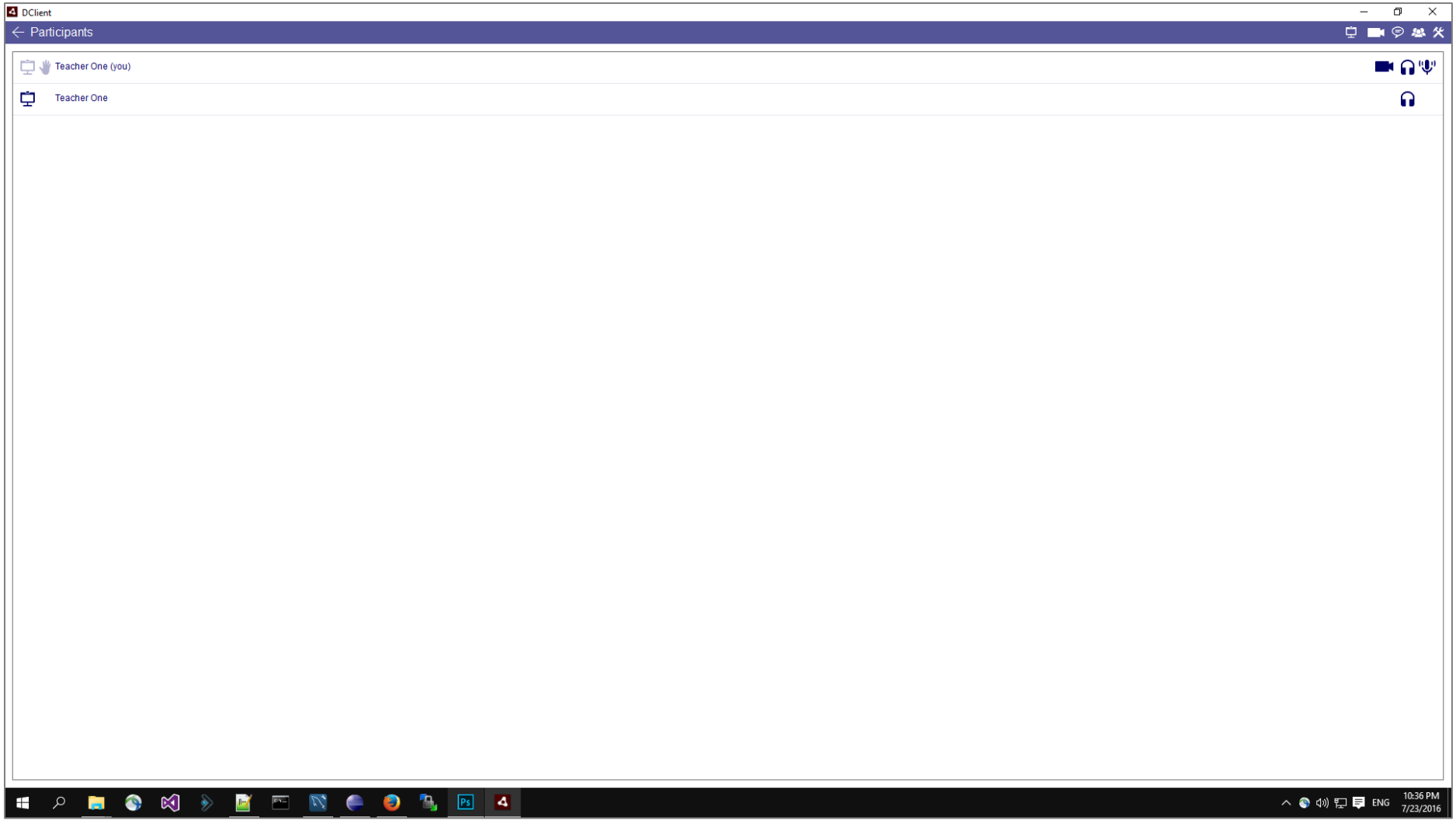

73°
83%
10:17 PM
Settings
Enable Audio
Enable Microphone
Microphone Level
Camera Settings

