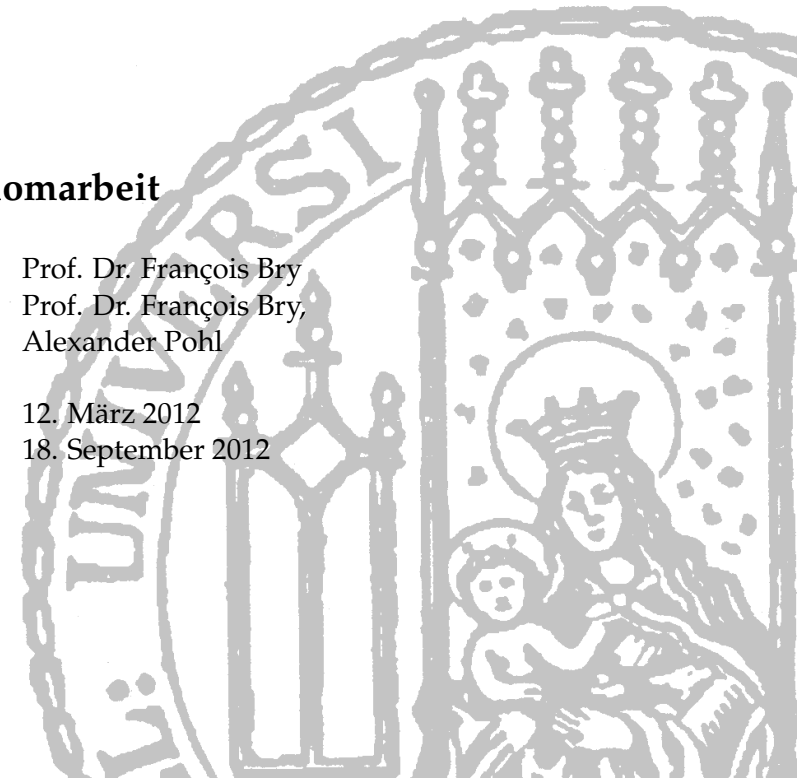
INSTITUT FÜR INFORMATIK
der Ludwig-Maximilians-Universität München

BACKSTAGE MOBILE

Merging Usability Guidelines with Educational Requirements

Julia Hadersberger



Diplomarbeit

Aufgabensteller	Prof. Dr. François Bry
Betreuer	Prof. Dr. François Bry, Alexander Pohl
Beginn am	12. März 2012
Abgabe am	18. September 2012

Acknowledgements

I would like to thank several people, who supported me throughout this thesis and helped me to reach my goal.
First and foremost, I would like to thank Prof. Dr. François Bry for offering this diploma thesis and the advice and ideas he provided.
Besides, I would like to thank Alexander Pohl not only for suggesting me this thesis, but also for his guidance during the development of my work. He helped me whenever I needed support and contributed a lot of ideas during fruitful discussions.
I want to express my eternal gratitude to my family, especially my parents for their support and love. It would not have been possible to study media informatics without them keeping me grounded.
Last but not least, I would like to thank Johann Kratzer, my eternal love, best friend and the best thing that ever happened to me. His love, help and humour kept me going throughout my studies. Completing this work would have been more difficult without his support and encouragement.

Erklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbstständig angefertigt, alle Zitate als solche kenntlich gemacht, sowie alle benutzten Quellen und Hilfsmittel angegeben habe.".

München, den 18. September 2012 Julia Hadersberger

Abstract

Smartphones and tablet computers enjoy great popularity because of their – in contrast to notebooks – light weight and small size. Therefore, it is not surprising that their ubiquitous use became common even for learning and that users claim a more sophisticated usability for websites and applications.
Backstage is an educational desktop application for the concurrent use during a lecture, enhancing the communication between the lecturer and the students. In the design process of the desktop application educational claims were focused and sometimes even preferred over general usability guidelines. Due to a growing need for the support of smartphones the desktop application needs to be transformed for mobile devices.
While usability and design guidelines for desktop applications are very well established, designing a user interface for smartphones and tablet computers is still challenging. Especially the much smaller display, which does not only serve as output but also as input, demands for an appropriate redesign if a desktop application shall be made available via mobile devices.
Accordingly, the purpose of this diploma thesis is to redesign the user interface in order to maximize usability, while preserving consistency with the design of the desktop application and its educational claims. As a result of literature research, guidelines and design principles for mobile devices in literature will be identified. These and the educational claims of the desktop version will be considered while designing a suitable user interface and possible conflicts discussed. Afterwards a paper prototyping study will reveal usability misconceptions and further ideas for improvement. The results of the study will be taken into account for creating and implementing the final design.

Zusammenfassung

Smartphones und Tablet-PCs erfreuen sich aufgrund ihres – im Gegensatz zu Notebooks – leichten Gewichts und ihrer geringen Größe großer Beliebtheit. Daher ist es nicht verwunderlich, dass ihr allgegenwärtiger Gebrauch sogar zum Lernen üblich ist und dass Nutzer eine hochentwickelte Benutzerfreundlichkeit für Webseiten und Anwendungen fordern.
Backstage ist eine bildungsbezogene Desktopanwendung für die gleichzeitige Nutzung während einer Vorlesung, um die Kommunikation zwischen Dozent und Studenten zu fördern. Während des Designprozesses für die Desktopanwendung wurden didaktische Forderungen fokussiert und manchmal sogar gegenüber allgemeinen Richtlinien zur Benutzerfreundlichkeit bevorzugt. Aufgrund der wachsenden Nachfrage nach Unterstützung für Smartphones muss die Desktopanwendung für mobile Endgeräte umgewandelt werden.
Während Richtlinien zu Benutzerfreundlichkeit und zum Design von Desktopanwendungen bereits sehr gut ausgearbeitet sind, ist es immer noch eine Herausforderung eine Benutzeroberfläche für Smartphones und Tablet-PCs zu gestalten. Insbesondere das sehr viel kleinere Display, welches nicht nur der Ausgabe sondern auch der Eingabe dient, erfordert eine entsprechende Neugestaltung wenn eine Desktopanwendung für mobile Endgeräte bereitgestellt werden soll.
Demzufolge ist der Zweck dieser Diplomarbeit die Benutzeroberfläche neu zu gestalten um die Benutzerfreundlichkeit zu maximieren und dabei die Konsistenz mit dem Design der Desktopanwendung und ihrer didaktischen Forderungen zu bewahren. Infolge einer Literaturrecherche werden Richtlinien und Designprinzipien für mobile Endgeräte ermittelt. Diese und die didaktischen Forderungen aus der Desktop Version werden während dem Entwerfen einer passenden Benutzeroberfläche berücksichtigt und mögliche Konflikte diskutiert. Anschließend wird eine Papierprototyp Studie Missverständnisse bezüglich der Benutzerfreundlichkeit und weitere Verbesserungsvorschläge aufdecken. Die Ergebnisse der Studie werden während der Erstellung und Implementierung des endgültigen Designs Beachtung finden.

Contents

1 Introduction **1**

2 Terms and Scientific Fields **3**
2.1 User Interface Design 3
2.2 Interaction Design 5
2.3 User Experience Design 6
2.4 Usability 7
2.5 Other Terms 8

3 The Desktop Application of Backstage **11**
3.1 Usability Guidelines for Desktop Applications 11
3.1.1 Guidelines by Ben Shneiderman 12
3.1.2 Guidelines by Donald A. Norman 15
3.1.3 Guidelines by Alan Dix 18
3.2 Educational Guidelines and Claims 19
3.3 Design and Functions of Backstage 20
3.3.1 Students' Interface 20
3.3.2 Lecturers' Interface 24
3.4 Evaluating the Design and Usability 26
3.4.1 The Usability of Backstage 26
3.4.2 Educational Claims in Backstage 32
3.4.3 Usability Guidelines vs. Educational Claims: A Conflict in Backstage? 33

4 Backstage Mobile **37**
4.1 Challenges of Mobile Devices 38
4.2 Usability Guidelines for Mobile Applications 39
4.2.1 Guidelines by Jun Gong and Peter Tarasewich 40
4.2.2 Guidelines by Jonna Häkkilä und Jani Mäntyjärvi 42
4.2.3 Guidelines by Daniel Su Kuen Seong 43
4.2.4 Guidelines by Erik G. Nilsson 45
4.2.5 Analysis of the Mobile Usability Guidelines 47

4.3 Design and Paperprototyping 48
4.3.1 First Design and Storyboard 48
4.3.2 Rapid Development by Paperprototyping 58
4.3.3 Resulting Design 64

5 Implementation **71**

6 Conclusion and Future Work **81**
6.1 Future Work and Challenges 81
6.2 Conclusion 83

Bibliography **107**

CHAPTER 1

Introduction

Mobile devices, like smartphones and tablet computers enjoy great popularity and reach more and more people. Reading the newspaper on an iPad or checking emails with an Android phone, while commuting to and from work or university has become routine, especially for young people. Therefore, the ubiquitous use of such mobile devices is also more and more supported, not only by providing the adequate infrastructure, but also by redesigning websites especially for the mobile access and offering user-friendly applications, so-called apps, customized for smartphones and tablet computers.

Due to the rapid development and upcoming market it is not surprising that mobile devices have also a great potential for educational and learning aspects. Several app-developers have recognized a market niche and offer, for example interactive language courses, so that customers can use their free time on their way reasonably. But the potential of mobile devices in education can be much greater, keeping in mind that many students in a lecture carry at least one electronic device with her, which is capable of wireless LAN or other Internet connections. It is remarkable that mobile devices are even sometimes preferred over notebooks due to their light weights and small sizes.

Backstage is an educational desktop application for the concurrent use during a lecture, uniting the audience and lecturer as one community, while enhancing discussions and questions at the same time. Most notably in lectures with a large audience Backstage can be very helpful to promote interactivity, not only by reading and writing messages, but also by rating the messages of the others and participating in quiz questions provided by the lecturer.

First studies with the prototype of Backstage show that there is a growing need for the support of smartphones and tablet computers. As a result the desktop version of Backstage has to be transformed adequately for mobile devices. During the redesigning process several aspects have to be considered: First, during the development of the desktop version of Backstage several educational claims were in focus and sometimes even preferred over general usability guidelines. Second, good

usability is of great importance, so that students will use Backstage on a regular basis. And third, special attention has to be paid to the small screen of smartphones, as the desktop design is too space-intensive to be directly transferrable to mobile devices without severely affecting usability.
The aim of this diploma thesis is to redesign and implement the user interface of Backstage for mobile devices, in order to maximize usability while preserving consistency with the desktop application and its educational claims. The challenge is to keep the balance between the concept of Backstage and user-friendliness on a small screen and to decide where old concepts of the desktop version have to give way to new ideas better suited for the mobile context.
Hence, this thesis starts with an overview of the different terms and scientific fields which are important for the research in Section 2. Section 3 mentions and explains the most important usability guidelines for desktop applications, as well as the educational guidelines and claims that are relevant for Backstage. Afterwards the desktop version of Backstage is presented and its design and usability is evaluated. In Section 4 the special characteristics of mobile devices are examined and mobile usability guidelines are presented. It is analyzed whether the mobile usability guidelines conflict with the educational claims and how they can be combined. Afterwards the design process of Backstage Mobile will be described, including a paperprototyping study and the resulting design. Section 5 discusses the final implementation of Backstage Mobile and its challenges. In the end Section 6 will not only give a summary of this work, but also an outlook on future work and further concepts that were not implemented until now.

CHAPTER 2

Terms and Scientific Fields

Many different terms occur in the fast developing scientific field of technology and design. Especially "user interface design" is a widely-used term that thus, without narrowing down to a certain perspective, can only be broadly defined. In fact, authors often use those terms without mentioning any definition or explication of it and hence, suppose that the reader knows and shares their opinion. This situation gets even more complicated, as only a few terms are concretely standardized by, for example, the ISO. In fact, the terminology of user interface design "[...] has always been fluid and subject to social, political and technological shifts" [27, p. 21]. Moreover, many scientific fields regarding design and hence many expressions including this term appeared during the last years. These include appearance design, interaction design, user interface design and digital product design to name but a few. However, it is not only difficult to distinguish the design terms among each other, but also to differ them from other similar non-design terms, like usability or information architecture. To get a better overview of all the interwoven terms and scientific fields, this section tries to define each one and estimate its importance for this thesis.

2.1 User Interface Design

As already mentioned, user interface design is a quite fuzzy term and its definition depends on who is using it, as each scientific field has another perspective [27]. The user interface itself is the "[...] part of the product (hardware and/or software) that the user perceives and must deal with" [2, p. 23], comprising ergonomics and aesthetics, although this opens up a lot of further ways for interpretation [2, 27]. It can also be defined by its five so-called user interface components: metaphor, mental model, navigation, interaction and appearance.
The metaphor is the fundamental concept that is conveyed to the user, trying to help her to understand the digital system. Usually it resembles something the user

knows or is familiar with, so that she can enjoy and remember its functions and entities. The most popular metaphor regarding computers is the desktop metaphor of modern operating systems. Further examples are the trashcan, windows, chatrooms, the electronic shopping cart and so on. [27]
Backstage has in fact two dominating metaphors: the annotation of slides and the chatroom. Attending a lecture, many students take printed slides along to add handwritten annotations to them. Additionally, arising questions during the lecture are usually related to content discussed in the slides, hence questions can be seen as a special form of an annotation. Therefore, messages in Backstage are represented by icons of different category and have to be placed on the related slide at the according position. On the other hand it can be said, that Backstage furthermore makes use of the typically chatroom metaphor, because students and lecturers can communicate with each other by writing short messages. In fact, the whole dashboard including the chat, slides and annotations represents the lecture hall, because every person participating in Backstage during that session is supposed to also be present in the lecture hall. These and further mentioned concepts of Backstage during this chapter are more concretely explained in Section 3.
The mental model component structures and organizes functions, data, tasks and roles. It is often implicitly included into the design process by making use of user models, where personas, needs, goals and roles are defined. Moreover, terms like user-centered or task-oriented design are closely related to the mental model. [27] The data or tasks are structured according to the information the user needs or the tasks she wishes to complete, thereby creating the appropriate mental model for the application. The mental model of Backstage is very multifaceted, as it can be structured regarding different aspects. For example, it can be said that the structure is given by dividing the users in the two roles student and lecturer, which also have slightly different interfaces. As Backstage is still a prototype, a task-oriented concept was pursued during its implementation, focusing on the tasks the user wishes to accomplish and the functions really needed. This is another form of structure, which can be recognized in Backstage by the different parts of the main view during a lecture. For example, messages for reading are placed on the left side, slides are in the middle and writing messages is at the top of the view. This structure abounds even more in Backstage Mobile by using tabs for different tasks.
As can be imagined, the navigation is the movement through the offered and structured tools and content, hence movement through the mental model. Typical user interface elements that enable movement within an application are all items that support some kind of dialogue, especially input-items like dialog boxes and menu items. [27] The accordingly necessary hardware and equipment for input- and output-mechanisms belongs to the component of interaction. It comprises not only local equipment like keyboards, mice and touchscreens but also wider aspects like usage scenarios, task activities and context issues. [27] In regard to Backstage, the interaction component is for example represented by the hardware the students use and the usage scenario of sitting in the lecture hall. Furthermore, even questions which are asked directly by raising the hand and speaking out loud could be implicitly part of the interaction component, as this situation would at least belong to the context.

The last component appearance regards to everything that can be perceived by ones senses, including among other things colours, sounds or vibration modes [27]. In Backstage the appearance consists of the visual perception of colours and pictures, as due to the context sounds and vibrations are impossible.

So the user interface design is the development, hence analysis, design, implementation and evaluation of a user interface and thus its five components [27]. On the one hand this definition opens up the way for many interpretations. Some authors claim that even non-digital products can have a user interface, for example a printed poster that has to be read and therefore interacted with by the user [2]. Although this seems to be partially true, as this interpretation is attended by the fact that everything a user interacts with has to have a user interface, it is questionable if the five user interface components can be applied to non-digital products. What is the metaphor, mental model and navigation of a poster?
Interaction designers tend to say, that user interface design is in contrast to their discipline about the statics of an application, like designing the screen layout and where to put the content and system information [21]. This perspective excludes the navigation and interaction components totally and focuses on the appearance of an application.
As there is no congruent definition of this term and each author uses and interprets it the way she likes, it comes to one's mind that the user interface design might be a disused term and should be replaced by more concrete terms like interaction and appearance design.
Whatever definition might be preferred, user interface design is definitely one of the main aspects of this thesis. It is necessary to not only design an appropriate appearance for Backstage Mobile, but also to reconsider new concepts for navigation and hence the mental model, as well as for interaction. The metaphor of the desktop version, a presentation and communication platform like a classroom, stays the same.

2.2 Interaction Design

Interaction design is quite as difficult to define as user interface design. The term has been used over many years by many different scientist involving different activities [21]. This discipline focuses on the behaviour and use of a product. The term interaction design should not be mixed up with the user interface component interaction, although this component is one part of it. However, interaction designers perceive their discipline consists of more than this single aspect.
According to the user interface components this scientific field consists of metaphor, mental model, navigation and interaction and hence, is a special case of user interface design, omitting the component of appearance to solely design the events of a product neglecting its look [2, 27]. Interaction designers try to create products that not only offer a good usability, but also a great customer experience, so that their products assist their customers everyday [21].
As customer experience seems to be almost the same as user experience, we pre-

fer to omit this aspect of interaction design. Hence, this discipline develops those components of an application or product which can be interacted with including all the effects one interaction can provoke.
Therefore, interaction design is of special interest in this thesis, as mobile devices support different ways of interaction and input. The concept for Backstage Mobile has to be adapted to those small devices and can benefit from new interacting modes.

2.3 User Experience Design

Another well-known term is the user experience design, sometimes also called experience design. To keep it simple, this term comprises everything related to the design of a product [2]. Therefore, it does not only include all of the five mentioned user interface components above, but also extends it by two more components: information and functionality. Information denotes the real content or data of a product [2]. The according example in Backstage are the messages, slides, statistics, quiz questions and answers. The component functionality stands for the functions a product has and the work it can do [2]. The functionality component of Backstage for example consists of the different tasks a user can accomplish. Hence, the components information and functionality could also be simply called content and tools.
Some authors criticize this discipline term, as it pretends to consciously influence, control and "design" the experience a customer has. This seems to be a vague statement, as hence each artist or architect could also claim to be a user experience designer influencing her customers [27]. On the other hand, further aspects like the brand identity, packaging of the product, purchase, customer service, product upgrades, compatibility and many more are strongly connected to the product itself and represent parts of the overall experience the customer has [2]. A product can be perfectly designed and offer a good usability, but an insufficient customer service or upgrade support might affect her over all user experience and she might buy another product the next time. The other way round, a strong and positive brand identity might help the user to overcome little deficits of the products. Hence, user experience is the whole package that is presented to a user and the way a product makes her feel [21].
Although there exist a lot of different definitions their mutual basis seems to be met by ISO 9241-210 of 2010, which defines the term as "a person's perceptions and responses that result from the use or anticipated use of a product, system or service" [41, 26].
As a conclusion it can be said, that the user experience can be influenced to a certain degree by the design of a product. But user experience designers have to handle many more influencing aspects and hence need the knowledge of all the different involving disciplines. Furthermore, it is complicated and almost impossible to measure user experience. Although some aspects of this discipline might be of interest not only for Backstage Mobile, but also for the desktop application, user experience design is not in the focus of this thesis. This can also be reasoned by

the fact that until now Backstage and Backstage Mobile will not be sold for money and economic aspects like product packaging and brand identity do not matter yet. As already mentioned, some aspects like customer service or upgrade support can help to promote the use of the application, so that as many students as possible will participate. Given the fact that we cannot measure the user experience of the students and that their experience is mostly influenced by the application itself, the discipline of user experience seems to be irrelevant for this thesis.

2.4 Usability

Another quite popular term is the usability of a product. This term is also defined by the ISO 9241-11 as the "extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" [4]. The important part of this definition lies within the words "with effectiveness, efficiency and satisfaction". It implies not only that a user should be able to reach her goals, but also in a quick way without circumstance. Furthermore, the user should be satisfied by the way the product works.
Jakob Nielsen defined usability more comprehensible as a "quality attribute that assesses how easy user interfaces are to use" [33]. He substantiates his definition by the five concrete quality attributes learnability, efficiency, memorability, errors and satisfaction. These components are often cited as his guidelines, because it could be interpreted that way, but in fact they are just a more detailed definition of usability.
Learnability defines how easy it is for new users to complete basic tasks the first time they use the system. Furthermore, efficiency denotes how fast users can perform different tasks after they have learned the design. In contrast, memorability determines how easily users can resume their skills when they return to the application after a period of not using it. The error component is a fundamental quality attribute for usability. It does not only consider how many errors occur and users make, but also how fatal they are and if users can recover from them. The last quality attribute called satisfaction measures how pleasant it is to use the design and hence, seems to be a bit vague. [33].
It is hardly possible to classify usability by the components of a user interface mentioned in the beginning of this chapter, because usability is connected to and affected by all of them. Misconceptions could lead to metaphors that are not understood by the user or mental models that do not match the tasks she wants to fulfil. Also navigation and interaction have to be designed with a focus on usability, as users will often make mistakes if the concept behind those components is misleading and not matching their perception. Even the component of appearance can lead to a limited usability. For example, if the design of an icon is misleading and cannot be understood by the user or if some text is hardly readable because of the disadvantageous colouring.

As can be seen, user experience design resembles usability, so that it is necessary to have a look at their differences. In contrast to user experience design, usability

is very limited to user cognition and performance [26]. The questions of usability tests focus more on problems and misleading concepts that might occur than how the participants like the product. User experience design is more focused on the way a product makes the user feel, the emotions that are connected to the product, what users like and dislike about it and the service and image of the responsible company.
It can be concluded, that usability is the main focus of this thesis. We want to inspire as many students as possible to participate in backstage because of the benefit for both, the students and the lecturer. Therefore, it is not only important that they get a chance to participate with as many different devices as possible, but also to enthuse them to use Backstage and Backstage Mobile on a regular basis. Hence, the user interface has to suit their tasks and offer good usability, so that the students stay interested in using this application and do not get annoyed by an insufficient design.

2.5 Other Terms

There are many more terms including design or connected to it. As they are not as regularly mentioned like the other four terms explained in the above chapters, they are just shortly described here.

The term appearance design "determines the surface form of a digital product" and is due to the domination of visual elements in digital product often also called visual design [2, p. 21]. This alternative name is misleading, as appearance design also comprises the design of sounds and other perceptions [2, 26]. In regard to the five components of a user interface, it would be connected to appearance and metaphor. Appearance design plays a minor role for this thesis, as most of the visual design was already done for the desktop application and will be adopted for the mobile version. The underlying reasons for this decision and further aspects of the appearance will be discussed in Section 4.

Another term that occurs is information architecture. It constitutes which data and content is used, including where and how it is represented and accessed. Therefore, it consists of the components navigation, mental model and information and as can be seen is not focused on the visual appearance of the data [2]. This discipline tries to structure the data and find a way to adequately provide it to the user. As will be seen in Section 4, the information architecture is less relevant for Backstage Mobile as the structure and organization of the data will be mainly adopted from the desktop version.

The last term discussed in this section is the digital product design, which occurs very seldom. It comprises the whole development of a digital product and therefore, not only the user interface design and hence its five components metaphor, mental mode, navigation, interaction and appearance, but also information and functionality of a product. This discipline does not only focus on the development

of the product including hardware and the user experiences that might be made with it, but also the specification before and marketing after the development [2]. Having this in mind, it can be concluded that this thesis could be part of a digital product design, but it is no standalone project of this discipline. There is no measurement of the user experience, no marketing for the product and no economic aspect.

CHAPTER 3

The Desktop Application of Backstage

In order to create a concept for the mobile version of Backstage it is advisable to analyse the existing desktop application. It has to be considered which concepts have proofed to be good and should be adopted by the mobile version and which need to be improved. Backstage was redesigned several times during its prototyping phase to fix bugs, get rid of obsolete concepts, add new ideas and solve usability problems. Nevertheless, the design, i.e. colours and the look of icons and the different modules, was changed with it several times. Although, usability was sometimes the reason to edit Backstage, new prototypes were never created according to usability guidelines. Instead, the developers trusted in their sense for usability without any research in literature.
To get a better overview of the way Backstage works and how the desktop application was designed, this section first summarizes the most popular usability guidelines found in literature. Afterwards some educational guidelines are mentioned and it is outlined which claims were made for Backstage and might still be important for a mobile version. The following chapter presents Backstage with its design and all the included functions, so that it is comprehensible which modules need to be adopted for Backstage Mobile. In the last chapter of this section the design and usability of the desktop application is evaluated, having in mind that Backstage is still a prototype.

3.1 Usability Guidelines for Desktop Applications

As already defined in Section 2, usability is more than just user-friendliness. Users need to be efficient and satisfied with the product. Misleading concepts, unnecessary trouble and often occurring errors frustrate the users and inhibit their motivation to use the application furthermore. Usability guidelines can help developers of applications to keep an eye on the user and her intuition instead of implementing concepts just the way they work. The efficiency and work quality can be improved,

furthermore, the users will need less support which leads to reduced costs [45]. A good usability of an application can influence the customer's experiences positively and hence improve the overall user experience of a product. This could have a positive effect on sales figures or on the frequency of use. Usability guidelines are suggestions for designing applications, patterns that have proven to be intuitive and comprehensible for the user. Furthermore, they not only show in which way to do it right, but also which way might be absolutely wrong. There is no standard solution for user interfaces as each application is very different, but usability guidelines can help to make the best of it. No one can tell exactly, when it is possible to neglect a guideline in favour of a certain function or where it is necessary to keep strictly to the guidelines. In fact, sometimes these guidelines even conflict with each other [47] and developers need to decide by themselves which one is more important and eventually prove their decisions by making use of a cognitive walkthrough, heuristic evaluation or usability study. Otherwise, gathering a lot of feedback about the use of an application during its prototyping phase makes it possible to diagnose usability problems in an early stage.
There are many usability guidelines and in addition each manufacturer has its own guidelines that concretely specify what standard applications should look like and for example which distance should be given between buttons in standard dialog boxes. As Backstage is a web application independent of the operating system and its recommendations, manufacturer guidelines can be neglected. Instead it is focused on the most famous general usability guidelines and best practices. Therefore, this chapter will present Shneiderman's "eight golden rules", Norman's "seven principles" and Dix's guidelines for usability.

3.1.1 Guidelines by Ben Shneiderman

One of the most famous guidelines for good user interface design are Ben Schneiderman's so-called "eight golden rules" [10, 21]. The following listing gives an overview of those eight rules, before they are explained in detail.

1. Strive for consistency
2. Enable frequent users to use shortcuts
3. Offer informative feedback
4. Design dialog to yield closure
5. Offer simple error handling
6. Permit easy reversal of actions
7. Support internal locus of control
8. Reduce short-term memory load

The first and one of the most important guidelines focuses on consistency. This comprises different ways of consistency and can be divided in three main aspects: internal consistency, consistency with other applications and consistency with real life experiences [21].

Internal consistency denotes every aspect that should be consistent within one application [21]. On the one hand it includes visual aspects, like the colour scheme or font used within the whole application should be consistent or that the same symbol or icon should not be used for two different functions. On the other hand it focuses on the structure of an application, including claims like the layout should stay the same whenever possible, so that the user will not get confused. As a concrete example it can be thought of a drawing application which offers its colour palette on the right side of the screen. It would be confusing for the user if the colours would be suddenly placed on the left or bottom of the screen, just because the user chose another tool for drawing a circle. In addition, the menu layout should stay the same, especially the order of the menu items. For example, if the menu item "import" is the fifth entry in the menu "file", it should always be placed there and not in another menu in the next screen. Furthermore, it is advisable to use the same terms for the same functions.
To stay consistent with other applications, it is necessary to follow common principles and to respect and reuse global commands [21, 1]. Well-known examples are the menu entries and shortcut keys for cut, copy and paste or that the application menu is always placed at the top of the window. Whenever possible existing patterns of other applications should be adopted, as the user might already be familiar with it. Contrariwise, it is not advisable to radically change existing concepts that have proven to be useful, unless it is necessary and reasonable.
The last form of consistency regards the personal knowledge and life experience of the user [21]. This kind of consistency is closely related to the metaphor component of user interface design, mentioned in Section 2, as each metaphor tries to transfer the knowledge and experiences of the real life to the virtual world. Hence, the user interface or more exactly the metaphor used for it should be consistent with the real world. For example, the recycle bin metaphor represents that a file is deleted, i.e. thrown away [1]. The real life bin keeps all the trash until someone takes it out, so that whenever something is accidentally thrown away, it could be fetched out of the bin. Thus, users associate some kind of undo-function with a recycle bin, therefore, this metaphor should be consistent with the real life and offer this possibility until someone "takes out the trash". Another example are the hand cursor in combination with a map. The hand cursor is obviously represents the hand of the user, so if someone clicks on a map and moves the cursor, i.e. the hand, to the left the map should also move to the left and the screen should show the right section of the map.

