TECHNISCHE UNIVERSITÄT DRESDEN

Redesign and unification of the web view of AMCS

Sinthujan Thanabalasingam

15.11.2019

Auditorium Mobile Classroom Service (AMCS) is an Audience Response system that is actively developed by several individuals at the TU Dresden since 2012. In the educational context, speakers and docents use it to enhance and manage their lectures during the semester. One of the main goals of AMCS is to improve the way knowledge is presented and transferred to the audience by offering interactive polls that students can participate in before, during and after a lecture. Several standalone front end applications for platforms like iOS, Android and web are provided that enable usage of the system. However, because of its ease of access, a majority of students use the system via their web browsers on different mobile devices such as laptops, tablets and smartphones. Regarding usability, design and consistency, the challenge lays in providing a unified and responsive user interface across all supported platforms that is intuitive to use.

1 Introduction

1.1 Motivation

AMCS (Auditorium Mobile Classroom Service) is an Audience Response System (ARS) currently under development at the TU Dresden. Several members of the Faculty of Computer Science initialized the project in 2012 and since then, the system and its community have grown continuously and countless features were added over the years.

In general, the system's main objective is to improve the way knowledge is transferred from a speaker to their audience. By providing interactive polls and evaluation mechanisms, AMCS aims at increasing the audience's engagement before, during and after a lecture takes place, overall strengthening the interaction between speaker and audience.

On the one hand, several features of AMCS are designed to support speakers in their role. Regarding the educational context, docents at university use it not only during their lecture to get immediate feedback from participating students. Furthermore, it is used during the whole semester to prepare and manage lectures, to analyze learning progress and to evaluate feedback given by students. More specifically, a speaker can gain insight on the understanding of their audience by evaluating answers to their polls. For example, poor results to a poll covering a certain topic might suggest that the topic was misunderstood or badly explained. AMCS enables the speaker to precisely use this information, helping to identify parts of their lectures that students might struggle with. Speakers can focus more easily on repeating and emphasizing these topics in the future.

On the other hand, AMCS provides a set of features that try to increase engagement and to close the gap between speaker and listener. One of the system's goals is to support a student's learning process by providing interactive polls, question pools and self-evaluation mechanisms that work on saved answers. While the polls are often used during lectures, the latter two tools can be used afterwards to identify and repeat difficult questions, to prepare for the next lesson or to study for the examination.

Several standalone front end applications for different platforms such as iOS, Android and web are provided to the audience. However AMCS is used by the majority of students via its web page across different mobile devices such as laptops, tablets and smartphones, likely because it is very easy to access.

Regarding the web page being the most common way AMCS is used, the challenge lays in

providing a unified and responsive web-based user interface across all aforementioned device types. The motivation of this work is to analyze the application's state in terms of usability, design and consistency, identify weaknesses in each category and provide a set of proposals that hope to improve the design currently in place.

1.2 Objective

The main objective of this work is to provide a redesign strategy that - when implemented - improves the web view of AMCS on mobile devices in order to create a more user friendly experience. AMCS does offer a variety of features to the audience and to speakers. They were designed and developed by different people over the years and therefore all differ variously in terms of UI design and layout. To keep the scope of this work manageable, this work will mostly focus on all features that audience members will come in contact with when using AMCS. Each proposal for itself is centered around improving usability of the application while all proposals as a whole aim at keeping a consistent and recognizable interface across all supported platforms. Therefore, the rest of this work is structured as follows:

Chapter 2 describes the current state of the application and briefly covers all features of AMCS that fall in the redesign strategy scope. Additionally, in section 2.7, relevant existing applications and work are analyzed to identify strategies that can be transferred and applied to AMCS.

Following up is chapter 3 that is divided in two main sections. In section 3.1, an analysis of the current state of the application is conducted, identifying weaknesses and inconsistencies that the design can be improved on. In Section 3.2 these weaknesses are subject of discussion in a redesign strategy that is described in detail.

Finally, in chapter 4 the redesign strategy it is implemented as a prototype that uses the existing back end system. Chapter 5 evaluates the prototype and compares it to the current state of AMCS.

Chapter 6 concludes this work by giving a short outlook on future work.

2 State of the art

At the time of writing, AMCS offers front end applications for iOS, Android and web that students can use. As already mentioned in section 1.2 this work will focus on the web view / web page of AMCS. This section elaborates on the current state of the system by identifying and analyzing all views that allow for access to the different functionalities of AMCS from a student's point of view.

2.1 Web Technology

The AMCS web page is written using Angular[1], a typescript-based front end framework for building mobile and desktop web applications.

2.2 Landing Page and Login

When accessing the website[2], students will be shown the landing page of AMCS (see Figure 2.1). A big login button is displayed that when pressed will reveal a login form (see Figure 2.2). In order to use the system, students have to create an account by providing credentials.

Additionally, a subscription to courses is mandatory in order to reasonably use the service. By typing in an optional PIN code, students will subscribe automatically to the corresponding course. From thereon, students have access to the system.

2.3 Main View

After successfully logging in, the user is presented with the Main View of the system (see Figure 2.3). It can be scrolled in the vertical direction and is divided into header and body. On top, the header consists of corporate branding on the left side and a burger menu on the right side. Below it, the view's body organizes information in different sections as follows:
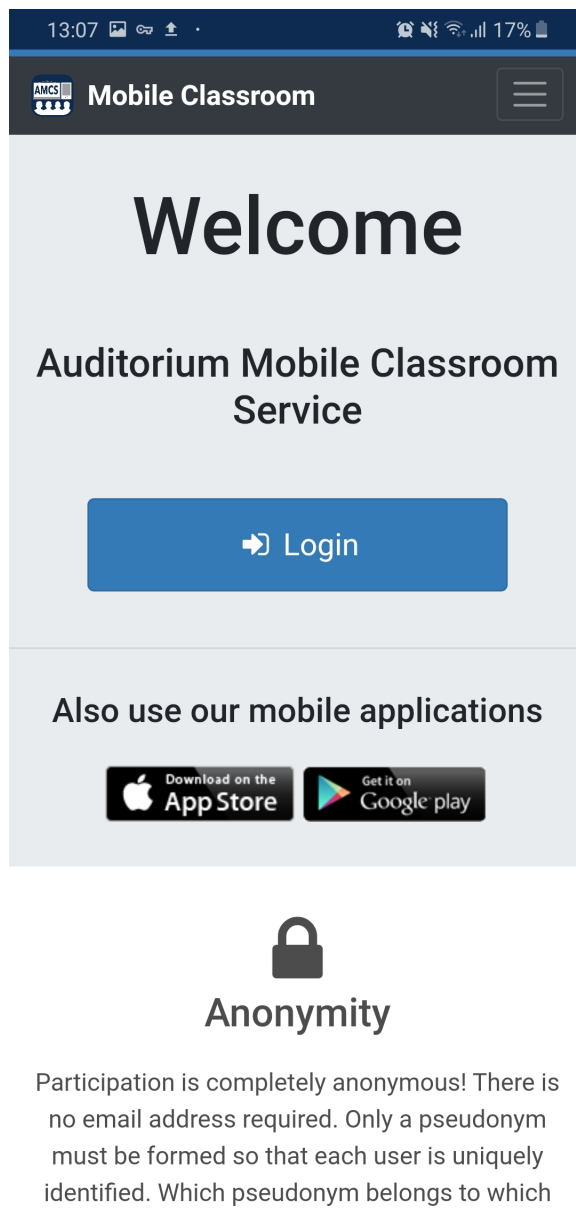
---