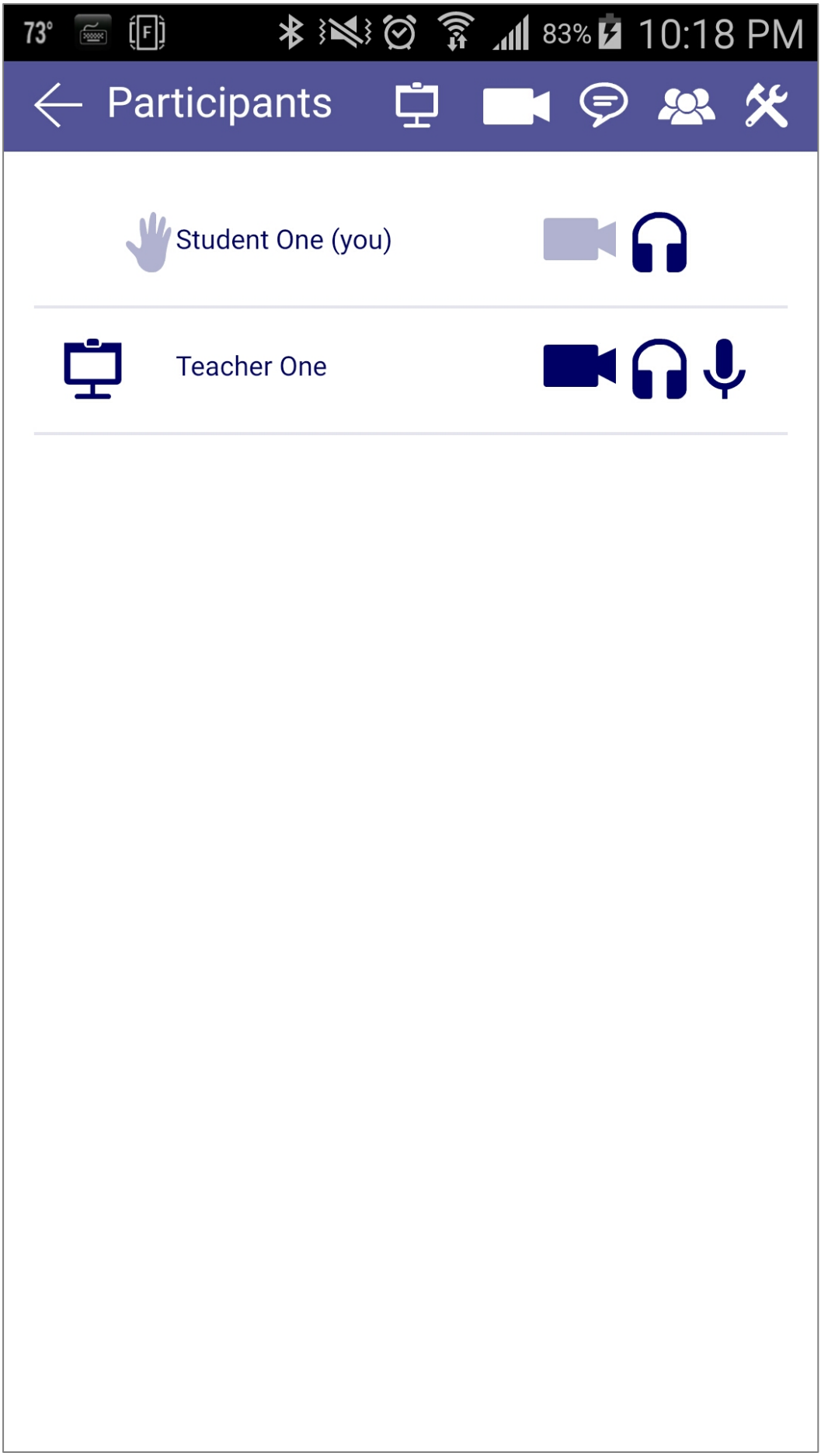

DClient
Chat
22:02
This server is running BigBlueButton.
Teacher One
22:18
Hello
Student One
22:19
Good morning!
Send
ENG
10:19 PM
7/23/2016