The second guideline postulates to offer experienced users shortcuts. This can be shortcut keys for menu commands or easy usable wizards for regular tasks [10]. Furthermore, customizable menus like the start menu or the favourites in an Internet browser can also be seen as shortcuts [21]. Shortcuts help experienced users to be even faster and more efficient in accomplishing the tasks.

Another very important guideline concentrates on offering informative feedback. For every action of the user appropriate feedback should be given. The more regular an action happens, the more unobtrusive the feedback should be [10]. For

example, if a user saves a file within a writing application by pressing the according shortcut keys, expressions like “document saved” within the status bar at the bottom of the window will give appropriate feedback. Otherwise, unusual tasks offer more prominent feedback and even additional information if the task will take some time. For example, if the wants to install a new application, the progress and remaining time is often shown in the centre of the window [21]. Of course, the feedback has to be easily understood by the user and therefore should avoid technical terms and long sentences.

The fourth guideline to design dialogs to yield closure claims that each interaction should have a beginning, middle and end [21]. By appropriate feedback at the end, the user will know when the task is completed [10]. Task wizards are a quite obvious example for this guidelines, as they have one screen in the beginning to explain what the wizard will do, several screens, asking the user to input data and show the progress and one screen at the end which explicitly informs the user about the completion. A more unobtrusive example is writing an email, which begins with opening a new email, proceeds in the middle by filling in the address, subject and message and ends by pushing the send button, which will effect that the window closes and the task is completed.

The next guideline, offer simple error handling, supposes that the prevention of errors is taken for granted [21]. Possible and especially serious errors should be avoided, by for example guiding the user through wizards and disabling functions and buttons which are not allowed. As not each error can be prevented, the interface should help the user to detect occurring errors, for example by checking if each form field is filled. Furthermore, the system should give informative feedback about where and why the error happened and how the user can recover from it [21]. Therefore, the error message should be easy to understand for the user and suggest constructive solutions. If possible, the system state should not be changed if an error occurs or at least offer an appropriate undo function. Which in turn leads to the next guideline “permit easy reversal of actions”, that is not only important for serious errors, but also minimal mistakes by the user, like for example a typing error. The effect of the undo function is even more extensive than supposed at first sight, because users who feel save in their environment as they can easily reverse any wrong action are more curious and likely to explore the environment on their own [10].

To support the internal locus of control is the seventh guideline and is based on the terms "internal and external locus of control", which are adopted from psychology. They determine whether a person thinks her behaviour is guided by her own decisions (internal) or by external circumstances or fate (external) [42]. Hence, by supporting the internal locus of control Shneiderman postulates that the user should always feel in control of the system instead of being controlled by the system [10, 21]. Therefore, the system should react to the actions of the user and not start procedures on its own. For this guideline it needs to be balanced where automation of tasks is reasonable and when to ask the user for confirmation. The user

will be unsettled, if the system obviously works on something without giving the user feedback about what is happening. On the other hand, applications often offer an opportunity to choose, if the user wants to be asked about tasks or if they should be done automatically, for example, installing system updates or new antivirus software.

George Armitage Miller, a famous psychologist once wrote an article about the fact that the capacity of the human short-term memory is limited [28]. In the following decades his experiments were often misinterpreted and argued that the short-term memory can hold "seven plus minus two" information chunks. Although several authors try to demolish this long-living myth [14], it was once mistakenly the basis for the eighth guideline, claiming to reduce the short-term memory load. Nevertheless, users should not be stressed by memorizing too much information, numbers or terms [21]. Therefore, windows should be kept simple and instead of asking the user to memorize something, the system should repeat the respective input [10].

3.1.2 Guidelines by Donald A. Norman

Norman's principles are based on his theory of the so-called "Seven Stages of Action" [36]. Accordingly, the way humans try to accomplish tasks is divided into seven stages, one for the goals and three each for the execution and evaluation. The execution is defined as doing something to the world, while the evaluation represents the comparison between what someone intended to effect by the execution and what really happened. The cycle of the seven stages can be seen in Figure 3.1 and starts with a certain goal the user wants to achieve, which therefore is translated into an intention to act. Afterwards a sequence of actions is planned according to the intention and finally executed in the world, in order to achieve the goal. This half of the cycle starting with the goal and ending in the world represents the execution aspect. In contrast, the evaluation aspect is shown by the other half of the cycle, starting in the world and ending with the goal. Therefore, the state of the world is perceived by the user and the perception interpreted according to the expectations. In the end the interpretation is evaluated by comparing what was originally expected (goal) and intended and what really happened.

In relation to his action cycle, Norman defines two crucial points where an action could fail. The first one, the so-called gulf of execution, regards the "difference between the intentions and the allowable actions" [36], that means the user has to know which actions can be made on a system and if they match the intended ones. Good user interface design can minimize the gulf of execution. The other one, the so-called gulf of evaluation, "reflects the amount of effort that the person must exert to interpret the physical state of system and to determine how well the expectations and intentions have been met" [36]. Good design also minimizes the gulf of evaluation by supporting the visibility of the system status and its interpretation.

According to the seven stages, designers can ask themselves seven different questions to check if both gulfs are bridged as far as possible. Table **??** shows for each

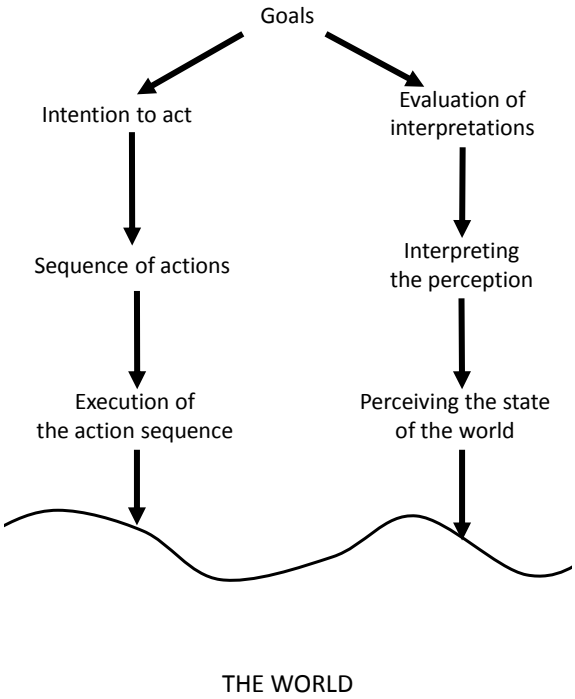


Figure 3.1: Norman's Seven Stages of Action [36]

stage in the action cycle the corresponding design question.

Summarizing those seven design questions leads to Normans four main principles of good design. First, ensure a high degree of visibility so that the user knows in which state the system is in and which actions she can perform. Second, present a good conceptual model with a high consistency between the presentation of operations and of its results. Third, offer good mappings so that the relationship between actions and results, controls and their effects and system state and visibility becomes clear. Fourth, provide continuously feedback for every preformed action [36, 10].

Furthermore, in conjunction with user-centred design, Norman suggests the following seven principles:

1. Use both knowledge in the world and knowledge in the head
2. Simplify the structure of tasks
3. Make things visible
4. Get the mappings right
5. Exploit the power of constraints
6. Design for Error

Stage	**Design Question: How easily can one ...**
Goals	... determine the function of the device?
Intention to act	... tell what actions are possible?
Sequence of actions	... determine mapping from intention to physical movement?
Execution of the actions sequence	... perform the action?
Perceiving the state of the world	... tell what state the system is in?
Interpreting the perception	... determine mapping from system state to interpretation?
Evaluation of interpretations	... tell if system is in desired state?

Table 3.1: Seven Stages and their Design Questions [36]

7. When all else fails, standardize

The first principle claims that interface designers should offer the required knowledge for a task either explicitly or by appropriate constraints. Furthermore, the effect of different operations has to be recognizable, so that it is easy for the user to build the according mental model and comprehend what is going on. In addition, this principle supports the efficiency of regular users.
Another important aspect is the simplification of the structure of tasks, thus preventing complex problems and high memory load. Therefore, continuous feedback and visibility of the system status are necessary and it might even be advisable to automate some parts of complex tasks.
To bridge the gulfs of execution and evaluation it is necessary to make things visible. Hence, it should concretely be shown what and how operations can be done by the system and in addition what actions effect.
The fourth principle claims to get the mappings right, which regards users intentions that should map onto the system controls. This does not only mean, that for every user intention the appropriate interface gadget has to be found, but also that minimal movements for example with a slider gadget should also have a small effect, while large movements have large effects.
In order to avoid errors, the power of constraints should be exploited. This could easily be compared with a jigsaw puzzle, where constraints help the user to do it right and complete the task. Whenever possible constraints should help the user to do nothing but the correct action.
As not every error can be avoided, it should be designed for error. Hence, possible errors should be considered and appropriate recovery functions provided.
The last principle called “when all else fails, standardize” determines that whenever there is no standard solution for a problem or design question, the designer should create one, stick with it and reuse it when possible. By this the user will have to learn them only once and know them the next time. For example, the critical controls of a car are always at the same place and work in the same way (brake, accelerator, steering wheel), while other more unimportant controls like air condi-

tioning and radio are constructed differently [36, 10].

3.1.3 Guidelines by Alan Dix

As Alan Dix published his principles to support usability, he already emphasized that this catalog might not be complete and will probably be extended in future, whenever new insights about usability are gained. Instead of claiming to publish a full list of guidelines, Dix focused more on their structure and divided those principles in three categories.

The first category is called *learnability* and defined as the "ease with which new users can begin effective interaction and achieve maximal performance" [10]. In order to support learnability, interfaces should be *predictable*, that is help the user identifying what future actions will effect by the knowledge she gained from past operations. In contrast, *synthesizability* claims that the system should help the user to recognize which effect past operations had on the current state. The *familiarity* principle is closely connected to the idea of a metaphor for an interface, as it is described as the degree to which user's real-world knowledge can be transferred to the interaction with a new computer system. *Generalizability* describes to what extend the knowledge and interactions of one system can be used in other similar applications and situations. The last principle for supporting learnability is *consistency*, which was already mentioned in association to Shneiderman's eight golden rules.

The second category for the principles is called *flexibility* and defined as the "multiplicity of ways in which the user and system exchange information" [10]. The principle of *dialog initiative* assumes that like each communication the dialog between a system and its user can be started from both sides. On the one hand, the system can require the user to answer a dialog box while it prohibits other interactions. On the other hand, the user can initiate any action towards the system. Dix emphasizes that user pre-emptive dialogs, that is communication initiated by the user, is generally preferred over system pre-emptive dialogs. Nevertheless, some situations require system pre-emptive dialogs in the majority of cases because of safety reasons.
Furthermore, *multithreading* should be supported, so that the user can work on more than one task at a time. According to Shneiderman's rule that the user should feel in control, Dix postulates *task migratability*, which means a user can do a given task on his own or pass the control on to the system. For example, the spell-checking in a document can be done manually by the user or automatically by the system. In fact, this situation will be solved by a cooperation of user and system in most cases, as the system will always make suggestions and ask the user which spelling to apply in uncertain cases.
Another principle supporting flexibility is *substitutivity* of equivalent values during input and output, which helps the user to prevent from errors of unnecessary calculations. For example, the user is allowed to specify the margin of a document either in inches or centimetres.

The last principle in this category is *customizability*, which can be either system-initiated (adaptivity) or user-initiated (adaptability). The former modifies the user interface according to the preferences learned from the user, for example file open menus that remember which files were lately used. The latter allows the user to customize the user interface the way she prefers it, for example, changing the order icons in the start-menu of Windows or modifying colours and font sizes for better readability.

Dix's last category is called *robustness* and determines the "level of support provided to the user in determining successful achievement and assessment of goals" [10]. This category takes up some of the ideas of Norman's action cycle. The first principle *observability* correlates to Norman's postulation to make things visible. Thus, the user should be able to perceive the state of the system by according representations. Furthermore, *recoverability* corresponds to Norman's principle of designing for error, which means whenever an error occurs the user should be able to recognize and correct it. The principle of *responsiveness* defines that the response time, so the duration until the system shows that the state has changed, should be as short as possible. If the situation requires a longer duration, the user should be informed about what is happening and how long it will approximately take. Furthermore, the response time should be stable, that is similar actions should require the same duration. Dix's last principle called *task conformance*, reflects Norman's basic conclusion derived of the action cycle. Task conformance consists of task completeness and task adequacy, and requires the system to support all of the tasks the user wishes to perform and in such way that the user understands it.

3.2 Educational Guidelines and Claims

As for each learning situation, educational guidelines can also be applied for computer-mediated lessons. With the rise of computer-mediated collaborative learning systems a lot of educational claims were analysed and important aspects to enhance the learning effect amassed. Backstage was developed to improve the learning situation in overcrowded lectures and facilitate a better communication between students and lecturers [7]. Therefore, it is necessary to take those educational aspects into account.

A first and quite simple educational claim is that system designers and programmers should focus on their educational goal and dispense with unnecessary functions and information [40, 48]. Furthermore, meaningful content should be chosen and presented in appropriate ways, so that the educational goal is supported accordingly [48, 45]. One very important and non-controversial theory claims that activity definitely enhances the learning effect [40, 48]. Hence, it is advisable to provide different ways of learning, for example, several kind of media types [48]. Conversations play a major role for enhancing the learning effect by activity. Some authors actually equalize learning as a form of permanent conversation with teachers, fellow students or even oneself, while dealing with the comprehension of the

subject matter [46]. Hence, the contact and communication between students, lecturers and teachers does not only enhance the social aspect of learning [48], but also facilitate the reflection and comprehension of learning content.
For further processing of subject matter, assessment issues like tests or quizzes, which can also be done externally and do not necessarily have to be part of the system, should be considered [40, 48]. Consequently, users should also receive constructive feedback, whereupon implicit, unobtrusive feedback is used for system input, while explicit feedback can be used for test solutions and learning results [48, 38].
Especially computer-mediated learning necessitates more guidance, because the sequence of actions within the system is determined and arranged by the educational curriculum [38]. Some authors suppose that implicit guidance should be preferred over explicit guidance, which means the necessary actions should rather be clear by the way the content is presented than by giving instructions [38]. On the other hand especially authors familiar with the educational science claim that external instructions, so-called external scripts, are necessary for learning [3]. The best solution will probably be a mixture of both theories, giving instructions where it is necessary and dispense with them where possible.
A last but very important claim in regard to computer-mediated learning postulates that such applications should be as easy to learn and self-descriptive as possible, because the learners will only use it for a limited time, hence less time on training should be needed [38].

3.3 Design and Functions of Backstage

As mentioned in the introduction, Backstage [7] is a microblog-based backchannel application, facilitating the communication between lecturers and students in large lecture classes. Users can write, read, answer and evaluate messages during the proceeding lecture. Simultaneously, the according lecture slides are shown and quizzes conducted, increasing the participation and awareness of the students and hence, supporting active knowledge construction.
Therefore, participants with laptops visit Backstage's web-address at the beginning of the lecture and login to their account. From this point on, the user interface looks a little bit different, depending on which role the user has (student or lecturer), hence the first description focuses on the students' interface, while the following chapter describes the differences in the lecturers' interface.

3.3.1 Students' Interface

As the students are logged in, the window contains a menu bar at the top which titles "Backstage" on the left and shows the user menu (shown by the user name and a down-arrow) and user avatar on the right. This user menu offers a logout function and the opportunity to edit the own profile (e.g. choose another avatar picture). Next to the user menu is a temporarily disabled speech bubble. This feature is not implemented yet, but it should show status messages for the user in

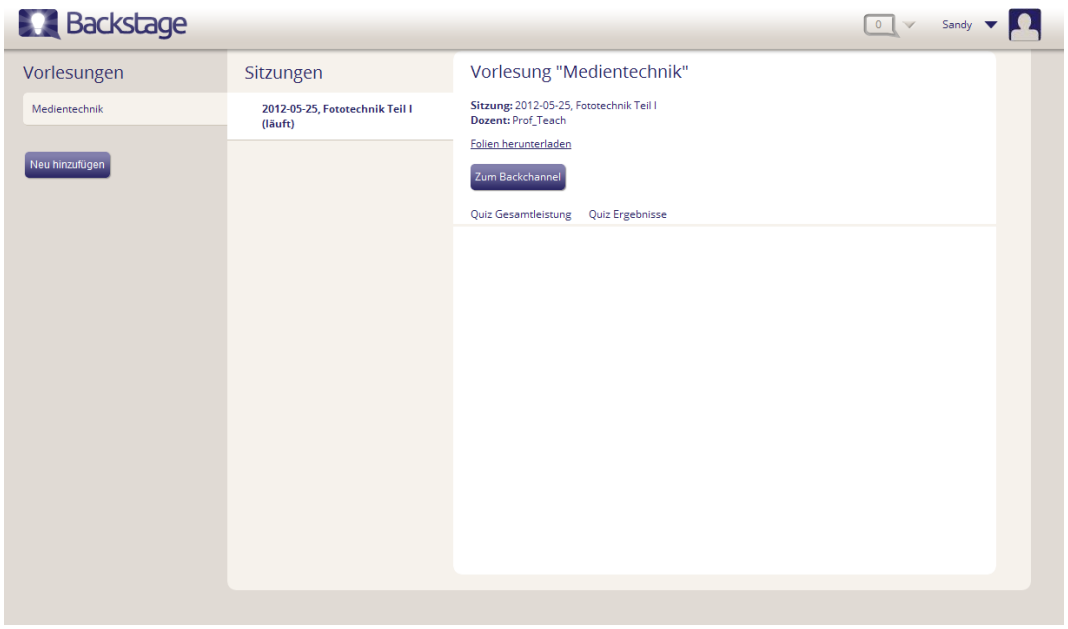


Figure 3.2: Lecture and Session Choice in Backstage

near future. In the main client area beneath the menu bar, students can choose an existing lecture by name in the left column or add a new one.
After clicking on the lecture the middle column represents the existing sessions, which are created by the lecturer and named by a date and title. As Figure 3.2 shows, more information about the lecture session (title, date, lecturer, quiz statistics) can be seen in the right column, as soon as one is chosen by clicking. Furthermore, the slides are offered for download and a button gives the opportunity to open the backchannel.
The main interface during a lecture, the so-called dashboard, can be seen in Figure 3.3. The slides are placed in the middle of the screen, as they are the main aspect during a lecture. Beneath the slide little thumbnails symbolize how many slides exists and which one is currently shown by the user (red frame). By clicking the left and right arrows between the shown slide and the thumbnails, the user can navigate through all of them. The down-arrow takes the user immediately to the slide, the lecturer talks about at the moment. It can also be navigated through the slides by using the offered shortcuts, namely the ALT-key in combination with the left or right arrow, the down arrow (actual slide of the lecturer) or the up arrow (first slide). Though, it is not implemented yet, it is planned to show the slide where the lecturer is by framing the according thumbnail with another colour. It is planned that the navigation through the slides offers a special feature. The lecturer can decide in the options, if students are allowed or not to navigate further than herself. For example, if the lecturer shows and talks about slide number seven, it is only possible to navigate in the range of slide one to seven for the students. This characteristic shall help the users to keep focused on the ongoing lecture instead of distracting the concentration by further looking ahead. Furthermore, it forbids the

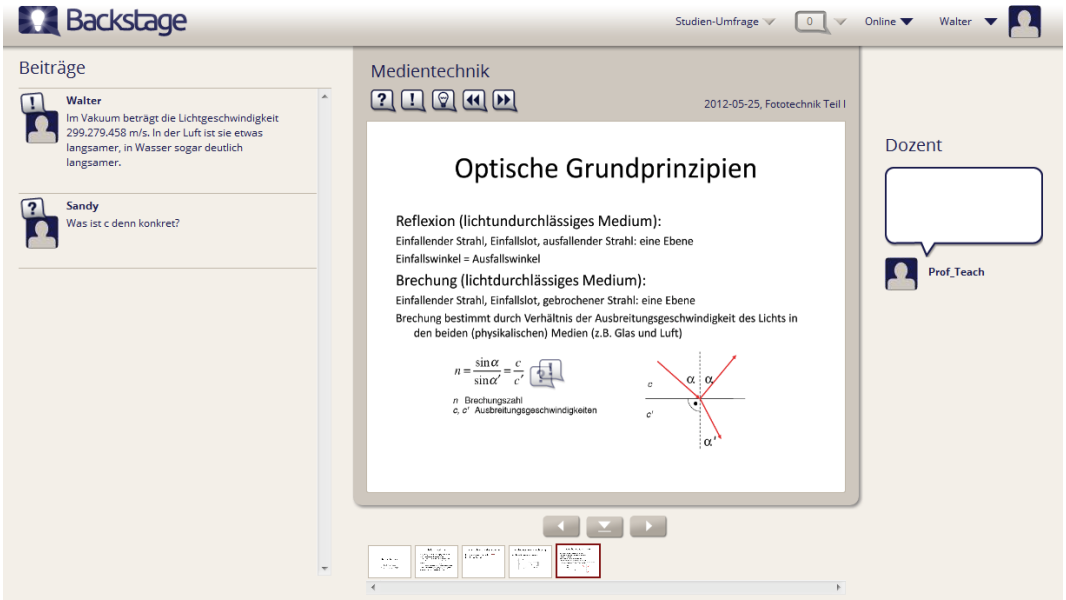


Figure 3.3: Students' Dashboard in Backstage

users to anticipate questions before discussing the content of the curriculum, in order to avoid unnecessary messages. Nevertheless, this kind of option will only be reasonable if the slides are not offered for download before the lecture has ended. The five icons above the slide, symbolizing the five message categories, are further explained by tool-tips and are used for writing messages.
If no certain message is activated, the left section of the screen shows all messages belonging to the activated slide ordered by its actuality with the latest message on top. Otherwise, if a message icon on the slide is clicked, all other icons on the slide are faded out and the left panel shows only the according message and its answers. It can be deactivated by clicking the same icon on the slide once again, so that all messages of the slide are shown.
The right side of the screen is used to show the lecturer's avatar and a speech bubble containing her statement, in case she declared one. This speech bubble can be used to share important information with the audience, for example for announcing omitted office hours. It can even be used to publish personal statements to communicate with the students and reduce the distance between lecturer and audience, for example indicating that the topic of this lecture session is very difficult.
The menu bar at the top is extended by further entries. A temporarily added menu called "Studien-Umfrage" provides a link to a survey for further studies on Backstage. The "Online"-menu offers the opportunity for students to switch between online- and concentration-mode. While the online-mode shows all panels of the dashboard as described so far, during the concentration-mode nothing but the slides including the according navigation buttons and thumbnails are shown. Naturally, the menu bar is also shown during concentration-mode in order to switch to online-mode again.

Figure 3.4: Input Textbox for Writing a Message

In order to write a message, the user has to click on one of the five category icons above the slide according to the kind of message she wants to publish. The cursor will immediately change to the chosen category symbol, so that the user can put the icon to the according position on the slide and place it there by clicking. Afterwards, as shown in Figure 3.4 the background is disabled and a box shown, where the user can type in her message restricted to 140 characters. The box is usually placed in the center of the window, which the screenshot does not show properly as it is clipped to show the main aspects of the interface. By clicking on the send button, the message is published and can be read by other participants in the progress of all messages according to this slide or by clicking the according icon on the slide. Although this procedure seems kind of cumbersome for the user, it follows the educational claim of providing guidance for learners. The student is first forced to think about the kind of message she wants to publish, reflecting about what she wants to say, by choosing the according category. Afterwards by placing it in the right position forces her to once again reflect about the context the message belongs to. The restriction to 140 characters avoid unnecessary information and force her to keep the message short and concentrated.
Answering a message works the other way round. First the message to be answered has to be chosen by clicking the corresponding icon on the slide. Afterwards again a category icon for the answer has to be selected, as it could not only be a real answer to a question, but also a comment (for example, a recommendation of a good book) or even a counter question. Immediately after clicking the category icon the input box is shown for typing a message. By clicking on the send button, the message is published in the progress of all messages belonging to this slide and in the progress of the primary message. Hence, no further icon is placed on the

Figure 3.5: Information about and Rating of Messages

slide for symbolizing an answer, instead the answer is hided within the primary question icon.

Due to the fact that, lecturers cannot read hundreds of messages produced by all participants, some kind of overview is needed to show the most important ones. Backstage uses a message rating and ranking to produce a top-k ranking of the best actual messages shown to the lecturer and produced by the rating of the users. Hence, each user has the possibility to rate each message once to express approval (plus), disapproval (minus) or that a message is outside the subject area (coffee cup). Therefore, the mouse cursor can be moved over the message to interact with and show further information about it, as can be seen in Figure 3.5. By this, the user can see the online-status of the author, date and timestamp of the message and how it has been rated so far. The backend functions for the favourite button (star) have not been implemented yet, but it should give users the possibility to save important messages for later use.

If the lecturer activates a quiz the question and answer possibilities are shown in the middle of the screen, instead of the activated slide, while all other panels are deactivated. Students are allowed but not forced to answer the question on their own. After a while the lecturer declares the question for closed, so that answering is not possible any longer and a statistic of the given answers will be calculated and shown instead of the question, simultaneously the other panels are activated again. The lecturer informs the users about the correct answer, as its displaying is not implemented yet.

3.3.2 Lecturers' Interface

The lecturers' interface differs from the students' interface in various aspects. The lecture and session overview is designed like the students' interface, but lecturers can add new lectures by specifying a title, lecture key and description. If students want to add a certain lecture to their overview they need to insert the according

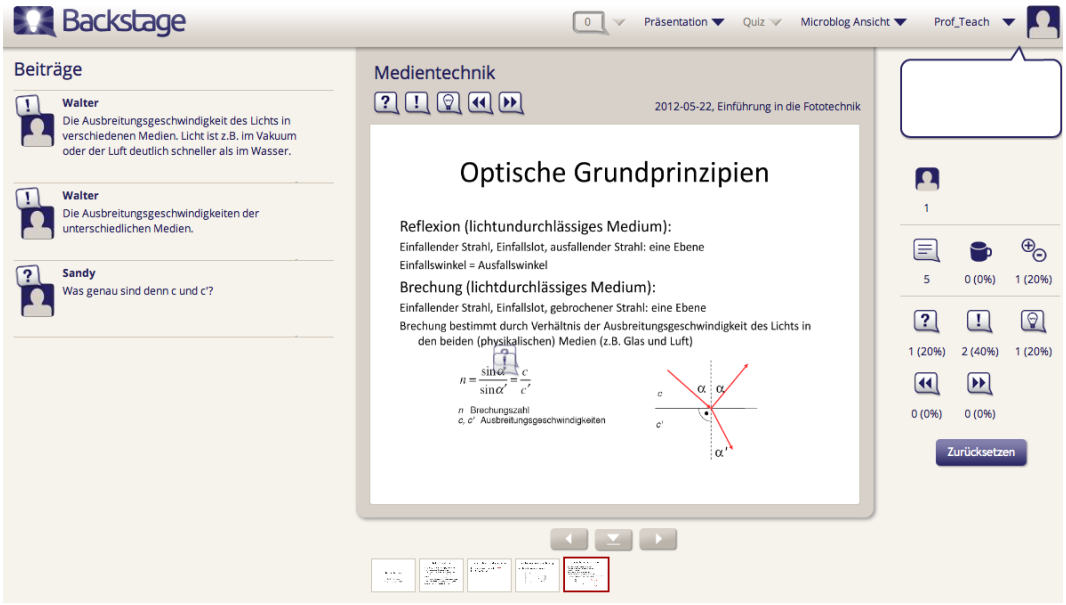


Figure 3.6: Lecturers' Dashboard in Backstage

lecture key. Furthermore, lecturers need to add sessions to their lectures, thus after clicking the according button they are asked to fill in the date and title of the session and additionally to upload the slides for the lecture. Just as the students, lecturers can open the backchannel by the corresponding button.
The dashboard for lecturers, which can be seen in Figure 3.6 is structured like the students' interface, where messages are shown in the left panel, slides in the middle and a menu bar on top of the screen. Additionally to the speech bubble, the right panel displays the statistics for the lecturer. It shows how many participants are online, how many messages have been written and their distribution among the categories. Furthermore it is shown how many messages are rated and marked as outside the subject area. The menu bar shows, like in the students interface, a profile menu and the deactivated speech bubble for personal status messages. The menu "Microblog Ansicht" gives the lecturer the possibility to switch between showing the messages in a standard way according to the slide or icon activated and instead displaying the message ranking on the left panel. Furthermore, the ranking sensitivity can be adjusted by a slider. The temporarily deactivated quiz menu will contain all options and commands for handling quizzes. The menu point "Präsentation" offers the lecturer four different options to adjust the properties for the microblog and message handling according to her preferences, for example, determining if students are allowed to communicate among each other. Furthermore, the slides can be shown in a second tab in fullscreen, which might be helpful during presentation, and the session can be started and stopped by this menu.
All other aspects, like writing, answering and rating messages are handled in the same way as in the students' interface.