[1]https://angular.io
[2]https://amcs.website

Abbildung 2.1: The landing page of AMCS. This is the initial screen shown when accessing *https://amcs.website*
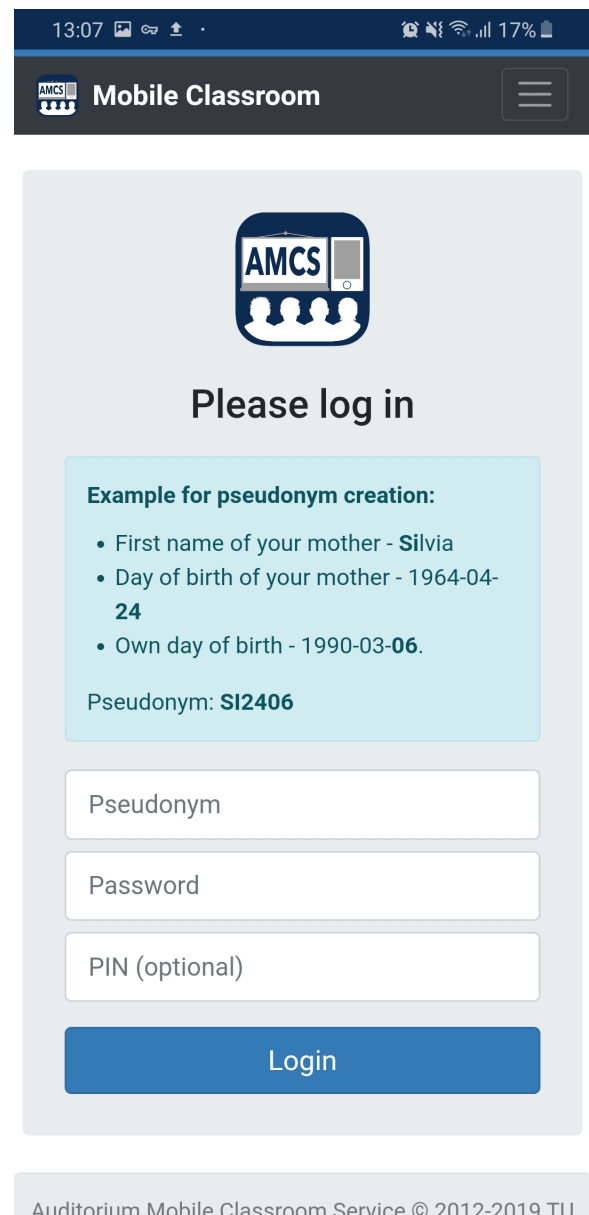
Abbildung 2.2: The login form of AMCS. Users can choose a synonym and a password to create an account.

Lectures

💡 Proposed lectures

Implicit Surfaces II — LIVE

📅 2019-11-10 - 13:00 h ⏱ 90 min

📄 We talk about implicit surfaces.

Computer Graphics II

Introduction — LIVE

📅 2019-11-10 - 09:20 h ⏱ 90 min

📄 Introduction to Application Development.

Application Development for Mobile and Ubiquitous Computing (English)

Requirements Analysis — AFTER

📅 2019-11-09 - 11:10 h ⏱ 90 min

📄 Requirements analysis for User Interfaces.

User Interface Engineering

⊞ Upcoming lectures (4)

⊞ Active lectures (2)

⊞ Upcoming lectures (4)

⊞ Active lectures (2)

⊞ Past lectures (2)

