

nuKIT – an Interactive Communication Tool via Smartphone Technologies

Friederike Pfeiffer-Bohnen, Fabian Kern, Lukas König, Hartmut Schmeck
Institute for Applied Informatics and Formal Description Methods
Karlsruhe Institute of Technology (KIT)
Germany

{friederike.pfeiffer@, fabian.kern@student., lukas.koenig@, hartmut.schmeck@}kit.edu

Abstract: The number of internet-ready mobile devices is steadily increasing. Today these devices are particularly well accepted among young people who build the main part of the audience in university courses. Using smart devices offers a great possibility to increase student-teacher interactions. In this paper, we present an e-learning tool called nuKIT (networked university KIT) which provides interfaces to smartphones and the web for establishing several types of interactions between a lecturer and the students. The tool offers students the possibility to give feedback regarding the presentation speed and to ask questions. As well, the lecturer can pose questions to the students. nuKIT is a lightweight tool which is particularly well-suited for use in academic courses, but also in related scenarios. We illustrate the technological background of nuKIT, give implementation and usage recommendations and present an evaluation of the tool originating from data collected in three courses at KIT.

Introduction

Recently, e-learning is becoming more and more popular especially for “brick and mortar” universities and in higher education (Shivett 2011, Rovai 2004). However, the usage of e-learning techniques can lead to complex and severe tasks which result in faculty members avoiding the usage of e-learning. Thus, an e-learning tool has to be systematically designed (Garrison 2004). Such an e-learning tool can be used to overcome the lack of communication and interactivity between teacher and audience in academic courses.

According to Ramsden (1992), communication and interactivity in teaching is important for the students’ understanding and learning process and overcomes the uni-directional teaching with passive students. However, especially in large courses, many reasons can be found that counteract the verbal communication. Some students may be too shy to ask questions or participate in the lecture; other students are kept from participation due to spatial distance from the lecturer, background noise or monotony of the lecturer’s presentation which may cause inattention, passiveness and decreased motivation. Such courses are especially known to affect students’ grades negatively when exceeding a threshold of about 100 students per class (Bandiera 2010b). The growth of modern universities in number and size in the last decade due to a world-wide increase in enrollment numbers (Bandiera 2010a) heightens this problem. An e-learning approach increasing the interaction and communication in academic classes and improving the individual learner support would counteract these negative effects on large courses.

In this paper we present the e-learning tool nuKIT program collection (*networked university KIT*) which is lightweight software for establishing communication between lecturer and students during the lecture. This software is platform-independent and relies on the usage of mobile devices on the student side. As new technologies are typically common among young people - about 30% of Germans own a smartphone-, considering people under thirty this number increases to 50% (Bitkom 2012) - such approaches seem particularly promising for academic classes.

On the student side, the software includes the possibility to ask textual questions in an anonymous way (if desired) and to give feedback regarding the presentation speed. On the lecturer side, the lecturer has the option to pose questions to the students that they can answer during the lecture or within a predefined time period. These questions can be set up with single or multiple choice answers or to be answered using free text.

The nuKIT program collection itself consists of three major parts; the lecturer tools, the student interfaces and the server architecture which is responsible for the communication management. The lecturer tools are independent from the projection system the lecturer chooses. Therefore, the lecturer can easily add the nuKIT tools to his actual presentation materials without changing anything. The student interface can either be accessed by the usage of an app or by a web application. Nevertheless, the independence from a special system or hardware can be assured. As almost all students carry a smartphone, they can easily post questions during the lecture and evaluate the lecturer regarding the presentation speed. Moreover, the nuKIT tools can serve as a tool for a query independent of time and place.

The nuKIT tools in their current version have been applied so far in three different computer science courses during the current lecture period with the target audience varying from 40 to 500 students, cf. Sec. “Preliminary

Evaluation". The evaluation indicates that by this rather simple communication tool, which does not require any heavy-weight e-learning technologies and is independent from special hard- or software, the interaction in the courses has improved significantly.

Related work

As the lack of interaction in large courses is a widespread problem and new technologies such as smartphones are typically common among young people, such approaches seem particularly promising for academic classes. In the last years there have been a number of publications proposing tools and strategies regarding the use of new media in large courses (Bonn 2003, Scheele 2005, Schön 2012a, 2012b, Pfeiffer 2012). Tools whose focus is to improve the interaction in lectures, vary greatly in the software they are based-on and related to, the hardware they use and the functionality they provide. Some of them only consider a specific part of the different interaction opportunities such as surveys or quizzes, feedback issues or are restricted to unidirectional communication. Moreover, many tools are often linked to a specific learning platform or require a particular mobile device like clicker systems which make them expensive, complicated in handling and can lead to a heavyweight communication tool. These aspects can lead to limited usage, time consuming preparation and complicate behavior.

Bonn et al. (2003) describe the *NUKATH* tools (*notebook university Karlsruhe (TH)*), which have been the predecessor of the here described nuKIT program collection. The NUKATH tools comprise the multiple choice questions, the student questioning and the voting. The specific design approach and original communication process within the client-server-architecture is described. It lacks the access through an app-technology.

Scheele et al. (2005) present a communication tool called WIL/MA (Wireless Interactive Learning at the University of Mannheim) based on a client-server-architecture for usage in the lecture. It offers a quiz, feedback and a call-in tool for student questions. The server software is based on Java which means the students may only use a java-capable end-device with the client architecture. The software is designed to be used in the lecture only.

The mobile quiz introduced by Schön et al. (2012a, 2012b) is based on the usage of students' mobile devices. Within this approach, a quiz is created with the learning platform ILIAS and is then forwarded to the students' mobile devices by dynamically creating Quick Response Codes (QR).