3.4 Evaluating the Design and Usability

The advantage of the opportunity provided by this work to review and discuss the current usability and future changes in Backstage is taken and its user interface reviewed. Section 3.1 and 3.2 mentioned the important usability and educational guidelines, which are now used to evaluate the desktop application. Of course it has to be reminded that Backstage is still a prototype, hence its usability will not be perfect.

3.4.1 The Usability of Backstage

In order to evaluate in what way Backstage supports good usability, Dix's guidelines (cf. Section 3.1.3) will be used as a basis and extended by the guidelines of Shneiderman (cf. Section 3.1.1) and Norman (cf. Section 3.1.2) to structure the analysis. Hence, as Dix suggests, this chapter is divided by the three main categories for usability: learnability, flexibility and robustness.

Learnability

Predictability and synthesizability, i.e. how easy users can predict the effect of a operation by their past experiences and even see which effect operations had, are the first guidelines to support learnability. Backstage fulfils these guidelines most of the time, for example the buttons always have the same effect and do not change their function spontaneously. Furthermore, operations that are not possible are disabled and visually greyed out. In two minor cases it is hard to see any effect: First, if someone registers as a new user, fills in the form and clicks the send button, she will be redirected to the login page without any further feedback about the completion and success of the registration. A short popup informing the user about the success and that she will be redirected to the login page now can produce relief. The second case regards the lecturers' interface, where it is not possible to distinguish if the user is in the standard or message ranking mode if no messages have been posted yet. The difference between both screens can be seen in the menu, as the selected view is marked by a bold font or if the messages in the ranking and those of the activated slide are not the same in the same order. The left panel wears the title "Beiträge", which means "Messages", regardless of showing the slide messages or the ranking. Therefore, to overcome this problem it will be helpful to change the title for the ranking view, for example to "Ranking".
Another guideline to support learnability is familiarity, which will be extended here by Norman's postulations to use the knowledge in the world of a user and to secure that the mappings are right. Hence, to fulfil these guidelines the user's expectations and knowledge should be recognized, appropriate metaphors and simple language used. For example, buttons should have a 3D-effect so that the user will realize it is a button and immediately know how to use it by her real experiences in the world. These guidelines are very well fulfilled in Backstage. Not only the 3D-effect of buttons is considered, but also the use of avatars and ratings, used in other online communities and websites too, is familiar. As known from other

applications and websites the logo is shown in the upper left corner, the personal menu and logout function can be found in the upper right corner. Furthermore, the menu bar is horizontally on top of the screen. Navigating through the slides by clicking the left and right arrows is also familiar to computer users, as for example PDF applications make use of this concept. Limiting the text messages to a certain number of characters is a usual characteristic of a microblog, which can also be seen in Twitter[1]. Other users are also familiar to this restriction by the length of a SMS. The conceptual model of Backstage divides the timeline of communication by the slides, which can also be seen as a metaphor according to the real world. In a lecture not making use of Backstage, students would probably sit in the lecture hall browsing in their printed slides and following the lecturer's explanations. If a question occurs the student would immediately raise her hand to ask and the lecturer would give an answer. It would be an exceptional case, if a student would ask a question about a slide long ago. Hence, dividing the communication timeline by the slides is not as far from the real world experiences as it might seem at first sight.
Another guideline is generalizability, i.e. to what extent the knowledge of other applications and situations can be transferred to the actual one. This guideline is not very suitable for Backstage, as only a few very special functions exist, which can not be transferred to comparable situations. The only aspect that could be argued to concern and fulfil generalizability is writing messages of different category. For example, if one knows how to write a question, one can easily transfer the knowledge and write a comment, too.
A last and very important guideline in this category is consistency, which does not only concern the application internally, but also other applications and the real world context. All three dimensions of consistency are fulfilled by Backstage. The internal layout and design, furthermore the used terms are consistent throughout the whole application. As described in relation to familiarity, Backstage is also consistent with other applications and the real world.

Flexibility

In order to support the flexibility of a user interface, the user should determine when a dialogue should be initiated and system dialogues should only be used if necessary. It is hard to evaluate this guideline according to Backstage as the application does not have any occurring dialogue boxes. The only box used in Backstage is the text input box for writing a message, which is initiated by the user herself. Feedback about occurring errors, for example forgetting to fill in a input field in a form, is shown on top of and embedded in the reloaded page. Hence, it can be said that due to the fact Backstage makes no use of them, the user is not interrupted by dialogues initiated from the system and therefore, the guideline fulfilled.
To what extend multithreading, i.e. offering the user to simultaneously accomplish several tasks, is fulfilled or not depends on the point of view. Backstage itself does

[1] http://www.twitter.com

not really support multithreading, rather the underlying operating system and Internet browser do. Users can perceive different information at the same time, for example reading the slides and according messages, but it is not possible to accomplish two different tasks in Backstage simultaneously. For example, users cannot rate one and answer another message at the same time. In fact, Backstage was intentionally designed that way, so that users focus on the task they want to accomplish and do not get stressed, confused or distracted by other functions. However, one could argue that showing the slides in an extra tab in the lecturers' interface supports multithreading. This is true, but it has to be recognized that the lecturer will always make use of multithreading, as the slides of a lecture will always be projected on the wall. So if she does not use Backstage's function for it, she might bring another notebook with her that will be used for the presentation. Furthermore, if multithreading is interpreted in such a non-technical way, Backstage already makes use of it, as the backchannel application is used simultaneously to the frontchannel communication within the lecture hall. Usability experts claim that multithreading should be supported so that users can be more efficient, but this guideline does not make sense in Backstage.
Another guideline that cannot be applied to Backstage is task migratability, which is the possibility to determine if a user wants to accomplish some parts of a task on her own or if it should be done automatically by the system. This makes sense for routine tasks, which do not occur in Backstage. It is well-defined which functions are offered by Backstage and which tasks have to be done by the user.
The guideline of substitutivity, to offer different forms of input and output, is only partially fulfilled by Backstage. In some cases it is not possible to offer alternative input mechanisms, for example writing a message needs a text input. Otherwise, some cases could make use of alternative mechanisms, like the interface for choosing a lecture. Backstage offers links to the registered lectures, but it could also be thought of a text input field, where the user types in the name of the lecture or a drop-down box containing all registered lecture titles. Since the number of lectures a user attends is manageable and one click seems to be a lot easier than typing several letters or clicking twice in the drop-down box, Backstage dispenses with alternative input mechanisms for the simplicity and better overview of the interface. The lecturers' interface provides substitutivity at one certain point, namely the date input for a session. If the lecturer clicks in the input field she could type in the date, additionally a calendar pops up beneath it, where the date can be clicked instead. Although these input mechanisms, shown in Figure 3.7, could be improved on the one hand by displaying in which format the date has to be typed in (e.g. yyyy-mm-dd) and on the other hand by more clearly visually distinguishing between the calendar and the background (as both have the same colour and the calendar is not framed). At another point in the lecturers' interface substitutivity is not supported but definitely necessary. In the presentation menu, the lecturer can adjust the sensitivity of the rating and ranking by a slider, which has a default value of 33%. A slider as input mechanism for this function is not exact enough for the user but very fast to use. Furthermore, if ever changed the slider can never be moved to 33% again as this level is skipped while moving it along. Hence, to offer the user a fast but inexact setup the slider should be kept, but to facilitate a precise adjustment

Figure 3.7: Date Entry for a new Session

whenever needed an additional text input field will be needed. Substitutivity in regard to the output can hardly be fulfilled in Backstage as in most cases there is only one suitable way to present data. The only reasonable case to offer different ways of output regards the absolute and percentaged presentation of figures, which is already done in Backstage. For example, the rating of messages and the statistics for the lecturer are shown in absolute and percentaged figures. Furthermore, the distribution of the quiz answers will not only be shown by percentage, but also as a bar diagram.
The last guideline in this category postulates that users should be able to customize the interface. This guideline is extended by Shneiderman's postulation that shortcuts should be offered to experienced users, so that they can be more efficient. As Backstage is still a prototype, the interface can not be customized yet, though it is imaginable that users might be given the opportunity to choose in which panel (left, middle, right) they want to show which function (messages, slides, statistics/speech bubble). Some customization is already facilitated by Backstage, for example students can choose between concentration and standard mode or lecturers can adjust the options for the microblog according to their preferences. Shortcuts are supported by Backstage, for example pressing the enter key instead of clicking the send button or the escape key to cancel the process. Furthermore, as mentioned in Section 3.3.2 there are shortcuts for navigating through the slides. However, Backstage already uses the shortest ways of input whenever possible with the exception of writing a message. Though, as mentioned in Section 3.3.1 a shorter way for writing a message is not offered, due to the educational claims and aspects.

Robustness

Robustness seems to be the most important category of usability, as all three authors have a main focus on it. The first guideline of Dix in this category claims a

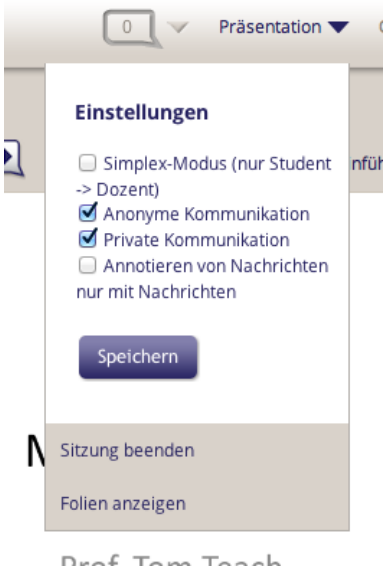


Figure 3.8: Lecturer's Options for Backstage

good observability. Hence, Shneiderman's postulation to give feedback and reduce the memory load seem to support this guideline. Norman's principle to simplify the structure of tasks and to make things visible also belong to this guideline. Thus, the first aspect to assure robustness is to keep the user informed about the system status, help her to orientate and give feedback. Ideally, the title and session of the lecture is always displayed above the activated slide and due to the thumbnail bar beneath it the user can easily orient herself. The more implicit feedback for routine tasks, for example after writing a message there is no feedback box, instead icon and message are just inserted in the communication, supports this guideline. Otherwise, there are several aspects where more feedback is needed, for example as mentioned above informing the user about the success of the registration. Furthermore, the system status is not shown and feedback needed in context of the rating buttons for a message. Users are only allowed to rate each message once, hence, feedback will be needed which messages have already been rated and which not. Backstage does not provide such feedback yet as it is still a prototype, but it is planned to include this function. As a result, the missing feedback burdens the user with memory load, as the already rated messages have to be memorized if the user does not want to waste her time, wondering why she cannot rate a certain message. This problem is known by the developers of Backstage and will be solved soon by disabling the respective controls if the user has rated a message. In the lecturers' interface it is possible to change the checkboxes for the options without pressing the save button, which can be seen in Figure 3.8. Hence, the system status looks as if the options would be changed, but in fact it is not saved that way. Therefore, it might be reasonable to omit the save button and save the options automatically whenever the checkbox is changed. In conclusion it can be said that observability and feedback are only moderately fulfilled in Backstage and could be further inserted.

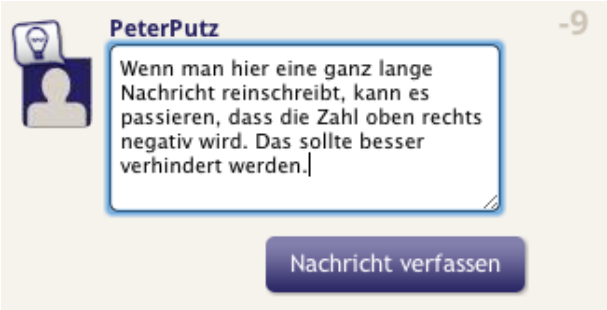


Figure 3.9: Number of Characters for a Message is not Constrained

Dix's guideline of recoverability coincides with Shneiderman's guidelines of error handling and reversal of actions, as well as with Norman's principles to design for error and to exploit the power of constraints. These guidelines postulate an easy recognition and diagnosis of errors for the user and moreover help to recover from them. In fact, errors should be avoided whenever possible by appropriate constraints and checks. This guideline is hardly fulfilled by Backstage, mainly because of missing constraints. In some cases constraints and further information could help the user to avoid errors, for example in relation to the date entry (cf. substitutivity in Section 3.4.1) or format of the slides for a session. Backstage apparently allows two sources for the slides: a Microsoft Office Powerpoint or OpenOffice Impress presentation and alternatively Latex. Trying to upload one of those office presentation formats (e.g. *.pptx) causes an error telling the user that only pdf-format is allowed. This error can be avoided by simply displaying the note that pdf-format is necessary. Another very important missing constraint concerns the limitation of messages to 140 characters. Although the remaining number of characters is displayed in the input text box in the upper right corner, users can enter more than those 140 characters. As shown in Figure 3.9 the number becomes negative and the message will be cut after the allowed characters when published. The entry in the input text box should really be constrained to 140 characters, so that the user is not able to enter more than those. By this it will be easier to recognize when the limit was reached and the message can be edited before sending. As already mentioned in relation to substitutivity in Section 3.4.1.2, the lecturer could possibly move the slider for the sensitivity of the ranking by accident and will never be able to adjust it to the default value. Thus, a default button recovering the standard properties for Backstage will be helpful for unexperienced users. The same problem, a missing undo-function, occurs in relation to the category icons above the activated slide. If a user accidentally clicks one of those icons, but does not want to write a message she has the opportunity to cancel the process by pressing the escape key. But if she does not know the shortcut, she has not immediately the possibility to get rid of the icon cursor by a perceived cancel button. Instead, she needs to click anywhere on the slide and cancel the procedure by clicking the close button (cross in the upper right corner) of the upcoming input text box. Furthermore, lecturers are not allowed yet to delete lectures or sessions, which is obvious on first sight, as students might want to log into Backstage long time after a lecture in order to look up old

questions and answers. On the other hand, typing errors or the upload of wrong slides during the creation of a lecture or session cannot be corrected by the lecturer afterwards. To overcome this problem, it is not necessary to offer a deletion but a edit function for lecture and session data.
The next guidelines supporting robustness is responsiveness, i.e. the system should react immediately and show feedback about the progress if procedures take a longer time. This guideline is fulfilled by Backstage, as there are no long-lasting procedures.
The last guideline, task conformance, i.e. how far does the system support the user in accomplishing the desired tasks. Therefore, the main tasks need to be identified and corresponding commands implemented. Shneiderman's guideline of closure is akin to this one and postulates that each task and wizard should have a beginning, middle and end. The task conformance of Backstage is definitely good, as every main task of the users has a corresponding command. Nevertheless, closure is sometimes neglected, for example the missing feedback in the end of registering a new user can confuse the user, so that the end of the task is not obvious. Possible solutions are mentioned above in the context of observability and feedback.

3.4.2 Educational Claims in Backstage

Backstage should not only fulfil the usability guidelines, but also respect the educational claims mentioned in Section 3.2. The most important guideline postulates that it should be focused on the educational goal and unnecessary functions be omitted. Backstage does definitely fulfil this postulation, as a prototype its functions are reduced to a minimum. For future implementations of Backstage it has to be deliberated which functions are really useful and which will not be advantageous for the learning situation.
Another claim is to use meaningful content and present it in appropriate ways. This educational guideline can hardly be applied to Backstage, as its learning contents are provided by the lecturers and cannot be influenced by the backchannel application. To assure that the content of the messages is still topic related, Backstage makes use of some kind of social self-regulation. The community, hence the users can mark messages as off-topic or express their disagreement by negative ratings, so that those messages will probably not be displayed in the top-k-ranking for the lecturer.
Backstage itself is the best factor for more activity in lectures to enhance the learning effect. It facilitates a better communication between the lecturer and her audience and moreover among students. Hence, the social aspect of learning is supported and further reflection and comprehension of the content guaranteed.
Besides, the quizzes and rating possibilities are additional activities motivating the students to review the subject matter. Even more, quizzes are not only another form of learning than the presentation by the lecturer and hence are conducive to the multiplicity of learning ways, but also support the claim for assessment issues. Explicit feedback about the success during the quiz is not only given by informing about the correct answer, furthermore, students can also compare their learn-

ing progress to their fellow students. Even the lecturer gets feedback about the progress of her students.
It is hard to evaluate to which extent guidance is necessary for a learning application, as even experts disagree about this topic. As yet Backstage does not make use of explicit guidance by for example instruction pop-ups, as such a behaviour is in stark contrast to keeping the interface simple and intuitive. Furthermore, too many instructions the user has to spend time on reading can hinder the efficiency in using the system. It is planned to integrate explicit guidance by external scripts, for example instead of filling their question into the text box users will be explicitly asked and instructed to choose a prepared question with certain placeholders. In contrast implicit guidance occurs in Backstage for example in the lecture overview screen. If the user has chosen a lecture in the left panel, the item is marked with a white background colour, which is also used for the panel in the middle, where the according session can be chosen. This visual flow from left to right effects an implicit guidance to choose a backchannel. This effect is further strengthened by the fact that Europeans are used to read from left to right. Implicit guidance is also given during writing a new message. Although there are no external instructions, the user is implicitly forced by the way the interface is designed to first choose a category, second a context and finally fill in the message. By this external requirement the user is forced to review which kind of message she wants to deliver in which context. Probably this behaviour will naturally not be given during a standard lecture without the use of Backstage.
The last educational guideline, claiming that learning applications should be easy to learn and as self-descriptive as possible, is fulfilled by Backstage. As mentioned above, explicit instructions are never used, rather implicit guidance helps the users to handle the system. Furthermore, good usability supports the intuitive use of the application.

3.4.3 Usability Guidelines vs. Educational Claims: A Conflict in Backstage?

As can be seen in Section 3.4.1 Backstage fulfils the general usability guidelines quite well, although in some cases the final perfection is missing. Several minor changes are suggested in order to improve Backstage's usability, but there are no severe violations of those guidelines. In most cases more feedback and constraints can help to overcome those little problems. Hence, the intuitive use of Backstage is not hindered in any way, it could rather be improved slightly. On the other hand, Backstage complies with the educational claims, too. As indicated in the section above, usability and educational guidelines seem to conflict at certain points. Depending on the specific situation it has to be discussed if a conflict is given and if so, which guidelines should have more weight.
One point, where both guidelines touch each other is multithreading. As mentioned above, multithreading was never considered within Backstage, as the students using it should focus on the lecture and its context without getting distracted. In fact, it is even some kind of multithreading that the students have to focus on the frontchannel (the lecture presentation and discussion itself) and the backchan-

nel (Backstage) at the same time. Nevertheless, this conflict can easily be solved for Backstage, because multithreading can hardly be applied to this simple application. On the other hand, it has to be mentioned that multithreading does not necessarily imply, that users do not focus on the educational goal. For example, the underlying operating system allows multithreading all the time. While using Backstage in the Internet browser, users could simultaneously visit other websites, write emails or play games. To control this kind of multithreading goes far beyond the competence of Backstage and is not possible. Nevertheless, those concurrent activities outside of Backstage are not necessarily disadvantegous for achieving the learning goal. It can be imagined that students use a writing application simultaneously to make personal notes during the lecture or use other websites to look up unknown terms and definitions. In general, the usability guideline for multithreading does not mandatorily conflict with the educational claim of focusing on a goal, actually it could even help. Depending on the concrete situation it has to be focused on wether the advantages of efficiency by supporting multithreading weigh more or less than the learning effect by focusing on one goal.
Another example where the intuitive handling suffers due to the educational claim concerns writing a message in Backstage. The desktop application uses a more complicated procedure for writing a message to force the users to review the content of their message before writing and sending. This procedure should not only help to gain more comprehension, but also to make it short and focus on the main aspect of their question, answer or comment. Nevertheless, this way of writing messages seems to violate two different usability guidelines. On the one hand, the postulation to offer shortcuts and customization shall help users to fasten regular tasks and be more efficient. The writing mechanism in Backstage cannot be changed in any way and needs at least three clicks and the typing of the message. Even more time is needed to figure out which category to choose and where to place the message. On the other hand, this writing procedure seems to conflict with the guideline of familiarity, as the participants are used to simply filling a text box and pressing the enter button to publish messages by the use of chatrooms, other microblogs and message boards. Moreover, the students are not used to rethink the context and category of their message, if they raise the hand during a lecture not using Backstage. Hence, it can be seen that a faster way for writing a message, might help the users to be more efficient in regard to the time they spend on writing, but as they do not review their own message, many more unnessary or imprecise messages will be published, which in turn will harm the efficiency of all participants while reading messages. Moreover, not only the number and quality of messages can be influenced by this complicated writing procedure, but also the orientation. Obviously, choosing a category and context helps other users to be more efficient while reading messages, as they can see on first sight where the message belongs to. If a student understood everything during a certain topic, she might not be interested in reading any messages corresponding to it and hence can ignore all message icons arround this topic on the slides. Furthermore, if she is only interested in reading the questions and according answers of her fellow students, she can keep an eye out for question-icons and ignore the other ones.

In conclusion, forcing the students to use the complicated writing mechanism, does violate the general conception of usability, though it not only pays off for the author as she reviews the content, but for all participants while reading and searching the messages. In general, adhering to guidelines supporting efficiency is a main aspect of usability, but it has to be reflected what is understood by "efficiency" in that specific case. In most cases, efficiency will be defined as accomplishing as many tasks as possible in as less time and with as less effort. Backstage's focus is not to offer students a fast way of learning, but to enhance the learning effect and communication. Hence, efficiency is not only defined by the rate of publishing messages, but also by their quality, which in turn can help others to be efficient while learning.
Both examples in this section show that conflicts between several guidelines are always given and need to be solved depending on the specific situation and application. Due to the fact that each application is different, applying such guidelines is often done subjectively and needs to be proofed by further user studies.

CHAPTER 4

Backstage Mobile

Although not advisable, many developers do not provide an extra design of an application or website for mobile devices, but adapt the desktop version for it [6, 21]. Figure 4.1 shows Backstage's main screen, the dashboard, displayed on mobile device. Although many mobile devices provide a zoom function to show the details of a website or application and ease the clicking of hyperlinks and buttons, using such an interface is laborious and has a negative impact on usability. Hence, a mobile version of Backstage has to be designed and developed in order to simplify using Backstage on small screen devices. Students often bring their notebook to the lecture and get detracted by activities which are not related to the topic of the lecture. Backstage tries to address those students, offering them an activity for the notebook, which keeps them focused on the topic. In contrast, many students also carry their smartphone with them and get detracted by it, but those students cannot be reached by Backstage, as it does not work properly on such devices. As mentioned in the introduction the development of Backstage Mobile necessitates the

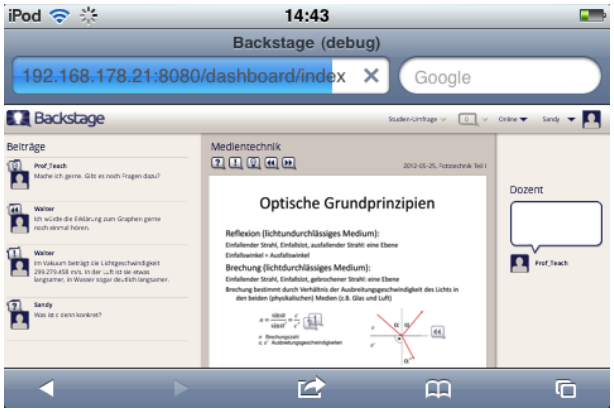


Figure 4.1: Displaying Backstage's dashboard on a mobile device

consideration of three main aspects: First, some educational claims were preferred over usability guidelines and it should be reassessed if this is still reasonable. Second, good usability is of great importance for the regular use of Backstage Mobile, but the desktop guidelines cannot be completely applied for mobile applications. Third, mobile devices pose new challenges, like the small display and different input mechanisms, that have to be kept in mind during development.
Therefore, this chapter comprises the whole development of a concept for Backstage Mobile, beginning with the challenges of mobile devices. Afterwards it is focused on usability guidelines for mobile devices found in literature. A first design of Backstage Mobile is discussed and a paper prototyping study conducted, leading to the final design.

4.1 Challenges of Mobile Devices

As mentioned above, the desktop applications cannot be adapted for mobile devices, due to the great hardware and software differences between computers and for example mobile phones [6]. In fact, the differences between those devices are great challenges for developers, as supporting both makes it necessary to develop two versions an application.
The first and most obvious difference is the screen size and its resolution, which poses the greatest problem [6, 8, 34, 40, 44, 45]. The lack of screen space requires the bigger part of the redesign, as not all information and interaction mechanisms can be shown at once. Although tablet computers are also mobile devices, their screen size and equipment support the use of usual websites designed for desktop computers. Nevertheless, if Backstage is redesigned for the use on small mobile devices, tablet users may choose between using the usual desktop website of Backstage and using Backstage Mobile, although the widgets might be oversized on tablets. It can be argued, that the development of a third version of Backstage, optimizing the use of space on tablets, might be necessary, but as users already have a choice between two working versions, a third design is not planned for now. Nevertheless, this topic is once again discussed in Section 6. Focusing on mobile phones, it can be concluded that the displays will not get much bigger in future, due to the fact that they have to fit in one's hand and be small enough to carry around in a trouser pocket [21].
Not only the small screens, but also the heterogeneity of the mobile devices in general is a great challenge. Different hardware and software conditions have to be considered, as for example, the not only small but also different screen sizes, low processing power and memory capacity. Furthermore, mobile devices provide different browser types and even operating systems, so that it is necessary to implement on each platform, if a native application for most of the mobile phones on the market is wanted [40, 44].
In addition, the special input and interaction mechanisms also differ depending on the mobile device and might be used with a stylus or a finger. While some mobile phones offer a hardware keyboard others only provide a software keyboard [8, 34, 40, 44]. Nevertheless, such mechanisms are often imprecise, so that the wid-

gets have to be optimized for their use, for example, icons and buttons need to have a bigger size if they shall be touched by a finger [34]. This challenge also implies that the navigation has to be completely revised and adapted to small screens. For example the usual menubar at the top of a desktop application cannot be used by mobile applications, because it often has to many entries to fit on a small screen and is to small to be appropriate for the touch of a finger. Another example is the context menu, which might be regularly used in desktop applications and cannot be easily used on mobile devices. Furthermore, users lose orientation to easily if the hierarchy withing a menu gets very deep [8, 21]. Hence, the developers have to provide more obvious information about where in the menu the user is or to omit some of menu points in favour of better orientation. It has to be considered if the provided menu entries can be displayed anywhere else than in a menu, for example by icons.
Due to its mobile nature, the last challenge of mobile devices is their changing context. It has to be considered that the mobile application might be used while moving, for example walking along the street. Hence, there is no more undivided attention on the mobile phone, so that the interaction becomes even more imprecise. Furthermore, a changing context implicates a different situation, where some features might not be available any longer or not appropriate. For example, the availability of WLAN or any Internet connection can be influenced. The brightness of the display needs to be adapted if the lighting conditions change. Using sound for feedback might not be appropriate in a library, hospital and many other places or even not recognized due to the loud environment [34, 40, 45, 13]. In preparation for creating a design for Backstage Mobile it has to be mentioned, that this is a special case of "mobile computing", as for now students using the mobile application will be present during the lecture and sit on a chair. Hence, some parts of the claimed context-awareness can be disregarded so far, for example moving while interacting with the mobile device, and may only be important for future work on Backstage and Backstage Mobile.

