Project report in Experimental System Development 2021

Group: Ikke lige på stående fod

Viktor Salling - 201906430
Mads Overgård - 201906515
Lucas Winther - 201905215
Rasmus Mikkelstrup - 201804618

May 27, 2021

Supervised by Henrik Korsgaard and Susanne Bødker

Department of Computer Science
Aarhus University
Denmark

Abstract

Several students at Aarhus University have exclaimed their dissatisfaction with the current platforms for obtaining information about their studies. These problems were investigated and a design and a working prototype were created to solve them.
A case-study and a Future Workshop were carried out to inform the design. A prototype of the design was implemented. The prototype was evaluated both internally (Streamlined Cognitive Walkthrough) amongst developers and externally (Think-aloud) with two user participants from the university.
During preliminary field studies, problems such as too many platforms and unnecessary features were identified. To alleviate this, a design for a web-based modular portal was created. During evaluation of the prototype, it was discovered to have moderate usability problems and major shortcomings.
It is evident that the current state of prototype does not make it clear whether the design solves the proposed problems. More time need to be spent on both design and implementation.

Contents

1 Introduction **5**

1.1 Problem statement 5

2 Field Study **5**

2.1 Case-study 6

2.1.1 Method 6

2.1.2 Where is it? 6

2.1.3 Too many complicated systems 7

2.2 Future Workshop 7

2.2.1 Method 7

2.2.2 Results 8

2.3 Summary 9

3 System presentation **9**

3.1 The idea 9

3.2 Style 9

3.3 Personalization 10

3.4 Requirements 10

3.4.1 Widget layout 11

3.4.2 Persistency 11

3.4.3 Mobile friendly 11

3.4.4 Design 11

3.5 Development process 12

3.5.1 First sprint 12

3.5.2 Second sprint 12

3.5.3 Last sprint 12

3.6 Prototype 13

4 Evaluation **14**

4.1 Internal Evaluation 15

4.1.1 Method 15

4.1.2 Analysis 16

4.1.3 Summary 17

4.2 External Evaluation 17
4.2.1 Method 17
4.2.2 Analysis 18
4.2.3 Summary 19
4.3 Results 19
4.4 Summary 20

5 Theoretical analysis **20**

6 Reflection **21**

7 Future work **22**

8 Conclusion **22**

9 References **23**

10 Appendix A: Walkthrough notes in danish **24**

11 Appendix B: Scenario in danish **28**

1 Introduction

The report describes the complete project that has been carried out throughout the spring semester of 2021 regarding the development of a prototype website, with the purpose of acting as a portal for the current, multiple sites and systems students at Aarhus University (AU) are using.

The first part of the report describes the field studies that have been completed in order to obtain information from the participants that have been a part of the project, regarding their use of the current platforms and their opinions on these. This served as a starting point for the project and aided with an understanding of the several problems that needed to be adressed.
From the results of these field studies, an idea for a prototype was proposed, which is explained in detail in the system presentation section (3). As well as a general idea, a more concrete design is presented together with arguments for how the system is to be used, namely through a widget-based interface, and why it was designed this way. The presentation of the system also discusses the multiple sprints this project has been through, and here the individual focus for each sprint is documented, revealing the timeline for how the system was developed with any delays and unexpected issues along the way. A final presentation of the developed prototype is found at the end of beforementioned section.

Following the description of the prototype, the report moves on to describe the work that has been done to evaluate the prototype in order to make sure that the learnability and usability of the prototype lived up to the expectations that were set in the beginning of the project, as the problems of the current systems were identified. This evaluation took place as both an internal evaluation among the authors of this report, as well as an external evaluation bringing in participants from outside the team, to test the prototype from multiple angles. This section describes the methods that were used in both evaluations and the results that arose from the evaluations.

As a conclusion to the prototype, the report discusses and analyzes the prototype from a theoretical standpoint and describes how some of the models that has been presented in the course have been used as well as the relevance of our prototype regarding central concepts such as articulation work (Schmidt and Bannon 1992) and activity-theory (Bardram 2009 and Bødker 2014) among others.

The report rounds off with a reflection of the project that discusses other approaches that could have been taken as well as issues that came along and how these could've been dealt with in conjunction with a short section on hypothetical future work that is to be done, were the project to continue.

1.1 Problem statement

The current platforms used by students at Aarhus University consists of multiple, redundant sites with various non-relevant features and poor guidance which confuses the student and adds unnecessary work to obtain the desired information. This can lead to frustration and the conseqence of giving up on finding certain information due to the current complexity. This project aims to develop a platform, through participatory design (Bødker et al. 2021) and relevant theory, that collects the information a student would find relevant, in a single system, giving the student a clear and concise overview that eases the process and removes unnecessary features that does not benefit the user.

2 Field Study

Tasked with creating a prototype of a new platform for students, the goal was to make sure that the problems students are currently facing would be remedied, thus creating a better experience for all. To do this, a case study (Lazar, Feng, and Hochheiser 2017) and a Future Workshop (Kensing and Madsen 1992) was conducted.

In section 2.1, the conducted case studies are described. Here the chosen method for carrying out the studies is explained and argued for, while the most important observations and results are discussed in section 2.3. The goal of these studies were to gain an in-depth understanding of the involved users' issues and experience with the existing tools and portals. These tools that support their studies are what, together with the Future Workshop described below, make out the foundation upon which the prototype was built.

Later on in section 2.2, the Future Workshop is covered. The workshop aimed to provide an establishment for the prototype through feedback and ideas that the participants provided. While the case studies explain the situation and current issues, the workshop provides an attempt at a theoretical solution which was applied in practice to design the first prototype. This also means that the participants were more involved in the Future Workshop as part of a participatory design (Bødker et al. 2021) compared to the case studies that were conducted and managed by the developers.

2.1 Case-study

2.1.1 Method

To provide data for the case study, several interviews were conducted. The interviewees for these were students were all enrolled to some education provided by Aarhus University. This ensured that they all had experience with the portal Blackboard. Nothing was asked of the interviewees in preparation of the interviews since the key thing we were looking for were day-to-day experiences.

The interviews were conducted one-on-one between one of the authors and an interviewee. This interviewee had been acquired for the interview by the member that interviewed them, thus providing a more familiar and safe environment for the interview to be conducted in.

The interviews conducted were semistructured (Lazar, Feng, and Hochheiser 2017), since while a general goal for the case study existed, it was at this point not narrow enough to conduct a fully structured interview with strict questions. The openness of the semistructured interview also allowed the interviewee to puill the interview towards the things they themselves found very interesting or troublesome, while still giving the interviewer a way to keep the interview on track. It also allows for the interviewee to provide more descriptive data and elaborations, which could be crucial in the analysis.

Since the case study covered more than one unique topic, such as how the interviewee conducts collaborative work or how they access course data, it is embedded (ibid.) in nature.

The purpose of the case study was to gain insights in how students use and access different information on and functionality of their study platforms, however whether the interview should be classified as intrinsic or instrumental is debatable. The result of the case study is both information regarding only Aarhus University related platforms, however it could be thought much of this could be generalized into rules that would apply to many digital platforms, as having to deal with multiple platforms each with their own way of presenting information isn't an unnatural experience (another example could be hospitals, see Bardram 2009).

2.1.2 Where is it?

When composing a handin at a higher education such as a university, it is often not as simple as writing an essay and calling it a day. It requires the use of many different systems, creating different parts and working hours upon hours to finish a single handin.

An interviewee told of a situation with a weekly handin from a mathematical course. Handins in this course often required work from previous handins, notes from the lectures from both the current and previous weeks, recorded lectures, personal notes etc. All of these things were to be found on Blackboard, the internal portal for AU (Aarhus University). To acquire all these prerequisites one would have to navigate to the course plan, then find the current week and from there open either the recorded lecture or the slides (the lecturers presentation from said lecture). These were also available from the sidebar under the menu item labeled "Materials (public)" not to be confused with the item "Materials (copyrighted)". Under the "Materials" item, all weeks were listed in a long list, but not always in a sorted order. This was combined with how the layout of the course plan differed from course to course as it depended on the concrete way the lecturer decides to "implement" the structure on Blackboard. Personal notes were to be found in a locally synced folder from OneDrive (or another cloud syncing service).