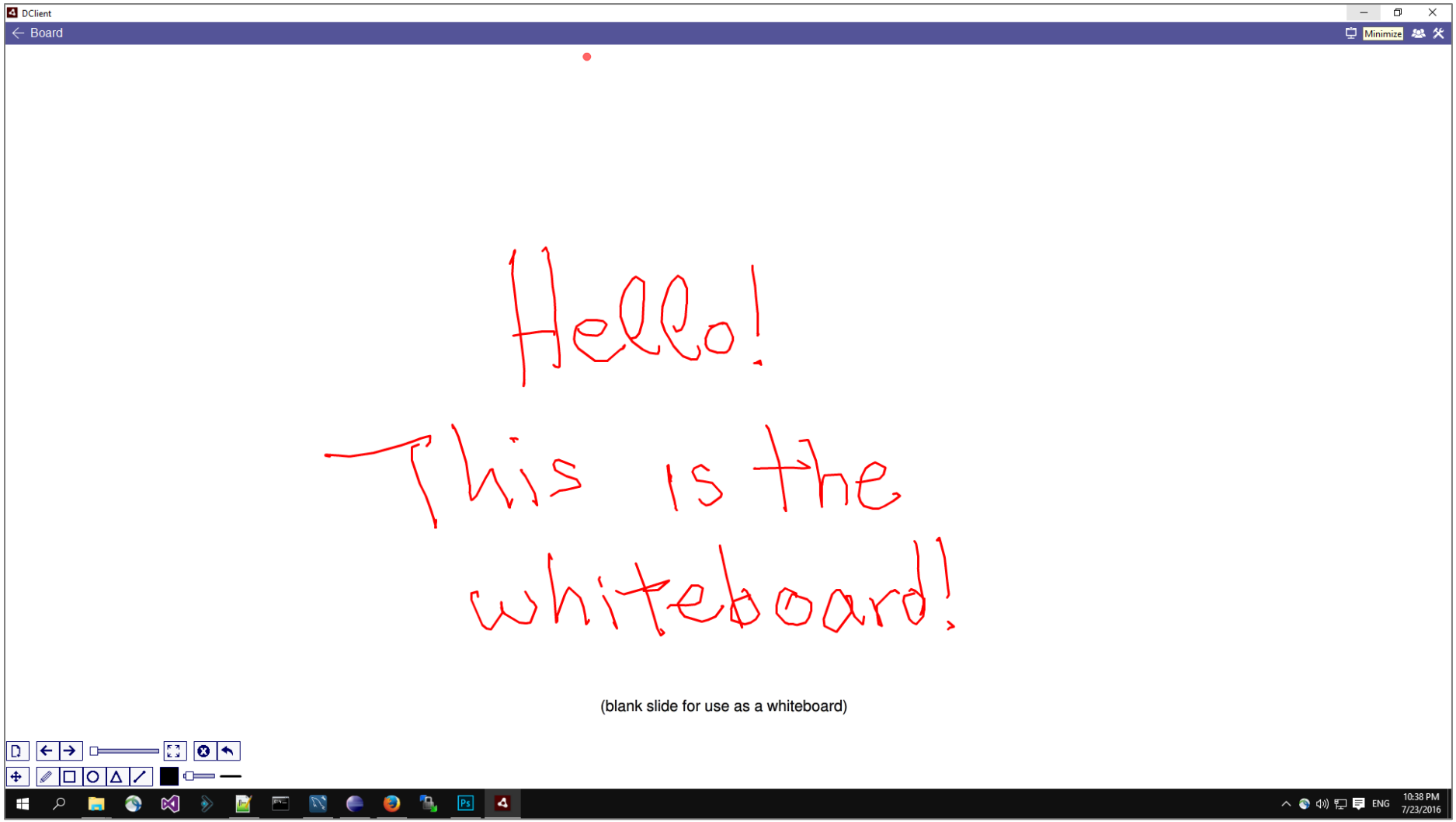

73°
83%
10:19 PM
Chat
22:17
This server is running BigBlueButton.
Teacher One
22:18
Hello
Student One
22:19
Good morning!
Send