4.2 Usability Guidelines for Mobile Applications

Since the daily use of mobile devices becomes even more usual, usability should not only be considered by manufacturers of mobile devices, but also by application developers. Like in the context of desktop applications, sticking to usability guidelines can help to improve the overall user experience and finally increase the sales figures. In contrast to the desktop applications, usability guidelines for mobile applications are not very elaborated yet. Just a few authors even mention the aspect of usability in literature and only a couple of it actually focus on creating or adapting guidelines for mobile devices. Though, many authors refer to usability in a very specific context, for example usability for context-aware mobile applications, this chapter provides a more general and short overview of the most prominent usability guidelines for mobile applications, sectioned by their authors.

4.2.1 Guidelines by Jun Gong and Peter Tarasewich

Gong and Tarasewich try to adapt Shneiderman's eight golden rules for mobile devices [13]. While some of the guidelines can easily be adopted, others need to be adjusted and some guidelines even need to be added for the mobile context.

Providing *shortcuts* for frequent users does also matter for mobile devices [13]. For example, older mobile phones, not equipped with touchscreens, do not only provide to reach a certain menu point by navigating through the menu structure by the arrow keys, but also by entering a shortcut number leading there.
Furthermore, mobile devices should also offer *informative feedback* for each action and pay attention to the *closure* of dialogues [13]. To pick up the example of mobile phones, users should be informed when a message is sent and the task completed.
The last rule that can be adopted from Shneiderman is the *locus of control*, i.e. whenever possible the user should initiate actions and receive responses instead of being interrupted by the system [13]. For example, usually the first time a SIM card is used in combination with a new mobile phone, the user is asked if she wishes to import the names and numbers saved on the SIM card to the address book of the mobile phone. This guidelines would be disregarded, if the entries are copied without even asking the user.

The four other rules of Shneiderman have to be slightly adopted to match the requirements of mobile devices. The important rule of *consistency* does not only include all the aspects already mentioned in Section 3.1.1, but is further extended by the consistency across multiple platforms. The "look and feel", colours as well as the input and output mechanisms should as far as possible be the same as in the desktop version of an application [13]. This guideline can be further extended by claiming consistency not only between desktop and mobile version, but also between the same app provided for different mobile operating systems. For example, if the colour scheme and logo of the desktop version is blue, the mobile applications should adopt it and also use that logo and the blue colour.
In regard to *preventing and handling errors*, the physical design of a device has to be taken into account [13]. For example, typing errors are more probable on small mobile devices as the small keys are harder to hit exactly. Therefore, buttons on touch devices should maintain a certain distance to each other in order to avoid misclicks.
In context of this, the claim to facilitate *easy reversal of actions* is more difficult to fulfil than for the desktop applications, as it is more complicated to track the past states of a mobile device due to less memory resource and computing power [13].
The last guidelines, inspired by Shneiderman's eight golden rules is the claim to *reduce short-term memory load*. Therefore, instead of memorizing where to find which command, users should get the possibility to recognize functions while scrolling through a menu. Furthermore, it has to be considered, that in contrast to the desktop version mobile applications do not receive the undivided attention of its user. In order to deal with this distraction, alternative interaction modes, as for example sound feedback, can be helpful [13].

Although Shneiderman's eight golden rules seem also to work pretty good for mobile devices, some additional usability factors have to be considered. Gong and Tarasewich [13] introduce seven additional guidelines helping to improve usability. The first of it, *designing for multiple and dynamic contexts*, handles the challenge of changing contexts mentioned in Section 4.1. For example, the lighting conditions and noise level can change depending on the environment. To solve this problem mobile applications should be context-aware and provide self-adaption for different contexts or at least give the user the opportunity to configure the output (e.g. text size, brightness) [13]. As will be seen in the subsequent chapter, self-adaption needs special considerations to meet usability guidelines, if it should not be a further source of frustration for the user.
As mentioned above, users do not solely focus on the mobile device any longer, hence, it should be *designed for limited and split attention*. Thus, the attention required by the mobile device should be as little as possible and maybe even hands- or eyes-free interaction be provided, so that the mobile device can be used easily, for example, while walking. Therefore, tactile or sound output and speech input might be preferred over the usual interaction mechanisms [13].
As referred to in the context of challenges of mobile devices, their small size is another problem, hence it should be *designed for small devices*. Accordingly, the just mentioned tactile or sound output or speech input can also be used to overcome the space problem. For example, sound output can be used for feedback that cannot be shown on a small display and speech input to replace further buttons or complicated text input by small buttons [13].
Due to the small size of mobile devices and hence the small display the information that can be shown at once is limited. For this reason, it is advisable to *design for top-down navigation*, i.e. high-level information should be shown first, while details are only shown if the user requests it [13].
Furthermore, it should be *designed for speed and recovery*, which in this case is related to time constraints and not to errors. In the mobile context users often want to change applications quickly or immediately start or stop an application or function, due to the current situation. Hence, mobile applications should avoid long loading or saving periods and moreover save the status of a task when it is stopped so that it can be easily resumed next time [13].
Just like Dix's postulation for customization, Gong and Tarasewich claim that *personalization* should be allowed. This becomes even more important for mobile devices, as there is usually only one user for a mobile device, while computers might be used by several different persons. Therefore, it can certainly improve the usability, if the user is allowed to customize the application according to her preferences [13]. For example, users suffering from red-green colour blindness might prefer another colour scheme or contrast level than users with perfect sight.
The last guidelines added especially for mobile devices is simply called *"design for enjoyment"*, claiming that mobile applications should be visually pleasant, usable and make fun [13]. It could be argued, if this guideline really belongs to usability, as fun and aesthetics seem to be factors regarding to user experience.

4.2.2 Guidelines by Jonna Häkkilä und Jani Mäntyjärvi

Häkkilä and Mäntyjärvi formulate usability and design guidelines for the special case of context-aware mobile applications [15]. Although the compilation of guidelines in this section should be kept more general, their guidelines include several interesting aspects and should be mentioned here. Furthermore, even though Backstage Mobile is not designed as context-aware application for now, it could be of interest for future releases. They formulate ten different guidelines in regard to context-aware mobile applications and ask several developers, which guidelines are most important to them [15]. Accordingly, the three most important guidelines, related to context-aware applications, are mentioned first.

The first guideline is already very specifically related to context-aware applications, postulating to *consider the uncertainty in decision-making situations*. As referred to above, especially a self-adaption mechanism needs special considerations to be sure in which situations it should be adapted automatically and when to ask the user for confirmation [15].
The next important guideline is to *secure the user's privacy*, which has two different meanings. On the one hand, privacy is related to the data about the user the application shares. Such information has to be treated carefully and it should be considered to offer the user a possibility to stay anonymous. On the other hand, privacy can also be related to the personal space of a user. Hence, the behaviour of the mobile device might be inappropriate in certain situations and make the user feel uncomfortable [15]. For example, loud sound alerts in a church can embarrass the user.
The *visibility of the system status* is also of interest for mobile devices. The mobile application should not only show the usual feedback and system relevant information, but also certain data related to context-awareness. For example, which data is shared with the context at the moment, which context was identified and if there are any uncertainties with the self-adaption to the context [15].

Another guideline that can also be applied to general mobile applications is the *prevention from interruptions*. Users should not be spammed with unnecessary messages and the priority of those should be considered before the interruption. In regard to context-aware applications it is imaginable that different situations, hence contexts, allow different levels of messages and interruptions [15].
Furthermore, users should get the possibility for *personalization* of the application to meet their preferences, which could also change over time. This customization is not only related to colour schemes, text size and privacy settings, but also to adjust preferences related to certain contexts [15]. Users may want to determine by themselves which behaviour is wanted in which context. For example, students may want to prohibit sound alerts while staying at a library.
Just like Shneiderman's rule to reduce memory load, Häkkilä and Mäntyjärvi present a guideline to *avoid information overflow*. Akin to the claim to prevent the user from unnecessary interruptions, unnecessary information should also be avoided. Hence, it has not only to be considered which information is important and has

to be presented, while other information might be disregarded, but also how the information can be presented in a reasonable and understandable way. In regard to context-aware applications it has to be especially considered if the user really needs to be informed about each context change [15].
As mentioned above in the context of Gong's and Tarasewich's guidelines, mobile devices do often not get undivided attention and are sometimes used while the user is moving. Similarly, Häkkilä and Mäntyjärvi take this into account by their guideline *remember mobility*. Although this guideline does not only regard the consideration of interaction while moving, but also the fact that certain context information might get lost or not be available due to the movement [15].
Another guideline that is already known from Shneiderman is to *secure the user control*, which means the user should feel in control and always be able to have the control over the devices decisions. Hence, it might be reasonable to offer a setting possibility, where the user can decide if the device should automatically adapt to the context, if she wants to be asked or if she will adapt to certain contexts on her own [15].
A further guideline, particular to context-aware applications is the postulation to *access the context*, which is closely connected to the guideline of personalization. Users should get the possibility to especially adapt the preferences of different context, which does not only include the settings and levels of attributes and values for changing contexts, but furthermore the naming of contexts for better understandability. This guideline can be seen as an extension to user control and should consider the guideline of information overload while being applied [15].
Häkkilä's and Mäntyjärvi's last guideline called *usefulness* meets with little response among the developers in the study. It is a very general guideline, claiming that developers of context-aware applications should consider wether a certain function, information or adaption is really necessary and useful. Hence, it has to be balanced, wether the advantages of a function (e.g. helpfulness) or disadvantages (e.g. interruptions) outweigh. A lot of developers participating in the study regarded this guideline as redundant, because developers would automatically consider the usefulness of a function and never implement unnecessary ones [15].

4.2.3 Guidelines by Daniel Su Kuen Seong

Seong discusses usability guidelines in regard to mobile learning platforms. Therefore, he uses Nielsen's criteria (cf. Section 2.4) as a basis for the guidelines and extends his considerations by conceivable constraints and the unique properties of mobile devices. Constraints can not only be capabilities and constraints of the users (e.g. cognitive constraints), but also characteristics of the environment and limits of software, hardware and the product. In contrast to the numerous challenges of mobile devices mentioned in Section 4.1, he suggests only two unique properties for mobile devices: the high dynamic of the interaction and the contextual awareness. As a result of those considerations Seong presents his usability guidelines in three different categories: *user analysis, interaction and mobile learning interface design* [45].

The category of *user analysis* consists of only one guideline, which is called *the user/learner*. In this guideline Seong claims, that the user is of great importance and therefore the users characteristics should be examined. This does not only comprise the demographic variables as age, nationality and disabilities, but also cognitive and general skills, motivation and skills in using mobile devices [45]. Obviously, this guideline is not only of interest for mobile but also for desktop applications, as the user is in most cases the central point of an application.

The second category called *interaction* comprises four different guidelines. The first one is simply called *human-mobile interaction* and further divided in three subcategories. This guideline claims that an application has to be *compatible with the mobile device* and its characteristics. Therefore, the developer needs to take the hardware and software limitations (e.g. screen size, bandwidth, processing power) of such an device into account while designing. Furthermore, it should be taken *advantage of the context* by adapting the functionality. The third part of human-mobile interaction consists of the *personalization*, i.e. offer the user settings to meet her preferences [45].
The second guideline of this category sounds very specific to learning applications and is called *map between mobile learning portals and the real world*. Nevertheless, it only claims that the presentation of system data and information should be understandable for the user and that it should map the conceptual model of the user, which is best done by using familiar phrases and concepts [45]. This guideline seems to summarize Norman's postulation to use knowledge of the user and the world and to get the mappings right (cf. Section 3.1.2).
The next guideline is also known from the desktop usability guidelines and called *help users recognise, diagnose, and recover from errors*. As already mentioned in the context of desktop applications, error messages should be written in simple language and provide constructive help for the user [45].
Besides, the guideline *visibility of status* is also known from the desktop applications and claims that the users should get feedback and be informed within a certain time frame [45].
Even the last guideline of the interaction category was already mentioned in relation to desktop applications and is called *minimise human cognitive load*. Based on the assumption that the human memory is limited, this guideline claims that especially the learning contents should be divided in little "information chunks", which can be shown at once in one screen [45]. In contrast to the desktop guidelines this is very specific and does not refer to the general burden of cognitive load.

The last category comprises four more guidelines and starts with *the small screen display*. In the context of this guideline Seong mentions that the size of the display affects the speed of reading, thereby also the efficiency, and that too many pages have a negative effect on interaction. Therefore, long pages should be divided in smaller ones and an appropriate mechanism for navigating and jumping between the pages offered, so that extensive scrolling and clicking can be avoided [45].
In conjunction with the small screen problem, the next guideline *do not overuse* postulates that only necessary information should be displayed to avoid overload. Fur-

thermore, Seong claims the most important information should be placed at the right top corner [45].
The guideline *navigation* claims a consistent navigation between the single screens. A simple hierarchy of the menus can help to reduce the number of needed clicks or keystrokes. In order to further reduce the number of keystrokes, text input should be avoided and be replaced by widgets for list selections [45].
Seong's last guideline is the fundamental *consistency*, i.e. "similar information and action should be located in the same location" [45]. Like Gong and Tarasewich emphasize in their consistency guideline (cf. Section 4.2.1), Seong also points out that consistency should be kept within and between different platforms [45].

4.2.4 Guidelines by Erik G. Nilsson

Although Nilsson calls his work "design guidelines" and does not explicitly claim to present usability guidelines, but practical advices how to solve certain problems. Nevertheless, his work can help to improve usability and therefore shall be mentioned here. Nilsson lists twenty-three concrete problems that can occur while developing for mobile devices, which are divided in three main problem areas [34]. As not each detail of this outstanding work can be analysed here, just the most important aspects are mentioned.

The first problem area concerns the utilizing of screen space and provides some advices how information can be shown and grouped effectively on small screens. Advices in this area also suggest how to handle flexible user interfaces, which allow orientation and/or layout changes [34].
In order to make the best of the less screen space, Nilsson suggests to group information according to an appropriate principle. He emphasizes that a task-oriented user interface works best for mobile devices, as long as the tasks are clear and there is no discrepancy between the tasks the user wishes to accomplish and the tasks offered by the interface. Furthermore, he provides different mechanisms to group information, for example tabs, collapsible blocks, dialogues, tree views or task wizards, which all have different advantages and disadvantages. Generally, a combination of all those mechanisms is possible and Nilsson suggests to use separate windows for different main concepts and tab folders for separate aspects of such a concept. To pack all needed information in one screen, whitespace should be reduced, a more compact font used, space-wasting widgets should be replaced by alternative smaller controls and text by visualizations whenever possible. Furthermore, Nilsson emphasizes that scrolling should be reduced, especially horizontal scrolling should be avoided, for example by changing the layout or making use of the landscape mode. To reduce vertical scrolling, lists should be optimized in their order and content [34].
Flexible user interfaces are needed in two certain situations: On the one hand, whenever the device is tilted the orientation changes from portrait to landscape mode or vice versa, so that height and width of the screen change. On the other hand, if a software keyboard is shown for text input, at least half of the screen is

covered by it, which makes it necessary to think about what should be shown in the remaining space. As a solution one could either implement one user interface for each situation to exactly specify where which widget shall be shown or determine certain constraints (e.g. the image width should be half of the screen width) and the rest be arranged by the device automatically [34]. In fact, the flexible user interface while showing the software keyboard is provided and set up by the operating system.

The second problem area regards the handling of input. Nilsson gives advice how input mechanisms can be optimized, so that they are as much efficient and accurate as possible [34].
Nilsson postulates to reduce text input when possible and replace it by other input mechanisms. If this is not possible, predefined values and autocompletion of the text input can help to be more efficient. As already mentioned in context of lists, order entry should be optimized, so that the user can be fast and efficient. It might even be possible to include a learning algorithm so that usual entries are saved and can easily be reused. Developers should also consider to use multimodal interaction mechanisms, for example a barcode reader, RF-ID technology or voice control.
In context of this problem area, Nilsson emphasizes the difference between guiding and forcing a user. Guidance helps the user to achieve what she wants by a natural order entry and automatically jumping into the next field, while forcing is unnatural and needs the disabling of all fields but one. Therefore, Nilsson postulates to avoid forcing and replace it with a wizard, with the exception of one case: Forcing the user to enter one keyfield before all the others is allowed [34].
In regard to handling input, it has not only to be thought of the mechanisms provided by the mobile device, but also how it is physically interacted by its user. If there is no stylus available for a mobile device, two alternatives have to be considered. First, using the finger seems intuitive, but it needs to be considered that certain widgets are not easy to use with it. Hence, some controls, as for example context menus, should be excluded and others preferred, like buttons or check boxes. Second, hardware buttons of a mobile device can also be used, but disadvantageously their use is not as intuitive and it has to be first learned which hardware button corresponds to which function. In context of this problem it also has to be thought about how to search for example a database if no keyboard exists for the text entry. Text entry can be made easier and faster by saving the history of searches or offering functions to narrow down the results. For example, buttons could be offered to filter by certain characters of the alphabet, e.g. “L-O”, as known from encyclopaedia [34].
The last main problem area is called “design at large” and comprises problems and advices that do not belong to any of the other problem areas or address general design problems [34].
Among other things, this chapter advices to visually code which form fields need to be filled (e.g. by an asterisk or bold font) and which are not allowed to be filled (e.g by disabling or sound effects). Furthermore, he postulates that standard solutions should be used where possible, while tailored solutions should only be used if they produce an usability gain. If there is any longer synchronization or loading

period, the user should be informed about what is going on in a simple language and the progress including the remaining time. A last guideline, which can not be integrated in one of the other problem areas concerns the log on and off for an application. It has to be balanced, whether the usability or security should outweigh. Shorter passwords, e.g. PINs, are easier to enter and do not need feedback if the right key was pressed, but a PIN of four numbers is not very secure. Otherwise, more complicated passwords, e.g. minimum eight upper and lower case characters, need explicit feedback to avoid typing errors, but such passwords are more secure [34].

4.2.5 Analysis of the Mobile Usability Guidelines

Before the above mentioned mobile usability guidelines are used in relation to the development of Backstage Mobile, their importance for it and even in general should be shortly discussed. Gong's and Tarasewich's [13] paper seems to be of great importance, as not only the typical desktop usability guidelines are adapted, but also further extended by the specific needs of mobile devices. In fact, their additional guidelines comprise many of the challenges mentioned in Section 4.1. They address the problems of screen size, special input mechanisms, changing context and the different navigation structure. It can be criticized that the authors claim to use a self-adaption mechanism for changing contexts without any further consideration of the situation.
In contrast Häkkilä and Mäntyjärvi [15] discuss this aspect very detailed in their paper, as it is focused on context-aware mobile applications. They postulate that the user should feel in control and that it has to be discussed where an automation of the adapting is useful and where it might harm the user control. Their guidelines give a great overview of the chances and risks of context-aware applications and could be very helpful if Backstage Mobile will ever make use of its context.
Seong [45] claims in his paper that his guidelines are based on Nielsen's definition of usability and extended by further constraints and the characteristics of the mobile device. However, his guidelines seem to be more based on Nielsen's famous ten heuristics [32, 30] used to evaluate the usability of applications rather than his definition of usability. Nevertheless, Seong repeats most of the guidelines and aspects which were already mentioned by Gong and Tarasewich [13]. Hence, his guidelines also pick up most of the challenges of mobile devices. Although he tries to group the guidelines in a reasonable way, it does not become clear why this kind of grouping was chosen and for example, "navigation" belongs to the interface design category and not to interaction.
Nilsson's [34] guidelines are a special case, as he never claimed to publish usability guidelines. Some of his guidelines are very specific and already partially implemented by the underlying operating system, so that developers are either way bound to it. For example, mechanisms for a flexible user interface while showing a software keyboard or wether a stylus, hardware keys or fingers need to be used for interaction. Nevertheless, his paper shows where problems and challenges can occur and gives advice to choose the best solution for concrete situations.

For the development of Backstage Mobile it is reasonable to stick to the guidelines by Gong and Tarasewich, as well as Seong and keep the ideas and suggestions of Nilsson in mind for problem solutions. Häkkilä's and Mäntijärvi's guidelines give a good overview of context-awareness, but those guidelines which can be adopted for "non-context-aware" applications are already picked up by the other authors.
Reconsidering the guidelines it has to be discussed, if they conflict with the educational claims made for Backstage and adopted for Backstage Mobile. The in Section 3.4.3 discussed missing multithreading is still not wanted for Backstage Mobile for more concentration on one task and will only be possible to the extent the underlying system allows it (e.g. changing the application). As multithreading is a special postulation of Dix [10], which is not mentioned in the context of mobile usability guidelines and additionally its advantages and disadvantages for Backstage have been discussed before, it can be disregarded for Backstage Mobile.
The second issue discussed in the same chapter regards the gain of forcing the students to write a new message in a certain way. Nilsson argues that forcing instead of guiding users is unnatural, should be avoided and can only be used in special cases. Nevertheless, the forced mechanism should be adopted for Backstage Mobile due to its advantages discussed in Section 3.4.3. Though, it has to be reconsidered in which way this mechanism can be adopted and how it can be designed to maximize usability despite this unnatural enforcement. The other educational claims do not collide with any of the mobile usability guidelines.

4.3 Design and Paperprototyping

Having in mind the different challenges of mobile devices, appropriate solutions and advices by usability guidelines, the design of Backstage Mobile can be developed. Therefore, a first design of the mobile application is created and analysed for improvements. In a second phase, the improved design is tested by real users in a paper prototyping study, where the results and qualitative feedback help to make Backstage Mobile more usable. In the end of this chapter the final design and concept of Backstage Mobile is presented.

4.3.1 First Design and Storyboard

As the user is an important factor for the success of a new product [9], it is advisable to make use of user-centred design techniques. In fact, Norman categorized his seven principles mentioned in Section 3.1.2 as a main part of user-centred design [36]. Creating personas and according scenarios is a typical method to analyse the requirements of the users.

Personas

Personas are user-models in form of a detailed description of an invented character [9, 35]. Each persona is a model of a certain user or customer group with specific characteristics, skills or behaviour [9, 21]. The invented characters help the developer team to know and understand the users, keep the focus on their customers

throughout the whole project and remember their needs whenever a design decision has to be made [9, 21, 35].
The concept of personas helps to overcome three often occurring problems while designing an application: First, during the design process each developer has her own imagination what the typical user will be and requires. As the characteristics of the imagined user are not determined and very imprecise, the imagination will not help during the design decisions. In one case, the developers might say, that the user is a computer expert and can easily handle a tree-view, while in the next case no computer skills are accredited to the same user to explain the use of a wizard. This kind of user is called "elastic user", as each developer interprets the user the way she wants because the characteristics are not determined. The elastic user can be avoided by the use of personas, as they seem to be real persons with certain characteristics and skills that will not change. Second, personas avoid the "self-referential design" by the developers. If personas are not used, developers tend to unconsciously suppose that their own skills, goals and motivations are the same as those of the users, hence creating a design that they can handle easily because of their insider knowledge, but the user does not get along with. In this case, personas remind the product team throughout the development and design that not their wishes and goals are of interest, but those of the users and who the users are. Third, edge cases can occur and need to be handled, but they should never be the main focus of the design. Therefore, personas can help to avoid such situations and keep the focus on main functions by asking oneself if persona x will even use a certain function and if so how often [9].
Usually personas are created on the basis of real qualitative data gained from user interviews or observations to be effective, which is not possible here due to the short time period of this thesis. Nevertheless, it is recommended to use at least provisional personas, also called ad hoc personas, than using even none, if there is no or not enough real data about the customers available [9, 35]. Provisional personas consist of the available data and for the most part of hypotheses and presumptions about the users [9]. Usually one persona is designed for each user group, which are extracted from the collected data. As there is no data available, it has to be supposed what characteristics and behaviour might occur. In case of Backstage Mobile it would be reasonable to invent at least one persona that represents the lecturer and one for the student. Nevertheless, in this thesis only a student interface will be designed and focused on, so that the persona(s) representing the lecturer are omitted here.

To create provisional personas the available real data has to be considered and complemented by hypotheses about the remaining characteristics. Therefore, it has to be discussed which sorts of people use the product and what there needs, behaviours and expertise might be [9]. The user data can first be divided by their existing roles, which are in case of Backstage the lecturer and the student. As mentioned above the lecturers are omitted for the creation of the personas, hence, regarding the demographic data the personas are all students of different subjects and therefore probably between about twenty and thirty years old. The students are of both gender and it can be supposed that they have different incomes, as some

might still live at home and do not need much money while others have to pay the rent.
Additionally, the students' behaviour has to be analysed, hence regarding Backstage it can be thought about which learning style (e.g. auditory, visual) is preferred and how much time is spent on learning. Actually, it has to be considered how much time is even available for learning. It can be thought of users, who have a lot of time for studying, while others have to work on the side and therefore are restricted in time. Thus, users might not only show different behaviour in spending free-time for learning, but also visit the lecture regularly or irregularly. Some students might learn by listening attentively to the lecturer, while others tend to be visual learners, needing to read information about a topic and make annotations for their own. Some extreme learners even suffer from panic attacks during exams and therefore fear to fail if they do not spend every free minute on learning.
For creating the personas, it has also to be considered which technical and domain expertise the users have [9]. As Backstage rebuilds the situation of a lecture and all students are familiar to such a situation the domain expertise of all users can be supposed to be high. Nevertheless, the technical expertise could highly differ. As nowadays basic computer skills are already required for participating in any studies, due to the often used online administration and the fact that theses have to be written on computer, it can be assumed that the user will also have at least basic computer skills. Nevertheless, as Backstage Mobile is created for mobile devices, the technical expertise regarding smartphones and tablet computers has to be considered. This expertise can range from almost no experience (for example, someone who just bought a new phone) to expert (for example, someone who uses her smartphone all the time).
Usually these hypotheses about the personas would be proofed by user interviews, focus groups or surveys. Afterwards behavioural patterns would be identified to build certain personas upon it, for example, interviews might reveal that persons who buy CDs also like to download MP3s [9]. As there is no real data to rely on it has to be guessed which behavioural patterns are reasonable. Therefore, a student who has to work on the side cannot spend as much time on learning as someone who does not have to work. A visual learner will probably make more use of Backstage Mobile than an auditory, who might get distracted by it from attentively listening to the lecturer. Someone fearing to fail in the exams will probably use every minute for learning and be thankful for every source of information on a topic. Such people might use Backstage Mobile to satisfy their inquisitiveness. Auditory learners and persons who fear to fail will probably visit the lectures regularly. These and further considerations lead to three different personas which are presented in the following.

Figure 4.2: Tom Huber [11]

Tom Huber

Tom is twenty-three years old and studies Informatics in Munich for three years now. He moved there from Cologne for his university studies and to experience something new. Tom shares a small flat with two other students and works for an online shopping company in the division of content maintenance on the side to afford the rent. As he works full-time on some days, he does not visit the lectures regularly, but catches up the missed subject matter by learning on his own.
In his free time he plays handball and spends a lot of time with his numerous friends. He loves to use social networks to stay connected with them and even uses those online applications regularly on his smartphone. He thinks that this is the best time of his life and does not want to spend all his free-time on learning and working.