Combining all the above, it took quite some time to get everything setup to start working on the actual handin. Some courses also required files to be available to it is students and while these could be hosted on the platform Blackboard itself, one interviewee explained that not all lecturers had discovered this. One time these files were

hosted on the file sharing service Dropbox, which yet again leads to another system (and interface) that one needed to become familiar with.

Zooming out a bit, things such as schedules for a students week containing the different courses, exercise classes and study cafés (spaces for qestions regarding handins etc.) were available using a few different methods, but none were really efficient. These might be found under the page of each course, if the lecturer has chosen to put it there. Maybe the student was required to find it on another AU site called MitStudie (translated "MyStudy"), however this information was known to contain errors. Lastly details for each course could be found using yet another portal studerende.au.dk (translated "student.au.dk"). This resulted in the student having to end up compiling their own schedule separately to make sure they were not missing anything.

During the COVID-19 pandemic, some lecturers had opted to use other tools (than Blackboard) as means of communication between themselves and the students. An example of a tool that had become widely popular was Discord. A social standalone platform which provided users the ability to create their own "servers" to which they could invite friends etc. This server provided options for creating text and voice channels. Such channels could have names reflecting their purpose, for example "announcements" and "questions". The only problem as told by an interviewee was that it simply added another information source to stay updated on.

Usually such information was sent out via email or written on Blackboard on the course page itself. Even with only the last mentioned places it was still quite the hassle of keeping up with information posted on all these social channels.

2.1.3 Too many complicated systems

As briefly discussed above, an issue that was mentioned multiple times across the interviewees was that there were too many tools and systems that had to be used, and some of them were too complicated. An example was Blackboard that did not provide a clear structure for where information, exercises and various links (to other systems) should be placed meaning that it would differ widely from course to course, further exacerbated when the lecturer or instructor chose to use a whole other platform e.g. Dropbox or Google Sheets. Another example would be how exercise classes could be carried out on either Blackboard (using their integrated classroom feature) or Zoom or Discord with seemingly no apparent reason for choosing one over the other and with no standard for which platform to use, although Zoom seemed to be the preferred.

This meant that the students had to learn to use multiple different tools with no official guide on how these tools worked. It was simply expected that the student would figure it out by themselves. If you wanted to find information on your course you would go to one system, whereas if you wanted to figure out what you had to read for the next lecture it would be on a second system, checking mail on a third system, attending lectures on a fourth, and so on. This is a good example of a case where technology that should be invisible to aid the user becomes visible and an obstacle because the user constantly has to consider which system they need to go to in order to get the information they need. This central system should also be more intuitive and while that is easier said than done, it was mentioned by an interviewee that while Blackboard did have the support for certain functions, they simply were not used because the people using the service did not know how to. Blackboard also contained a lot of functionality that was not relevant which made it a jungle of functionality the user had to navigate through, resulting in the tendency to dismiss functionality that could actually be beneficial or simply missing it entirely. This suggested that it should be carefully considered exactly which tools the product developed should support in order for all of them to be as relevant as possible, perhaps with the possibility to easily hide tools that will not be used by the individual user.

2.2 Future Workshop

2.2.1 Method

As previously described, we also held a Future Workshop (Kensing and Madsen 1992) to get a better idea of the needs of potential users of the prototype. The Future Workshop was conducted with the same participants described in the case study above. The reason for this comes to the knowledge that most of the participants have experience with the same systems in regards to their daily workflow (since they are students at AU) which leads to a better critique and fantasy phase of the workshop, since the participants can draw upon a lot of the same experiences and software frontends.

While conducting the workshop we acted as guides for the discussion, and controlled the overall structure. To get the most of the workshop, it is key to try and get a discussion going between the participants, observing only as a "fly on the wall", intervening only if it seems there are some people dominating the discussion, or helping with metaphors if the brainstorm is becoming stale.

2.2.2 Results

In contrary to the case studies, the Future Workshop is more focused on the tools that the participants are using, instead of focusing on the participants themselves. Being students of Aarhus University, this includes software like the previously mentioned Blackboard, mitstudie.au.dk and Zoom. We wanted to learn from the mistakes the the previous platforms made, and make an improvement into the new combined prototype platform.

Giving that we at this point had already done the case studies, we had a general idea of some of the problems that the students ocurred. One of the most pertinent things to mention, is that there was a large amount of critique on the layout and presentation of information. A general dissatisfaction with the requirement of multiple different webplatforms, all having their own method of presenting their information and their own purpose, leads to all participants forming the consensus that they have a hard time finding the information they require.

A solution the participants came forth with, was using a single platform that could perform the tasks that many of the systems can right now. There were comparisons to platforms that many of the students used before university, such as "Lectio"[1] or "It's Learning"[2]. Putting the most important parts of each platform they are currently using into a quick and easy to access platform with a more streamlined interface. The primary focus here was on handins, non-handin homework, Zoom links and notifications from Blackboard, the schedule from mitstudie.au.dk and the mail from post.au.dk.

This is combined with an idea of better prioritization when delivering information. A lot of notifications from Blackboard (which also get sent to the users emails) were not very important, and the onslaught of these unimportant information leads to users getting tired out of keeping up with notifications and their email. The ideal platform would in this case allow either more categories of notifications, or to disable some type of notifications entirely, making email and notifications worth checking again. This is also impacted by how applications like Discord, which were previsouly only used for free time for most of the students, now also has a formal use by the student faculty which "pesters" students with studicious notifications after school. This would, in an ideal world, not be happening, and formal messages would be confined to this all-in-one platform.

A thing that was also problematic (but less so) was the horrendous performance and user experience of the webapps the participants experienced while browsing using their phones. Be it on the bus, walking around on campus or laying comfortably on their bed, having the ability to check where they need to go for their next lecture, or if they need to get up and open their computer to redo a handin, having easy access, both as in easy to get to and as easy to read quickly, would be a really helpful improvement for the students.

The layout of a proposed replacement for all of these webapps is also a really important issue. Many of the participants disapproved of the teachers use of Blackboard, describing that it was so customizable that the structure of the different courses often ended up being a maze to navigate. This was a particularly interesting thing to hear, as it promotes having the information students need to be presented more from a plan-perspective than a situated action one (Suchman 2007). Having too much freedom leads to the students performing articulation work (Schmidt and Bannon 1992) when navigating this maze of information. A collected solution would therefore have a much simpler and unified way of presenting information to the students.

For communication during the pandemic, the participants mainly used Discord and Zoom to do their work. Participants compared the applications to each other, explaining how each application had some features that they would love in the other. Discords voice communication felt unpersonal and lacked ways of interaction during conversations, be it in groupwork or lectures. Zoom however lacks a lot of the text capabilities and multi-room structure that Discord excells at, one of which is that the chat of Zoom disappears after a lecture, and messages shown before a participant enters a Zoom meeting does not appear for this participant.

One final thing to note from this workshop, is a feeling of vexation when it comes to discovery of functionality. The participants described how many of the webapps had functionality that, while useful, was way too hard to

[1] https://www.lectio.dk/
[2] https://www.itslearning.com

discover, and that there should be an easier way to discover this functionality.

2.3 Summary

It is clear to see, that a lot of valuable information and design ideas can come from users when they are involved in the design process.

The key problem expressed by the users, found both in the case study and the Future Workshop, was the confusingly made layout and uncollected nature of the information they were presented with. Firstly, there is the problem of finding information asking questions such as "where do I go to find X, do I use Blackboard, discord, mitstudie, ...". There is also the question of how to prioritize between information, as not all information is presented equal, and of equal importance. Missing hierarchy and overarching structure of information leads to a bad user experience for the users, creating the need for uneccesary work finding specific information, or simply just asking someone else to get the information for you (eg. "I cannot find the course slides from monday, could you send it to me?"). Lastly, grouping information and making sure the user is not spammed with the same information over and over is important too, as getting the same announcement on email, as a notification on Blackboard and on its own seperate section on the course page is arguably enough to make the users just turn off the notifications all together.