Goals of nuKIT

nuKIT is a collection of tools based on a simple smartphone app or a web application on the student side, to be used in academic courses as a communication platform between student and teacher during a lecture. The idea and basic design principles originated from the communication tool *NUKATH (Notebook University Karlsruhe (TH))* which was developed at the University of Karlsruhe in 2003 (Bonn 2003). Although having been used regularly and unmodified for almost 10 years in the same courses at the University of Karlsruhe and the KIT, the NUKATH platform received little interest on the student side until about 2011. Apparently due to an increased availability of mobile devices such as smart phones, laptops, or tablets in the classroom, the platform achieved a major revival in 2011. This led to a lot of online communication between students and teachers during lectures, and, in addition, to a new interest in the platform by different teachers of different courses; it also pointed out several drawbacks of the original platform. The most important ones were a lack of interfaces with new technologies such as mobile apps, and a too specific design approach making it complicated to adapt the platform to different courses and to the various requirements of different teachers. Therefore, nuKIT has been developed as an upgrade from the old platform -in particular- a mobile app-based communication tool on the student side, and enhancing several technical aspects of the platform such as security, stability and usability. Altogether, the development of nuKIT was driven by the following goals:

1. Provide a lightweight communication tool for within-lecture use which is easy to manage on the student side requiring no further equipment than a common mobile phone.
2. Provide a generic platform which is not bounded to a specific learning platform but which can be ported to different platforms and easily adapted to different courses which can take place in parallel.
3. Provide a stable and secure platform that is intuitive to set up and use.
4. Offer the students the possibility to ask and answer questions during lectures in an anonymous way.
5. Provide a tool which can be combined with arbitrary lecture presentation systems.
6. Provide lecturers with student feedback regarding the presentation speed.
7. Provide the lecturer with an easy-to-use functionality to ask single choice, multiple choice or free text questions in class which can be prepared before lecture for getting immediate response.
8. Provide a method to conduct surveys or tests in predefined time periods outside of lectures.

Structure of nuKIT

The nuKIT program collection can be divided into three major parts, (1) the tools on the *lecturer side*, (2) the

interfaces on the *student side*, and (3) the *server-side* programs for the communication management. In this section, an overview of the lecturer and the student sides is given, cf. Figure 1. The server side is described in detail in Sec. “Implementation of nuKIT”.

The overall nuKIT structure is partially derived from the former NUKATH tools. The main structural changes in comparison to the NUKATH tools include the creation of the *Student App* and a password-secured access both on the student and the lecturer side. Furthermore, additions such as parallelization of the questioning process and the password structure have been added. Detailed information regarding the original basic structure of NUKATH is given in (Bonn 2003). Since that paper was written in German, we summarize again in the following the basic structure of NUKATH as well as the additions made for nuKIT.

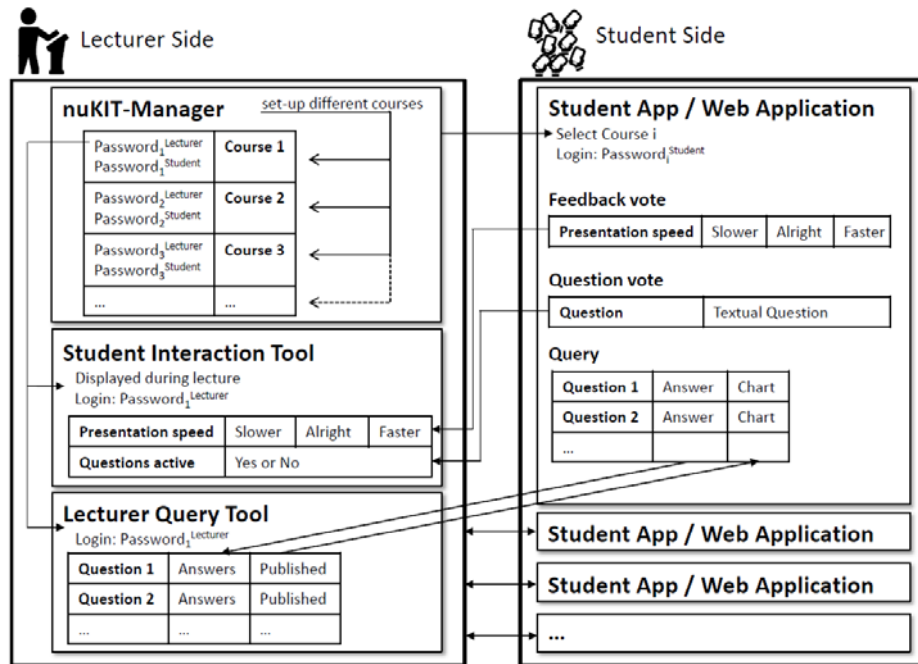


Figure 1 - Structure of nuKIT. Right side: layout of the lecturer side consisting of the nuKIT-Manager for creating courses, the Student Interaction Tool and the Lecturer Query Tool. Left side: layout of the Student side with several students using the Web Application or the App. The communication structure is illustrated by arrows.