Enrolled courses

Example: PIN_123 | **Enroll**

Application Development for Mobile and Ubiquitous Computing (English) 🗑

Computer Graphics I 🗑

Computer Graphics II 🗑

Computer Networks 🗑

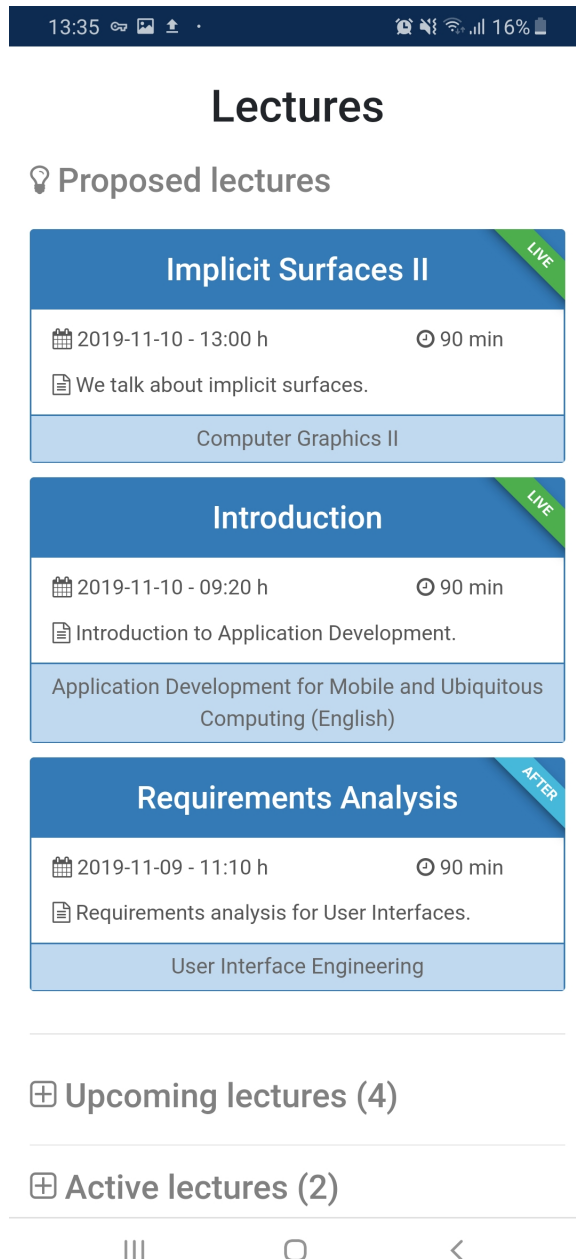Scientific Visualization 🗑

User Interface Engineering 🗑

Abbildung 2.3: Main View: Lecture information is provided in different sections for each temporal context. Per default, a section with proposed lectures is expanded.
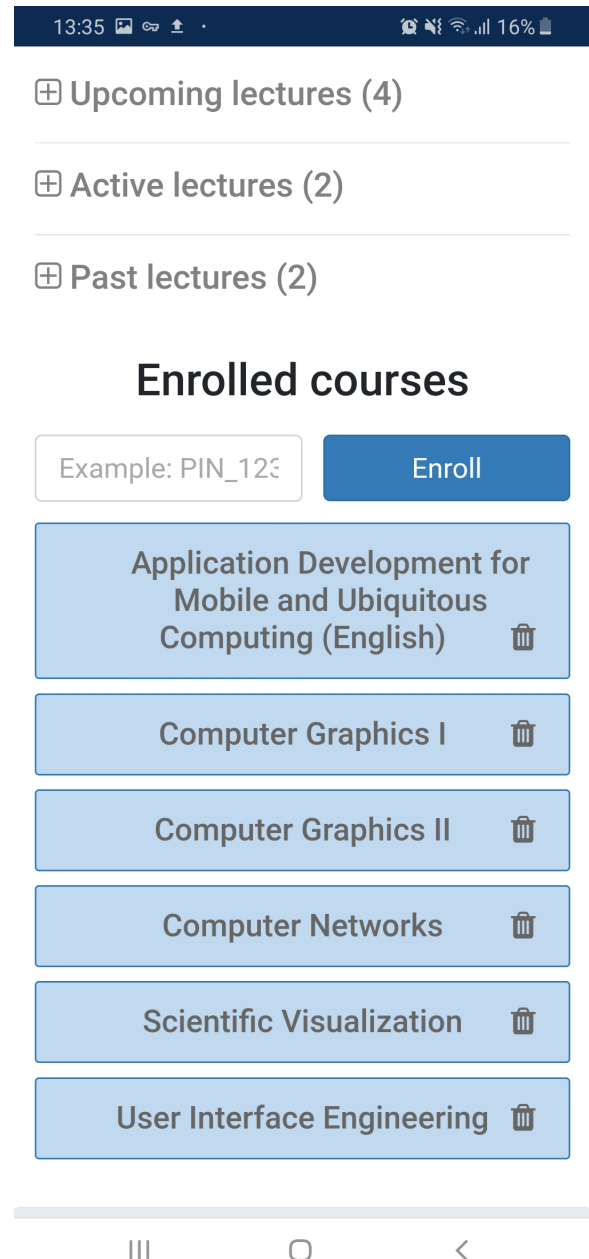
Abbildung 2.4: Course management below Lectures: Each course is listed below an enrollment form consisting of a PIN input and a submit button.

7

2.3.1 Lectures

This section lists all lectures that the user subscribed to (see Figure 2.3). It is organized in subsections that indicate the temporal context of each lecture. These include:

Upcoming lectures   - Lectures that will take place in the future are shown here.

Active lectures   - Lectures that take place right now are shown here.

Past lectures   - Lectures that have already taken place are shown here.

Rendering of lectures   Each of the aforementioned subsections is organized in a list that contains all corresponding lectures. For each lecture, a box is rendered that uses all horizontal space available to it. The box consists of a blue header with the lecture's name, a white info/detail area and a light blue footer that contains the course name. A color-coded badge on the top right of the boxes serves as an indicator for the temporal context of the lecture.

add graphic of badges

2.3.2 Course management

Further down on the page, the section *Enrolled Courses* can be found (see Figure 2.4). It serves two purposes: Primarily, it provides a way to enroll into a course. An enrollment form is shown that consists of a text field to enter the course PIN and a blue button to trigger the enrollment. When provided with a valid PIN, pressing the button redirects the user to the *Course View* (described in section 2.5) on successful enrollment. Secondly, the view shows all courses the student is currently enrolled in. They are rendered as light blue buttons in a vertical list. A trash can icon on each button provides a way to leave the given course.

2.4 Poll View

Answering polls is one of the main functionalities of the system that users will engage with. Polls can be reached by clicking on a lecture box from either the *Main View* or the *Course View*. Each poll consists of a set of questions the user can answer. They are rendered in a view that is reused by the system depending on the situation and context. This means that the view might only be accessible under certain circumstances, for example when the lecture reaches a specific point in time, making it a slide poll (SP). SPs are shown when a specific slide is on display and can only be answered in this very moment. Other types of polls include "global" course polls (CP) that are always accessible during the semester and lecture polls (LP) which can only be answered during the life time of a lecture. Active polls are displayed all at once in sections designated to each poll type. The different types of polls that occur in AMCS are further summarized in Table 2.1.

If no polls for a given lecture are available, the user is presented with the hint shown in .

add graphic here

2.5 Course View

The *Course View* is shown when the user selects one of the courses from the course management section (see Figure 2.5 and 2.6). Its purpose is essentially to provide a filtered view on the lectures of a single course. The course name and PIN, it's description and lists of upcoming, live and past lectures are visible from top to bottom in this order. It reuses the lecture section component described in section 2.3.1.

13:41 ⬆ 🔒 ✓ 📷 🔕 📶 16% 🔋

AMCS Mobile Classroom ☰

← **Computer Graphics II (CG_2_2019)**

❝ Students have to make sure that their solution is handed in at the submission date. Theoretical submissions must be handed in in written form on A4 paper by 16:00 to the chair staff. We recommend handing in the solutions at the end of the previous lecture. Practical submissions must be uploaded via Opal by 23:59. To get points for their practical submission, each team has to present their work to a tutor on the evaluation date. For this evaluation, each team gets an individual appointment with their tutor (Friday, 3rd - 5th DS). Note that for technical reasons, every team member has to upload a copy of their solution to OPAL in order to get scored.

Overview of all lectures of Computer Graphics II. Click on one of the lecture to open the detail view.

⊟ **Upcoming lectures (1)**

Implicit Surfaces II LIVE

ⅠⅠⅠ   ◯   ‹

13:43 🖼 ⬆ ⚙ · 🔒 staging.amcs.website/cours ③ ⋮

Overview of all lectures of User Interface Engineering. Click on one of the lecture to open the detail view.

⊟ **Upcoming lectures (2)**

Sketching and Prototyping BEFORE

📅 2019-11-23 - 09:20 h   🕒 90 min
📄 Sketching and prototyping

History of User Interfaces BEFORE

📅 2019-11-16 - 11:10 h   🕒 90 min
📄 The history of User Interfaces.

⊟ **Past lectures (1)**

Requirements Analysis AFTER

📅 2019-11-09 - 11:10 h   🕒 90 min
📄 Requirements analysis for User Interfaces.