3 System presentation

To present the system created in this paper, the core aspects of it has been split into three sections. These sections are shown below. A link between the functionality and design choices explained below and the reasoning is not directly established. However it should be clear when reading the following how it connects to the information gained from the preceding case-study (2.1) and Future Workshop (2.2).

3.1 The idea

From the information gained by the case-study and Future Workshop, it was decided to construct a widget-based platform. More precisely, a frontpage consisting of widgets with support for customizability to fit the user's needs. An example of such a design can be found on most smartphone handhelds which comes with support for widgets on the home screen. These widgets come in several variations supporting multiple different views with different focuses and data for the same application. Take, for instance, the Notes app from iOS, which supports both a widget of a single note, but also a widget for a list of all notes. The idea builds on this concept but instead of these variants of a widget that the user can choose themselves, it is the size of the widget that decides what the widget should show. This both provides advantages of a simpler experience for the user, but one might argue that there are disadvantages due to the user not being able to have full control over the widgets. However to accomodate for this possible disadvantage, a settings menu will be present for each widget, to provide options (see Figure 2) for e.g. how to sort lists and whether to show full dates etc.

To get a clearer common understanding of how the imagined prototype is used, we created a scenario which can be read in full at Appendix B (Section 11). This was also used as a basis for creating the user-stories in the development phase of the project.

Having a frontpage consisting of only widgets can be a disadvantage since if one does not have a given widget present, the information this widget would provide will be available. However, the problem has been identified although currently no solution is suggested.

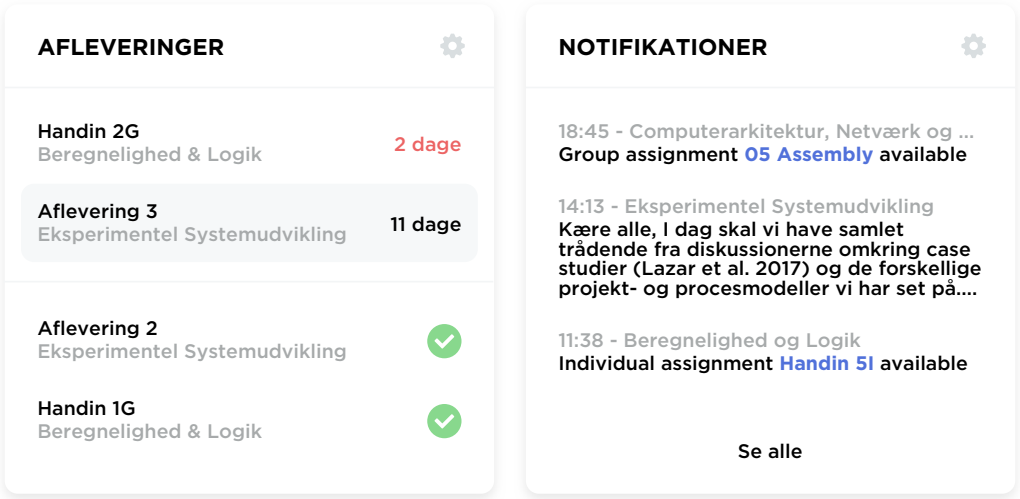
3.2 Style

The design for the platform (see Figure 1) has been chosen based on the simplicity and the aim to provide a non-cluttered interface. The colors are mostly grayscale to avoid interfering with important information, such as due dates, accepted handins, schemas etc. However certain elements have been highlighted using contrast colors, such as neon blue, grass green and faded red. This helps draw the user's attention to elements highlighted

in these colors, such as deadlines, links etc. This also removes the problem of having to manage all the different colors, but at the same time might make it harder to quickly retrieve the information. Color theming might be added at a later point to support seperation of different main aspects of the platform such as the student's different attended courses.

It is also worth noting, that since most of the site is in grayscale, the problem of color blindness is gracefully avoided. In the cases where color is used, as in conveying whether a handin is a re-handin or not, there is also a use of symbols, making the same information available in two different ways. With the previously described color theming, it could also be a good idea to make a color blind color scheme.

The widgets have been given rounded corners to make it more playful and welcoming for the user. The hope is to invite the user to interact with the interface, move things around and customize it to their liking.



(a) Handins-widget (b) Notifications-widget

Figure 1: Widget design

3.3 Personalization

A major selling point of the platform is customization. Blackboard also supports this but somehow none of the widgets that are available are the ones that people actually want. On top of this some widgets are required to be present and thus cannot be removed. On the new platform, this should not be a problem with a much larger widget selection ranging from a simple clock to a fully fledged calendar. Likewise it shall be possible to have no widgets at all, if somehow anyone wants this.

This is also the main reason for choosing a widget based interface. If all the information was fixed in place, like many popular platforms, customizability is not possible or atleast very tricky to figure out and implement. With widgets it becomes intuitive for the user to adjust and move things around, since it resembles most mobile operating systems whether it is apps or widgets.

To allow for maximum customization it is essential that each (large) widget can be resized. A classic problem with many widget-based interfaces is that one large widget, such as a calendar, takes up practially the entire screen. This prevents the user from having said calendar data available along with other information. To accomodate for this, each large widget must be resizable such that, if wanted, it can be shown fully for maximum information, while also making it possible to shrink it to a much smaller more practical size, while not missing the (albeit less) information the widget provides.

3.4 Requirements

To make sure the focus was kept on track during the development process, specific requirements valued to be the most important for the platform was described. These are concrete descriptions of key aspects of the

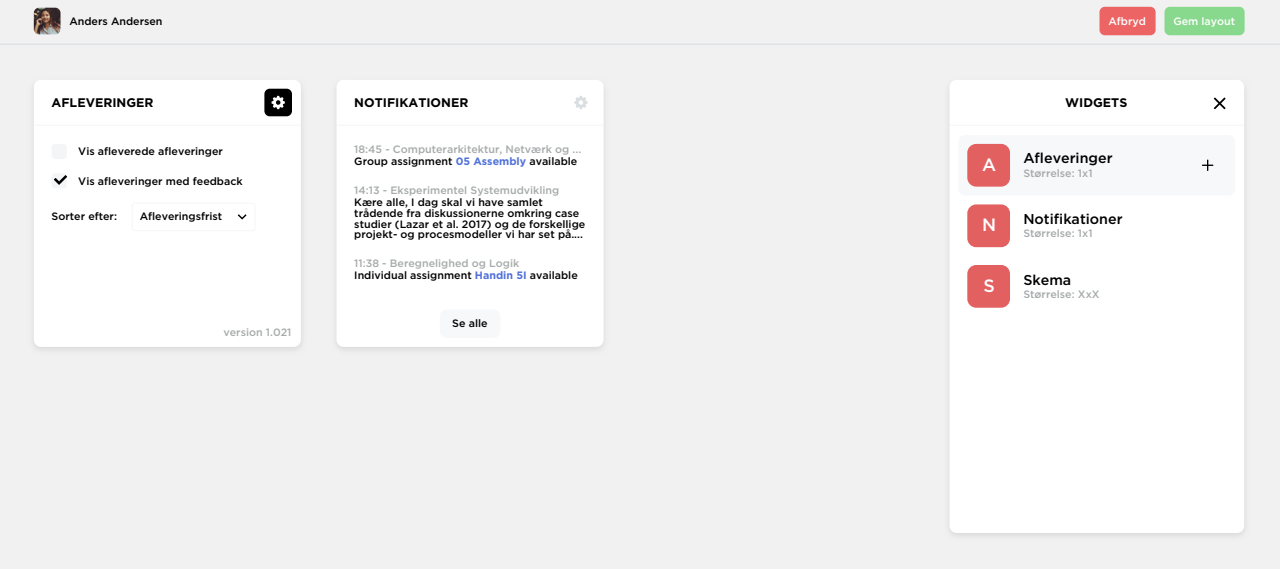


Figure 2: The platform with a few widgets and the widget drawer open.

platform that are a direct requirement for the final product. Even though only a prototype was developed, it was important to keep these requirements in mind.