Lecturer Side: As shown in Figure 1-right side, the lecturer side contains three major parts that are implemented as separate programs: (1) the *nuKIT-Manager* (*nM*), (2) the *Student Interaction Tool* (*SIT*) and (3) the *Lecturer Query Tool* (*LQT*). The *nM* is the administrative program of the nuKIT collection, and is particularly required for setting up new courses or altering existing courses in the system. The set-up process includes the selection of a lecturer password ($\text{Password}^{\text{Lecturer}}$) and a student password ($\text{Password}^{\text{Student}}$). These passwords administer the access to a specific course in nuKIT for lecturers and students, respectively. $\text{Password}^{\text{Lecturer}}$ grants access to the SIT and the LQT. $\text{Password}^{\text{Student}}$ grants access to the Student App and the Web Application respectively (cf. Sec. “Student Side”). The SIT and the LQT are communication tools which are used during a lecture and provide the fundamental functionality of nuKIT. At startup, they initiate the download of corresponding data from the database, e.g. predefined questions. The SIT, as a lecturer client user interface, visualizes two types of student communications: (1) When students ask questions over the platform, these are stored in a queue managed by the SIT, and the lecturer is informed about new questions using a visual aid. There, an unobtrusive visualization is chosen, making it acceptable (to all teachers using the tools so far) that the client shows a notification about all questions asked during a lecture in real time. (2) Students’ feedback regarding the lecturer’s presentation speed is visualized constantly during a lecture by showing a small feedback chart in the same visualization window as the students’ questions. The SIT can be displayed, beside the set-up mode, as an unobtrusive presentation mode which guarantees a transparent feedback system. The LQT is intended to initiate in-class communication by the lecturer by presenting questions to the students that are supposed to be answered immediately and discussed afterwards. For this purpose, the lecturer creates single choice, multiple choice or free text questions, usually before the course takes place, and publishes them during the corresponding course. The questions are stored in a server-side database as part of the course data of a specific course. Therefore, a lecturer can use his/her pre-generated questions from different computers or different lecturers can use the same questions

when they teach the same course. During a lecture, the questions can be chosen and displayed on the lecturer's computer as well as on the student clients (cf. Sec. "Student Side") where the students can answer them. The LQT gathers all responses from the students and presents the results from the survey visually in a chart (in the non-free text cases). These graphs can be published to the students and uploaded to the student clients. Another application of the LQT is the usage outside the classroom, for example between lectures during a semester. There, the LQT can be configured to publish a set of questions (*survey*) at a certain arbitrary time, and to give the students a chance to answer the questions for a predefined time interval. After closing the survey, the teacher can analyze the questions and, for example, give feedback to the students during the subsequent lecture. While this is a rather straight-forward application of the survey functionality with regard to the LQT, the tool can be used in a flexible way to perform various types of surveys. For example, a survey may last for a whole semester allowing the students to comment on general "time-less" course-topics. Another application is a "virtual consultation-hour" where students can answer questions such as "How can I help you?" in a free-text way by posing their teacher course-related questions without having to actually visit his/her consultation-hour.

Student Side: The corresponding application on the student side (cf. Figure 1-left side) consists of two different client user interfaces which offer basically the same functionality. There is an Android App that can be downloaded from the Google play store (an iPhone App is currently being developed), and a web application that can be accessed by simply visiting the corresponding website (<http://46.137.181.121:10080/nukit>). Before being able to use the student client, the students have to select a course and log-on using the Password^{Student}. The student client accepts an arbitrary user name meaning that the students can act in an anonymous way. The course is online while a lecture tool (at least one of them) is online and logged on to the corresponding course. Therefore, the questioning and feedback process is only available if this has been initiated by the lecturer. The student interface is divided into three parts: (1) feedback vote, (2) question vote and (3) query. The feedback vote offers the possibility to evaluate a lecturer's presentation speed on a three-level scale: "slower", "faster" and "alright". When a vote is given, it is forwarded to the SIT of the lecturer where it is displayed in real time. Therefore, students can give the lecturer in a simple and anonymous way feedback regarding the presentation speed. If a question arises during a lecture, a student can use the question vote to send a textual question which is also redirected to the lecturer in real time. The third functionality of the student client is activated by the lecturer-side LQT when he/she sets up questions to be answered. The students can answer the questions with a single or multiple click or by a textual answer depending on the kind of question, using standard visual elements provided by the Android architecture and the HTML structure respectively. The answers are sent to the server database and forwarded to the lecturer's LQT where they are graphically displayed. The according results can be published to allow the students direct access via the student client user interface. The number of students using the tools simultaneously is not limited meaning that large courses can use the platform without hesitation (of course, an appropriate internet connection has to be available in the classroom).

To allow for a lecturer-side analysis of the communication with students and for scientific analysis of the usage statistics of nuKIT (as performed for this paper), all communications between teachers and students are logged and stored in the database in an anonymous fashion to protect data privacy for all involved parties.

Implementation of nuKIT

This section describes the implementation of the nuKIT program collection, which comprises the lecturer tools (nM, SIT and LQT), the student tools (Android app and web application) and the server implementation (Web service and database).

nuKIT is realized as a *client/server architecture* as shown in Figure 2. The server application and the lecturer tools are using Microsoft's .NET Framework and are programmed in C#. For the *Web Frontend* Microsoft's ASP.NET framework is used and for the web server the *Internet Information Server (IIS)* from Microsoft. To enable empirical evaluations a MySQL Server hosting a MySQL database has been integrated. The communication between the clients and the server uses a SOAP web service based on HTTP and XML. The nuKIT Android application is programmed as a native Android application in Java. In the following, the diverse parts of the architecture which are shown in Figure 2 are explained in more detail following the server/client architecture.

Server: The *Server* application coordinates the clients and fulfills the *Business Logic*. It is responsible for all created courses and manages questions and votes corresponding to each course. Each course has a pre-defined standard duration in which the lecture is marked as active and students can log-in. A lecture is set active when the lecturer logs-in to at least one feature of the lecturer tools. To prevent the system from being flooded by the clients, a list of all IP addresses of users who have participated is temporarily saved. This list is cleared after a predefined time which can be adjusted in the manager tool. The courses with all attributes are saved in an XML file.