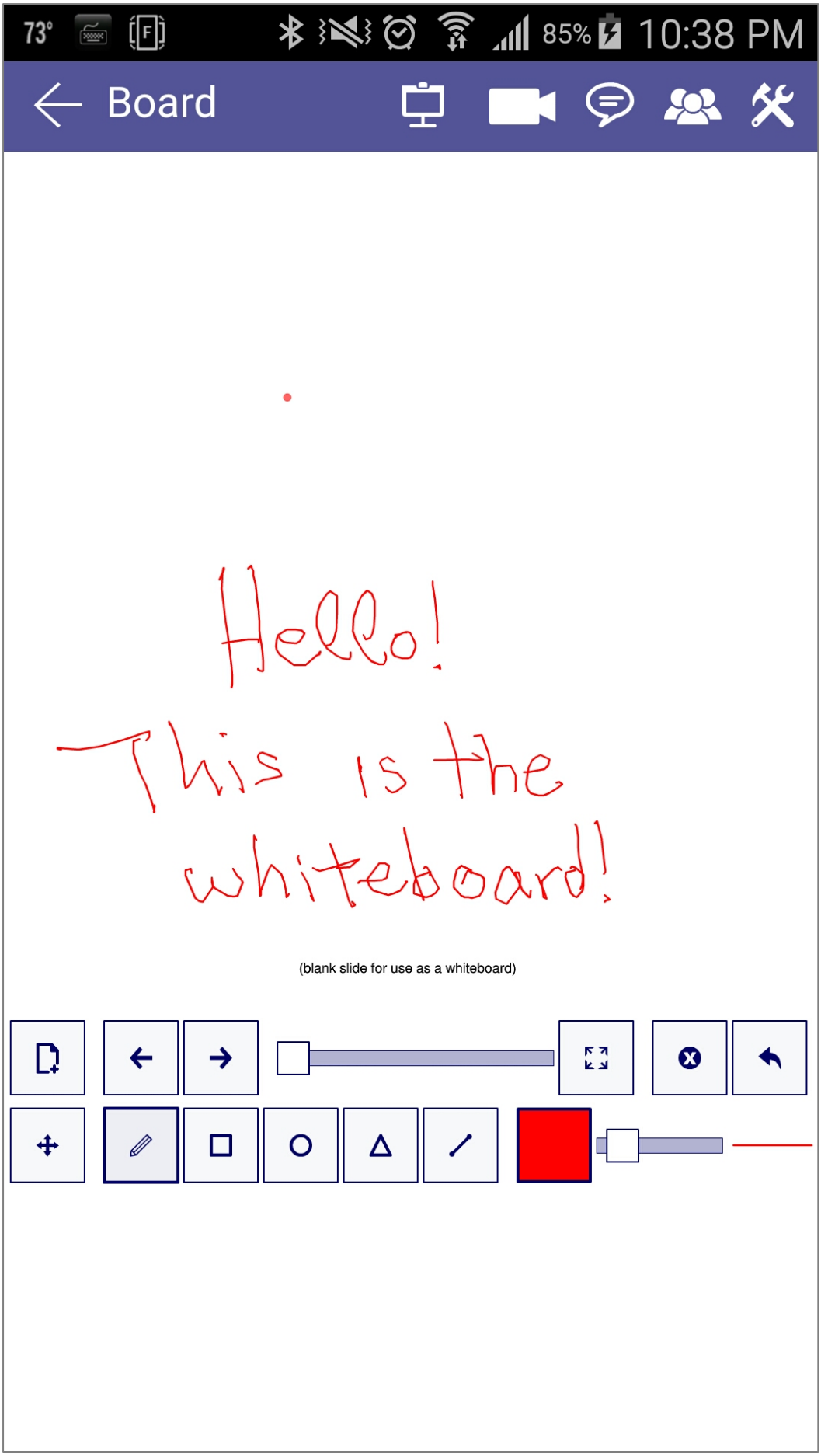

DClient
Camera
Follow Speaker
1280x720
Stop Sharing
Switch Camera
ENG
10:24 PM
7/23/2016