3.4.1 Widget layout

The platform shall revolve around a widget-based layout. Widgets should be movable and resizeable with the possibility to remove and add them to the users liking. A list of all widgets shall be available in a drawer-like menu where it will be possible to add more widgets.

The widgets available shall be atomic versions of much of the data available on the Blackboard platform. Hence it shall be possible to only add what is actually needed without being required to add bloat.

Each widget may come with settings, that allows for further customization. These settings shall be linked to the widget instance. Furthermore these settings shall be understandable and easy to use.

3.4.2 Persistency

Since the widgets are movable amongst other things, the placement, size etc. shall be stored for the user. Thus upon opening the platform on a new device or browser the same layout shall be visible. Likewise for settings shall these be saved for the user.

3.4.3 Mobile friendly

The platform shall be viewable on a mobile device. Likewise on a wide range of device-sizes, such to allow the user to use the platform on any device available.

3.4.4 Design

The whole platform shall be in a minimalistic design, that focuses on actual content in regards to the user's study programme and not layout content that explains how to use the layout and contains no actual value to the end user. Meaning that things such as borders and shadows shall not be dominant, but rather be as discrete as possible to not draw attention. On the other hand, things of importance such as deadlines, titles, timestamps etc. shall be highlighted and immediately available to the user's attention.

3.5 Development process

To tackle the development process of the working prototype, the development framework Scrum (Beyer 2010) was loosely used. Using this with the time given for the project, it was possible to create three sprints each with a unique focus area as listed below:

1. Layout and design
2. Essential widgets
3. Bug fixing, further development of existing widgets and creation of new ones

3.5.1 First sprint

In this sprint, the focus was primarily on getting the layout and functionality of the dashboard to a level that was presentable, focusing on things like adding and removing widgets, and creating a framework so developing new widgets would be painless. In combination to this, the basic design style was also implemented.

In the end, the adding and removing functionality was not implemented in this sprint due to time constraints and was moved to second sprint.

3.5.2 Second sprint

Most of the first sprint was implemented and working at this point. However fixes and adjustments to the previous were be remedied here. Furthermore the most essential of widgets such as the assignments, calender / upcoming and discussion board, were planned to be added and made functional with correct styling according to the design.

During production, the focus of this sprint diverged a bit from the original plan to instead focus more on customizability and persistence. Adding and removing widgets, some fleshing out of the existing widgets in addition to changes to styling were implemented as planned. In addition, settings functionality for widgets, saving of layout and and a 'starting soon'-widget were added. The aim of these changes were to make the page be, or at least appear, more persistent and dynamic.

3.5.3 Last sprint

The original plan for this sprint was that the most important parts of the system should have been mostly done by this sprint. All essential widgets should have been implemented and working. Bug fixing have commenced, since the project had some major bugs. In addition to this, a few more of the less advanced widgets should have been added to add to the whole of the system. Minor adjustments to the styling of elements might should have been carried out to match that of the designed system perfectly.

The plan mostly succeded, and most of the essential functionality was done by the beginning of the last sprint. To accommodate the evaluation that was to be carried out by the end of this sprint, some major usability problems identified during development were fixed. This includes more consistent behavior when showing the widget drawer, consolidating the edit toggle and utility buttons into a top bar and fixing the height of grid (prior, the grid was always a bit to tall, allowing for a tiny amount of scrolling).

In addition, the system was styled to be more consistent with the conceptual design, increasing the clarity of the interface.

Common for all these foci was that the required design should have been ready for the given sprint, thus implying that is had to have been created in the previous sprint. In practice, it was not feasable to finish designs for all features before implementation since some added functionality was largely unplanned until the start of that sprint.

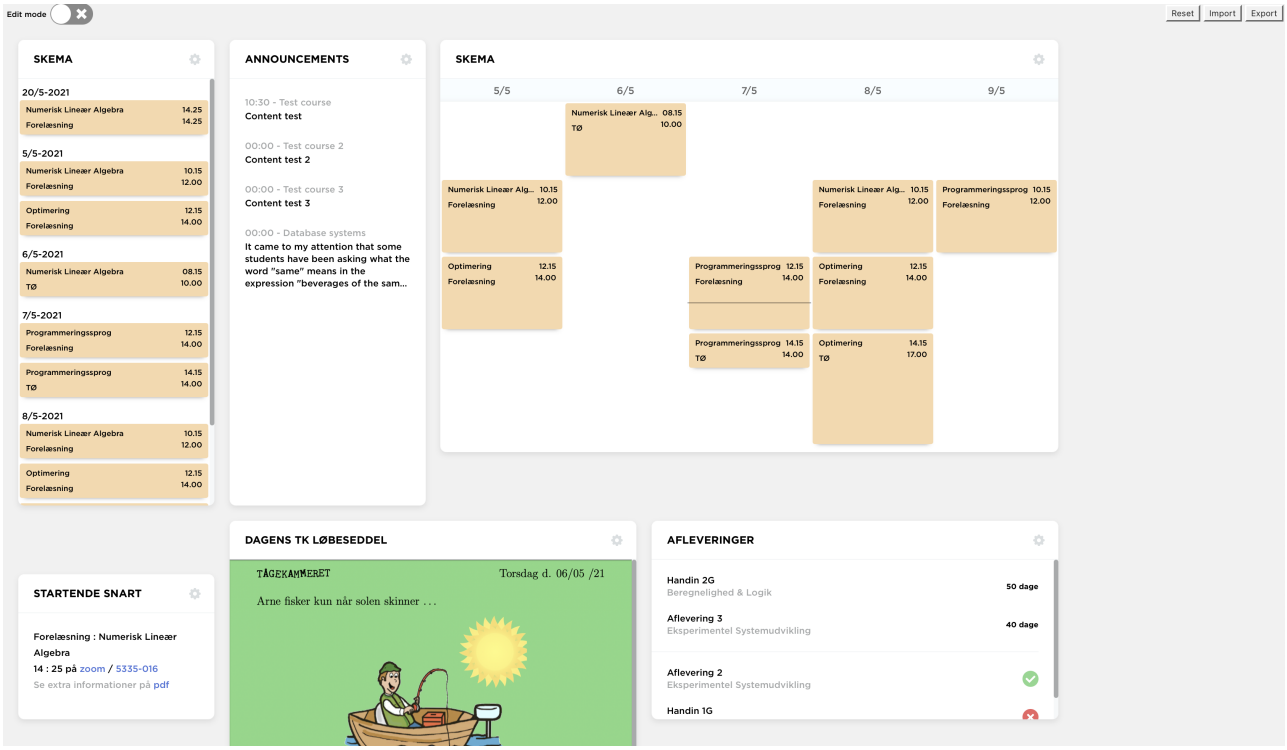


Figure 3: The current state of the platform

3.6 Prototype

The prototype consists of several working components. Each of these components have been described previously in the system presentation chapter (3). These include

- Widgets as a feature
- Settings per widget
- A layout system with resizability
- Saving and loading of widget layouts
- Adding and removal of widgets

Widgets as a feature

Several widgets are currently present on the platform, while none are ready to be marked as completely done. The handins-widget is the most complete widget as it has been fully designed and that design fully implemented, albeit far from ready to production. All widgets conform to the same super-widget [3], which ensures that all widgets inherit the same properties, such as resizing, settings and design.

Currently there exist 5 widgets:

- Handins
- Schedule
- Announcements
- Starting soon
- TK flyers

It was originally planned to have more than these ready for the current stage of the prototype, however time limitations amongst other things prevented this.