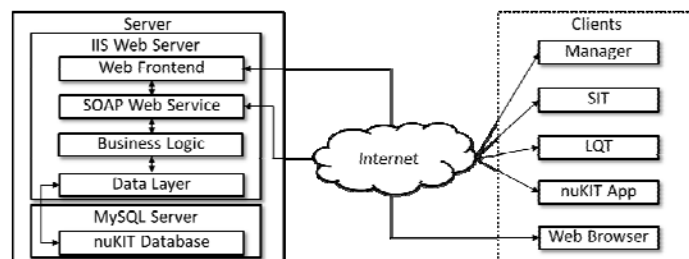


Figure 2 - Client/Server architecture of nuKIT. The different parts of nuKIT are displayed and the whole communication is illustrated with arrows.

An interface providing the functionality of the server application is provided by a *SOAP Web Service*. SOAP is a lightweight protocol which exchanges structured information in a distributed, decentralized environment. The HTTP protocol is used to encapsulate the SOAP protocol. This has the advantage that HTTP is routed through the internet and it is usually not blocked by firewalls. Such a protocol fits our requirements very well because of the necessity of communication between different kinds of systems like the Android application programmed in Java and the web service and lecturer tools implemented in C#.

The third big part on the server side is the *MySQL Database*, using the *MySQL Server*. The *Data Layer* interconnects the *MySQL Database* to the business logic and provides all necessary data to the business logic. As mentioned above, all communication between the tools and the server application is logged to the database. Additionally, all questions created with the *LQT* are saved in the database. This has the advantage, that every lecturer can create a different question catalog for each course and can access his/her question catalog from different devices.

Client: The *nuKIT Manager* is the administrative *client* which manages lecturers' courses. This includes the creation and deletion of courses, changing the passwords for lecturer and students, requesting new passwords in the case that the lecturer forgot his password and modifying some course-specific settings. A status bar which shows the result of an executed action has been implemented. The web service URL is saved in an XML file as an encrypted string, using the Advanced Encryption Standard (AES). This has the advantage that it is only possible to change the web service address via the *nuKIT Manager* but it is not possible to figure out the current web service by examining the XML configuration file. Creating a new course requires providing information about the course as the name, the lecturer's email address, and the two passwords. The name of the course has to be unique as it is used as a primary key on the server side. Deleting courses or changing settings requires the current lecturer password to prevent from manipulation by others. If a lecturer forgets his password, he can request a new password for the selected course. This triggers the server to create an email with a new random password which immediately replaces the old password. This email is sent to the lecturer's stated email address.

The *SIT* displays the student speed votes and their questions in real-time. The votes are visualized as a bar chart where each bar is representing the votes of one speed category. Additionally, the absolute number of votes for each category is shown. New student questions are indicated by a button which turns yellow in the instant the question arrives. To achieve real-time behavior without slowing down the responsiveness of the Graphical User Interface (GUI), the communication actions are running in a separate thread which regularly queries the server for new questions and votes. Clicking the button in the *SIT* which indicates an unanswered student questions displays the first question asked. Additionally, the number of subsequent questions is shown. The lecturer can choose between answering the question- which removes the question from the queue-, and putting off the answering, which shifts the question to the end of the queue. If there are more questions in the queue, the lecturer can show all questions at once. By highlighting different questions, the lecturer is able to answer multiple questions at once. Notifications about the success or failure of the mentioned actions are shown in a status bar.

Posing questions to the students is realized with the *LQT*. To pose a question an entry of the question catalog can be chosen. By clicking the "publish question" button, the question is sent to the server where it gets associated with the course. The answers to a question are by default not published to the student clients but can be published manually. The answer statistics are visualized as a bar chart. The updating of the answering statistics is done in a separate thread, due to process speed reasons. The web service URL can be set in the same way as with the *nuKIT Manager* and *SIT*. Due to the restriction that currently only one question is allowed to be set simultaneously, the realization of surveys consisting of multiple questions is under progress. Surveys can be created with the questions from the question catalog and can be started immediately, but also by setting a starting and ending point.

The *nuKIT App* for Android is the implementation of the student part of the *nuKIT* program collection. It supports every Android version since Android 2.1 which covers almost all of today's active Android devices. Google introduced a new window feature called Action Bar with Android 3.0, which is recommended to use

within Android applications. To provide a consistent user experience for Android devices before version 3.0, the open-source library ActionBarSherlock had to be used to provide the action bar also for devices before 3.0.

Figure 3 shows the architecture of the nuKIT App. The app consists of seven Activities which represent the features of the app. An Activity is an element which builds-up a screen for the interaction of the user and provides a GUI which is described in XML. The most important Activities are described in the following. In the Business Logic Layer the data processing is executed. This comprises in particular the preparation of the received data from the SOAP web service for displaying within the Activities, but also the handling of the lifecycle of the Activities and changing the Activities. The Data Layer provides the data which is processed by the Business Logic. For communication with the web server, a SOAP web service client has been implemented.

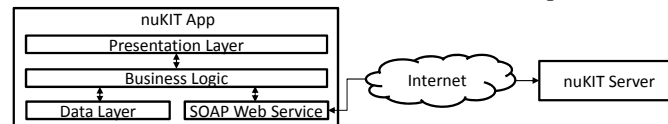


Figure 3- nuKIT App architecture. The different parts important for the communication within the nuKIT app are displayed and the communication is illustrated with arrows.