Auditorium Mobile Classroom Service © 2012-2019 TU Dresden.
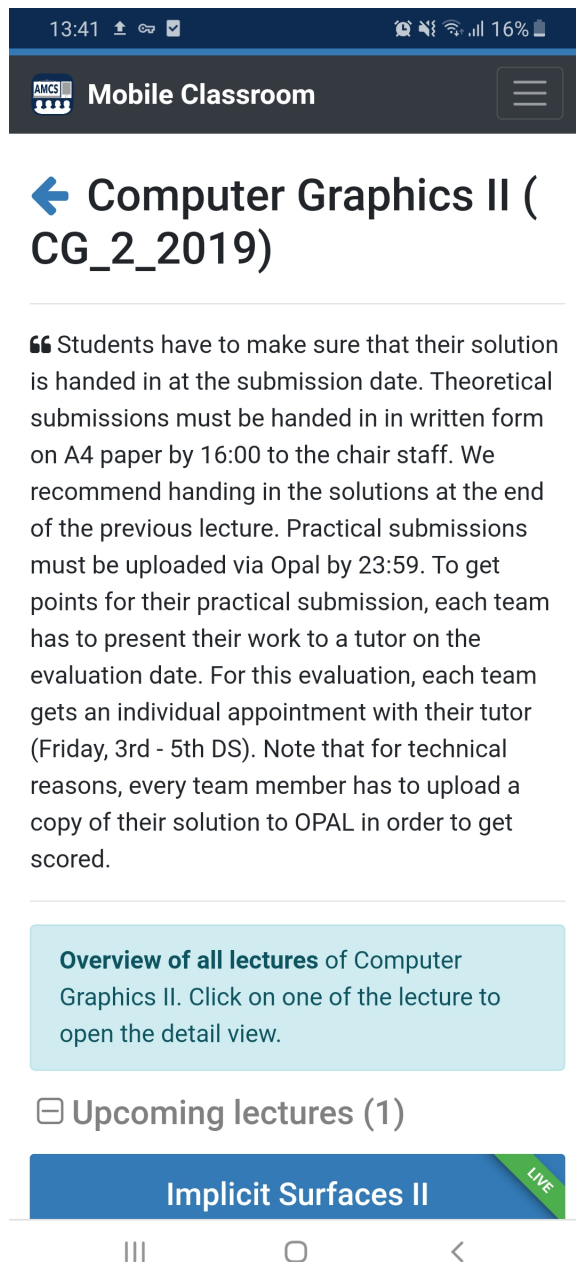
ⅠⅠⅠ   ◯   ‹

Abbildung 2.5: *Course View*: Details like the course name, description and PIN are displayed.
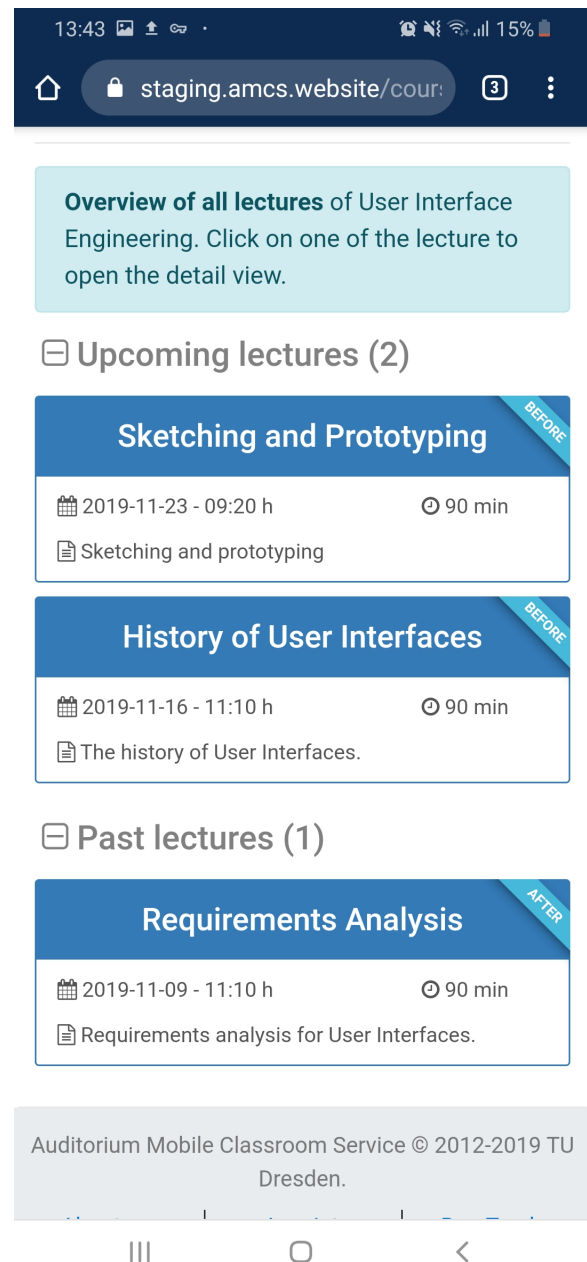
Abbildung 2.6: Lectures that belong to a certain course displayed in the *Course View*.

Poll Type	Explanation
Slide Poll (SP)	Active when a specific slide is shown. Commonly used for pop quizzes after a difficult section in a lecture to make sure that students understood everything correctly.
Preparation Poll (PP)	Active before the lecture takes place. Is commonly used to instruct students to prepare for a certain topic.
Lecture Poll (LP)	Active during the lifetime of a lecture.
Post Processing Poll (PPP)	Active after a lecture has taken place. Commonly used to check gained knowledge.
Course Poll (CP)	Active during the whole lifetime of the course (commonly during the whole semester).

Tabelle 2.1: Different poll types that the user might encounter when using AMCS.

2.6 Menu and Navigation

Besides using the *Main View*, additional functionality can be reached by navigating the burger menu that is shown in the upper-right corner of the screen. It reveals a sub menu which expands vertically, offering three additional sub menus (see Figure). In the following, these sub menus and their functionality are briefly explained.

add section here

2.6.1 Student

This is one of the most important buttons that connects a subset of main functionalities of AMCS. Upon pressing this button, the menu expands again vertically, showing a list of further sub menus. Most of the functionalities shown in this list will be touched by the proposals for improvement that are presented in chapter 3. The functionalities in questions are:

1. Question Pool

2. Evaluation of answers

3. Edit account

2.6.2 How it works

Pressing this button will redirect to a page that shows tutorial instructions on how to use AMCS. This help page is rendered identical on all mobile devices and therefore falls out of the scope of this paper.

2.6.3 Logout

As the name already states, pressing this button will logout the user and end the session. If logging out was successful, the *Landing Page* is displayed.