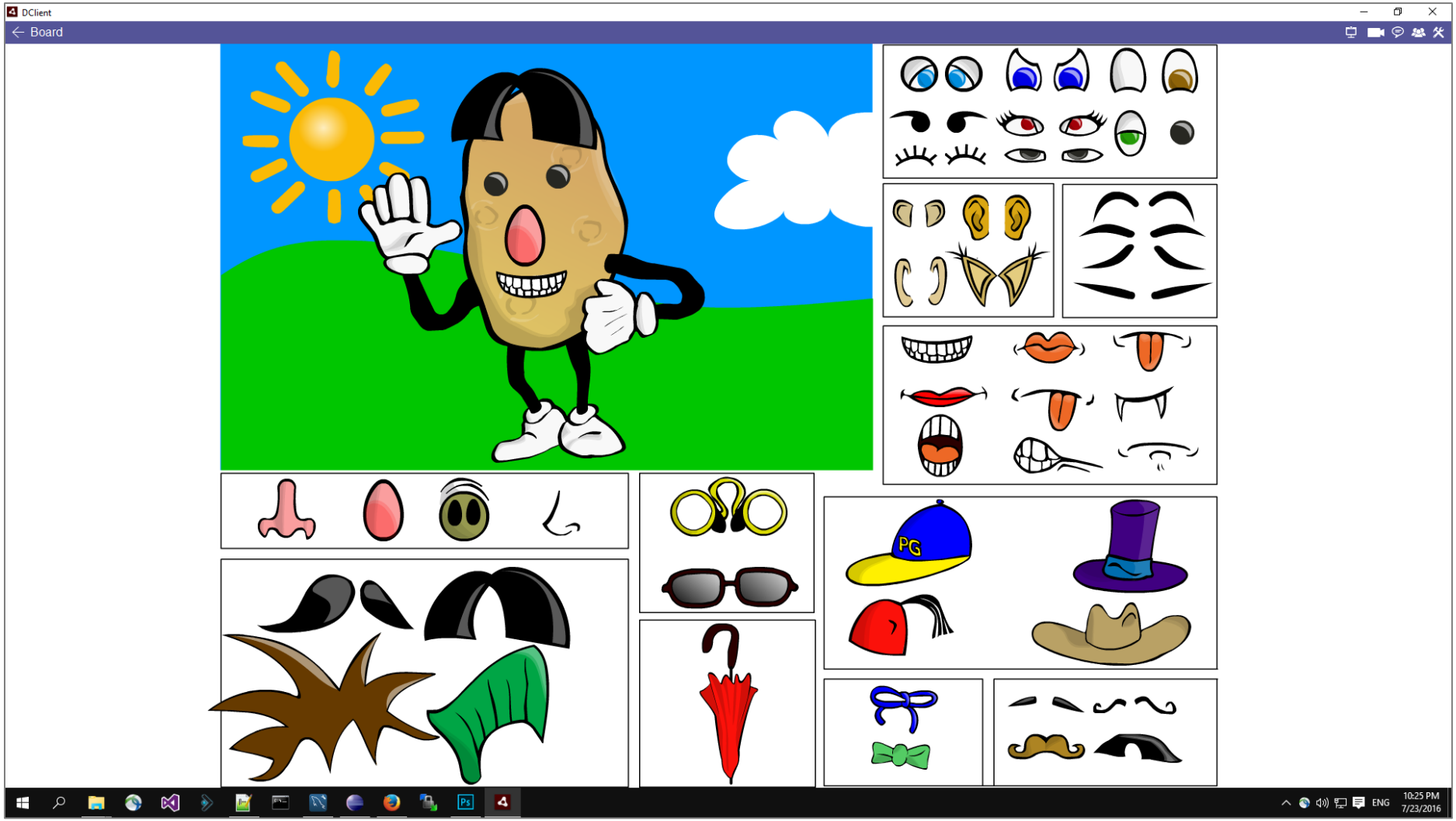

73°
83%
10:20 PM
Camera
Follow Speaker

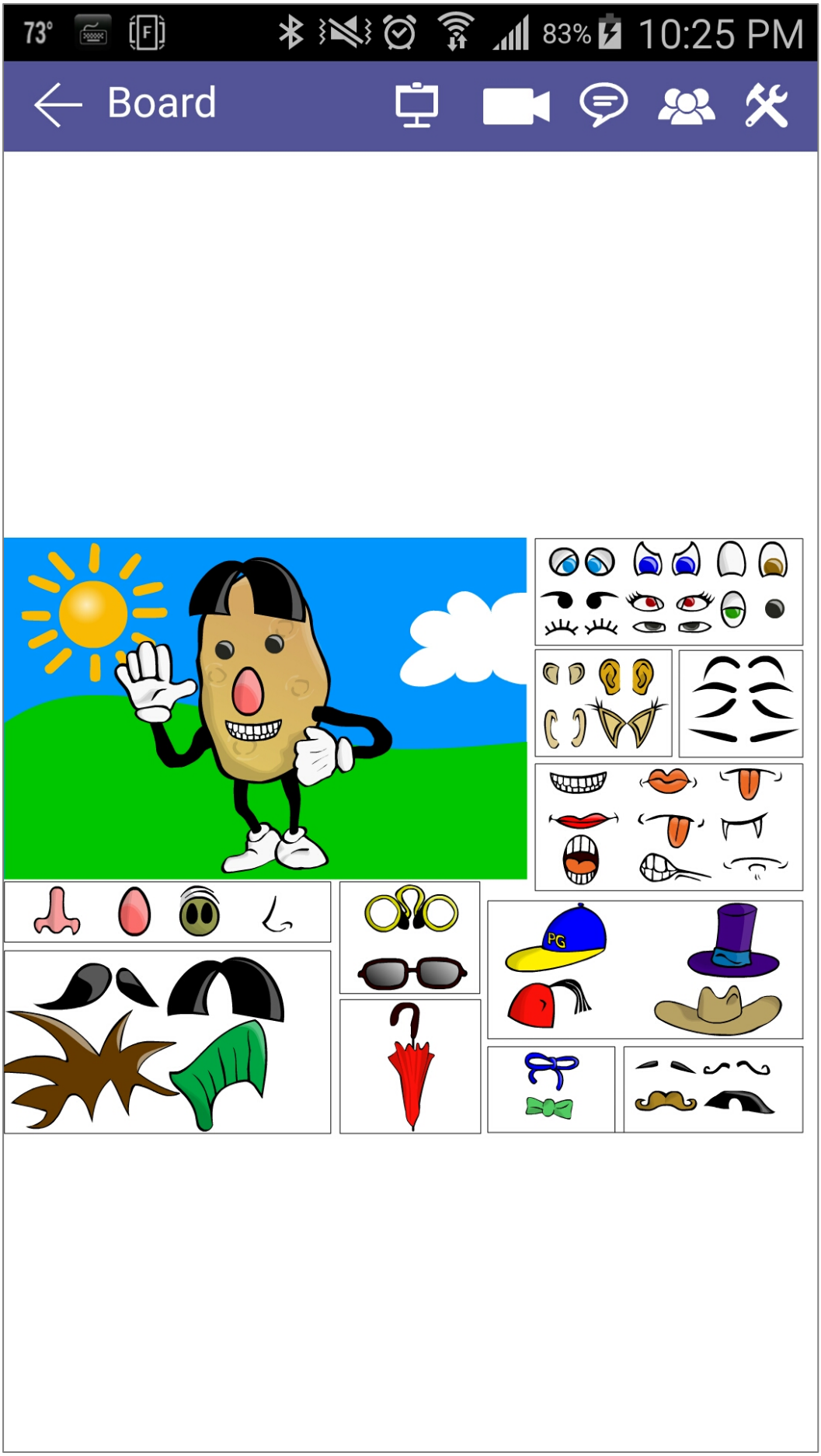

DClient
Participants
Teacher One (you)
Teacher One
ENG
10:36 PM
7/23/2016


73°
83%
10:18 PM
Participants
Student One (you)
Teacher One


DClient
Board
Minimize
Hello!
This is the whiteboard!
(blank slide for use as a whiteboard)
ENG
10:38 PM
7/23/2016


73°
85%
10:38 PM
Board
Hello!
This is the whiteboard!
(blank slide for use as a whiteboard)


DClient
Board
PG
ENG
10:25 PM
7/23/2016


73°
83%
10:25 PM
Board
PG

ΕΛΛΗΝΙΚΟ ΑΝΟΙΚΤΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ

HELLENIC OPEN UNIVERSITY
SCHOOL OF SCIENCES AND TECHNOLOGY

WWW.EAP.GR