Figure 4.3: Jennifer Meyer [11]

Jennifer Meyer

Jennifer, also called Jenny by her mother, is twenty years old and has just started to study in Munich to become a physics and english teacher. She still lives there with her parents, who support her, so that she can be a full-time student and does not need to work on the side. Consequently, Jenny wants to finish her studies as fast as possible to sponge off her parents only for a short time. Therefore, she visits the lectures regularly and spends a lot of time on learning. Jenny prefers to sit in the first rows of the lecture hall, as she can see and hear everything the lecturer says, but the lecturer will probably overlook her and be concentrated on the uninterested students in the last rows.
In her rare free time she loves to go horseback riding or to watch a film with her best friend Susan. Although she is not interested in modern technologies and has even little computer skills, her grandparents gave her a new smartphone as a birthday present a month ago. Jenny fears that the private school she wants to teach at in a few years

will not employ her without perfect final marks.

Figure 4.4: Peter Kruse [11]

Peter Kruse

Peter is twenty-four years old, studies law in Munich and lives in a granny flat at his parents' house. As he works regularly in the evening as barkeeper to finance his hobbies, he tries to use the daytime effectively for learning. Therefore, Peter visits the lectures regularly and listens the lecturer attentively. Although he has medium computer skills, he is sceptical about the supposed advantages of using computers during lectures and prefers to take handwritten notes.
In his free time at the weekends Peter loves to motorcycle and go on day trips with his friends. Since his old mobile phone broke last year, he bought a new smartphone. Most of the time he uses it for calls and text messages, but sometimes he even looks up terms in the Internet. Nevertheless, Peter thinks that people reveal to much about their own personality in the Internet and that this should rather be a place of knowledge than of gossip.

It is recommended to choose the so-called primary persona out of all the created ones, by comparing the goals of the personas. The primary persona is the one that will not be satisfied by any of the interfaces designed for the goals of the other personas, while the other personas can also use the interface of the primary persona without at least being dissatisfied [9]. In this case, Jennifer is the primary persona, as she will probably use most of the functions of Backstage and due to her less smartphone and computer skills needs much support. It has to be mentioned that there could be further personas combining the different learning styles with another kind of expertise regarding mobile devices. For example, a fourth persona could have no experience with smartphones and also be not interested in using Backstage, as she is a auditory learner and shares a negative attitude towards it. Nevertheless, in contrast to all other personas Jennifer has the highest expectations and needs the most support while using Backstage, so that she can be chosen as primary persona. Tom will probably also make use of the functions, but as he is familiar to smartphones and computers he will also get along with an unintuitive interface. Due to his high experience it might be a good idea to offer Tom some special features like shortcuts. Peter is very sceptical and will probably make even no use of Backstage because of his negative attitude towards modern technologies in lectures. Nevertheless it could happen that he might use Backstage, but probably he will participate very passively. Whatever interface will be designed for Jennifer,

will likely be also satisfying for Tom and Peter.

Scenarios and Requirements

Usually designers create several scenarios for the different personas to identify the requirements for the design framework [9]. Such a persona-based scenario can be a context also called "day in the life" scenario, which is a written narrative about a day or situation in the life of the persona, the typical tasks she wants to perform and how the product helps her [9, 21]. Due to the limited timeframe of this thesis it is not possible to write down ten to thirty usual scenarios [37]. Instead, to show how the method works, one context scenario for the primary persona is presented in the following.

At monday afternoon Jennifer walks into the lecture hall and takes a seat in the front row. In a few minutes the lecture about optics will start and Jennifer has a bunch of of questions in her head. Next week the exams are going to take place and she is already very nervous and fears to fail. While the other 300 students gradually enter the lecture hall and take a seat, Jennifer prepares her records and recognizes that her notebook battery needs to be charged but she forgot the charging cable.
Fortunately, she remembers that the backchannel application Backstage can also be used by mobile devices and that her best friend helped her installing Backstage Mobile on her new smartphone some days ago. She starts the application and logs in with her username and password. After choosing to open the optics lecture of the actual date in Backstage Mobile, she waits for the lecture to begin.
As Prof. Bernstein starts the lecture, Jennifer can see the questions and answers of her fellow students. Just as she wants to ask a question on her own, she recognizes that a fellow student called "Paul89" published a similar one regarding the actual topic. Jennifer gives Paul's message a positive vote to push it forward. After a while many other students have also voted for Paul's message and Prof. Bernstein sees the urgent question. He answers it shortly by talking about the topic and explaining it again.
Later in the lecture, Jennifer has a question about the velocity of propagation in different material. She uses the according function to specify and write a message. Prof. Bernstein announces to start a quiz now and Jennifer sees the question with three different options to answer. She decides to choose the second one and waits for the quiz to finish. As Prof. Bernstein stops the quiz and informs the students about the correct answer, Jennifer is happy to recognize that she was right. Afterwards she takes a look, if someone answered her question. Actually, she has two very helpful answers by her fellow students.
In the evening, as Jennifer is at home, she learns for the coming exams. While learning for optics she remembers that someone asked an excellent question she also wanted to pose. Instead of turning on her computer Jennifer uses her smartphone and logs in to Backstage Mobile once again. After choosing the according lecture and date she browses the questions of the afternoon to find what she was looking for. The additional information helps her to learn for the exams.

This and further scenarios can show what tasks and activities are needed for an

application by analysing the text [9]. Accordingly, Jennifer's main requirements for Backstage Mobile are summarized in the following list.

- Log in to Backstage Mobile with existing account
- Choose the according lecture
- Read messages of the fellow students
- Vote for messages of others
- Ask questions
- Read answers
- Answer quiz questions
- Browse questions after the lecture

Further scenarios for Jennifer and the other secondary personas may bring up further activities that need to be considered. In fact, Backstage Mobile is derived from Backstage and differs only in the user interface. Both applications are even used in the same context for the same tasks. Therefore, the tasks that Backstage Mobile needs to offer can be adopted from Backstage.
Transferring a design from the desktop to a mobile application effects the same "look and feel" on both platforms. By this users who are familiar with one of both versions will recognize and get along with the same function on the other platform, so that it is almost unnecessary to "re-learn" all functions [21]. Thus, the main functions of Backstage, already mentioned in detail in Section 3.3, comprise the following activities.

- Registration as a new user
- Log in to Backstage
- Add a new lecture
- Choose an existing lecture
- Choose an existing session to a lecture
- Read and browse the slides, know which is the actual slide
- Read the messages according to the slides
- Evaluate messages by voting
- Send messages
- Answer messages
- Participate in quiz questions
- Change personal avatar
- Log off Backstage

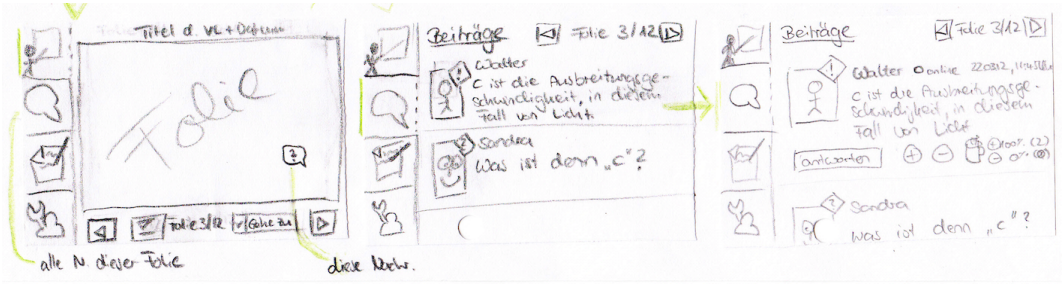


Figure 4.5: Part of the Storyboard of the First Design for Backstage Mobile

The First Design of Backstage Mobile

According to this basis, a first design is developed, which corresponds to a great extent to the desktop version. Hence, the mechanism for writing a message should be adopted, which means that the slides have to be presented in order to place the message icons. As the slides are usually in landscape format, it suggests itself to use landscape and prohibit the portrait mode for the layout. As Nilsson suggested [34], a task-oriented design works best for mobile devices, hence tabs are used to divide the interface by its different tasks and save space. To show the slides as large as possible the tab bar is best arranged vertically on the left side.
If users open Backstage Mobile on their mobile device everything is portrayed in landscape mode. The first screen shows two input fields (username and password) and a button to login, furthermore a link to the registration form. If the users have successfully logged in the tab bar on the left side appears with four different icons. The first icon, a figure presenting something on a wall, is activated after the login and the corresponding screen on the right side shows drop-down menus and a button to choose a lecture and according session for opening. If the user has opened the backchannel of a certain lecture and session this icon represents the slides. Hence, on the right side the users will see the current slide eventually with message icons on it and some buttons for navigating. Figure 4.5 shows three views of the storyboard (whole storyboard cf. Appendix A), where the first one shows the slide screen.
The second icon in the tab bar, a speech bubble, represents all existing messages. If a user tabs this icon, all messages are shown according to the slide chosen in the slides-tab, which can be seen in the middle in Figure 4.5. The messages are shown with the user's avatar and a category icon, furthermore username and the message itself are displayed. If a user taps a message it will be expanded, as can be seen in the last view of Figure 4.5. The expanded space shows status information about the message (date, time, rating, online-status of the author) and buttons for voting and answering.
The third button of the tab bar shows a piece of paper with a pencil and is a symbol for writing messages. If a user wants to write a message she taps this icon and is shown the five category icons, which she has to choose from. Tapping a category icon leads her immediately to a screen showing the current slide, where she is asked to place her icon on it. Afterwards a field for the text input occurs, where

she can finally write the message and push the send button.
The last icon of the tab bar shows head and shoulders of a figure and a screw-wrench to symbolize the profile and general setup. There the user can change the lecture, log off Backstage Mobile or change her avatar picture.

However, adapting the desktop version to the mobile device as described above bears several issues. It is not only unusual to create an app which is not game but rather textual in landscape mode, but also placing or tapping a small icon on the slide is very difficult for touchphones. Furthermore, it has to be thought about wether all functions are really needed in a mobile app, for example changing the avatar picture will probably be of no interest, as many people do not save appropriate pictures in the right format on their mobile device. Moreover, the button for changing the lecture backchannel provided in the setup menu can also be omitted, as users will usually have to change the lecture hall for visiting the next lecture and therefore will log off anyway.

Recently it has been considered to support the lecturer by so-called filter boxes [39]. The lecturer can draw boxes on the slides to be informed about message icons which are placed in this area. For example, a lecturer could draw a box around a formula to be especially informed about questions and messages in this area, as the formula is of great importance for the future topics and definitely needs to be understood by the students. This concept opens up many new ideas and a new solution for writing messages in Backstage Mobile. The lecturer has to place boxes on the slide according to the underlying topics and give them a precise name, for example “Quadratic Formula”. For the desktop version, users can still place their icons on the slide and if the category icon is placed within a box, the message will belong to that topic. For the mobile version, users are asked about the category and context, i.e. topic box, of their message, before they are allowed to put in the message text and send it. An appropriate algorithm, for example on the server, can place the messages and according icons of mobile users at any free place within the box so that desktop users are able to interact with them. In contrast, mobile users should get the ability to filter messages by the according topic box.
Keeping this concept in mind and thinking about the context in which Backstage and its mobile version shall be used, it becomes obvious that the slides are dispensable for guiding the user in Backstage. On the one hand, the new box concept can be used to introduce a new writing mechanism which is independent of showing the slides, especially on a small screen. On the other hand, not only will some students bring printed copies of the slides with them to annotate them during the lecture, but furthermore the slides are also projected on the wall during the lecture. Hence, it is no longer necessary to present the slides on the mobile device, where they probably would be anyway much too small for reading. Omitting the slides renders the landscape mode unnecessary, so that it can be switched to portrait mode for the application. Moreover, the tab bar, which seems to be very useful and should be kept, can now be shown on the bottom of the screen, as the users might already be familiar with it from other applications. The first design mentioned above did not include any possibility to participate in the quiz. As the slides are excluded, the

free tab can now be used to introduce a “quiz tab” in the bottom bar.

In conclusion, the new concept is used in portrait mode and shows a login screen on startup, with an additional link to a registration form for new users. After the login, a tab bar on the bottom of the screen is shown, which includes four different icons. The first one, the speech bubble, is activated and the screen shows drop-down menus to choose a lecture and session for opening. Furthermore, a button is provided to allow users to add new lectures by entering a lecture key. The next two icons in the tab bar, the piece of paper with the pencil for writing messages and a icon showing “a) b) c) ?” symbolizing the quiz are disabled as long as no session is opened. In contrast, the last icon showing a figure and gear wheel to symbolize the profile and general setup, is enabled as the user is already logged in and could change her settings. Furthermore, a button is provided in this screen to log off Backstage Mobile.
If the user has chosen a session, the speech bubble tab is still activated and now shows the slide and topic, where the lecturer is on top of the screen. The screen for the lecture choice is not shown in an extra tab, as it is a precondition to choose a lecture and session first, before the according communication can be shown. The rest of the screen shows all messages belonging to that topic. A tap on a message will expand it and offer further information about it and buttons for voting and answering. Consistent to the planned feature in the desktop version, this screen shows the actual messages and will automatically change to the new topic if the lecturer changes it. Hence, if the user wants to read older messages she has to swipe from left to right to show a second message screen where she can filter older messages by drop-down menus according to their category, slide or topic. A swipe to the other direction will lead her back to the actual messages. The same concept is used in the quiz tab, which shows first the actual quiz question, its answer possibilities with radio buttons and a send button. If the user wants to look up older quiz questions and solutions she has to use the swipe again, on the next screen all older quiz questions are listed, where a tap on one will bring up the according details. To visualise the swiping possibility the live pages provide a left arrow and “older quizzes” or “older messages” at the top of the screen. Vice versa, the old pages show a right arrow and “actual quiz” or “live messages”. After answering a quiz question a popup occurs, informing the user that her answer has been sent and that she has to wait for the quiz to end. When the lecturer has stopped the quiz, the correct answer and a bar diagram about all given answers by the students is shown.
As mentioned above, to write a message the user has to tap the paper and pencil icon, which lead to a screen, where one enabled and two disabled drop-down menus are shown, furthermore a disabled text input field and send button. The user has to choose first a category in the enabled category drop-down menu, afterwards the slides drop-down menu is enabled to choose the according slide number for the message. Next, the topic box drop-down menu is enabled offering the titles of the topic boxes on that slide. After choosing the according topic the text input field and send button are enabled, so that the user can type her message and send it. As feedback, the screen changes back to the actual messages, as the user will

probably write only one message in a row.
Although the design already resolves many of the above mentioned issues, we are sceptical if it will work the way it is intended and if the users understand the box concept. To optimize the usability and discuss such design questions a user study has to take place.

4.3.2 Rapid Development by Paperprototyping

The main focus of this thesis is to create a mobile interface, which does not only fulfil the educational claims, but also serves good usability. While creating the second design, many new design questions come up, which can not be answered by the developers without feedback of the users. Hence, a so-called paper prototyping study is conducted to get fast and qualitative feedback of the users. Prior to the presentation of the designed study, the concept and advantages of paper prototyping are presented.

Paper Prototyping and Preparations

"Paper prototyping is a variation of usability testing where representative users perform realistic tasks by interacting with a paper version of the interface that is manipulated by a person 'playing computer', who doesn't explain how the interface is intended to work." [49, p. 4].

Advantageously paper prototyping is a cheap possibility to get qualitative user feedback in an early state of a project. Therefore, the user interface is drawn on different pieces of paper by hand without using colours. This is of great importance, as users often do not dare to criticize a neat prototype because they already see all the effort in there and do not want to harm any of the designers. Furthermore, a neat prototype using colours and real pictures provokes the people to give high-level feedback, while the rare paper prototype encourages them to give low-level feedback as the graphical design has obviously not been specified yet. To get feedback in an early state of the project can save time and money, because as long as nothing is implemented yet, changes can easily be incorporated, decisions tested and in the worst case the developer has to dump some pieces of paper that costed her some hours but not thousands of lines of code that costed her weeks [49].
Usually the developers invent some tasks according to the issues they are uncertain about and need feedback on. Afterwards most of the parts of the prototype needed to accomplish those tasks are created and prepared (whole paper prototype cf. Appendix C). According to a developed user profile, four to six users have to be recruited for participating in the study. Although this number seems to be very small, it represents the best trade-off. As Figure 4.6 shows, less than four users produce too less feedback, hence some usability mistakes might not be found, while more than six will probably not mention any new findings, which the other six have not already found. Usually such a study is conducted by at least three persons, one plays the computer, another observes the actions of the user and takes notes and

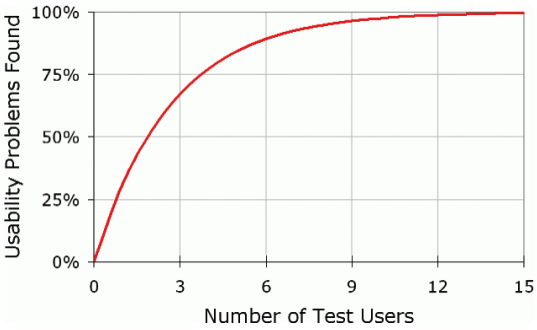


Figure 4.6: Trade-Off between Number of Tested Users and Found Usability Problems [31]

a third person represents the facilitator introducing the user and explaining everything [49].
Although it is not recommended, but due to the conditions of this thesis, the paper prototyping study was facilitated by only one person and videotaped for later analysis. It has to be reminded that paper prototyping does not result in quantitive data in form of statistics, but in qualitative feedback by the users. It helps to find usability issues, as for example misleading concepts, and recognize that a function might be missed by the users. Furthermore, it can show which design alternative the users prefer and which functions are indispensable, while others might not be needed [49].

To create the tasks for the paper prototyping all feared risks and design questions in regard to the user interface have to be collected and prioritized, according to their impact on the success of the product [49]. Reviewing the second design of Backstage Mobile five questions occur, with the highest priority first:

1. Does the user recognize and understand the connection between the topic boxes shown on the slide and the drop-down menus symbolizing the topic boxes in the mobile application?
2. Does the user understand that the live messages are automatically bound to the navigation of the lecturer and recognize how she can navigate through the messages on her own?
3. Does the user recognize if a message has already be answered and how to read those answers?
4. Is the user able to intuitively use the rating?
5. Does the user understand the icons in the tab bar?

According to those design questions or risks, concrete tasks have to be constructed and filled in a template which summarizes the goal, the needed input, a time estimation for export, the single steps to solve it and a concrete instruction for the user (cf. Appendix B) [49]. For this prototyping study four tasks were created to

answer the questions above and one more simple task added in the beginning to introduce the user to the concept of paper prototyping. According to Backstage's and Backstage Mobile's user profile, the users recruited for the paper prototyping study were all between twenty-four and twenty-eight years old and students of different subjects. All of them owned a smartphone they use regularly, three of them have even a touch phone and three of them a tablet computer. The paper prototyping study took place in a single session and at home of the users, to make them feel comfortable. Before declaring to participate in the study each user was informed about the procedure, especially that the recorded data is anonymous and even the videotape will only show their hands and the paper prototype. As recommended the test persons were also informed that this study does not test their behaviour, opinion or success but just the user interface [49], so that they are not afraid to criticize aspects or even feel that they "failed".

Conduction and Results of the Paper Prototyping Study

The paper prototyping study started by a short introduction of Backstage and its main functions and goals, as the participants are not familiar with this application. Afterwards a short warm up task should help the participants to get familiar with the concept of paper prototyping. Therefore, the users were asked to log in to Backstage Mobile with a given username and password and open the lecture "Medientechnik" on a certain session date. All persons claimed that this task was very easy to solve. Only one of the five participants was insecure what the term "backchannel" means on the button "open backchannel". When the users chose the lecture a drop-down menu with several fixed date-entries was presented. One user suggested that she would prefer a calender to pick the date, while another one criticized that it would be reasonable to immediately show the date of the next session if a lecture was chosen as it is probable the user wants to log in for the new and actual session and not to an old one. In combination with this task, the users were asked afterwards where they suppose to find a function to log off Backstage Mobile. All five users, supposed it as expected in the setup tab.

In the second task, the users were asked to pose a certain question on Backstage Mobile in regard to a specified topic. All participants first tapped the speech bubble icon in the tab bar, where nothing happened as this is the actually activated one. Most of them tried the paper and pencil icon next while two of them tried it as third possibility. All participants thought that the meaning behind the icons can be easily learned the first time and will be clear from that point on. This problem could also be caused by the paper prototype, because it does not properly show which tab is currently active. Hence, this problem will probably disappear in the final implementation.
Four of the participants did not criticize the way the user is forced to write a message while one would prefer to automatically enter the number of the current slide and topic, as it can be supposed that someone who wants to write a message will probably want to write one according to the recent topic. If required, the entries

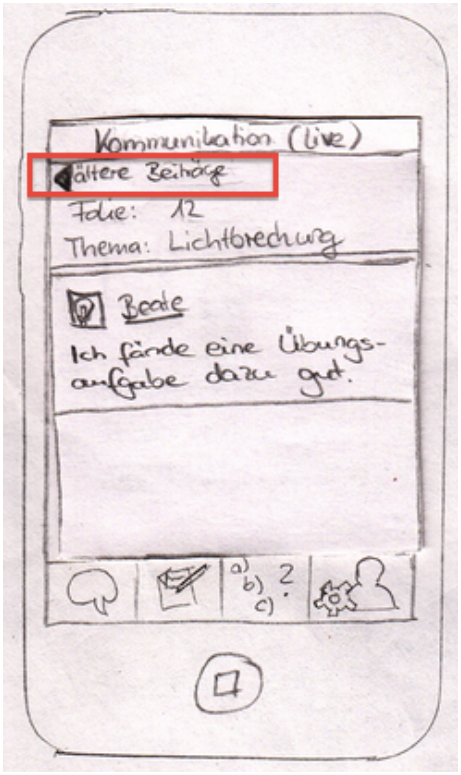


Figure 4.7: Live communication screen. The arrow, marked by the red rectangle, gives a hint for swiping.

can be changed manually by the user. Evolving from this statement it came up, that the desktop version actually also automatically chooses the actual slide (as the icon has to be placed on it) and that for reasons of consistency this idea should be considered to be integrated.
Although the users are not familiar with Backstage only one of them was uncertain about the meaning of "category" until she tapped the menu and saw the possibilities. One critical point is entering the slide number, because one person would prefer to enter it herself using the keyboard, while another prefers scrolling through the numbers as long as there are not more than a hundred.

In the next task, the lecture went on and the users were asked to rate a certain message positively, although the actual slide has already changed. Therefore, the users had to change to the old messages by swiping, enter some information into the filter, tap on the according message and on the plus button. All test persons tried tapping on the arrow saying "old messages", as can be seen in Figure 4.7, because they thought that this was a button. Once again this problem could be caused by the characteristics of a paper prototype. Nevertheless, the users were asked, if this was not a button, what they think how they could get to those messages. Two of the participants suggested swiping, a third person named the same after suggesting other menu buttons. Interestingly those three participants are also the three

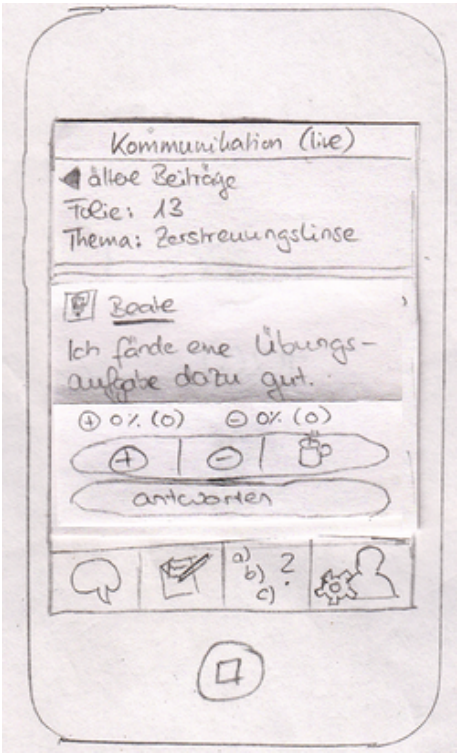


Figure 4.8: Rating and Answer Buttons in the Paper Prototype

owning a touchphone, which could indicate that users used to this gesture will more likely try to use it. Though, this hypotheses cannot be verified here, as five users are not enough data for a statistical analysis.
Independently of the swiping effect, three of the participants claim that it is hard to recognize the hint with the arrow and that it should be more clearly separated from the other information. One of them suggested that if this was a button it would be disadvantageous to place it on the top, because he regularly uses his mobile phone with one hand and would not be able to reach it that way.
Maybe due to the way the task was posed, all participants used the slide number to navigate to the wanted message. Only one participant claims that she would prefer to see all messages at once without entering any data and filter the messages herself if required. Although it is comprehensible that entering no data might be easier than being forced to enter at least one, it has to be considered that receiving all messages of one session at once from the server will probably occupy the mobile device for a long time. Hence, the search should be narrowed before downloading the messages.
All participants knew intuitively where to find the rating, three of them would also press the answer button, which can be seen in Figure 4.8 after tapping the plus. Once again this could be caused by the concept of paper prototyping as "playing the computer" to disable the buttons (grey them out) does not work as fast as the users press another button. Nevertheless, it might be reasonable to reorder the but-

Figure 4.9: Alternative Speech Bubble Icon for Backstage Mobile

tons, to avoid misconceptions. If the answer button is illustrated by an icon instead of the text it would be small enough to put all four buttons in one row, symbolizing some kind of "interaction bar" for each message. Most of the students recognized that their rating has been accepted because the buttons greyed out and the rating statistic for the message changed.

In the fourth task, the users were asked to find out if someone has already answered their question and if so read the messages. Afterwards they should check which slide the lecturer talks about at the moment. Before the participants started to work on this task, a speech bubble icon with a 2 on it was shown to them instead of the original speech bubble as shown in Figure 4.9. As they were asked what they think this icon means, two persons thought it was the number of messages belonging to the actual slide, one thought it was the number of personal messages and one concretised more precisely that it the number of answers one got on her own questions. The remaining person said that it depends on what is taught to her, but due to the low number she would suppose that it is the number of personal messages. In conclusion, symbolizing answers on own questions this way could be misinterpreted so that another solution has to be found. It can be imagined to show a status icon in the title bar, in Android's notification tray or iOS' notification centre.
All participants easily found the required message by navigating through the older messages. Three participants were not sure, whether they should tap the envelope or the message itself and decide to tap the message first. As they recognized that this expands the message for rating and answering options they tapped the envelope. Two users suggested to be provided with an overview of one's own messages anywhere, to simply look for answers, but there should be no annoying popup if an answer comes in. A single person was disappointed that only the beginning question and not all according answers are immediately shown. Therefore, she wished the rating of a question was obvious from the beginning to decide which question and answers to read.
After reading the answers all users found their way back to the screen with the live messages and recognized the right slide number and topic the lecturer talked about. One participant claimed that it was confusing, why the speech bubble icon in the tab bar does not lead the user back to the screen with the live messages as some kind of "home-button". In fact this is a very interesting aspect which could be easily included into the design without destroying or disturbing any other function.

In the last task users were asked to give a certain answer on the actual quiz question. This task was very easy for all users, as maybe due to the process of elimi-

nation all five user knew which was the quiz icon. Although all participants recognized which answer was declared right in the result and how many percent of their fellow students gave the right answer, they agreed that it would be good to show anywhere which answer was given by them. One person suggested to display "right" or "wrong" by corresponding colours, like green and red, or symbols, as check mark and cross, instead of using the words themselves. This idea was not claimed by the other participants, but as they were asked all would understand these symbols the right way. Another person complained about the popup giving feedback that the answer has been sent. In his opinion, this would be implicitly clear by greying out the buttons or a displayed message while waiting for the quiz to end. The additional popup is superfluous and annoying.
One participant commented that the results by percentage of the fellow students are "nice to have", but that he would not be interested in it and rather only know whether his answer was correct. Although this idea was only mentioned by one participant it is of great interest. The slides are not shown in Backstage Mobile due to their large size and because they are also projected on the wall. In fact, the bar diagram about the quiz results will get very small on the mobile interface and can also be omitted because the results will also be projected on the wall.

4.3.3 Resulting Design

The paper prototyping study gives an interesting insight to possible problems and misconceptions in the second design. Hence, Backstage Mobile is redesigned to overcome the detected problems and include the mentioned ideas of the users. Furthermore, it has to be considered if Backstage Mobile fulfils all requested usability guidelines and the educational claims.