Component	Description
Landing Page	test
Main View	test
Poll View	test
Course View	test
Burger Menu	test
Question Pool	test
Evaluation of answers	test

Tabelle 2.2: UI Components of AMCS.

2.6.4 Question Pool

By selecting this option from the *Burger Menu*, the student is offered the possibility to create collections of already answered questions. The intent is to provide a way for students to collect and repeat questions that they had difficulty in answering. Similar to , the student is prompted with a drop down menu to select a course they are interested in. After selection, the student is presented with a list of all lectures and their polls respectively. All questions of each poll are grouped and shown to the student in a vertical list. From this list, the student can select all questions that they might be interested in to create a pool of questions. These pools are composed into polls that the student then can answer again. These polls are rendered in the same manner as stated in 2.4.

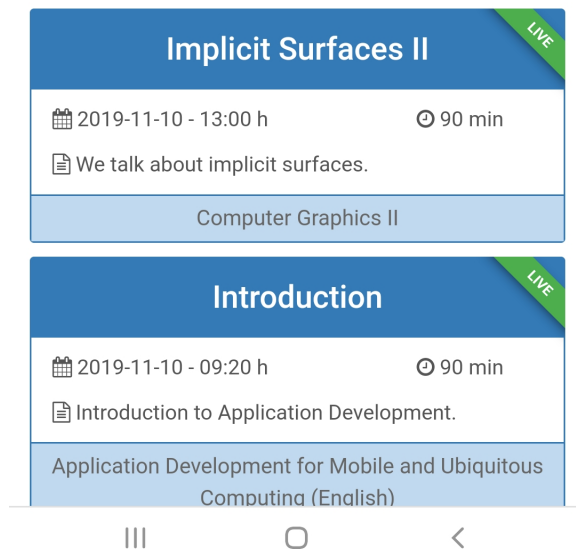add section here

2.7 Related Systems
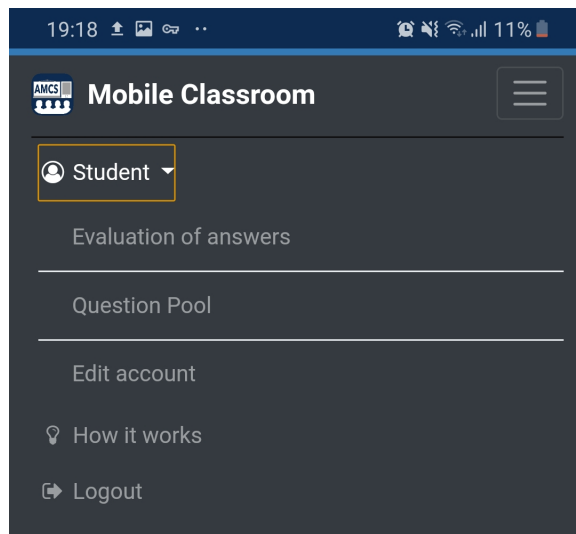
Add this

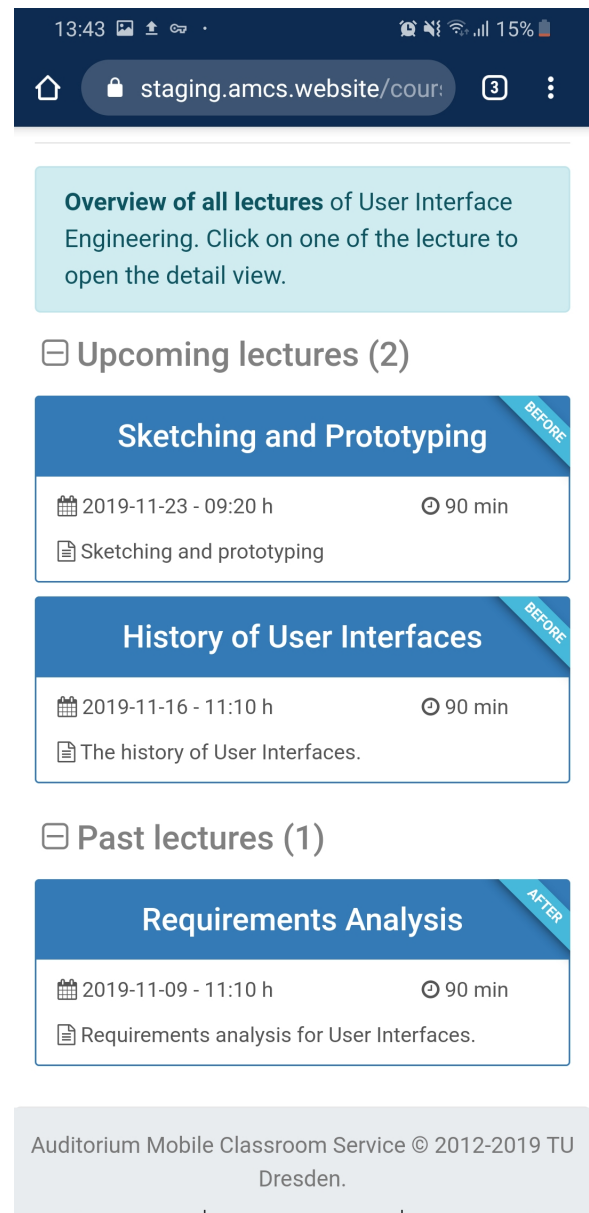Abbildung 2.7: Expanded *Burger Menu*.

Abbildung 2.8: Lectures that belong to a certain course displayed in the *Course View*.

3 Concept

This chapter will cover identified problems that occur when using AMCS via a web browser on mobile devices. To narrow down the extend of this work, the system is analyzed from the point of view that an audience member like a student has while using AMCS on their smartphone. When doing so, students will interact mostly with the *Main View*, the *Poll View* and the *Navigation / Burger Menu*. Therefore, this work is centered around these components.

3.1 Problems of the mobile view

The back end of AMCS reacts on requests coming from mobile devices such as smartphones or tablets by providing a responsive mobile view to its clients. However, in some aspects, AMCS struggles to offer a uniform UI experience that guarantees high usability. One challenge lays in the fact that the system has to deal with limited screen space to visualize information as effectively as possible. Additionally, users might approach the application with different ways of interaction and navigation in mind that are typical for mobile devices. For example, a smartphone user might expect to be able to use swiping gestures to navigate a menu or that information is organized in views consisting of separate tabs. This section lists key issues that lower usability or might cause confusion to mobile users.

3.1.1 Main View

As already described in section 2.3, the *Main View* relies on a vertically scrolling list view, consisting of different sections. The layout in place causes problems regarding usability aspects of the application.