After starting the app, the user has to enter an arbitrary user name and select a course. The list of the current active courses is the result of a web service call. To maintain the app responsively, all network calls are implemented as asynchronous calls. Choosing a course requests the user to enter the course's password. The entered name and the course's password are persistently saved, so that the user does not have to reenter it in the future. Each Android device has a unique identifier which has to be consistent with the German Federal Data Protection Act. The identifier consists of a random 32 characters string which is generated on the first time the app starts plus the name of the selected course. Out of this string a SHA-512 message digest is calculated which is used as the identifier string. Hence, data protection regulations are fulfilled.

If the login was successful, the activity for sending questions and votes is shown. Coincidentally, the server is prompted for any existing lecturer questions but sending such a request manually is also possible. The success or failure of a sent question or vote is confirmed by a short notification. If a question from the lecturer is available, a new button appears to switch to the Activity showing the question.

The view of this Activity depends on the kind of question. If the question is of the type free text, a text field to enter the answer is shown. If it is of the type single or multiple choice, a list with all answering options is presented. Depending on whether one or more answers are allowed, the list entries have check boxes for multiple choice questions or radio buttons for single choice questions. The entered or selected answers are sent by pressing a commit button. The users' given answer is saved locally on the device. So if the user closes the activity and reopens it, his/her chosen answer is selected and entered again. If the lecturer allows publishing results, a button for opening the results activity is shown. In the results, the answer with the most votes is highlighted.

To support non-Android devices a *web frontend* has been implemented which is accessible by every conventional *web browser*. The web application provides the same features as the Android app and is designed for usability from notebooks and even for mobile devices with smaller screens. The identification of a device is done by generating a random 32 characters string the first time it visits the Web application. This string is saved in a cookie on the local device. The identifier sent to the server consists of the SHA-512 message digest which is calculated from the random string plus the selected course name. Despite the fact that the Web application is running inside the same IIS as the server application, communication happens only via the web service.

Preliminary Evaluation

This section gives statistical analyses of the usage and acceptance of the nuKIT platform as observed in three courses during the first half of the winter semester of 2012/13 (starting on October 14th 2012 and lasting until February 15th 2013; nuKIT data collection until December, 4th) at the KIT. The platform in its current, largely revised version has been broadly tested for the first time in these courses which made some minor technical startup difficulties unavoidable. Overall, however, the platform has been introduced successfully and there has been a high acceptance on the sides of both student and lecturer.

Scenario

In the following, the teaching environment of the nuKIT tools is described regarding the lectures in which it was first introduced. The platform has been evaluated in three courses: Course 1 is a first-year bachelor course on "*Programming and Introduction to Computer Science*". The number of attending students ranged approximately from 300 to 500. This course took place every Tuesday and Wednesday during the semester. Course 2 is called "*Foundations of Informatics II*" and addresses mainly students in the third semester of their bachelor program of

study. It took place on Mondays and Wednesdays and the number of attendees varied from 20 to 470 students. Course 3 is part of the master's degree covering the topic “*Algorithms for Internet Applications*” and taking place every Wednesday. The number of attending students varied between 5 and 50. The exact number of present students is only known for course 2.

Figure 4 summarizes the days when the three courses took place during the evaluation period. All courses took place regularly except for the course 2-lecture on November 28th which was cancelled. Course 1 and 3 were each given by one lecturer, Course 2 was held by two different lecturers. Due to the size of the audience, the teaching strategy has been ex-cathedra teaching in all courses.

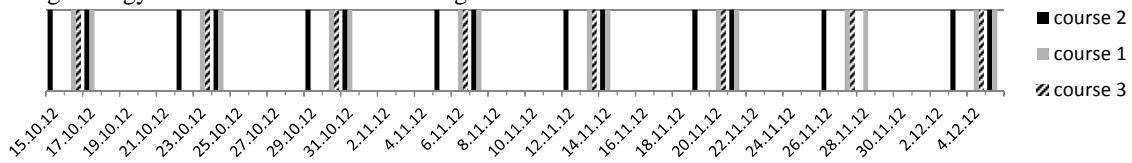


Figure 4 - Overview of the distribution over time of the courses. Course 1 (“Programming and Introduction to Computer Science”), course 2 (“Foundations of Informatics II”) and course 3 (“Algorithm for Internet Applications”).

Methodology of evaluation

As the nuKIT platform in its current version was introduced in Mid-October, the evaluation period refers to approximately 2 months, only. However, as this preliminary data collection and analysis shows promising results at several levels, we present it here carefully, being aware that more data is needed in several areas to gain statistically significant results. The nuKIT tools are evaluated in the following four respects.

1. General acceptance and registration data: the absolute and cumulative numbers of downloads of the platform are analyzed.
2. Usage of the SIT: the usage statistics of the *SIT* are analyzed with a focus on the data of student questions and votes.
3. Usage of the LQT: the number of logins is compared to the login data, the number of attending students and the acceptance of lecturers’ questions by the students.
4. General usage of mobile devices: the usage of mobile devices within the nuKIT platform is presented.

1. General acceptance and registration data. Figure 5 shows the progress over time of registrations to the platform in all three courses. A registration means either the installation of the nuKIT app or the first usage of the web application. The dark grey lines indicate the set-up of nuKIT in the three courses (ordered from left to right: course 2, course 3, course 1) at the beginning of the evaluation period. The distribution of the registrations to the days is, of course, highly correlated to which day the lecture took place (cf. Figure 4).