Analysing the Risks and Design Questions for Backstage Mobile

In order to invent the appropriate tasks for the paper prototyping study, design questions and risks were listed in Section 4.3.2. These risks and questions can now be evaluated and reasonable changes suggested.

The greatest risk, which will lead to a total failure of Backstage and Backstage Mobile, is that users might not understand the connection between topic boxes on the slide and the topic drop-down menu for writing and filtering messages. In the paper prototyping study the users were given two versions of the slides. On the one hand, they got a printed version of the slides without boxes which represented the slides where they can write down their annotations. The other slides, including the boxes, were printed much larger and symbolized the slides which are projected on the wall. Whenever the lecturer would switch to the next slide, the new one was placed on top of this slide stack by the facilitator of the study. Unexpectedly, the users immediately understood how to use those boxes.

The subsequent design question regarded the problem that users might not understand that the live messages are bound to the navigation of the lecturer and that

older messages can be reached in another screen. Surprisingly, the users were never confused about which topic the lecturer is talking about, but the study revealed that the swiping mechanism seems to be not that obvious on first sight. Nevertheless, the study also showed that the users were confused by the nature of the paper prototype and thought that the writing was a button. Hence, the swiping will probably be more explored if there is not any hint, which is not unusual in mobile applications. If the users once know that the swiping mechanism exists they prefer to use it.
As mentioned above, to show no old messages without entering any data is not only reasonable for mobile devices, but also consistent to the desktop version and therefore should be retained in the mobile version. Furthermore, it may be advantageous to make it a little bit "harder" to reach older messages, to avoid asynchrony and facilitate to keep the focus on the current topic at hand.

The third design question was about recognizing and reading answers. In fact, users recognized that the occurring envelope symbolizes answers and easily found out how to read them, although some of them first tapped the message instead of the envelope. Nevertheless, they were able to recognise and correct the error, therefore this mechanism can be retained.
Nevertheless, users claimed that their messages should be shown in an extra overview to easily access answers to their own questions. As mentioned before it can be thought of some kind of status symbol or message in the title bar or system status bars, as Android's notification tray or iOS' notification centre [39]. In fact, the latter idea will be consistent with the desktop version, where the - for now – disabled speech bubble in the menu bar will show status messages in future.
The study revealed that it is reasonable and consistent to the desktop version to show all messages at once and one message accordingly with all its answers isolated in one screen, if an envelope is tapped.

If the rating mechanism was not intuitive for the users, so that they would not make much use of it Backstage's and Backstage Mobile's filtering system for urgent messages might suffer. Surprisingly, the rating mechanism is very intuitive and the users got along with it very well. Probably due to the nature of the paper prototype users often also pressed the answer button additionally. To avoid this misconception the answer button could be renamed as "answer message". Instead, an answer can also be seen as an interaction with and reaction to a message like the rating. Therefore, it can also be consistently shown as a button with a symbol instead of a text, for example an arrow as known from answering emails can be used. Hence, it is possible to show all "interaction"-buttons for a message within one bar. Nevertheless, to avoid that users still might tap first a rating button and afterwards the answer button, because they might think they need to press some kind of "send" after rating, the answering button should be shown first in the row.

The last design question concerned the interpretation of the icons in the tab bar. In the beginning the users were confused about the icons and mistook the speech bubble icon for writing messages. This is once again probably caused by the nature of

Figure 4.10: Tab bar at the bottom of the screen. Grey icons are disabled, white icons are enabled and activated tabs are highlighted by a blue background colour.

the paper prototype as there is no highlighting for the activated tab. Colours and highlights for the activated tab, as shown in Figure 4.10 will solve this problem. Nevertheless, if each icon was at least used one time, the users know exactly where to find which function.
Additionally, the study revealed the idea to use the icons more consequently. For example, if a user has navigated to the "old messages" screen, no tab should be highlighted in the tab bar and a tap on the speech bubble icon should lead back to the "live messages" screen. By this kind of navigation users will always find their way back, if they are not aware of the intentional use of the swiping mechanism. The same should be implemented for the old and actual quiz screen.

The study also revealed some necessary changes and further ideas that were not considered in the design questions. It is reasonable to automatically preselect the current or next date in the session drop-down menu, when a lecture was chosen. In most cases users want to login to an ongoing or next lecture.
Furthermore, preselecting the slide number if a user wants to write a message is also consistent to the desktop version and makes sense. It is likely, that an appearing question is related to the actual slide and not to topics some slides ago. If the user wants to respond to a message, the slide number and topic should be preselected, but not the category of the message. So that this function is consistent to the desktop version.
Participating in a quiz revealed that the statistics can be omitted, consistently to handling the slides and instead the own answer can be saved and shown. In fact, this is not consistent to the desktop version, where own quiz answers are not shown in order to secure the user's privacy and avoid that other users might cheat. In contrast, mobile devices are small enough to be covered by one hand, if the user wants to conceal her answer from her fellow students. Additionally, the cheating is impeded as it is more difficult to read an answer on a mobile device which is not directly in front of the user.

Fulfilment of the guidelines in Backstage Mobile

Prior to the implementation it has to be considered, whether the mobile usability guidelines and educational claims are fulfilled where possible. Nilsson's design guidelines [34] are ignored in this analysis, as they are more design advices for concrete problems than "dos and don'ts". A lot of his suggestions, as using a tab bar, creating a task-oriented user interface and to avoid text input help to design Backstage Mobile and are adhered to in the interface.

One of the most important guidelines, consistency [13, 45], is fulfilled by Backstage Mobile in several ways. The design is not only consistent to the desktop version, but also different principles in regard to functions. For example, that any data needs to be entered before showing older messages and that the relationship between the messages is only shown if required (by tapping the envelope). Suggesting a status area at the top of the application where notifications about own messages are shown is also consistent to the desktop version. Furthermore, the communication is still bound to the slides.
Backstage Mobile is also consistent with other mobile applications, as the tab bar is presented at the bottom of the screen and a swiping mechanism is used without further hints.

Regarding navigation, guidelines as offering shortcuts, using a simple top-down navigation showing details on request and avoid to many clicks [13, 45] are very important. The navigation of Backstage Mobile is kept very simple, by using only four tabs for different tasks. As suggested, only the most important aspects are shown first and details presented if required. For example, at first only the message with the name of the author and a category symbol is shown. On request (by tapping) the user can not only read further information about the message (date, time, rating) but also use interaction features (answering, rating). The swiping mechanism is consistently used for both messages and quiz questions.
Furthermore, highlighting activated tabs and adequate title bars help the users keeping the orientation and know where they are.

In order to facilitate closure dialogs should have a beginning, middle and end and furthermore interruptions should be avoided [13, 15]. Backstage Mobile does not make use of disturbing pop-ups and tries to give implicit feedback in order two signalize the end of a task. For example, after sending a message the screen automatically changes back to the live messages. Especially, for informing the user about incoming answers to her questions no pop-up will be used, but a status bar that unobtrusively notifies the user on responds to her messages.
In one situation the user is possibly interrupted by the system, namely during an actual quiz question. In consistency to the desktop version it is planned to disable any other function than the actual quiz and logout function, if a quiz question is activated by the lecturer. By this the interface forces the user to at least take notice of the quiz question and probably more students will participate in it, if they are not able to do other activities in Backstage Mobile. This could probably interrupt a student just writing or reading a message. Though, the majority of the users is supposed to stay focused on the lecture and hence be ready for a quiz question.

It is not only necessary to prevent errors, but also to support the user recognising and handling them, for example by undo-functions [13, 45]. Backstage Mobile tries to avoid text input, which advantageously also eliminates a great source of errors. Whenever possible, the user is advised about the error and how to solve it. For example, if the user forgot to fill a field during the registration, she gets not only

informed about the fact, but also which field has to be filled. The view for writing a message also includes a cancel button next to the send button, so that the user can easily change her mind. Whenever possible, the user is guided implicitly by the interface, for example it is impossible to choose a session before a lecture has been selected, accordingly the button for opening the backchannel is disabled until all information is available.

Another guideline is to secure the user control [13, 15], which is fulfilled by Backstage Mobile in the majority of cases. The only aspect where the user is forced by the system, is the view with the live messages. On the one hand, this is partially consistent with the desktop version if the regarding option would be activated, as students using Backstage cannot have a look at slides, the lecturer has not yet talked about. On the other hand, this kind of force shall help to keep the focus on the actual topic and stay concentrated. Although this mechanism might be strange and getting to old messages might be laborious it serves to keep the focus. As the user can easily change the screen to navigate through the messages on her own, she is never completely at the mercy of the system.
Nevertheless, forcing the user to have a look at the activated quiz question withdraws the control from the user. Although a user is not forced to participate in the quiz, she cannot do anything else until the quiz has stopped. By this, the quiz also serves as a point of synchronization, where users engaged in discussing older topics are forced to have look at the current one.

In order to reduce the memory load and avoid information overflow [13, 15, 45], the user interface is kept very simple. Furthermore, a filter is provided to get a better overview of older message. Additionally, the relationship between messages can be recognized, if the envelope icon is tapped. Further information overloading the screen is only shown if required, for example, the rating information.

Another important guideline postulated to give understandable feedback and show the status of the system [13, 15, 45]. This guideline is fulfilled by Backstage Mobile as the user is always informed about which topic and slide the lecturer talks about. Furthermore implicit feedback is given for every action, for example the rating buttons are disabled after voting for or against a message.

Considering the multiple and dynamic context for mobile devices [13, 15, 45] does not matter yet for Backstage Mobile, as the context does not change. Nevertheless, in regard to this guideline it has to be thought about which behaviour of the mobile device might be appropriate for this context. Therefore, Backstage Mobile does not make use of any vibration or sound feedback, as this will be annoying and distractive during a lecture.

The physical characteristics of mobile devices as the hardware and small screen [13, 45] have to be considered for the development of Backstage Mobile. Therefore, Backstage Mobile avoids text input and complicated navigation gestures by more appropriate widgets. Whenever possible knowledge about the context and

user is used to make the handling easier. For example, preselecting the actual slide for writing a message or the next session date for opening the lecture backchannel. Furthermore, the already mentioned task-oriented tab bar is very appropriate for mobile devices and easy to handle on touch screens. Moreover, horizontal scrolling is avoided and alternative controls used to avoid further buttons (swiping). The collapsing boxes used for showing and hiding information and interaction buttons according to messages help to keep a better overview on the small screen and save space.

A guideline not fulfilled by Backstage Mobile, is personalization, i.e. that users can change settings and adapt the interface to their preferences [13, 45, 15]. In fact, there are no features yet the user could possibly change, hence the interface does not serve any changes yet. Nevertheless, the setup/logout tab could offer such settings in future.

In order to design for enjoyment [13], the interface especially the widgets should be visually appealing and animated transitions used for switching between screens. A sliding transition representing the swiping effect can help the user to recognize this navigation.
Furthermore, one author claimed that only useful functions should be implemented [15], which is fulfilled by Backstage Mobile. The design comprises no function which does not serve any certain purpose.

In context of this, it can also be seen that the user interface focuses on the educational goal of Backstage Mobile. The content presented is needed and meaningful without any unnecessary data. The process of writing a message remains fairly structured: the user is guided through this process by the layout of the user interface (selection of slide, box and category at the top and thus before writing the message when handling the view from top to bottom). The slides are not shown within the user interface, but as they can be seen on the wall or in printed documents they can easily be omitted. Like the desktop version Backstage Mobile facilitates activity in form of providing appropriate ways of communication, rating and quiz questions. Forcing the users have a look at the quiz question will probably lead to even more participation and activity. The quiz also serves as assessment and self-monitoring function for the students, but also as feedback for the lecturer. Whenever possible the user is guided by the interface, which is due to its great usability easy to learn. The writing mechanism used in Backstage was appropriately adopted for the mobile user interface, so that a great quality of messages can be achieved.

CHAPTER 5

Implementation

Prior to the implementation of Backstage Mobile it has to be considered which devices are targeted. 85% of the sold smartphones worldwide in the second quarter of 2012 are shared by devices using Android or iOS. Accordingly, Samsung and Apple are the leading smartphone sellers in the world [50, 51]. Hence, it can be concluded that most of the students are probably using an Android or iOS device. Nevertheless, focusing on one of these will inevitably lead to disregarding the other ones. Implementing for both platforms would be an enormous effort and furthermore ban all other users, who have no Android or iOS device, from participating via Backstage Mobile. Therefore, it seems suitable to rely on a mobile development framework which supports as many operating systems as possible by writing just one code.
The open-source development framework Adobe PhoneGap , also known as Apache Cordova, is used to create Backstage Mobile [16, 17]. By using PhoneGap, mobile applications can be written once in HTML 5, CSS 3 and JavaScript and be build for seven different mobile operating systems (iOS, Android, BlackBerry OS, WebOS, Windows Phone 7, Symbian and Bada) [16, 18]. PhoneGap offers access to the native functions and features of the supported mobile phones, for example accelerometer, storage, notification, camera and contacts [18].
In order to build appropriate widgets which are suitable for the navigation by fingers, it is additionally reasonable to make use of a touch-optimized framework for smartphones. The two most famous according frameworks are Sencha Touch and jQuery Mobile [20, 23]. In case of Backstage Mobile we are using jQuery Mobile as Sencha Touch supports only iOS, Android and BlackBerry OS [19]. In contrast jQuery Mobile is not only suitable for eight different mobile operating systems (iOS, Android, BlackBerry, Bada, Symbian, Web OS, Windows Phone and MeeGo) but also for mobile and desktop web browsers [24].
By using PhoneGap and jQuery Mobile, Backstage Mobile can not only be developed quite independently for all smartphone operating systems, but also share the same look and feel on all mobile platforms [12]. To keep it simple, the implementa-

tion was started with two platforms, Android and iOS, as further operating systems could easily be added in the end by copying the code. For the implementation of an Android application the following prerequisites were downloaded and installed on a desktop computer running Windows 7:

- Eclipse Helios 2 IDE [1]
- JDK 1.6.0_22 [2]
- Android SDK 20.0.0 [3]
- Android ADT 20.0.0 Plugin for Eclipse
- PhoneGap SDK 1.8.1 [4]

The implementation of an iOS application requires an Mac OS X system, hence a MacBook Pro running Mac OS 10.7.4 (Lion) was used and Xcode 4.4.1[5] and the PhoneGap SDK 1.6.0 installed. The Android application used jQuery 1.7.2 minified[6] and jQuery Mobile 1.1.0 minified (JavaScript and CSS)[7], while the iOS application used jQuery 1.7.1 minified and jQuery Mobile 1.1.1 minified (JavaScript and CSS).

Due to the fact, that Backstage is still a prototype and that the boxes concept is a totally new approach, the development of Backstage Mobile had to begin independently of the desktop application. We hoped that the development of Backstage would progress and incorporate the boxes approach while Backstage Mobile was still in the concept and user study phase, so that both platforms could be merged in the end of this thesis. Nevertheless, there were not enough developers to expedite the development of Backstage as far as we expected. Further complications made it difficult to implement Backstage Mobile the way we intended.
As Backstage Mobile's design phase was completed and a connection to the server was needed, Backstage did not integrate the boxes concept. Only basic concepts of Backstage were prepared for communicating with mobile devices, so that only those were implemented yet. Furthermore, the development of the Android application got stuck, when we tried to connect the emulator to the server available under the address http://localhost:8080. The server cannot be connected that way as localhost targets the emulator itself. Although several websites and bulletin boards suggest to use the ip 10.0.2.2 instead [5], this did not solve our problem. As the time of the thesis is restricted, the development of a version for Android was deferred and focused on developing a prototype version for iOS.
Furthermore, as the desktop version of Backstage makes use of DWR[8] and no extra connection interface shall be implemented for the mobile devices, Backstage Mobile

[1] http://www.eclipse.org
[2] http://www.oracle.com
[3] http://developer.android.com
[4] http://www.phonegap.com
[5] https://developer.apple.com/xcode/
[6] http://www.jquery.com
[7] http://www.jquerymobile.com
[8] http://www.directwebremoting.com

also has to use DWR. Nevertheless, using DWR with mobile devices led to further problems. We figured out that we had to include a timeout of one second before including our different modules via JavaScript, as otherwise the DWR-Session-ID will not be returned yet by the server and hence cause a failure. We know that setting a timeout in the code is not suitable for such a problem, but it serves as workaround until another solution is found, as a while statement waiting for the id to be set did not work.
The code and all screenshots of Backstage Mobile can be found on the enclosed DVD (cf. Appendix D).

In order to implement a tab bar at the bottom of the screen, a so-called navbar, custom icons are needed to match the intended ones, as jQuery Mobile offers only some standard symbols [22]. In most cases, professionally made icons for mobile devices require payment or at least providing attribution. In order to avoid judicial conflicts, custom shape brushes and Photoshop are used to create the icons [29, 25]. The icons of the resulting navbar, which can be seen in Figure 4.10, match quite exactly the ones used in the paper prototyping study, where they proved to be suitable.

Each screen of Backstage Mobile offers a title bar, telling the user where she is. When the mobile application is started, the login screen is shown, which can be seen in Figure 5.1, where username and password have to be entered. To support consistency, the known Backstage logo of the desktop application is used. Furthermore, this screen offers a button leading to another form for registering a new user. As users not knowing about Backstage are likely to not have their laptops with them for the first lecture, a possibility for registration is needed in order to participate in Backstage Mobile. In contrast to the desktop version, the registration form asks the user to enter the password twice because typing errors are more likely on small devices. Furthermore, exact error messages inform the user constructively if she forgot to fill in a field. If the user has successfully registered, she is informed about it and the upcoming forwarding to the login page by a popup message. As Backstage Mobile is a native and no web application, an option has to be offered where the server address can be changed. Therefore, a link in the bottom right corner of the login page leads to a screen where the address can be entered and saved on the local storage of the mobile device. Nevertheless, changing the address leads to a further conflict while statically including the "root.js" script from the server. Hence, another function has to be included to unload the script of the old address and load the new one, if the address was changed.
After successfully logging in, the above mentioned navbar is shown on the bottom, where the speech bubble icon is highlighted, the icons for writing a message and participating in the quiz are disabled and the icon for logout and setup is enabled. The rest of the screen shows the two drop down menus, as can be seen in Figure 5.2, where the first one is used to choose a lecture, and the second one shows the according sessions. By tapping on the backchannel open button, the user is forwarded to the live communication screen and the disabled buttons are enabled. Although we would prefer to offer the user the native drop-down menu of the de-

Figure 5.1: Login Screen of Backstage Mobile

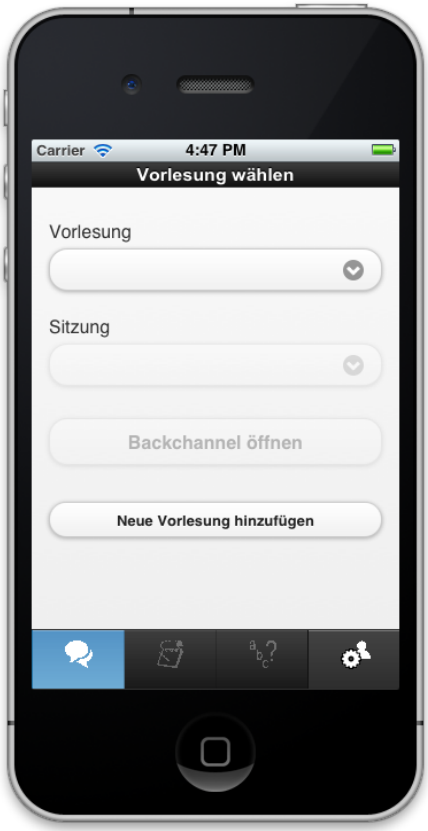


Figure 5.2: Screen for Choosing a Lecture and Session in Backstage Mobile

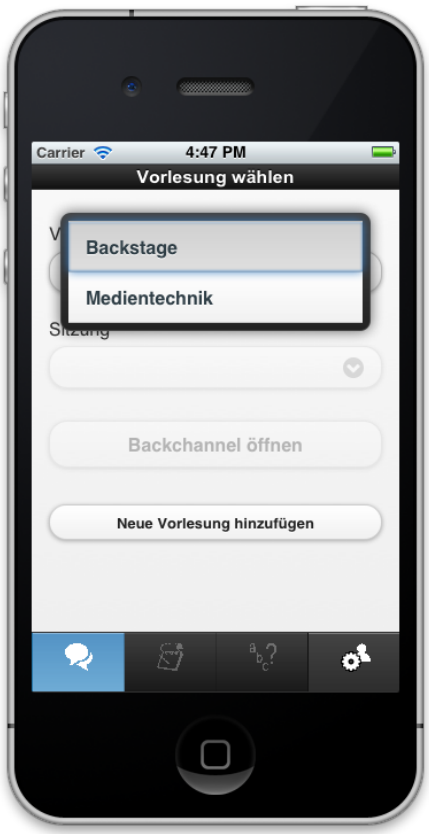


Figure 5.3: jQuery Mobile's Select Menu with Placeholder

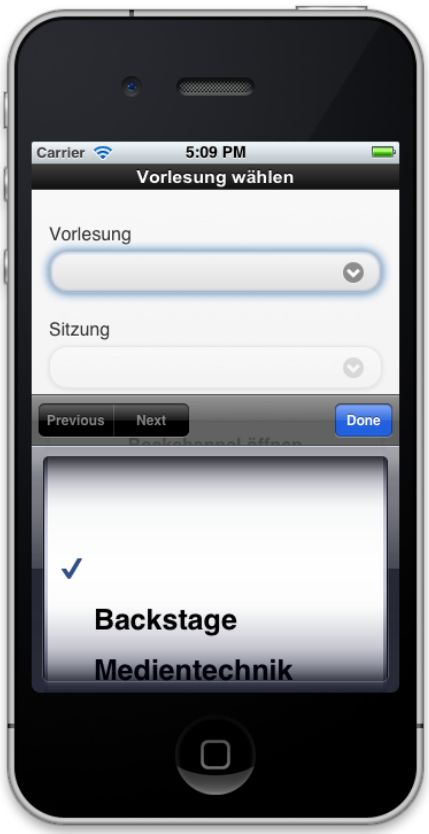


Figure 5.4: Native Select Menu with Placeholder

vice, as she is familiar with it, this was not possible with jQuery Mobile. The jQuery Mobile select menus offer a placeholder option, which is an empty or named element that titles the select menu, but does not belong to the selectable options. As can be seen in Figure 5.3 this works just fine for the jQuery Mobile theme, as the placeholder option is empty so that nothing is preselected and the user is forced to choose between the other selectable options. However, using the native select menu the placeholder is shown, but also selectable, which can be seen in Figure 5.4. As this does not only lead to further lines of code to avoid that users try to login with the empty placeholder as name of the lecture, but will also confuse the users the native menu is omitted and the jQuery Mobile select menu used.
In order to add a new lecture to ones personal list, the user can tap on the "add new lecture" button, which leads to another screen where the according lecture key can be entered.

While login and logout are animated by a so-called pop transition, for changing between the main screens of the application (login, lecture choice) and those which are will not be visited that often (server setup, registration, adding a lecture) a flip transition was chosen to show their difference implicitly. Between the main screens

which can be reached by the navbar icons, no transition was chosen so that an immediate switch is shown. As the hint for swiping is omitted, because it led to confusion during the paper prototyping study, an according slide transition was chosen to support the metaphor of sliding from one screen to the next. This slide transition is used between old and live communication and between old and current quiz screen.
Implementing these transitions once again led to further complications during the development of Backstage Mobile. The developers of jQuery Mobile 1.1.x added a fade effect to all transitions, which causes a confusing visualization of the sliding transition. As omitting the slide transition is no option for Backstage Mobile in order to support the swiping effect, a solution had to be found, how to use the new jQuery Mobile with the old transitions. Thankfully, one developer offers code snippets, which overwrite the new jQuery Mobile transitions with those of version 1.0.1 so that the slide transition can be used without the flickering fade effect [43].

The main view of Backstage Mobile consists of the four tabs mentioned above. When the backchannel is opened, the live communication screen is shown. On the top of the screen, the number of the current slide of the lecturer and the number of total slides is displayed. This area will also show the current topic in a future version of Backstage Mobile. Underneath this information, a list is shown, containing all messages according to the current slide. The list can be vertically scrolled, while the navbar stays fixed in its place.
Each post consists of a category icon, the author's nickname and the text of the message. Furthermore, an envelope with a number is shown in the upper right corner to symbolize existing answers to a post. As this functionality is not implemented yet, the icon was included for each message for testing purposes regarding the layout. Although the listview of jQuery Mobile is a powerful tool, it is difficult to handle in such a special case as Backstage Mobile. While the listview works fine without the category thumbnails, including them needs some adoption by appropriate CSS properties. As can be seen in Figure 5.5, dynamically loading the listview leads to a further problem, as it covers the area, where the current and total slides number is shown.
In order to show the post and its according answers, the envelope icon has to be clicked. Otherwise, to show further information about a post and the interaction bar including rating and answer buttons, the post itself has to be tapped. Therefore, two eventhandler were included, extracting the id of the according list element, which represents the id of the post. It is planned to include a certain div-element to each list item, which can be hidden and shown by the according jQuery functions (hide( ), show( )). This div-element shall include the further information about the post and the interaction bar consisting of four buttons in a row (answer, vote plus, vote minus, vote off-topic). Nevertheless, representing such a bar by the horizontal button controlgroup of jQuery Mobile seems not to work in a listview, as can be seen in Figure 5.6. As not even the buttons work, it was not possible to include the according function.
Although the listview seemed to be a good idea in the beginning of the development, each further function that was included raised new complications and prob-

Figure 5.5: Live Communication Screen. Listview hiding Slides Display (cf. red circle).

Figure 5.6: Horizontal Button Controlgroup does not work in Listview.

lems. Therefore, it has to be considered for a future version of Backstage Mobile to use individual div-elements or similar instead of the listview. Refreshing the listview is another problem coming along with it. As Backstage does not offer any event yet, which informs the clients about the lecturer changing the slide, Backstage Mobile calls this information manually from the server. As a temporarily solution, this information shall be called whenever the screen with the live messages is shown anew in Backstage Mobile. Hence, an eventhandler was added on showing this page which works fine, the first time the page is loaded, but if the user switches to another function and back to the live communication, the appearance of the listview crashes. Therefore, listviews might not be the appropriate instrument for displaying posts.
A swipe from left to right leads from the live communication screen to the old messages. This screen is not implemented yet, as the server does not offer the according filtering functions. Nevertheless, a future version of Backstage Mobile shall show the messages of the last slide and a filter to get messages by their slide number, topic, category or combinations out of it. Furthermore, the same functions on a post as in the live communication screen need to be offered. A swipe from right to left leads back to the live communication screen.

The second tab of the main screen leads to the screen for writing a message, which shows two select menus, a text input field and a send as well as cancel button. The first select menu is for choosing a category, afterwards the second select menu for choosing a slide number is enabled. The text input field should be disabled in the beginning and enabled by changing the first select menu. However, disabling a textarea in jQuery Mobile does not lead to the desired effect of greying out, which will confuse the users. Furthermore, enabling a disabled textarea does not seem to work, too. Therefore, this function is not included here and needs to be replaced by individual code and CSS. The cancel button resets the form, which does not work for select menus yet, and leads back to the live communication screen.

Due to the time restriction, it was not possible to include the quiz function. Furthermore, some of the desired functions are not supported by the server yet, for example checking if the own answer is correct or not. Nevertheless, a quiz mockup was created in HTML to show how it could look like, as can be seen in Figure 5.7. A swipe on the live quiz page leads to the old quizzes screen, where old questions of the sessions shall be archived.

So far the last tab, the logout and options screen, does only show an logout button which leads the user to the login screen again. Nevertheless, future options for the users or profile preferences could be offered within this screen.

The implementation of Backstage Mobile was more complicated than expected. Due to the delayed supply of a connection for the mobile devices and several missing functions it was not possible to implement all features of Backstage Mobile. Furthermore, working with code that is hardly commented or even documented is very complicated. Although Backstage Mobile is not completed yet, one can be

Figure 5.7: Login Screen of Backstage Mobile

confident that the further implementation can be promoted as soon as the necessary features are incorporated into Backstage.