General Visualization Problems

Section 2.3.1 covered the fact, that lectures are rendered by displaying the title of the lecture in the top section of the box using white letters on a solid blue background. It is followed by details about the lecture such as time, duration and a textual description, visualized in gray letters and icons on a white background. Finally, at the bottom of the box, the course name is shown in gray letters on a light blue background. The order *lecture name, details, course name* can cause confusion.

   The most coarse grain piece of information - the course name - is displayed at the bottom of the box rather than at the top. Generally, when seeking information about active lectures, a student will most likely remember the course name rather than the name of a single lecture,

as timetables used by students only contain course names. Therefore, displaying the course information in this order might cause confusion and students take longer time to find the pieces of information that they are looking for. In addition to that, inappropriate background colors and font sizes are used to differentiate between course name and lecture name, further increasing the ambiguity described above.

Indirection Problems

The boxes that represent each lecture claim a lot of screen space in relation to the information that is displayed to the student (see Figure 2.3). The layout causes a lot of indirection, because per default, the sections for upcoming and active lectures are expanded fully. This might be handy when quickly gathering information about lectures that are or soon will be active, but in every other case it slows navigation and overall interaction, because the course management section is pushed down to the bottom of the page. A list of only four boxes causes a scroll bar to appear on the very common screen resolution of 1920x1080 pixels. A student that navigates to the *Main View* to enroll into a new course therefore always has to scroll to the bottom of the page before they reach the enrollment form. The same problem likewise occurs when simply seeking information about what courses a student is already enrolled in or when trying to leave a course altogether.

Further more, if a student looks for information regarding a specific course, no filter or search functionality is offered by the lecture list. Instead, they have to scroll down to the bottom of the lecture list, scan the course list manually with their eyeballs, find the course and click on the corresponding item. Finally, they are redirected to the *Course View*. This level of indirection is only further increased the more courses the student is enrolled in. To sum it up, the hierarchical and logical relationship between courses and lectures is disregarded.

Redundancy

Some visual redundancy is added by the badges that are displayed on the upper-right corner of each lecture. These badges are used to visualize the temporal context of the lecture for each item in the corresponding section. It seems that the badge's intention is to help conveying the temporal context of the lecture, because sorting the lectures in their respective section alone fails to do so. Furthermore, the badge's names do not match the section's names, e.g. an upcoming lecture's badge reads *BEFORE* instead of *UPCOMING*.

In the course management section below, for each course the student is enrolled in, a box is shown with the course's name along with an unsubscribe button, represented by a trash can icon. This is a redundant way of rendering the courses which adds noise to the overall look and uses a lot of the vertical screen estate.

add graphic here

3.1.2 Course View

Since the *Course View* reuses the lecture list along with the sections *Upcoming lectures*, *Active lectures* and *Past lectures*, likewise the same issues arise as for the *Main View*, as already described in section 3.1.1. The *Course View* has a very important function in terms of usability as it acts as a filter for lectures belonging to a certain course. Problems arise, when a user wants to switch quickly between different courses. Doing so requires to leave the *Course View* by tapping the back button and then scanning the course management section for the element of interest, which is slow and cumbersome. This is explained in more detail in section 3.1.4

3.1.3 Poll View

Rendering

Section 2.4 describes the rendering of questions as boxes that are aligned in a vertical scrolling list. Namely the extensive use of vertical space on the screen is one problem introduced by this layout. Bigger polls that consist of multiple questions unnecessarily take a lot of vertical screen estate. The view also lacks of basic information such as total number of questions or number of remaining questions. This information might be useful in bigger polls if students want to gain an idea on how many questions are left. Answering one question usually does not require to see the neighboring questions, but most of the time, two to three questions are in view simultaneously (see Figure). This might be distracting to some students. Additionally, the layout lacks of separation and distinction between types of polls. Different types of polls are simply appended to the bottom of the list, s its length even further.

> add section here

> add graphic here

> add graphic here

Local navigation

The vertical list is difficult to navigate as it requires scrolling between questions. If a student wants to jump from the first to the last question, or vice versa, several swiping gestures are needed to reach the top or the bottom of the list. Similar to the lecture list described in 2.3.1, the question list is also segmented into different sections. Lecture and course questions are similarly appended to the bottom of a *Slide Poll*. This means again, that a student who wants to view these questions has to scroll all the way to the bottom of the list. Again, this layout introduces a lot of indirection and is unintuitive.

3.1.4 Burger Menu and Navigation

At the time of writing, the ways of navigating the application can be described as problematic and partly confusing. Several layers of indirection introduce problems and may worsen the user experience. Figure 3.1 illustrates click paths a user must take in order to reach different views (illustrated in blue) within AMCS. In general, some views are connected via the *Burger Menu* as the overarching element of navigation (illustrated in green). In contrast, other views are interconnected and can be reached by clicking on elements inside a view such as a course or lecture. The following paragraphs elaborate more on both aspects of navigation.

On the one hand, the interconnected graph of views as it is described in Figure 3.1 contains two main issues. One example of unexpected behavior is the fact that when a user wants to return from the *Poll View* to the *Main View* in order to chose a different poll, they are first redirected to the *Course View*. However, it is possible to reach the *Poll View* from the *Main View* with one tap, by selecting a lecture for example. Furthermore, some views like the *Question Pool View* and the *Answer Evaluation View* do not offer buttons that allow to navigate back. The navigation relies partly on the corporate branding on the upper-left that when tapped will redirect to the *Main View*. This means that the interconnectedness between different views might not be strong enough.

On the other hand, the *Burger Menu* connects several aspects and functionalities of AMCS in an overarching manner, as it is part of every view. One visual problem that arises is the fact that the menu uses a lot of vertical screen space and delocates the rest of the content that is currently shown when several sub menus are expanded. An example as to why this is problematic might be the following scenario: a student wants to evaluate their answers to polls for a certain lecture. When opening the menus on their phone from the *Main View*, information like course name and lecture title are pushed down by the menu. But this information is required

ID	Name	Category	Summary
Slide Poll (SP)	Test	Test	test

Tabelle 3.1: Classification of issues identified in the usability analysis of AMCS.

in the Answer Evaluation View because the user is asked to chose their course and lecture of interest from two drop down menus. This could lead to users having to return to the *Main View* to look up the lecture name or other details again. In addition, the menu can confuse students because the first menu entry is labeled as *Student*. This implies to users in general that different roles exist that a user can play in AMCS, which is the case. However the user's role should not be the label of a sub menu in the navigation as it is not intuitive that functionality like account management, the Question Pool and Answer Evaluation can be accessed via a button labeled *Student*.