Settings per widget

[3]The concept of individual elements inheriting structure and functionality from a superior, more abstract element, used to ensure consistency and compatibility.

Each widget presents a settings menu via a small button in the header, which toggles the settings menu. This menu contains settings related to the given widget, such as ordering of list items, visibility of certain elements etc. However these settings have yet to be implemented for all widgets as they have to be implemented on a per widget basis. The only widget with working settings is the handins-widget, but only some of the settings actually do something and contains a few bugs regarding browser-specific issues.

A layout system with resizability

Widgets can be moved around while snapping to a predefined grid, which allows for flexibility but not too much freedom. If a widget is moved to overlap with another widget, the non-moving widget is moved out of the way to make space for the moving widget. However this system currently has some unexpected behavior as the non-moving widget will always move downwards until there is space available. But doing this, the page will scroll downwards continue happening. It is possible to resize each widget, but there is currently no rules in regards to what widgets can conform to what sizes. An example: the handins-widget, or other list-based widgets, can become very wide, which makes no sense as no more content is shown. Other widgets like the calendar-widget changes view-mode depending on the size, as shown in the Figure 3.

Saving and loading of widget layouts

Functionality to export and import layout configurations is present in the prototype. The design, however, is practically non-existing as it was implemented without having been designed prior. It is accessed via the buttons in the top right corner. Clicking "export" simply shows a default browser alert with only a JSON object. This is then used in conjuction with the "import" button which shows a popup with an input field. The "reset" button simply resets the layout to the default layout.

Adding and removal of widgets

Widgets can be added and removed using the "edit mode" found in the top left. Clicking this does a few things. Firstly, every widget now show a cross that can remove said widgets when clicked. A drawer becomes visible on the right, and when hovering over the drawer it slides open to the left. The drawer contains a list of all available widgets. It is possible to add widgets via this list. However adding a widget inserts it a the top leftmost corner of the layout, moving all overlapping widgets away, sometimes resulting in unexpected visuals.

A focus area was selected midway into the development process as to channel the focus of the prototype towards a specific area. The selected area was required to be a component, but as the platform contained many smaller widgets, which contained not much potential for longer development, it was decided to focus on the concept of being able to rearrange, add and remove and customize the widgets available on the platform.

In regards to data, it was decided not to include any real information, as this would require an external database and filling this with scraped information. Instead dummy data (fake data) was written in a simple data format, for making the platform more real. By using this approach the structure of the system became much smaller, as only a frontend was needed.

Many of the components currently visible on the platform have been directly developed without a preceding design process. Thus many components might not conform to the style described earlier, but rather visually unstyled (default browser styling).

4 Evaluation

In this section, the evaluation that has been carried out to get a better understanding of the usability of the final version of the prototype, developed throughout this course, is discussed and analyzed. The evaluation is made up of two parts: the first is the internal evaluation that the authors of this report have carried out. For this evaluation, the Streamlined Cognitive Walkthrough (Spencer 2000) method was chosen which is described in the following section. The second evaluation is an external evaluation where test participants were invited to evaluate the system. Here the Think-aloud (Nørgaard and Hornbæk 2006) protocol was chosen with two participants, one of them having participated in the earlier user involvement phases in this project, namely the Future Workshop and case study.

At last the results of the aforementioned evaluations are discussed as well as how these results could provide

guidance in creating a plan for what should be achieved in the future, were the development of the project to continue.

4.1 Internal Evaluation

4.1.1 Method

To evaluate the system internally, ie. among the authors of this report of the system, the Streamlined Cognitive Walkthrough (SCW) (Spencer 2000) was chosen. This method builds upon the Cognitive Walkthrough 3 (CW3) (Mahatody, Sagar, and Kolski 2010) method, but with a few modifications to streamline and simplify the process. The original CW3 method is a way to evaluate the learnability of a user interface in a cheap and simple way. The method involves the participants of the evaluation to carry out a set of tasks in the system. Each task should be analyzed with the team asking the following questions (ibid.):

1. Will the user try to achieve the right effect?
2. Will the user notice the correct action is available?
3. Will the user associate the correct action with the effect that user is trying to achieve?
4. If the correct action is performed, will the user see that progress is being made toward the solution of the task?

The SCW method attempts to solve some of the issues that arise when using the CW method. The first problem is time pressure: teams seem to find the process too slow with redundant questions and unnecessary dwelling on the questions, causing participants to feel they're not spending their time in a productive manner. The second problem is the inclination to spend time on discussing design and fix design problems *during* the CW process, and at last the third problem, continuing from the second problem, is that participants who have created the UI are likely to defend their interface and argue against critics of it (Spencer 2000). This is quite counter productive considering the purpose of the process is to identify UI learnability issues and correct them.

SCW solves these issues by laying out some ground rules and streamlining the questions that are asked during the task analysis. The four ground rules of SCW are (ibid.):

1. No designing
2. No defending a design
3. No debating cognitive theory
4. The usability specialist is the leader of the session

These ground rules aim to deal with the second and third problem where teams tend to design and defend said designs during the walkthrough. The first problem, that the sessions take too long with redundant questions and unncesessary dwelling, is solved by changing the questions that the team asks itself during the walkthrough. The first three questions are reduced to the following question (ibid.): "Will the user know what to do at this step"?
The fourth question is reformulated to (ibid.): "If the user does the right thing, will they know that they did the right thing, and are making progress towards their goal?".

This simplification of the questions removes the redundancy that could occur using the previous questions and generally speeds things up, especially for the tasks where the UI is working as intended. The questions are arguably also easier to understand which makes it eaiser for non-HCI experts to apply this method to their own case. This is one of the primary reasons for the selection of this method.

A total of 5 scenarios were created in preparation for the internal evaluation. They did not cover the application's functionality completely, but tested most of the main functionality including some known problem areas. Each task had between 3 and 6 subtasks/subgoals describing the action sequence of the given task.

The internal evaluation was carried out with all four authors of this report present in the time span of about an hour. The overall process during the evaluation followed the order described in Table 3 of (ibid.).

4.1.2 Analysis

A document was created during the SCW session which can be found in Appendix A (10).

The 'Identification of users' step was partially skipped after an informal discussion to confirm the shared view on who the users were: "Danish university students with a varying level of technical skills". The "Description or implementation of interface" was also partially skipped after briefing the authors of this report about the latest changes.

The goals of the walkthrough were determined to be to "find usability problems in the interface". What was to be done were agreed upon to be "finding usability problems and proposing simple solutions". What was not to be done during the walkthrough was "(thoroughly) designing solutions". The participants were briefed in points 2.d, 2.e and 2.g to ensure agreement and reduce the possibility of breakdowns (Suchman 2007) in the process. During point 2.f one participant was chosen to be sessions leader and another was chosen to note down the problems found.

As the SCW process prescribes, only failure stories were written down. These failure stories can be found in Appendix A (marked with red). The first 3 scenarios were "walked through" before it was decided that sufficient material had been collected, leaving 2 scenarios untouched. Care was taken to enforce the ground rules outlined in SCW, especially to prevent early designing of solutions. This was found to be easily achievable after a small adjustment period with some explicit requests to follow the ground rules.

Most subtasks were found to have at least one significant failure story and a total of 18 failure stories were found throughout the 3 scenarios. Examples include:

- 1.3 - "Remove components": The user cannot find the delete button since the button has become very small (because of some layouting).
- 2.1 - "Reset layout": The user could not identify a way to reset the layout, since the label "Reset" does not indicate what will be reset by pressing it.
- 3.3 - "Insert the import string": The user could not identify what kind of input the popup needs.

The failure stories noted down were primarily caused by low visibility/feedback and both the concrete layout and the layouting problems of the users. The layouting problems were found to be particularly serious. When moving a widget over other widgets, the other widgets would become permanently displaced. This could easily ruin a user's layout.