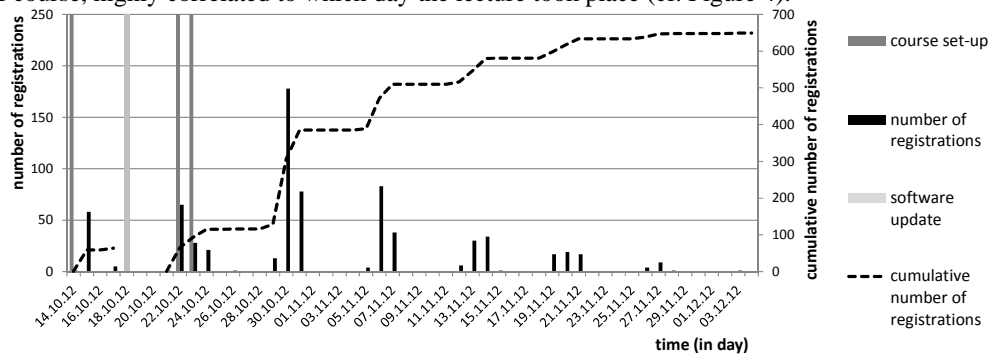


Figure 5 - Number of nuKIT registrations in the period of reference. Left y-axis / black bars: absolute number of registrations. Right y-axis / dashed line: cumulative number of registrations. The drop of the cumulative number of registrations between the first and the second lecture week is due to a software update in October, 18th which required a completely new download or registration of the nuKIT platform.

The dashed line (reference: right y-axis) shows the cumulative number of registrations. It shows that the number of registrations increased heavily during the first lecture weeks. The increase in registrations slowed down in the week starting at November, 12th which can be ascribed to the fact that almost all students willing to use nuKIT already had installed it before. This drift is also reflected in the decreasing absolute numbers of registrations. The number of registrations within the period of evaluation reached a total of 649. Relating this number to the maximum number of attendees of the three courses (500, 470 and 50, respectively) shows that about 65% of the attending students are using the nuKIT platform.

2. Usage of the SIT. The *SIT*, as described above (cf. Secs. "Structure of nuKIT", "Implementation of nuKIT"), offers the possibility to interact with the professor during a lecture by the means of questions and lecture-speed voting. This feature of nuKIT has been used in the period of reference in the three different courses.

speed	faster	alright	slower	sum
number of votes	359	641	447	1447

Figure 6 - Overview of the usage of the voting technology divided in "faster", "alright", "slower".

The overall number of lecture-speed comments is illustrated in Figure 6. It shows that the voting functionality is a highly-used feature of nuKIT. As more than 55% of the comments ask for a change in the presentation speed, the feature is apparently used for constructive criticism during a lecture which otherwise might not have been expressed. Whether this feedback is actually capable of changing a lecturer's presentation speed is not within the scope of this analysis and has to be evaluated in further studies. From the total sum of 1447 votes, only 39 votes result from the application of the nuKIT tools in lecture 3 (which is caused by the fact that all the lectures are recorded and freely available to the student for individual replay). This is due to the fact that the number of students attending this lecture mostly did not exceed 10 students. Thus, the use of nuKIT did not achieve high participation in this course and the tools were used irregularly. Therefore, course 3 will not be considered further in the following analysis. This shows that nuKIT is particularly suitable for the use in large courses.

Figure 5 shows the overall analysis of the *SIT*, divided into the accumulated number of lecture-speed comments on the right axis related to the right bar and the overall number of questions on the left axis related to the left bar. The left chart shows results from course 1, the right those from course 2, respectively. The usage of the nuKIT tools in course 1 did not start before November, 30th. The missing data in course 2 is due to technical problems (November 14th and 21st) and the cancelling of the lecture on November, 28th. The overall number of lecture-speed comments during the lectures exceeds the number of questions posed by students. It can be seen that the high participation at the beginning of both lectures with 349 (77) flattens over time. This trend appears to be both due to the flattening of the initial enthusiasm, as can be seen in course 1 where the attendance was rather stable between 300 and 500 students, and due to the decrease in students' attendance, as was particularly the case in course 2. It has to be taken into account that the visual aid for the feedback tool was not made transparent in course 1 which might have also led to a decrease in participation. In the latest lecture on December, 12th the GUI was visible to all students which resulted in an increase of participation. This aspect of transparency has to be evaluated further. The numbers under the two bars in the charts show the number of attendees. Whether the nuKIT platform may have a positive influence on the number of attendees in the lectures has to be analyzed in addition.

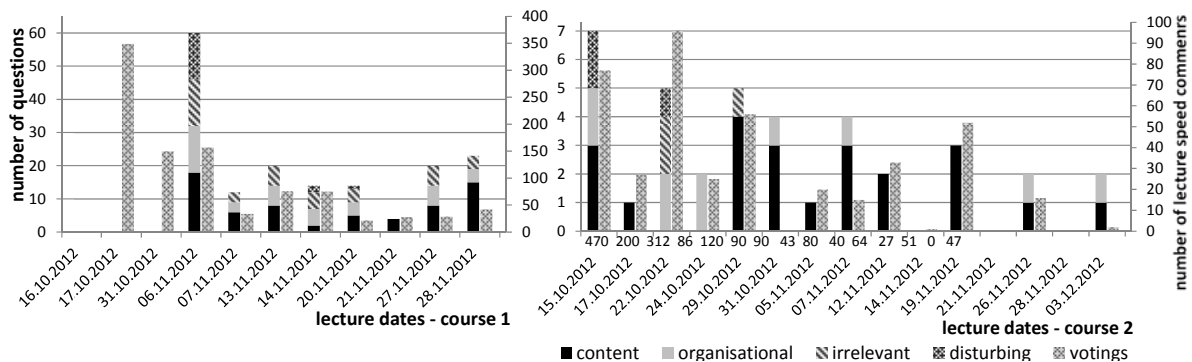


Figure 7 - Usage of the SIT. Right axis / right bars: number of lecture-speed comments. Left axis / left bars: number of questions (divided in contentual, organizational, irrelevant and disturbing questions); Left side: course 1; Right side: course 2.