Evaluation of answers

Clicking on the option *Evaluation of answers* in the expanded *Burger Menu* leads to a view with a drop down menu from which students can choose a course that they are interested in. Afterwards, a list of expandable items is shown, where each item represents a lecture. Clicking on one or multiple of these items will expand a vertical list of questions similar to the regular question list described in section 3.1. Likewise, answers given by the student are shown as well (see Fig.). Multiple problems occur on this view: First of all, the navigation path to reach this view contains unnecessary indirection and might not be intuitive enough, which is again illustrated by Figure 3.1. Students might expect this functionality to be located at the *Main View* attached to the elements of the course list or inside the *Course View* itself. Instead, every time evaluation of given answers is attempted, this functionality can only be accessed by using the *Burger Menu*, choosing the appropriate item from the sub menu, selecting the course in question and afterwards expand the lecture and the corresponding question list. Moreover, the question list suffers from the same rendering and navigation problems already described in . Questions are poorly navigable and a lot of scrolling is required to jump between questions.

Question Pool

The question pool suffers from the same navigation problems described in the preceding section. Again, a drop down menu for selecting a course is shown before students can see the overview of the *Question Pool*. Once more, students might think that access to this functionality is located near the *Main View* or the *Course View*, which is not the case.
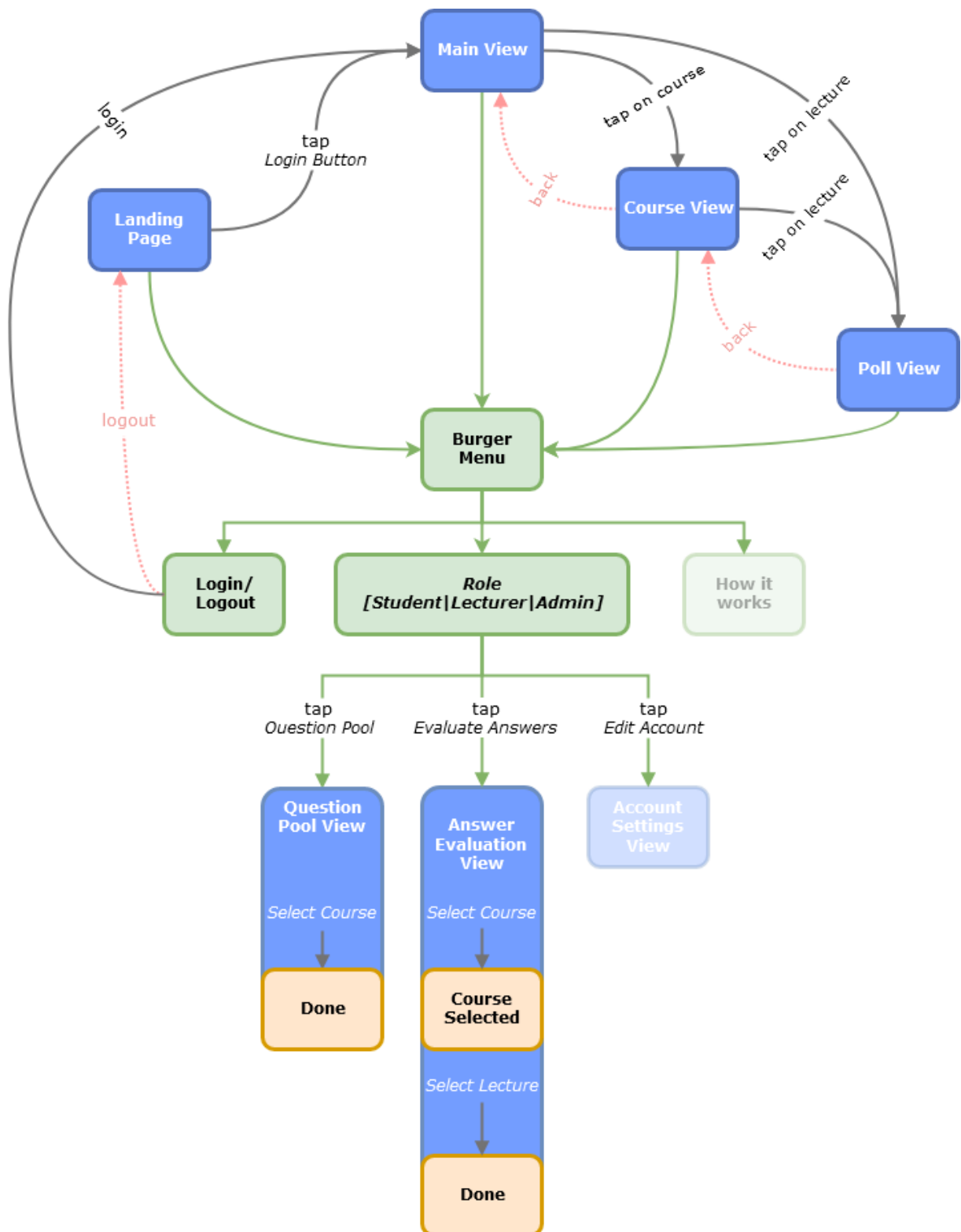
Abbildung 3.1: Navigation concept of AMCS: Every arrow represents a tap / click the user has to do to reach the desired destination.

3.2 Redesign Strategy

Several weaknesses and flaws of the web view of AMCS have been identified and analyzed in the previous section. They range from issues regarding visualization, layout and space usage or user navigation. A set of proposals that aims at solving these problems is introduced in the subsequent section. Mainly, the focus will lay on using the available screen space more efficiently, improving local navigation inside polls and global navigation between different views and reducing indirection to a minimum.

3.2.1 Main View

Layout

As mentioned in section 3.1.1, the *Main View* suffers from using the vertical space available inefficiently. Most noticeably, the course overview and enrollment form are placed below the list of lectures. In order to find information about relevant courses or to enroll / leave a course, students are required to scroll all the way to the bottom. Therefore, one proposal is to compress this view by using drop down menus and tabs (see Fig). First of all, the view is re-structured to follow the hierarchical concept of courses containing lectures: In the top (1), a button for a drop down menu is shown next to the currently selected courses' name and two additional buttons (6) and (7). The functionality of the drop down menu and the buttons is explained later. Following the heading, selectable tabs for *past*, *upcoming* and *live* lectures are laid out side by side (2). The tab bar is followed by a numbered horizontal list of clickable dots (3) that each represent one lecture. The currently selected lecture is highlighted with a bold blue border to enhance visibility and orientation. Finally, the information section of the view follows (4) with the title of the lecture, time and duration details and the lecture description. In the details section, an additional button is placed (5) that is labeled as "Evaluate". This button is only available on past lectures, as shown in Figure . This layout reduces the amount of vertical space used drastically. The placement of the "Evaluate" button (5) is motivated by reducing vertically occupied space as much as possible, but it could be reasonable to place it below the description text of the lecture. On most devices, the amount of scrolling required should be reduced with the proposed layout. As much redundancy as possible is eliminated from the view, as the badges that indicate temporal context of the lectures now miss completely.