CHAPTER 6

Conclusion and Future Work

Prior to drawing a conclusion of this diploma thesis, the ideas and different aspects for the future work on Backstage Mobile have to be discussed. Especially including the boxes concept will raise new questions.

6.1 Future Work and Challenges

The development of Backstage Mobile shows that there are several obstacles that can occur during the implementation of a design, which have not been considered before, like the select menu in the lecture choice screen.
Another challenge will come along with the boxes concept, that has been chosen. Advantageously, it is not only suitable for mobile devices and the users of the paper prototyping study could cope with it, but also seems to be combinable with the existing desktop concept and application. Nevertheless, the user study did not comprise the effect of this concept on the users of the desktop application. It will be a great challenge to find an appropriate algorithm to arrange the category icons produced by mobile users automatically on the slides of the desktop users. The placed icons should not only be near the according topic, but also avoid several icons upon each other or placing so much icons at one spot that nothing can be read anymore. It has to be analysed, whether a "rough" algorithm, i.e. one that does not place the icons exactly next to the topic, leads to any constraints for the users. For example, it could happen that the "roughly" placed icons of the mobile users do not attract the interest of the desktop users as a precisely placed icon would do. Hence, mobile users could suffer from the disadvantage that their messages are not read and voted as often as the other ones. Otherwise, the desktop users could be confused by the way the "rough" algorithm places the icons of the mobile users, as it does not seem reasonable for them.
The boxes concept bears another difficulty. It has to be chosen, if the according arrangement algorithm shall be executed on the server- or client-side. Both parties

have all data they need to implement such an algorithm. One approach could be that the algorithm is executed on the client side, so that the server will be informed about the slide number, topic box as well as the x and y coordinates where to place the icon on the slide, as it is the case for the desktop clients using the iconic drag-and-drop approach. The advantage of this approach is that the interface within the server for receiving the messages does not have to differ between handling mobile devices and desktop computers. On the other hand, this approach is of great disadvantage for the mobile devices, because of the additional load. As mentioned in this thesis, mobile devices have a lower processing power and memory and should be spared whenever possible. The second approach envisages including a small if-statement on the server-side. Hence, the mobile device will send the slide number, topic box and special coordinates, for example each -1, so that the server can easily differ between mobile and desktop users by checking those coordinates. In case of a mobile user the arrangement algorithm has to be started to calculate the new coordinates.

Offering Backstage Mobile for the students will also show, if there is a need for supporting tablet computers. These users could be comfortable with using Backstage Mobile on their tablets, but also wish to rather choose using the desktop version. If Backstage was redesigned again, it could be made a little more space-saving and omit the on mouse events, so that it will be suitable for both, desktop and tablet users.

The design guidelines considering the context-awareness of mobile devices gave a new idea for the future development of Backstage Mobile. If it was possible to extract certain information from the context, for example the date and time as well as the lecture hall, it would be possible to automatically choose the lecture and session for the users. Furthermore, in case of a version of Backstage that will support users who are not present in the lecture hall, for example by video streaming, context-awareness might be even more interesting. Nevertheless, context-awareness brings up a lot of new challenges, especially when to adapt to the context, which have to be discussed carefully.

A last and major aspect for the future work on Backstage Mobile includes the missing design of an interface for the lecturer. As discussed in the beginning of this thesis, the lecturer will probably always need a notebook to at least project the slides on the wall. Furthermore, the elaborated and broad functions in Backstage offered to the lecturer would hardly fit on a small screen device. Therefore, Backstage Mobile could rather serve as an extension to the lecturer's desktop interface, trying to support her while she is not staring at her notebook. It can be imagined that the lecturer walks through the hall talking about a topic and thereby forgetting about the backchannel communication. As mentioned in the context of the boxes concept, the lecturer could be informed about certain activities on the backchannel by her mobile device. For example, an alert can draw her attention to the backchannel if an interesting message comes up. Even more, a short statistical overview and the current ranking can give her an impression what is going on in the backchannel by

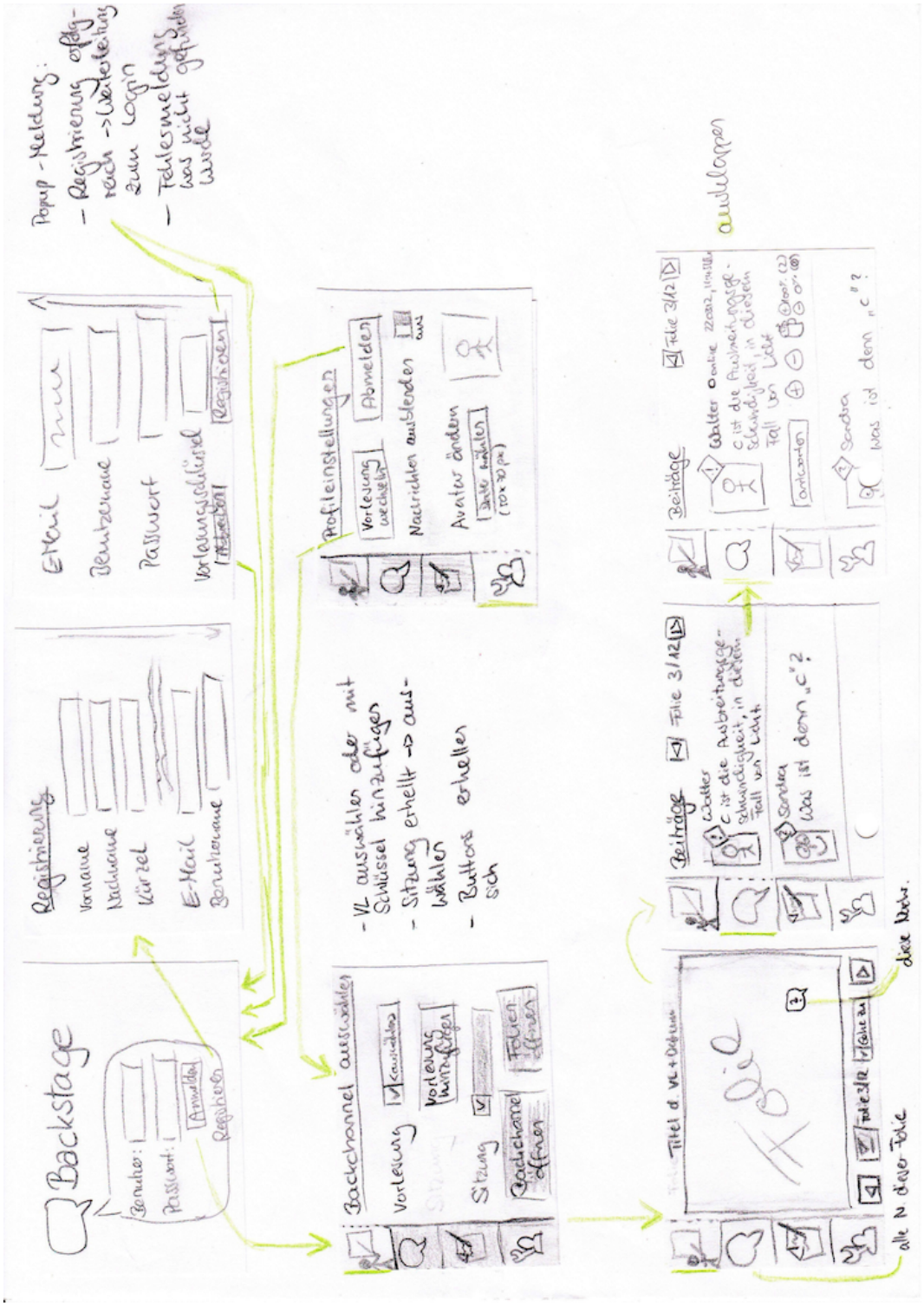
one glimpse on the mobile device. Furthermore, Backstage Mobile could be some kind of "remote" for the desktop backchannel, by offering buttons to navigate in the slides or start and stop quizzes. This idea would free the lecturer from staying at the notebook all the time.

6.2 Conclusion

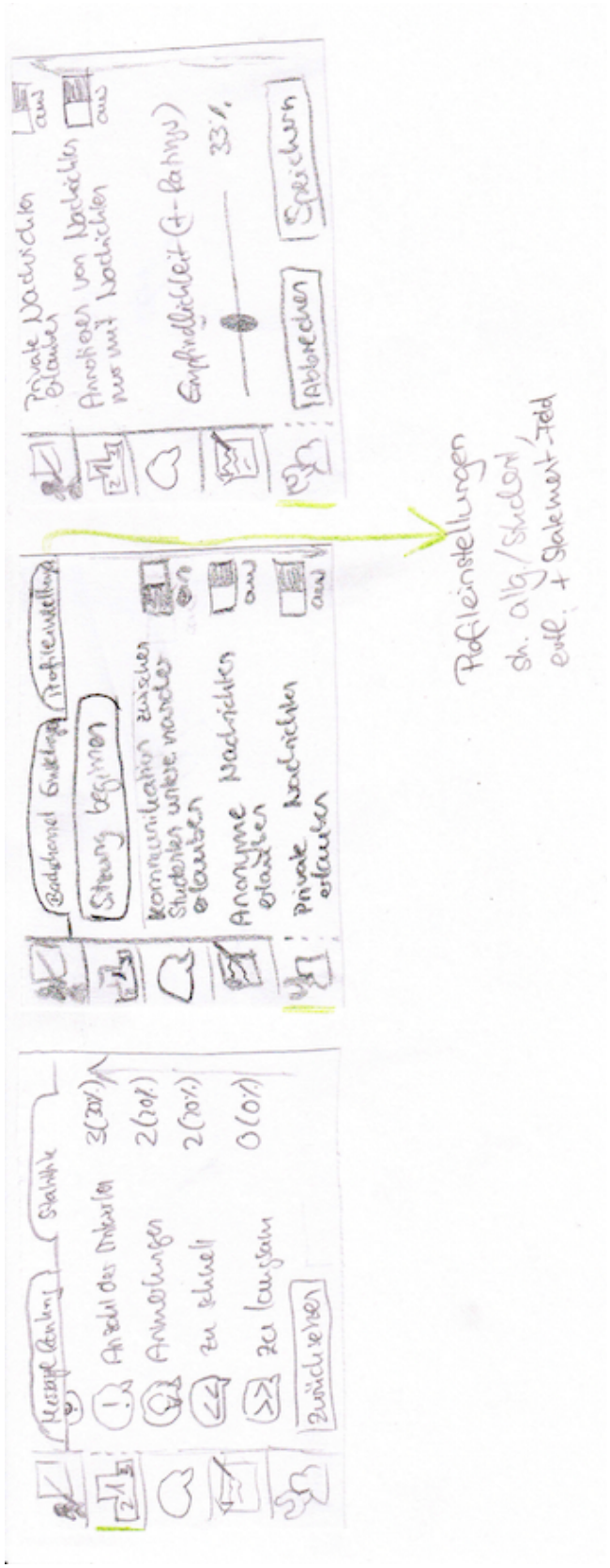
This diploma thesis does not only give an overview of the most important usability guidelines for desktop applications, but also how they can be adapted for mobile phones and which further challenges have to be considered when designing for small devices. Applying usability guidelines to concrete situations is no clear yes or no decision, but rather following the advice to use solutions that have proved to be suitable in certain situations. Thus, educational claims and usability guidelines can also be combined whenever needed. Nevertheless, in certain cases usability guidelines have to be disregarded in order to support an important educational claim. Whenever these guidelines conflict a careful discussion of the advantages and disadvantages of different solutions has to take place.
The development of Backstage Mobile shows that designers tend to transfer desktop versions to mobile devices - although this is not recommended - because this is the most obvious approach itself. The analysis of the first design shows that, especially due to the small screen, designers need to replace some concepts of a desktop application by others that might be more suitable for mobile devices. Furthermore, the results of the paper prototyping study prove, that it is very important and profitable to communicate with the users. The study did not only reveal certain misconceptions, but also gave an impulse for many good ideas.
Unfortunately, the implementation of the necessary features in Backstage is delayed, so that Backstage Mobile is not completed yet. Nevertheless, the basic modules and concepts have been designed and tested in the paper prototyping study, so that they can be implemented whenever needed. Further developments and ideas might turn Backstage and Backstage Mobile into an outstanding application for educational use.

Appendix A

Storyboard of the First Design


Backstage
Benutzer:
Passwort:
Anmelden
Registrieren
Registrierung
Vorname
Nachname
Kürzel
E-Mail
Benutzername
E-Mail
Benutzername
Passwort
Vorlesungsschlüssel
Registrieren
Popup-Meldung:
- Registrierung erfolgreich -> Weiterleitung zum Login
- Fehlermeldung was nicht gefunden wurde
Backchannel auswählen
Vorlesung
Vorlesung hinzufügen
Sitzung
Backchannel öffnen
Folien öffnen
- VL auswählen oder mit Schlüssel hinzufügen
- Sitzung erhellt -> auswählen
- Buttons erhellen sich
Profileinstellungen
Vorlesung wechseln
Abmelden
Nachrichten ausblenden
Avatar ändern
Datei wählen
Titel d. VL + Datum
Folie
alle N dieser Folie
diese Nachr.
Beiträge
Folie 3/12
Walter
Sandra
Was ist denn „c"?
Beiträge
Folie 3/12
Walter
Sandra
Was ist denn „c"?
auslappen

hier zwei Feedback mgl.
a) explizites Popup zur Bestätigung
b) Wechsel zu Nachrichten-tab mit neuer Nachricht
Nachricht verfassen
Wähle eine Kategorie für deine Nachricht
Frage
Antwort
Anmerkung
zu schnell
zu langsam
Nachricht verfassen
Platziere deine Nachricht
Folie
Folie 3/12
OK
Nachricht verfassen
Textengabefeld
Nachricht senden
PROF:
Backchannel auswählen
Vorlesung
Vorlesung löschen
Vorlesung hinzufügen
Sitzung
Sitzung löschen
Sitzung hinzufügen
Backchannel öffnen
Folien öffnen
Vorlesung hinzufügen
Schlüssel
Titel
Beschreibung
Abbrechen
Sitzung hinzufügen
Datum
TT-MM-JJJJ
Titel
Folien Upload (PDF)
Powerpoint/Open Office
Latex
Datei wählen
TT-MM-JJJJ
Titel
Folien Upload (PDF)
Powerpoint/Open Office
Latex
Datei wählen
Abbrechen
Sitzung erstellen
Message Ranking
Statistik
Wallet
Message Ranking
Statistik
angemeldete Nutzer 2 5
Anzahl alles Nachrichten 10
Anzahl off-topic 0(0%)
Anzahl Bewertungen 2(20%)
Anzahl der Fragen 1(10%)


Private Nachrichten erlauben
Annotieren von Nachrichten nur mit Nachrichten
Empfindlichkeit (+-Rating)
33%
Abbrechen
Speichern
Profileinstellungen
Sh. allg./Student, evtl. + Statement-Feld
Bachelorrel Gruppe
Profileinstellungen
Sitzung beginnen
Kommunikation zwischen Studenten untereinander erlauben
Anonyme Nachrichten erlauben
Private Nachrichten erlauben
Statistik
Anzahl der Antworten 3(30%)
Anmerkungen 2(20%)
zu schnell 2(20%)
zu langsam 0(0%)
Zurücksetzen

Appendix B

Tasks for the Paper Prototyping Study

Introduction

The platform Backstage serves to facilitate the communication among students and the lecturer in large lectures. In Backstage the participants can pose questions about the topic of the lecture, give answers or make annotations. Additionally, these messages can be voted, to express agreement or disagreement with a message. Quiz questions, posed by the lecturer every now and then, serve as self-monitoring for the students and allow the lecturer to gain insight to the level of knowledge of her audience.

Task No. 1 - Login (warm up)

Goal/output:
The user is logged in for a certain lecture and session. The current live communication is shown.
Inputs:
Username, password, lecture name, session date.
Assumptions:
The user is already registered for Backstage and for three different lectures. The lecture "Medientechnik" offers four different session dates.
Steps:

1. Tap input field for username and enter username
2. Tap input field for password and enter password
3. Tap login button
4. Tap lecture drop-down menu and tap on "Medientechnik"
5. Tap session drop-down menu and tap on "25.06.2012"
6. Tap open backchannel button

Time for expert:
One minute

Instructions for user:
Your name is Tom and you participate among others in the lecture “Medientechnik”, where Backstage is used. You just arrived late in the lecture hall and want to use your smartphone to log in to Backstage.
Log in to Backstage (username “Tom”, password “12345”) and open todays (“25.06.2012”) session.
Notes:
Ask users where to find logout function

Task No. 2 - Write a Message

Goal/output:
The question is sent and shown on the live communication screen.
Inputs:
Category, slide number, topic, messagetext (question)
Assumptions:
User is logged in. The live communication screen is shown.
Steps:

1. Tap icon for writing a message in the tab bar
2. Tap category drop-down menu and choose question
3. Tap slide number drop-down menu and choose slide number
4. Tap topic drop-down menu and choose "Lichtbrechung"
5. Tap text input field and type message in
6. Tap send button

Time for expert:
two minutes
Instructions for user:
The lecturer is explaining light refraction (“Lichtbrechung”) on slide 12, but she does not mention what the variable “c” stands for. Pose the question “What does ’c’ stand for in the formula?” (“Wofür steht ’c’ in der Formel?”) on Backstage.
Notes:
—

Task No. 3 - Vote for a Older Message

Goal/output:
The rating button is disabled (grey), the rating value is refreshed.
Inputs:
—
Assumptions:
The message that should be voted has to exist. User has to know about slide number, topic or category of the according message for filtering. The live communica-

tion screen is shown.
Steps:

1. Swipe from left to right
2. Tap category, slide number or topic drop-down menu and choose according data
3. Tap Beate's message
4. Tap the positive rating button

Time for expert:
Two minutes
Instructions for user:
Although the lecture has proceeded and the lecturer is already on the next slide, you want to support Beate's message, that she would prefer an exercise about this topic, on slide 12. Vote positively for Beate's message.
Notes:
—

Task No. 4 - Read an Answer

Goal/output:
User has read the answer and switched back to the communication live screen.
Inputs:
—
Assumptions:
An answer to the question exists. The live communication screen is shown.
Steps:

1. Swipe from left to right
2. Tap category, slide number or topic drop-down menu and choose according data
3. Tap envelope symbol
4. Read answer
5. Swipe from right to left

Time for expert:
Two minutes
Instructions for user:
Find out, if anyone has answered your question of slide 12 meanwhile and if so read the answer(s). Afterwards check out on which slide the lecturer actually is and if there exist any new messages related to that slide.
Notes:
Ask user which slide the lecturer is currently talking about. Ask user about alternative speech bubble icon.

Task No. 5 - Participate in a Quiz

Goal/output:
User has participated in the quiz and the results are shown.
Inputs:
Answer for the quiz.
Assumptions:
Quiz question has to exist. The live communication screen is shown.
Steps:

1. Tap quiz icon in the tab bar
2. Tap "Reflexion" (radio button or text)
3. Tap send button
4. Close pop-up window by tapping OK
5. Wait for quiz to stop
6. Read results

Time for expert:
One minute
Instructions for user:
The lecturer announces to pose a quiz question now. You are sure about the answer and want to participate in the quiz. Answer the quiz question with “Reflexion” and check if your answer is correct.
Notes:
Ask user if her answer was correct or not. Ask user how many percent of the fellow students answered correctly.

Appendix C

Paper Prototype

Printed Slides

Optische Grundprinzipien

Reflexion (lichtundurchlässiges Medium):

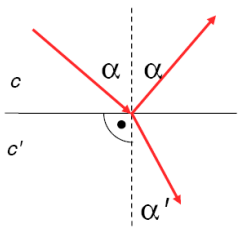
- Einfallender Strahl, Einfallslot, ausfallender Strahl: eine Ebene
- Einfallswinkel = Ausfallswinkel

Brechung (lichtdurchlässiges Medium):

- Einfallender Strahl, Einfallslot, gebrochener Strahl: eine Ebene
- Brechung bestimmt durch Verhältnis der Ausbreitungsgeschwindigkeit des Lichts in den beiden (physikalischen) Medien (z.B. Glas und Luft)

$$n = \frac{\sin\alpha}{\sin\alpha'} = \frac{c}{c'}$$

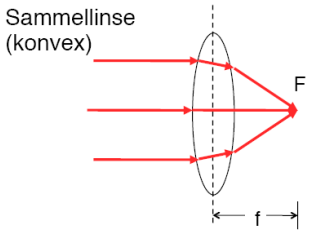
n Brechungszahl
c, c' Ausbreitungsgeschwindigkeiten



Optisches Glas: definierte Brechungszahl

LMU München – Sommer 2012 Prof. Hußmann: Medientechnik Kap. 1 Teil a – Folie 12

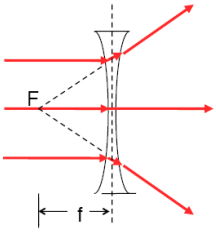
Linsenoptik, Brennweite



- f = Brennweite *(focal distance)*
- F = Brennpunkt *(focal point)*

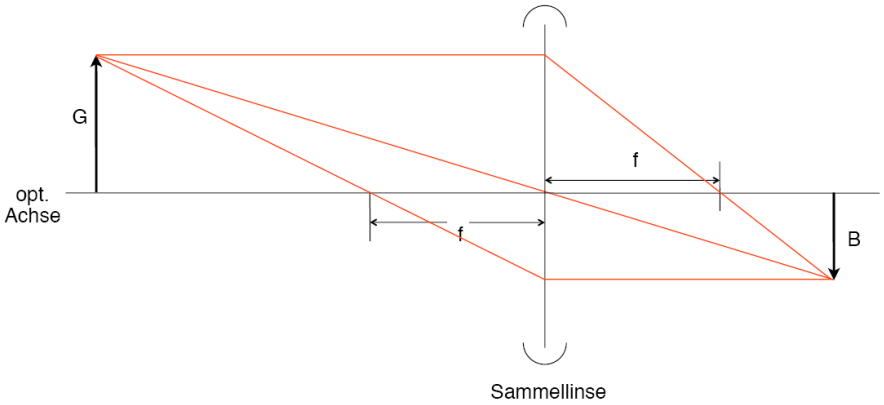
- *Objektive* sind komplexe Kombinationen von Linsen mit der Gesamtwirkung einer sehr guten Sammellinse
- Brennweite kann fest oder verstellbar sein *(Zoom-Objektiv)*

Zerstreuungslinse
(konkav)



LMU München – Sommer 2012 Prof. Hußmann: Medientechnik Kap. 1 Teil a – Folie 13

Strahlengang an einer Sammellinse



Strahlen, die von einem Punkt eines Gegenstandes G her auf eine Sammellinse fallen, schneiden einander nach dem Durchgang durch die Linse in einem im Bildraum gelegenen Punkt.

LMU München – Sommer 2012 Prof. Hußmann: Medientechnik Kap. 1 Teil a – Folie 14

Projected Slides

Optische Grundprinzipien

Reflexion

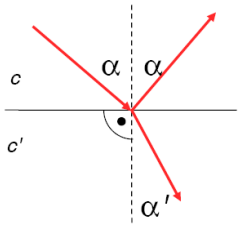
Reflexion (lichtundurchlässiges Medium):
- Einfallender Strahl, Einfallslot, ausfallender Strahl: eine Ebene
- Einfallswinkel = Ausfallswinkel

Lichtbrechung

Brechung (lichtdurchlässiges Medium):
- Einfallender Strahl, Einfallslot, gebrochener Strahl: eine Ebene
- Brechung bestimmt durch Verhältnis der Ausbreitungsgeschwindigkeit des Lichts in den beiden (physikalischen) Medien (z.B. Glas und Luft)

$$n = \frac{\sin\alpha}{\sin\alpha'} = \frac{c}{c'}$$

n Brechungszahl



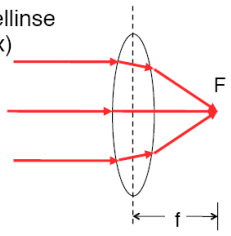
Optisches Glas: definierte Brechungszahl

LMU München – Sommer 2012 Prof. Hußmann: Medientechnik Kap. 1 Teil a – Folie 12

Linsenoptik, Brennweite

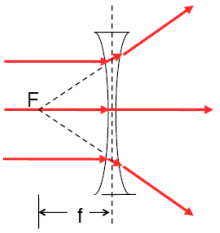
Sammellinse

Sammellinse (konvex)



- f = Brennweite *(focal distance)*
- F = Brennpunkt *(focal point)*

Zerstreuungslinse



Zerstreuungslinse (konkav)

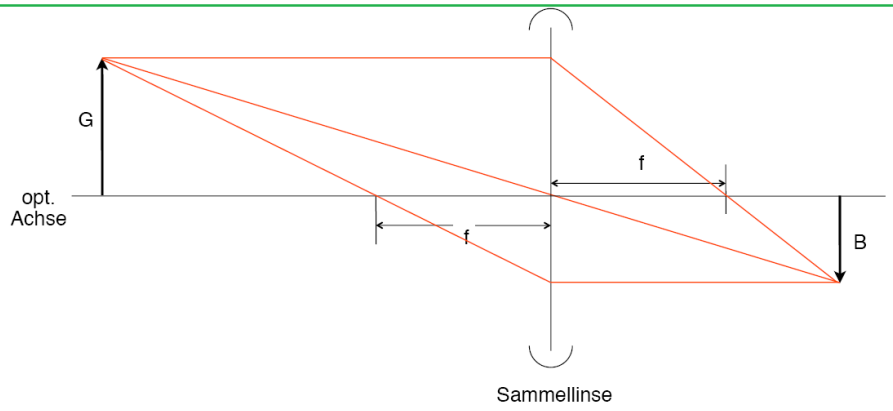
Objektive

- *Objektive* sind komplexe Kombinationen von Linsen mit der Gesamtwirkung einer sehr guten Sammellinse
- Brennweite kann fest oder verstellbar sein *(Zoom-Objektiv)*

LMU München – Sommer 2012 Prof. Hußmann: Medientechnik Kap. 1 Teil a – Folie 13

Strahlengang an einer Sammellinse

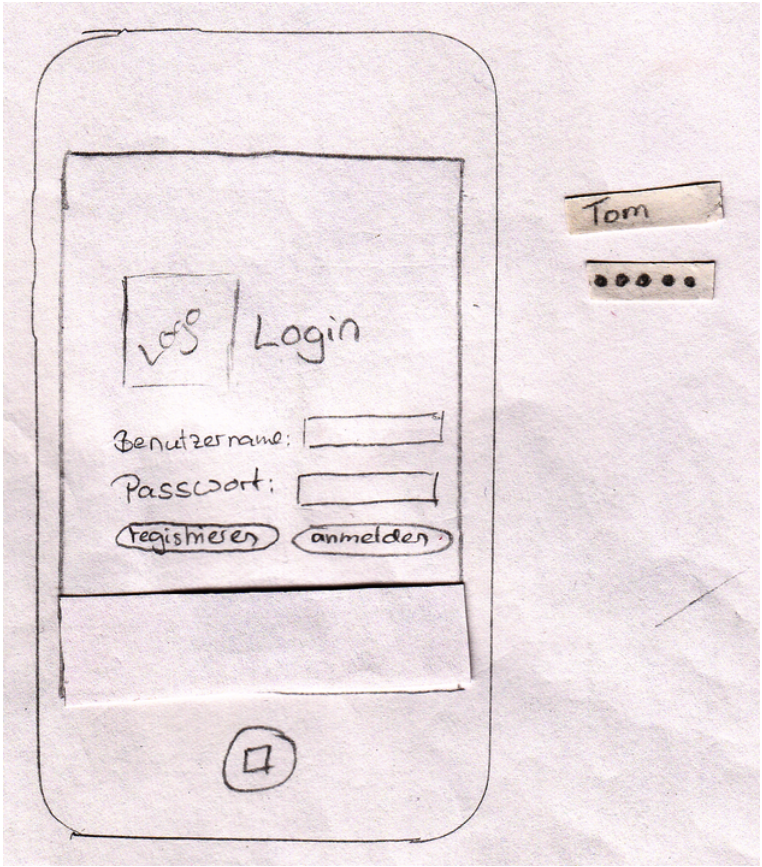
Strahlengang

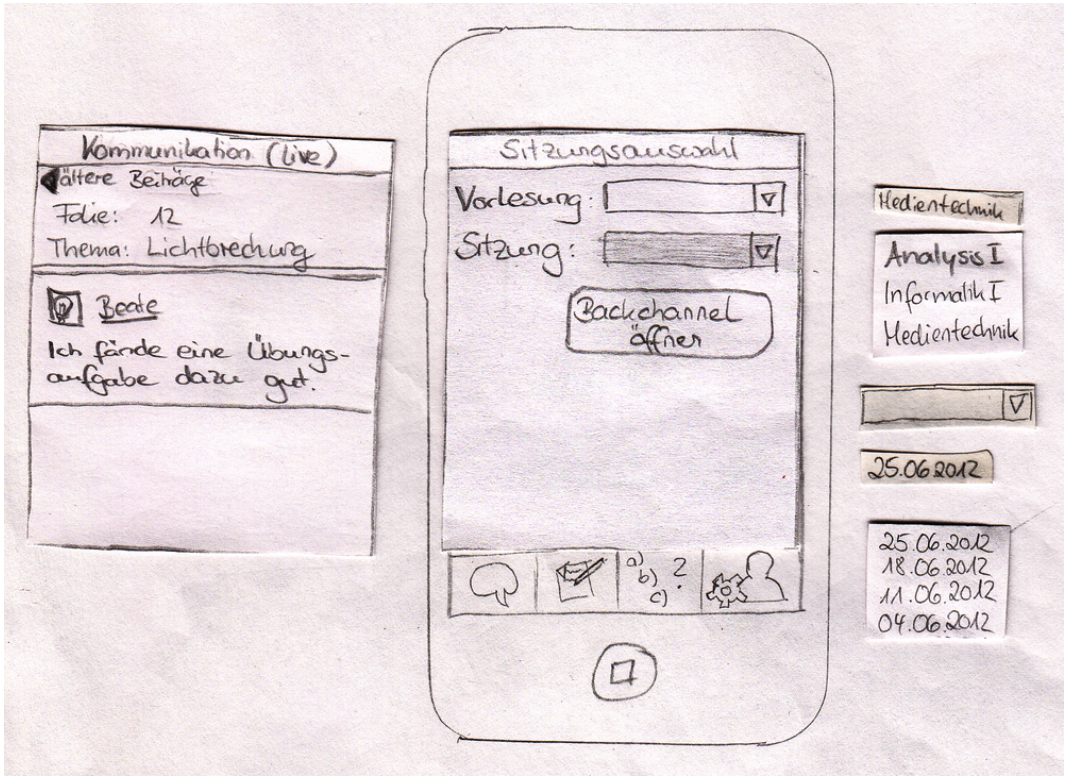


Strahlen, die von einem Punkt eines Gegenstandes G her auf eine Sammellinse fallen, schneiden einander nach dem Durchgang durch die Linse in einem im Bildraum gelegenen Punkt.