Similar tasks shared similar failure stories. An example of this is the buttons "Reset", "Import" and "Export", which all shared vague titles and no visibility when the page was scrolled. When duplicate subtasks were present across scenarios, for example the "Go into edit mode" task, the failure stories were found to be the same or similar in all cases.

During the 4.a stage (Critical information - Possible learnability problems), 4 points with subpoints were noted down:

1. Missing information on buttons
   (a) Vague titles
   (b) Visibility
      i. Not visible when page is scrolled
      ii. Too small in edge cases
2. Layout problems
   (a) Widgets push eachother around
3. Feedback to user

To remedy these problem points, some design ideas were created:

1. Label for the button group on the right (fixing 1.a)
2. A less destructive layout system (fixing 2.a)
3. Contextual popups/modals to indicate success/failure of actions (fixing 3.)

The following design gaps were also identified:

1. Menu bar at top not fixed to view (Solution: Fix the menu bar to view, fixing 1.b.i)
2. Button sizes had no lower bound (Solution: Give the buttons a static size, fixing 1.b.ii)

These design ideas and solutions to design gaps fix all of the found problems and appear to be easy to implement. The participants agree that the SCW had been very fruitful in that it allowed the to quickly find some major and minor problem points with the usabilty of the application.

The focus on problem points caused the participants to largely gloss over the subtasks with clear and credible stories. This allowed the participants to focus on the areas that needed to be fixed, but as a consequence, it was not a focus to make clear and note down why something worked or what worked particularly well. This might cause the proposed design ideas to be mostly fixes of what did not work and not as much design improvements to really make a feature shine. In the end, the SCW method seems to be an efficient and worthwhile evaluation method to quickly find and propose solutions to problem points.

4.1.3 Summary

The internal evaluation, which took place as a Streamlined Cognitive Walkthrough involving all authors of this report of the project, has shed some light upon some of the issues that can be found in the prototype. Some of these issues were not too surprising given the time that has been invested in the programming of the prototype, but they were still issues that had not been noticed before this evaluation.

The problems were generally concerned with layout issues on the page itself as well as lack of communication with the user, eg. some features such as exporting and importing layout had some implicit knowledge in them, expecting the user to know what it is they're importing, and how they would carry it out, with little information on how they worked. Design solutions to fix these issues were proposed after the evaluation had been carried out. The SCW method was useful to evaluate the prototype, and especially the streamlining of the task analysis questions seemed helpful, making the proces simpler, giving the team time to focus on identifying issues rather than discussing how the questions should be interpreted.

How these results could aid in laying down a plan for future improvements is discussed later in the assignment.

4.2 External Evaluation

4.2.1 Method

For the external evaluation, the Think-aloud protocol (Nørgaard and Hornbæk 2006) was applied. This protocol was used with two different participants both enrolled in a bachelors degree at Aarhus University. The protocol was applied by the standard definitions: Multiple scenarios were presented to the participant, for them to act out using the platform provided. As the participant went along they were instructed to "Think-aloud" such that they would speak out loud whatever was on their mind. This being questions, frustrations, appreciations etc. These thoughts were then noted down by the evaluation conductor for later analysis (as presented in the next section). Using this protocol it is possible to gain insights of how a user traverses the platform in question and how easy the platform is to learn and use.

The same scenarios was used on both participants and were as follows:

1. Add the announcement-widget to the dashboard.
2. Get the starting soon-widget to be on the top left of the dashboard.

3. Change the settings of the handins-widget to not show completed handins.

As the platform does not present itself with many different scenarios possible, due to it being a dashboard-style platform, a semi-structured interview (Lazar, Feng, and Hochheiser 2017) was also be conducted after the Think-aloud session. This made it possible to gain more information on how the participants viewed the platform in question. The interview was planned to be a short one, as it was meant to act as a supplement to the Think-aloud protocol. Hence only a few questions were prepared and asked to the participant to spark conversation. These questions mainly revolved around if and how the problems presented by the participants from the pre-development phase had been successfully remidiated. The goal of the interview was to gain knowledge regarding if these problems had been resolved.

The participants for this evaluation consisted of two students, both enrolled in the Bachelor's Programme in Computer Science at Aarhus University. One of the participants were part of the initial case study conducted before the development phase began. Important to note is that both participants were used to work with computers and are thus expected to "perform" better, regarding perception of the platform.

4.2.2 Analysis

4.2.2.1 Think-aloud

The user evaluation using the think aloud process was somewhat of a mixed bag of feedback. The participant who was previously in the case study and Future Workshop seemed to glance over the user interface, and navigate around the described scenarios with ease. The other participant, who did not participate in the earlier ideating of the new platform, had more difficulties navigating the layouts/interface of the dashboard. This might indicate that the group of participants chosen in the earlier parts of the participatory design process (Bødker et al. 2021) were not chosen too broadly based on their methods of study, but more on their availability, which increased the difficulty of recruitment. In the words of the PACT model (Benyon 2014), the focus have been on the activies, contexts and technologies of the potential people using the platform without looking deeper at the people themselves. This new user had, for instance, expressed that while platforms like Blackboard had a lot of useless information, the participant did not find it that hard to navigate, and would prefer if the platform functioned as an improved dashboard, but then had links and references to other sites like Blackboard and STADS, becoming more of a helper tool than an all-out replacement for Blackboard.

Concentrating more on the Think-aloud experiment itself for the participant, who showed greater struggle with the platform (the one who was not a part of the initial design process), the experiment yielded a greater understanding of how to support the dashboard layout as described in the focus area. It seemed that generally, after doing the first scenario of adding a widget, the rest of the dashboard proved to be very self explanatory. The greatest difficulty encountered was the participant missing the edit-mode switch button at the top left of the application, and instead opting to use the import button in the top right, writing the name "Announcements" in the import modal, and clicking import. This import functionality is meant to import layouts that can be exported from other students, but by the lack of feedback and explanation of the feature, the users' mental model (Craik 1943) did not line up with the design of the application. This could also have something to do with control buttons, like minimize, fullscreen and close often being in the top right corner. Once in edit mode however, the other tasks of the Think-aloud flew by easily. A solution to the encountered problem was proposed and discussed both with the participant and the interviewer. The solution was making the "confusing/unlabeled" layout-related import/export and reset buttons into the edit mode menu, and switching the position of the buttons and the edit-mode toggle.

The new participant also noted a few other minor issues and things they werent fond of, such as how new widgets are added on a weird place on the dashboard, and how he said it was not intuitive that widgets could be moved and resized, and that the starting layout of the dashboard implied that there was some empty space on the right of the screen that could not be used, as this was a wrong assumption.

4.2.2.2 Interview

In regards to the initial problems disclosed from the users in the ideating session at the beginning of the project, some of these are still present. The problem of being able to find the correct data seems to be partly solved, as the currently implemented widgets allow for a quick overview of a vast amount of data without having to engage

with the platform. But despite that, since there are a lot of widgets that has yet to be implemented, some data is not yet available, thus making this data impossible to find, meaning we can not reach information parity with the current platforms (our dashboard can not replace the other platforms yet). As stated it is possible to solve by adding more widgets to accomodate for these specific types of data. The same story repeats itself for several of these initial problems, such as "Having to access many different services to find the right information".

As for the problem regarding the learning curve of previous system and/or multiple systems, this seems to currently be vastly improved. Understandable icons, buttons and easy to understand widget-system. However, this could be likely to change, as when more features gets implemented it could make the platform more complex.

It was pointed out that given information was sparse, such as for the handin-widget, where only a title, date, and course name was given. It was proposed that more information should be available upon clicking on a given handin. This idea has already been discussed/planned for the future in the shape of a complete overlay with full information and actions regarding the item clicked, for almost every clickable item in the different widgets. However since this feature is not included in the main focus for this project, and Blackboard does not provide the data easily, it was never designed nor implemented for the prototype.

4.2.3 Summary

The selection of participants for all external evaluation and participation has shown to be rather important, as some things seem trivial to certain people while others find them confusing. Participants were selected mostly in the same field as the authors of this report, which might have been hindering in picking up on basic miscommunications of design elements, likely causing breakdowns (Suchman 2007).