Navigation

Tabs (2) should separate lectures by their temporal context. Selecting a tab will only display lectures that share the respective temporal context, meaning that it should be easier to switch between *past*, *upcoming* and *live* lectures. The *Button Bar* (3) in Figure 5 is used to ease navigation between lectures that share the same temporal context. A student can use the bar for example to switch quickly between the oldest and newest past lecture by selecting the corresponding button from the bar. This should improve navigation within the *Main View* as well as within a lecture context. Additionally, this proposal aims at removing indirection as much as possible on a global scale by providing certain buttons that serve as shortcuts for the functionality that is currently found in the *Burger Menu*. Button (6) serves as a shortcut to the *Question Pool* for the selected course. On click of button (7), the student will be removed from the selected course. Both of these buttons are placed in the header of the layout next to the course's name to indicate that both referenced functionalities operate on a course scale, whereas the "Evaluate" button (5) operates on a lecture scale. The *Evaluate Button* eliminates the need to select a course from a drop down menu and even choosing from a list of lectures is not necessary anymore, as all of these information needed to send the request to the backend are

present. Therefore, multiple layers of indirection are removed from the established workflow. All three buttons try to remove the indirection introduced by the *Burger Menu*ch as possible. Functionality that is associated with a course or lecture is triggered from a view that deals with courses or lectures respectively. The *Burger Menu* would then only have to deal with the profile editing and logout functionalities.

Embedded Drop down menu

The drop down menu (see) is introduced to help reducing usage of vertical space even more. Clicking on it reveals its two functions: for one, the enrollment form is now embedded in the drop down menu - a text field and a button are shown next to the text *Enroll…*. The close proximity to courses the student already enrolled into makes this functionality potentially more intuitive to be found by users. The enrollment form will only be embedded in the drop down menu when the student is enrolled in at least one course beforehand. Otherwise, in place of the *Main View*, only the enrollment form should be shown. Besides the enrollment form, a list of courses a student is already enrolled to is shown below. Selecting an item from this list will bring up the corresponding course view, similar to Figure . The *Drop Down Menu* acts as a filter to the *Main View* and essentially covers the responsibilties that the *Course View* currently in place has. An idea to further enhance the drop down menu would be to show notifications in little bubbles beside the course title to indicate new or upcoming lectures.

3.2.2 Poll View

Layout

Several issues have been identified regarding layout and visualization of polls in section . Main problems include the ineffective use of vertical space in this view and a lack of separation between types of polls. Different types of polls are simply appended to the bottom of the list, making it even longer. In order to solve these issues, some proposals that were made in can be applied here as well. Figure shows an instance of a slide poll showing a SCC-question using a concept that is proposed in this section. Beginning at the top, the course's name is displayed in white font on a blue rectangle (1). Following up, to separate the type of polls from one another, a tab based menu is used (2) that differentiates between slide polls, lecture polls and course polls. Each tab therefore corresponds to a poll type. Selecting one of the tabs will cause the layout to show only questions of said type. This will improve the effectiveness of vertical space used greatly. The tabs act as a filter to what is currently displayed.

A further enhancement would be to show notifications in little bubbles besides the name of a tab to indicate new content. Below the tab menu, a navigation bar (3) is displayed that contains the name of the current topic and additional information such as the number of the current question. In case of a slide poll as shown in Figure 7, the current slide number is shown as well. Also, the dot bar of section 3.2.1 is reused here (4). The dot bar serves as a means to navigate between questions more easily and faster but also reduces vertical space used significantly. It uses color coding and icons to convey information to the student. A blue bold border is used to indicate the current question selected in the dot bar, light blue dots signify, that the corresponding question has not been answered yet, where as bold green or red dots indicate correct and wrongly answered questions respectively. The dot bar can be further enhanced by using icons that represent the state of a specific question. Green arrows and red crosses can be used to visualize correctly or wrongly answered questions respectively to ease usability for colorblind students. Below the dot bar, only one question at a time is displayed to the student to avoid overwhelming them (5). The question is displayed in a blue rectangle with white text. Below the question, an instance of an answering mechanism is displayed (6). In the case of SC-,

add graphic here

add section here

add graphic here

Rewrite the sentence above. Its not true as we aim to eliminate the course view

add section here

add section here

add sec-

MC-, SCC- or MCC-question, the button to answer the question is omitted. Multiple ideas for triggering the request to the AMCS backend exist here: Either selecting the answer will trigger a modal that asks the user whether or not he is sure with his choice of answer, and confirming this dialog will send the data to the server, or the mere selection of an answer will trigger the request. A wrong answer is highlighted as before in red, a correct answer in green and it will still be possible to answer twice. Finally, space for textual feedback is given in a box (7) below the answers. This view is reusable and can therefore can also be used to display already answered questions when using the "Evaluate answers" functionality. This view then will also profit from the reduced amount of vertical space used.

Navigation between questions

Navigation between questions should be made easier for students and focus on one question at a time. Therefore, an improvement would be to introduce two buttons in the navigation bar that can be used to navigate one question forward or backwards. Pressing the respective button will cause to show the next or previous question, regardless of whether the current question has already been answered. This leads to the same level of freedom when navigating polls that the current state of the application allows.

Swiping is a widely spread way of interacting with a user interface on smartphones or tablets. Consequently, a student might find it intuitive to use these gestures while using AMCS. Navigating between questions should be possible by swiping left to go forward or right to go backwards. The combination of swiping and the provision of buttons to navigate in the navigation bar helps not to break uniformity between different platforms. This will result in the fact that students on smartphones and tablets have buttons and swipe gestures available to them, while users on laptops and PCs without touchscreens can use the buttons. In addition to that, the student can use the dot bar to freely select a question they wish to answer or review. This eases navigation within a poll, no scrolling is required anymore.

3.2.3 Course View

The *Course View* has an important function in terms of usability since it acts as a filter for lectures belonging to a certain course. Users must be given the opportunity to sort and filter a list of elements, which is why this function has to be preserved in the redesign. However, as described in 3.1.2, the *Course View* has a redundant nature as it looks and feels nearly identical to the *Main View*. Furthermore, some potentially confusing click paths lead to the *Course View* as elaborated in section 3.1.4. As outlined in segment 3.2.1, the *Drop Down Menu* serves as a filter for courses, rendering the *Course View* obsolete. It is therefore dropped by the redesign. This results in an additional side effect in form of reduced amounts of click paths and stronger interconnectedness between all the views (see Figure).

3.2.4 Navigation

4 Implementation

5 Evaluation

6 Conclusion

.1 Appendix Title

This is the text of the appendix, if you need one.

Literaturverzeichnis

[Smith et al.(2009)Smith, Jones] P. Q. Smith, and X. Y. Jones. ...reference text...