The students' question-posing statistics in Figure 5 follow the same decreasing characteristics as the lecture-speed voting statistics. However, the question structure can be analyzed in more detail. Therefore, the questions are divided into four categories: contentual, organizational, irrelevant and disturbing questions. This classification is done by a manual analysis with the help of the knowledge of the content of the two courses. The two graphs of Figure 5 show that at the beginning of the use of the platform the number of disturbing questions was rather high; this number, though, decreased quickly while the questions concerning organizational or contentual topics increased in relation. This shows that -after the initial playing around- the demand for a serious questioning tool is given and the usage regarding questioning is with the given data similar over time.

3. Usage of the LQT. The *LQT*, as described above (cf. Secs. "Structure of nuKIT", "Implementation of nuKIT"), offers the lecturer a possibility to interact with the students by the means of single choice, multiple

choice or free text questions. This technology of nuKIT has been used in the period of reference only in course 2. The corresponding data is displayed in Figure 8. The left chart shows the absolute number of logins of course 1 as a reference. The right chart shows the number of logins of course 2 ranging between a maximum of 70% and a minimum of 10% with an average of 34.6% of all attending students. In the period of reference, there have been four lectures where at least one question was set by the lecturer. These questions have been answered by 7% to 30% of the attending students. It has to be examined in future studies if this number can be increased by offering different mobile apps for more platforms such as iPhone. Moreover, it has to be evaluated further whether there is a correlation between the posing of a question and the number of logins, which is uncertain with the given data.

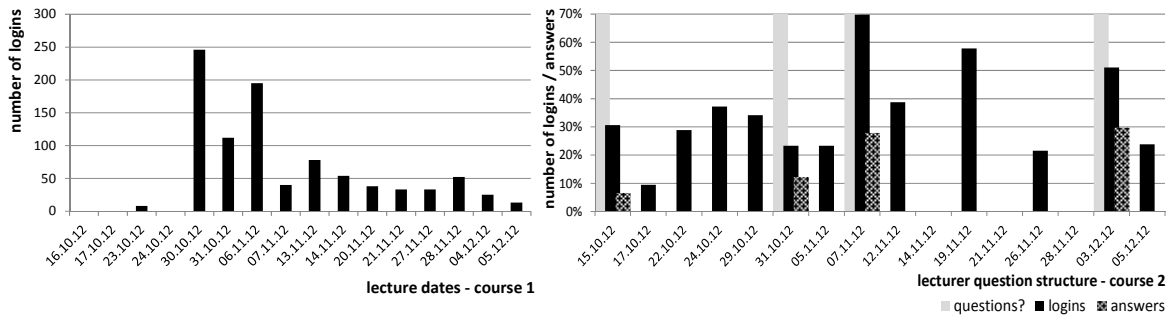


Figure 8 - Usage of the LQT number of logins / answers. Left side: LQT not used, only logins displayed. Right side: number of questions, answers in comparison to logins displayed related to the whole number of present students.

4. General usage of mobile devices. As the Android app on the student side has been one major improvement of nuKIT it remains to be analyzed if there is an influence on the acceptance of the nuKIT platform if it is accessible via a smart-phone app or whether a web application is sufficient. Figure 9 shows an overview over the mobile devices used in the period of reference to access nuKIT from the student side.

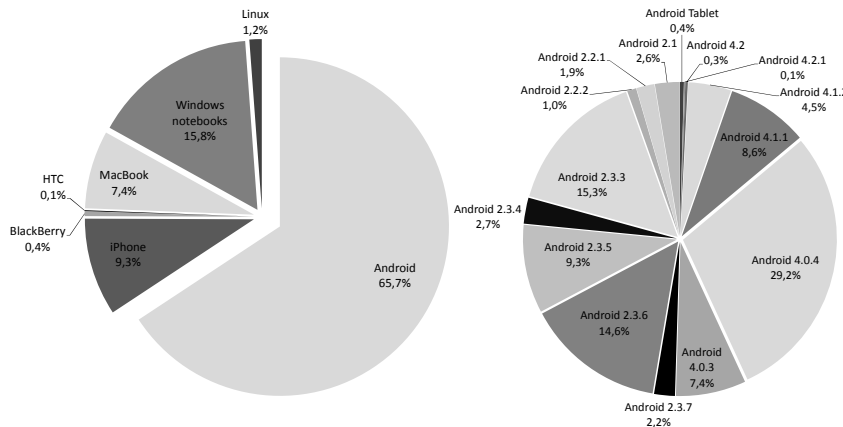


Figure 9 - Overview of the mobile devices which were used to access the nuKIT tools from the student side.

The pie chart on the left side shows that nuKIT was mainly accessed by Android phones. Only about 37% of accesses resulted from other sources platforms using the web application. This result indicates that students prefer the access with the app. This hypothesis can be further confirmed by an analysis of the distribution of mobile devices in the corresponding courses. The pie chart on the right shows a detailed distribution of Android phones.