The decision to use the Think-aloud protocol (Nørgaard and Hornbæk 2006) might not have been the best decision as the platform was not really the ideal application of use, due to the focus area selected earlier in the process, since the current state of the platform does not allow for very different scenarios and some scenarios included unfinished elements, essentially predicting the outcome of the protocol (ibid.) before it even started.

During the interiews it became clear that the platform was far from finished. While the current state proved to be a step in the right direction regarding usability and learning curve, many of the expected features among participants had yet to see the light of day. The lack of "essential" widgets was pointed out as being a major problem, which was to be expected as the platform indeed far from it's finished state.

4.3 Results

From the results of the internal and external evaluation, it is evident that there are some key points of improvements that could be implemented.

There are evidently some issues that have come up both during the internal and external evaluation, which concerns usability and learnability of the prototype platform. There are some clear solutions to these problems, which help fix the platform's leaky pipe (*Bad usability is like a leaky pipe.* 2006), and would help most of the users to navigate the page with ease. These are things like adding more feedback to users' actions, such as when trying to import a layout that is invalid or adding new widgets, as well as directing their attention better towards the features that has proved to be important in design workshops and ideating sessions such as the deletion of information the users find unnecessecary and resizing widgets to create an easy to read hierarchical structure the users can understand.

It is also apparent that some of the problems discovered are an effect of the protoype nature of the project. For example the fact that there is a need for more widgets before we reach information and function parity with the current systems that the platform is aiming to replace. Functions like the group feature on Blackboard, exam and grading feedback, registration for exams and courses and more. While these functions are important to note, it is arguably not evaluating the core idea and already developed part of the platform, and more describing problems with the fact that we are evaluating on a prototype.

4.4 Summary

An internal evaluation has been executed using the Streamlined Cognitive Walkthrough method (Spencer 2000), from which it has become clear that many smaller problems are present in the platform.

Furthermore an external evaluation has been carried out using the Think-aloud protocol (Nørgaard and Hornbæk 2006), which has shown that for individuals with above average technical knowledge, the platform is intuitive and easy to use. However for the less technically experienced people the platform contains many ambigous elements which has proven to create confusion. A reason for this is perhaps due to the current non-complete state of the platform.

As the state of the platform was evident to the authors of this report, it was expected that many of the already known problems were to be brought up in both evaluations. Still, the evaluations provided useful insight into the usability of the platform, and the ideas for improvement brought up by the two evaluations act as a good starting point for fixing the problems identified.

5 Theoretical analysis

Prototyping

When looking at the final state of the platform that was the prototype, it makes sense to consider the Prototyping Theory (Lim, Stolterman, and Tenenberg 2008).

In regards to filtering dimensions, the prototype was very focused on the spatial structure, functionality and appearance aspects.

The appearance dimension was prioritized not necessarily to evaluate it directly, but to make sure that it did not distract the end user during evaluation. Since the prototype is manifested as a website (which is very visual in nature), it was not possible to filter out appearance. The thought was that a visually developed interface allowed the users to focus more on the functionality of the page, rather than a unfinished design.

The functionality and spatial structure dimensions were at the core of the focus of the prototype. The main priority was to test the layouting system, including adding and removing widgets. This required major work on the functionality and spatial structure of the prototype, including adding a drawer menu and integrating a grid system (see system presentation 3). Still, the functionality of the individual widgets was not prioritized.

Considering the manifestation, the scope of the dashboard was close to what was originally planned (see system requirements 3.4), as most of the core widget functionality was present. However, even though the core layouting functionality works, there was a lack in the total number of widgets, as well as lack in the functionality of each of the widgets, that would be required to produce the full user experience that was designed. This reduced scope was, as previously described, due to time constraints during the project. Furthermore the reduced scope could have influence on the feedback from the user evaluation, which has been an effort to keep in mind.

The resolution of the prototype was high, since a design was ready from the beginning. The prototype was built with the intention of being a presentation prototype/prototype proper, and as such, the system was presented as highly realistic. Internally, the prototype used dummy data and had no connection with any backend for features such as user login, data persistency or proper reflection of the users' real world school data. This lowered the overall resolution, though since the focus was on testing how the core functionality worked, this was deemed to be the correct choice for the prototype.

It might be a valid consideration to explore the design space with different filtering dimensions, if the prototype was to be redone. Creating simpler manifestations which could filter the different design qualities, could in this case perhaps have given way to a better end product (see Fundamental Prototyping Principle, ibid.).

Activity-Theory

It is also interesting to consider activity-theory when looking at the constructed prototype in a more theoretical and analytical way. If one were to look at Bødkers Triangle model (Trekantsmodel) (Bødker 2014) with the given scenario with the Object (Arbejdsgenstand) being attending a Zoom lecture, the developed platform takes the role of the Worksharing Technology (Arbejdsdelingsteknologi). If the Community (Fælleskab) contains students

as well as lecturers amongst other administrative personnel, the mediation between said community and the object becomes the information regarding how to attend the lecture in question. Since the platform is intended to either work along or replace Blackboard, it will be used as a place to publish and gather information about upcoming lectures. This facilitates the ability for students to gain knowledge of e.g. the link for the Zoom lecture along with the time and date. It can be argued that since the Zoom link is required to attend the lecture, our platform also takes the role of a Tool (Redskab), while not being the primary tool, that is Zoom.

One could make multiple similar analyses where the platform acts as a Worksharing Technology for an external Object. It is also interesting to look at the situations created by the platform itself. Consider the case where the User (Bruger) wants to change the layout of their dashboard to a specific one, i.e. the layout is the Object. In this case, the platform acts both as a Tool to change the layout directly, but also as a Cooperation Technology (Samarbejdsteknologi), since the User and the Community can share layouts with each other.

Articulation work

A key element of information platforms such as the one described in this report is the extra Articulation Work (Schmidt and Bannon 1992) that students often do to acess information in different scenarios as mentioned previously. There are different ways to imagine inputting new data and information into the prototype platform. Since the platform in question will either act as a relay system that fetches information from the underlying platform, Blackboard, or simply a replacement for Blackboard, the instructors and lecturers will probably have to do the same amount of articulation work. It could be interesting to delve deeper into this side of the platform, but because of the lack of current experience, it was decided to focus on the students.

The platform has in that way focused on lowering the Articulation Work (ibid.) of the students, trying not to become superfluous to the students to a point, where theyd rather just use text chats to communicate the same information again, that is described in the platform. This will be caused by breakdowns (Suchman 2007) both in more functional ways ("the website is down, can someone give me the zoom link") but also in more core ways ("I cannot seem to find the zoom link on the website, Ill just scroll up in our group chat..."). This is one of the core problems the prototype platform tried to remedy, but it has to be kept in mind in the future that this core functional breakdown does not reappear to a bigger extent.

6 Reflection

What direction would the platform have taken if the focus area chosen (see prototype 3.6)? It was chosen to focus on the concept of being able to customize the frontpage by moving, adding or removing widgets. This focus made it possible to tackle more difficult development aspects as well as keeping it rather wide, to allow for small changes in focus direction. Had the decision fallen on a single widget, firstly the development process would have come to a stop quite abruptly as a single widget can only contain so much functionality, before becoming a platform of itself. Furthermore the main motivation for the whole project was that of customization and allowing the user to filter information, which would have died off rather quickly had the focus been set on a single widget instead of the broader concept.

Since the participants in the different external evaluations have all been enrolled at Aarhus University, it is hard to say much about how the current and future state of the platform would be received. The received feedback from evaluations have been that of AU (Aarhus University) students and with the authors themselves being AU students the platform as a whole might be biased towards solving university specific platform problems, instead of broader ones pertained to such platforms in general.

Albeit only one university has been part of this report, having multiple participating unversities might increase the complexity and thus difficulty of said project. One might say that having only a single participant have helped keep the focus relatable and motivation high.