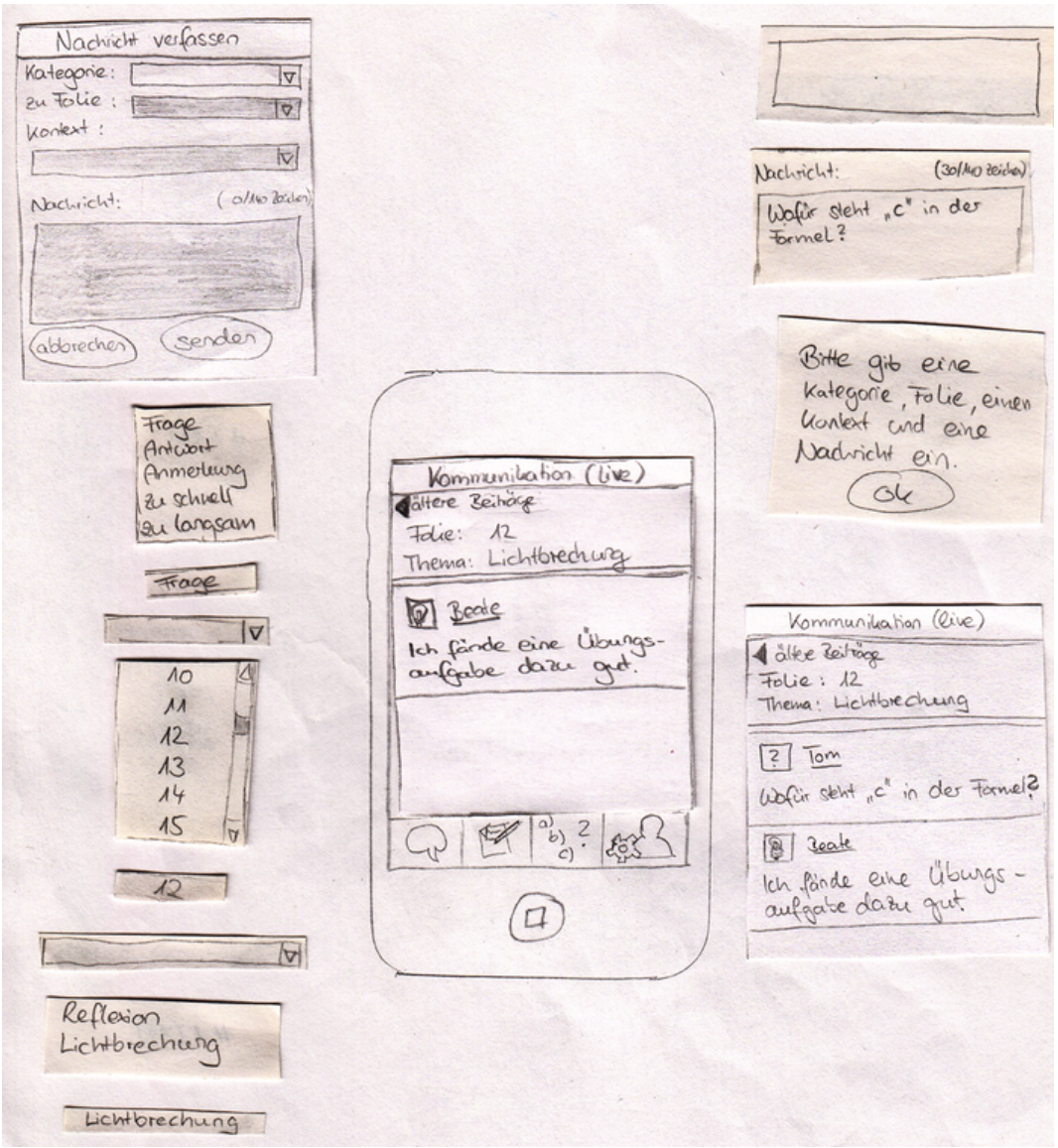
LMU München – Sommer 2012 Prof. Hußmann: Medientechnik Kap. 1 Teil a – Folie 14

Parts of the Paper Prototype for Login and Lecture Choice

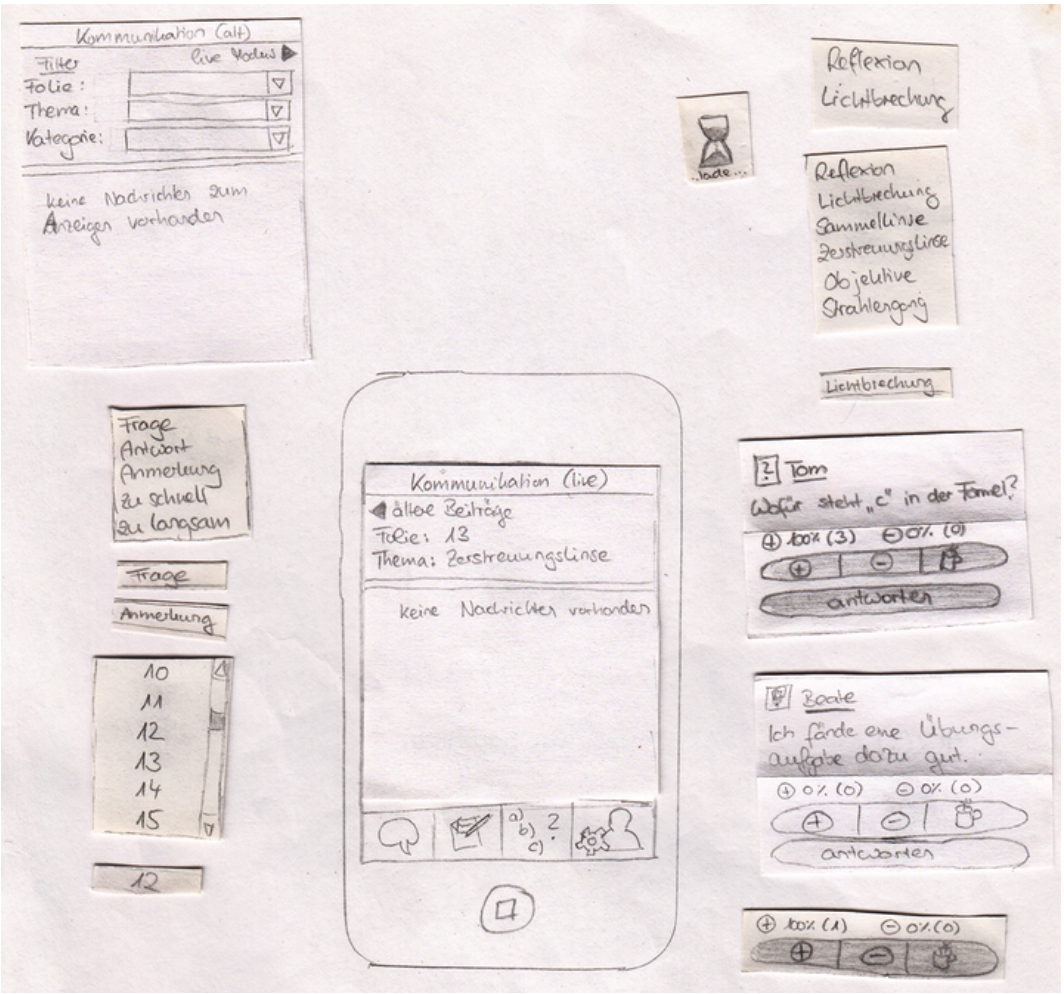



Kommunikation (live)
ältere Beiträge
Folie: 12
Thema: Lichtbrechung
Beate
Ich fände eine Übungs-aufgabe dazu gut.
Sitzungsauswahl
Vorlesung:
Sitzung:
Backchannel öffnen
a)
b)
c)
Medientechnik
Analysis I
Informatik I
Medientechnik
25.06.2012
25.06.2012
18.06.2012
11.06.2012
04.06.2012

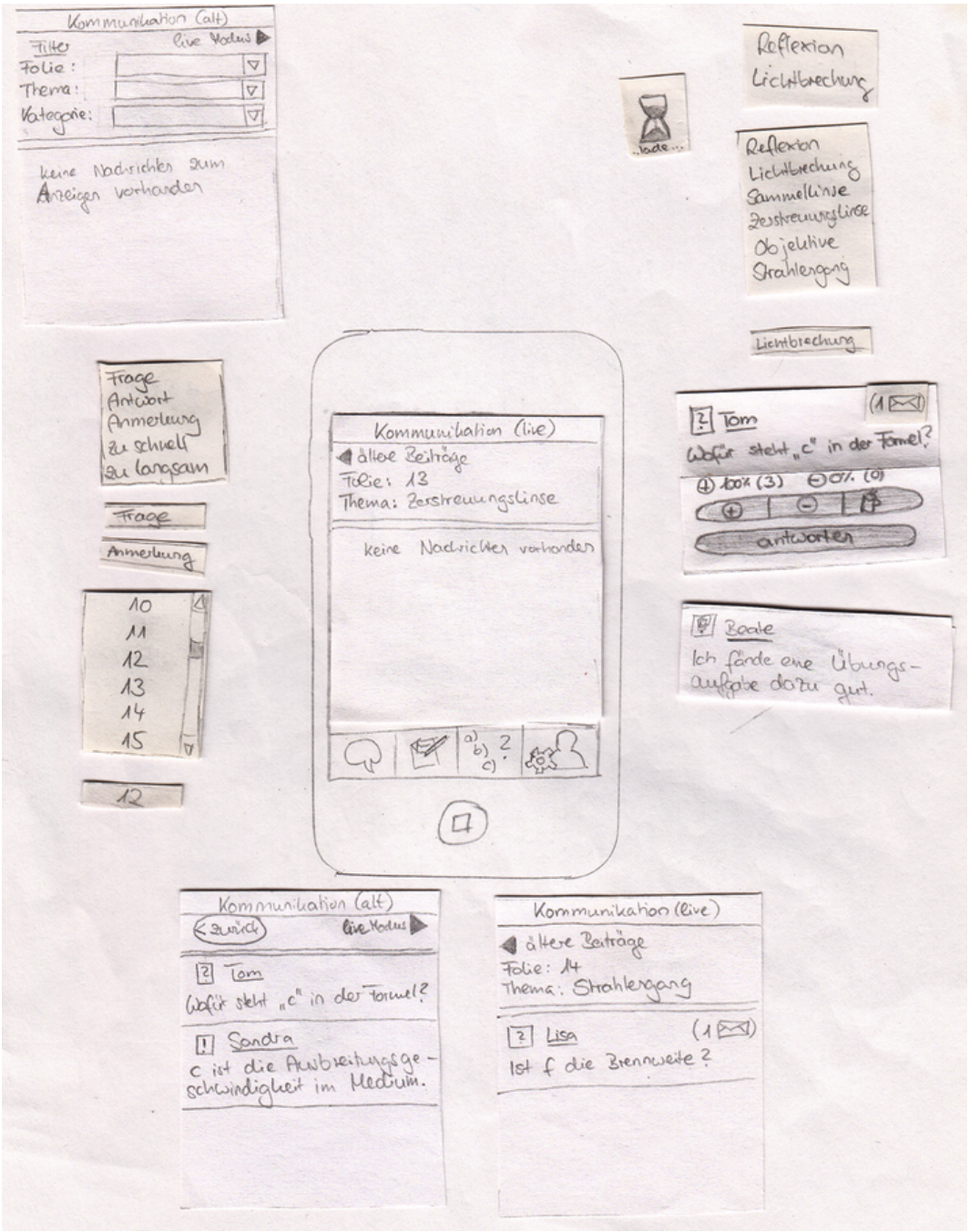
Parts of the Paper Prototype for Writing a Message



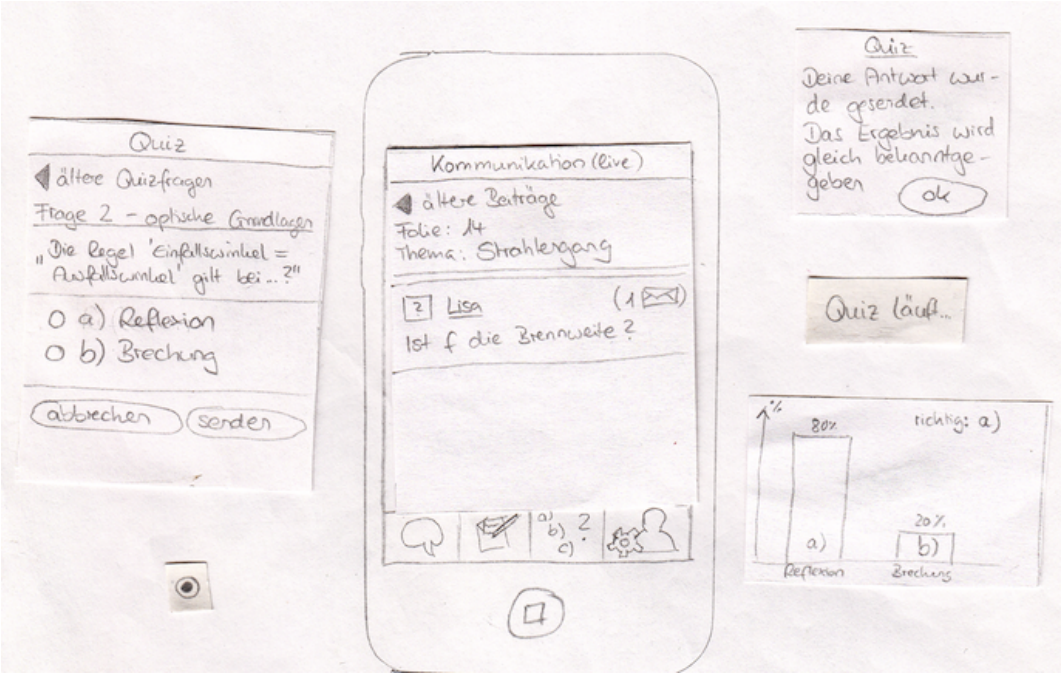
Parts of the Paper Prototype for Rating an Older Message



Parts of the Paper Prototype for Reading Answers

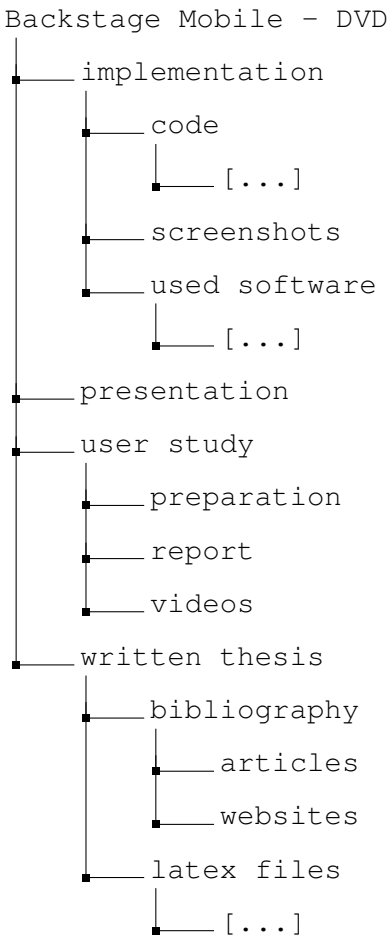


Parts of the Paper Prototype for Participating in the Quiz



Appendix D

Content of the enclosed DVD



List of Figures

3.1 Norman's Seven Stages of Action [36] 16
3.2 Lecture and Session Choice in Backstage 21
3.3 Students' Dashboard in Backstage 22
3.4 Input Textbox for Writing a Message 23
3.5 Information about and Rating of Messages 24
3.6 Lecturers' Dashboard in Backstage 25
3.7 Date Entry for a new Session 29
3.8 Lecturer's Options for Backstage 30
3.9 Number of Characters for a Message is not Constrained 31

4.1 Displaying Backstage's dashboard on a mobile device 37
4.2 Tom Huber [11] 51
4.3 Jennifer Meyer [11] 51
4.4 Peter Kruse [11] 52
4.5 Part of the Storyboard of the First Design for Backstage Mobile 55
4.6 Trade-Off between Number of Tested Users and Found Usability Problems [31] 59
4.7 Live communication screen. The arrow, marked by the red rectangle, gives a hint for swiping. 61
4.8 Rating and Answer Buttons in the Paper Prototype 62
4.9 Alternative Speech Bubble Icon for Backstage Mobile 63

4.10 Tab bar at the bottom of the screen. Grey icons are disabled, white icons are enabled and activated tabs are highlighted by a blue background colour. 66

5.1 Login Screen of Backstage Mobile 74
5.2 Screen for Choosing a Lecture and Session in Backstage Mobile . . . 74
5.3 jQuery Mobile's Select Menu with Placeholder 75
5.4 Native Select Menu with Placeholder 75
5.5 Live Communication Screen. Listview hiding Slides Display (cf. red circle). 77
5.6 Horizontal Button Controlgroup does not work in Listview. 77
5.7 Login Screen of Backstage Mobile 79

Bibliography

[1] *Maintaining consistency - frontend - user experience design consultancy*, http://www.frontend.com/design/maintaining-consistency.html, 30.08.2012.

[2] J. Armitage, *From user interface to über-interface: a design discipline model for digital products*, interactions **10** (2003), no. 3, 18–29.

[3] D. Baumgart, A. Pohl, V. Gehlen-Baum, and F. Bry, *Providing guidance on backstage, a novel digital backchannel for large class teaching*, Education in a Technological World: Communicating Current and Emerging Research and Technological Efforts, Formatex, Spain (2011).

[4] N. Bevan, *Usabilitynet: International standards*, http://www.usabilitynet.org/tools/r_international.htm#9241-11, 30.08.2012.

[5] Martin Breuer, *Ever tried 'localhost' with the android emulator? on android development*, http://www.droidnova.com/ever-tried-localhost-with-the-android-emulator,102.html, 13.09.2012.

[6] S. Brewster, *Overcoming the lack of screen space on mobile computers*, Personal and Ubiquitous Computing **6** (2002), no. 3, 188–205.

[7] F. Bry, V. Gehlen-Baum, and A. Pohl, *Promoting awareness and participation in large class lectures: The digital backchannel backstage*, Proceedings of the IADIS International Conference on e-Society, Avila, Spain, 2011, pp. 27–34.

[8] G. Buchanan, S. Farrant, M. Jones, H. Thimbleby, G. Marsden, and M. Pazzani, *Improving mobile internet usability*, Proceedings of the 10th international conference on World Wide Web, ACM, 2001, pp. 673–680.

[9] A. Cooper, R. Reimann, and D. Cronin, *About face 3: the essentials of interaction design*, Wiley-India, 2007.

[10] A. Dix, *Human-computer interaction*, Prentice hall, 2004.

[11] FreeDigitalPhotos.net, *Free photos - free images - royalty free photos - free stock photos - freedigitalphotos.net*, http://www.freedigitalphotos.net, 06.09.2012.

[12] R. Ghatol and Y. Patel, *Beginning phonegap: Mobile web framework for javascript and html5*, Apress, 2012.

[13] J. Gong and P. Tarasewich, *Guidelines for handheld mobile device interface design*, Proceedings of DSI 2004 Annual Meeting, Citeseer, 2004, pp. 3751–3756.

[14] Zoltán Gózca, *Myth #23: Choices should always be limited to 7+/-2 - ux myths*, http://uxmyths.com/post/931925744/myth-23-choices-should-always-be-limited-to-seven, 30.08.2012.

[15] J. Häkkilä and J. Mäntyjärvi, *Developing design guidelines for context-aware mobile applications*, Proceedings of the 3rd international conference on Mobile technology, applications & systems, ACM, 2006, p. 24.

[16] Adobe Systems Inc., *Phonegap, home*, http://phonegap.com, 13.09.2012.

[17] ———, *Phonegap, license*, http://phonegap.com/about/license, 13.09.2012.

[18] ———, *Phonegap, supported features*, http://phonegap.com/about/feature, 13.09.2012.

[19] Sencha Inc., *Built on html5 - features of sencha touch 2, features, sencha touch, products, sencha*, http://www.sencha.com/products/touch/features/, 13.09.2012.

[20] ———, *Mobile app development platform, sencha touch, products, sencha*, http://www.sencha.com/products/touch/, 13.09.2012.

[21] M. Jones and G. Marsden, *Mobile interaction design*, Wiley, 2006.

[22] The jQuery Foundation, *jquery mobile docs - navbar*, http://jquerymobile.com/demos/1.1.1/docs/toolbars/docs-navbar.html, 13.09.2012.

[23] _____, *jquery mobile, jquery mobile*, http://jquerymobile.com, 13.09.2012.

[24] _____, *Mobile graded browser support, jquery mobile*, http://jquerymobile.com/gbs/, 13.09.2012.

[25] Oksana Khristenko, *12 pretty chat photoshop shapes, photoshop custom shapes*, http://www.shapes4free.com/photoshop-custom-shapes/chat-photoshop-shapes/, 13.09.2012.

[26] E.L.C. Law, V. Roto, M. Hassenzahl, A.P.O.S. Vermeeren, and J. Kort, *Understanding, scoping and defining user experience: a survey approach*, Proceedings of the 27th international conference on Human factors in computing systems, ACM, 2009, pp. 719–728.

[27] A. Marcus, *Dare we define user-interface design?*, interactions **9** (2002), no. 5, 19–24.

[28] G.A. Miller, *The magical number seven, plus or minus two: Some limits on our capacity for processing information.*, Psychological review **101** (1994), no. 2, 343.

[29] myPhotoshopBrushes.com (Nathan Blenke), *Nate's shapes - photoshop custom shapes*, http://myphotoshopbrushes.com/custom_shapes/id/2024/, 13.09.2012.

[30] J. Nielsen, *10 heuristics for user interface design*, http://www.useit.com/papers/heuristic/heuristic_list.html, 02.09.2012.

[31] _____, *Usability testing with 5 users (jakob nielsen's alertbox)*, http://www.useit.com/alertbox/20000319.html, 10.09.2012.

[32] _____, *Usability engineering*, Morgan Kaufmann, 1994.

[33] ———, *Usability 101: Definition and fundamentals - what, why, how (jakob nielsen's alertbox)*, http://www.useit.com/alertbox/20030825.html, 30.08.2012.

[34] E.G. Nilsson, *Design guidelines for mobile applications*, SINTEF ICT, http://www.sintef.no/uploadpages/25723/design%20guidelines%20for%20web.pdf, 30.08.2012, 2008.

[35] D.A. Norman, *Don norman's jnd.org / ad-hoc personas & empathetic focus*, http://www.jnd.org/dn.mss/personas_empath.html, 04.09.2012.

[36] ———, *The psychology of everyday things*, Basic Books (AZ), 1988.

[37] U.S. Department of Health & Human Services, *Creating scenarios, usability.gov*, http://usability.gov/methods/analyze_current/scenarios.html, 05.09.2012.

[38] R. Oppermann, *User interface design*, AdCP2001 (2002).

[39] A. Pohl, F. Bry, J. Schwarz, and M. Gottstein, *Sensing the classroom: Improving awareness and self-awareness of students with backstage*, Proceedings of the International Conference on Interactive and Collaborative Learning, 2012.

[40] R. Romero, T. Zarraonandia, I. Aedo, and P. Díaz, *Designing usable educational material for english courses supported by mobile devices*, HCI in Work and Learning, Life and Leisure (2010), 373–383.

[41] V. Roto, M Lee, K. Pihkala, B. Castro, A. Vermeeren, E. Law, K. Väänänen-Vainio-Mattila, J. Hoonhout, and M. Obrist, *User experience definitions - all about ux*, http://www.allaboutux.org/ux-definitions, 30.08.2012.

[42] J.B. Rotter, *Generalized expectancies for internal versus external control of reinforcement.*, Psychological monographs: General and applied **80** (1966), no. 1, 1.

[43] Stephen Russell, *Fixing the transitions in jquery mobile 1.1.0, zsprawl's zapps*, http://zsprawl.com/iOS/2012/05/fixing-the-transitions-in-jquery-mobile-1-1-0/, 13.09.2012.

[44] C. Ryan and A. Gonsalves, *The effect of context and application type on mobile usability: an empirical study*, Proceedings of the Twenty-eighth Australasian conference on Computer Science-Volume 38, Australian Computer Society, Inc., 2005, pp. 115–124.

[45] D.S.K. Seong, *Usability guidelines for designing mobile learning portals*, Proceedings of the 3rd international conference on Mobile technology, applications & systems, ACM, 2006, p. 25.

[46] M. Sharples, D. Corlett, and O. Westmancott, *The design and implementation of a mobile learning resource*, Personal and Ubiquitous Computing **6** (2002), no. 3, 220–234.

[47] B. Shneiderman, *Designing for fun: how can we design user interfaces to be more fun?*, interactions **11** (2004), no. 5, 48–50.

[48] R.S. Smith, *Guidelines for authors of learning objects.*, New Media Consortium (2004), 32.

[49] C. Snyder, *Paper prototyping: The fast and easy way to design and refine user interfaces*, Morgan Kaufmann Pub, 2003.

[50] IDC Corporate USA, *Android and ios surge to new smartphone os record in second quarter, acording to idc - prus23638712*, http://www.idc.com/getdoc.jsp?containerId=prUS23638712, 13.09.2012.

[51] _____, *Strong demand for smartphones in second quarter continues to derive the worldwide mobile phone market, according to idc - prus23634612*, http://www.idc.com/getdoc.jsp?containerId=prUS23624612, 13.09.2012.