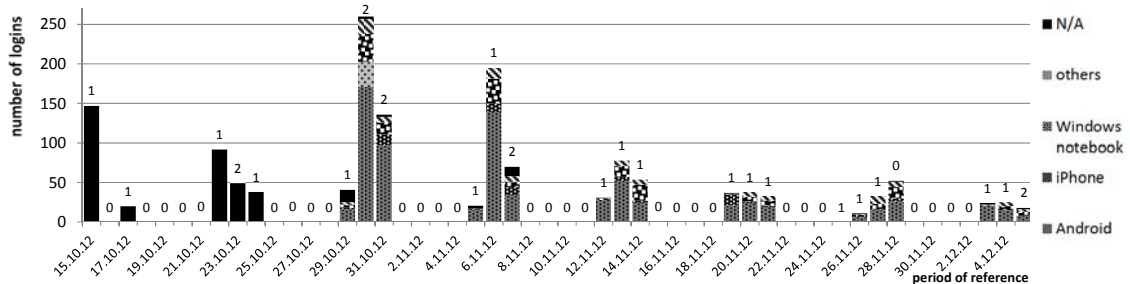


Figure 10 - Distribution over time of the use of mobile devices. Small numbers above the bars indicate the number of lectures taking place on that day.

Figure 10 illustrates a distribution over time of the use of mobile devices to access nuKIT. It also shows the number of logins within the period of reference. The allocation of a user to a mobile device could first be carried out after a software update on October, 25th. Therefore, the logins before this software update cannot be allocated to a special mobile device. These numbers also depend on the number of attending students. The small number above each bar indicates the number of lectures taking place at each particular date. As the number of attending students is only known for Course 2, this has to be evaluated further regarding values on a percentage basis.

Conclusion and Future Work

In this paper, we presented a lightweight interactive communication tool via smartphone technologies, which is currently used in three different courses at the Karlsruhe Institute of Technology. The nuKIT tools were used to increase the interaction and communication and to improve the individual learner support. They offer students the possibility to give feedback regarding the presentation speed of the lecturer in real time and to ask questions in an anonymous way, if desired. In addition, the lecturer has the possibility to pose questions to the students that they can answer during the lecture or within a predefined time period.

A preliminary evaluation of nuKIT has been presented based on a 2-month data collection. The data indicates a positive reaction. About 65% of the attending students of the observed courses registered for the nuKIT platform. Moreover, approximately 35% of the present students (of course 2) logged-in every lecture. This shows a good acceptance by the students. The tool has also been used for constructive criticism with the help of the feedback functionality, especially in large courses. The higher usage of the app compared to the web application indicates that students prefer the app access. Further surveys regarding the distribution of smartphones in the courses are necessary to settle this assumption.

In future, more detailed evaluations have to be performed. Further analyses have to be done, for example, if the usage of nuKIT has a positive influence on the number of attending students or if the transparency of the questioning and feedback interface has an influence on usage statistics. Another open question is if the number of students answering a lecturer question can be increased by offering various mobile apps. Moreover, we intend to extend the nuKIT tools by further features. Besides offering an iPhones app, we aim at expanding the application area, e.g. by surveys outside the lecture, so that the benefits of nuKIT for students and lecturers can be increased.

References

- Bandiera, O., Larcinese V. & Rasul, I. (2010a). *The impact of class size on the performance of university students*. VoxEU.org. <http://www.voxeu.org/article/impact-class-size-performance-university-students> [viewed 07.12.2012].
- Bandiera, O., Larcinese V. & Rasul, I. (2010b). *Heterogeneous Class Size Effects: New Evidence from a Panel of University Students*. *Economic Journal* 120 (549), (pp 1365-1398).
- Bitkom (2012). *Presseinformation – Jeder Dritte hat ein Smartphone*. Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e.V. http://www.bitkom.org/files/documents/BITKOM_Presseinfo_Besitz_von_Smartphones_16_04_2012%281%29.pdf [viewed 11.12.2012].
- Bonn, M., Dieter, S. & Schmeck, H. (2003). *Kooperationstools der Notebook Universität Karlsruhe (TH)*. In (David, K., Wegener, L., Ed.): *Mobiles Lernen und Forschen. Beiträge der Fachtagung an der Universität Kassel am 6. November 2003*, (pp 63-71) Kassel: Kassel university press.
- Garrison, D.R. & Kanuka, H. (2004). *Blended learning: Uncovering its transformative potential in higher education*. *The Internet and Higher Education*, 7(2), (pp. 95–105).
- Pfeiffer, F., König, L., Wiesner, A. & Schmeck, H. (2012). *An Interactive Pool of Exercises for Individual Learning Support in Academic Courses*. In *Proceedings of World Conference on Educational Media and Technology (EdMedia)*.
- Ramsden, P. (1992). *Learning to teach in higher education*. London: Routledge.
- Rovai, A.P. & Jordan, H.M. (2004). *Blended learning and sense of community: A comparative analysis with traditional and fully online graduate courses*. *International Review of Research in Open and Distance Learning*, 5 (2).
- Scheele, N., Wessels, A., Effelsberg, W., Hofer, W. & Fries, S. (2005). *Experiences with Interactive Lectures - Considerations from the Perspective of Educational Psychology and Computer Science*. International Conference on Computer Supported Collaborative Learning (CSCL), Taipei, Taiwan.
- Schön, D., Kopf, S., Schulz, S. & Effelsberg, W. (2012a). *Integrating a Lightweight Mobile Quiz on Mobile Devices into the Existing University Infrastructure*. In *Proceedings of World Conference on Educational Media and Technology (EdMedia)*.
- Schön, D., Klinger, M., Kopf, S. & Effelsberg, W. (2012b). *MobileQuiz - A Lecture Survey Tool using Smartphones and QR Tags*. *International Journal of Digital Information and Wireless Communications (IJDIWC)* 2(3), (pp231-244). The Society of Digital Information and Wireless Communications.
- Shivetts, C. (2011). *E-Learning and Blended Learning: The Importance of the Learner A Research Literature Review*. *International Journal on E-Learning*. 10 (3), (pp. 331-337). Chesapeake, VA: AACE.