An external evaluation was only conducted once, and as can be read in the Theoretical Analysis 5 the appearance of the prototype was essentially ignored as a dimension, thus not receiving appropriate feedback on this. It would perhaps be beneficial to have executed multiple external evaluations along the way to gain more insigt into reality of our prototype and thus included the end user in the design process along the way (Bødker et al. 2021). Having done this, a more fitting design might have been achieved, as users would've been more active in the design of the UI, but at the cost of having to spend more time executing these, thus stealing time that could have been spent on the development of the platform.

When the authors wrote this report, it was noticed that there were very few elements relating to CSCW, expecially the collaborative technologies described in (Schmidt and Bannon 1992). It could be interesting to explore whether or not it would be beneficial to create features directly based on the models and terms used in CSCW. On one hand, it is easy to see that the product is not devoid of collaboration; Articulation work is at the heart of the platform. Whether or not collaborative features would be helpful very much depends on how general the platform tries to be. A Messenger-like IM system would seemingly be out of place. One should at least carefully consider whether or not a feature is added simply for the sake of it.

7 Future work

Looking forward, the first step is to implement the suggested solutions to the unearthed problems from the internal evaluation (4.1.2). These problems, albeit individually small, together create a very unfulfilling user experience (Hassenzahl 2010), which leads to a loss of users.

Many of the features presented in the current state of the prototype (3.6) are yet to be marked as complete and would thus also be of priority to be finalized.

Furthermore, spending more time in the design process of the platform so it becomes more thought through and perhaps tested, to improve usability. Many more widgets are to be added to the platform, such as to add more of the features and information from the original platform, Blackboard, and other university subsites.

8 Conclusion

A platform that acts as a portal, for conveying relevant information for students enrolled at Aarhus University, has been developed. A case-study and Future Workshop were carried out to identify problems. These include multiple different platforms, too many unnecessary features and issue-prone platforms. A design for a web-based modular portal was sketched, aimed to solve the problems mentioned above. A prototype was developed, which was evaluated both internally and externally to uncover problems in the design and implementation. This revealed several bugs and usability problems. Furthermore it was discovered that the current state of the prototype had yet to solve the main goals as it was in too early a state. While the platform has yet to reach it's fullest potential it touches many ideas that differs from the current platforms the students use. As the available time had run out, no subsequent changes were made to the prototype. Future work, including usability improvements, more widgets and an actual backend integration, would be required to achieve the goal of a platform that solves the stated problems.

9 References

Bad usability is like a leaky pipe. (Nov. 2006). en. URL: https://www.90percentofeverything.com/2006/11/13/bad-usability-is-like-a-leaky-pipe/ (visited on 05/13/2021).

Bardram, Jakob E. (June 2009). "Activity-Based Computing for Medical Work in Hospitals". In: *ACM Trans. Comput.-Hum. Interact.* 16.2. ISSN: 1073-0516. DOI: 10.1145/1534903.1534907. URL: https://doi.org/10.1145/1534903.1534907.

Benyon, David (2014). "Designing interactive systems: A comprehensive guide to HCI, UX and interaction design". In:

Beyer, Hugh (2010). "User-centered agile methods". In: *Synthesis lectures on human-centered informatics* 3.1, pp. 1–71.

Bødker, Susanne (2014). *IT-systemer og menneskelig aktivitet.* Institutfor Datalogi, Aarhus Universitet.

Bødker, Susanne et al. (2021). *Participatory Design.* Forthcoming.

Craik, Kenneth J. W. (1943). *The Nature of Explanation.* Cambridge University Press.

Hassenzahl, Marc (2010). "Experience design: Technology for all the right reasons". In: *Synthesis lectures on human-centered informatics* 3.1, pp. 1–95.

Kensing, Finn and Kim Halskov Madsen (1992). "Generating visions: Future workshops and metaphorical design". In: *Design at work: Cooperative design of computer systems*, pp. 155–168.

Lazar, Jonathan, Jinjuan Heidi Feng, and Harry Hochheiser (2017). *Research methods in human-computer interaction.* Morgan Kaufmann.

Lim, Youn-Kyung, Erik Stolterman, and Josh Tenenberg (2008). "The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas". In: *ACM Transactions on Computer-Human Interaction (TOCHI)* 15.2, pp. 1–27.

Mahatody, Thomas, Mouldi Sagar, and Christophe Kolski (2010). "State of the art on the cognitive walkthrough method, its variants and evolutions". In: *Intl. Journal of Human–Computer Interaction* 26.8, pp. 741–785.

Nørgaard, Mie and Kasper Hornbæk (2006). "What do usability evaluators do in practice? An explorative study of think-aloud testing". In: *Proceedings of the 6th conference on Designing Interactive systems*, pp. 209–218.

Schmidt, Kjeld and Liam Bannon (1992). "Taking CSCW seriously". In: *Computer Supported Cooperative Work (CSCW)* 1.1, pp. 7–40.

Spencer, Rick (2000). "The Streamlined Cognitive Walkthrough method, working around social constraints encountered in a software development company". In: *ACM Special Interest Group on Computer-Human Interaction.* ACM Press, pp. 353–359.

Suchman, Lucy A. (2007). *Human-machine reconfigurations: Plans and situated actions.* Cambridge university press.

10 Appendix A: Walkthrough notes in danish

Walkthrough noter

1. Inputs
    a. Enige, ikke noget skrevet ned
    b. Lavet af Mads og Viktor
    c. Lavet af Mads og Viktor
    d. Enige, ikke noget skrevet ned
2. Convene
    a. Finde brugervenlighedsproblemer
    b. Finde problemer, simple løsninger
    c. Ikke designe løsninger i dybden
    d. Briefet
    e. Briefet
    f. Rollefordeling
        i. Mads: Ordfører
        ii. Viktor: Problemnotering
    g. Briefet
3. Udført, se “Problemer med SCW”
4. Critical information
    a. Problemer
        i. Manglende information på knapper
            1. Titler
            2. Synlighed
                a. Scrolling
                b. Knapper, der bliver for små i edge cases
        ii. Layoutproblemer
            1. Widgets, der skubber til hinanden
        iii. Feedback på handlinger
            1. Import og eksport
    b. Design ideas
        i. Label til knapper til højre, løser i.1
        ii. Et mindre destruktivt layoutsystem, løser ii.1
        iii. Kontekstuelle popups/modals til at indikere success/failures, løser iii
    c. Design gaps
        i. Låst menubar, løser i.2.a
        ii. Lås knappestørrelser, løser i.2.b
    d. Problemer med task analysis
        i. Punkter som instruktioner frem for mål, rettet løbende
        ii. Nogle features ikke test, forhåbentligt testet under brugerundersøgelse
        iii. For mange tasks, nogle blev sorteret fra (4 + 5)

Problemer fra SCW

Scenarie 1

Brugeren vil gerne have en side der består af ”større skema” og ”afleveringer” widgets

Task list

- Gå ind på forsiden
- Vælg ”Edit mode”
    - Edit mode er ikke synlig hvis brugeren scroller ned ad siden
- Fjern de komponenter der ikke skal bruges
    - Brugeren kan ikke finde krydset på widgets hvis vinduet bliver for lille
- Flyt ”skema” og ”afleveringer” komponenterne op i venstre hjørne
    - Brugeren rykkede “Afleveringer” widget for tæt på “Skema” widget så denne blev flyttet nedad
- Skift størrelse på ”skema widget” til at være større
    - Brugeren resizede “skema” widget så den skubbede til den anden widget
    -
- Gå ud af ”Edit mode”
    - Edit mode er ikke synlig hvis brugeren scroller ned ad siden

Scenarie 2

Brugeren vil gerne gå tilbage til default layout, opsætte et nyt bestående af ”Stort skema”, ”Announcements”, ”Afleveringer” og ”Startende snart” og eksportere det

Task list

- Reset layout
    - “Reset” fortæller ikke hvad der resettes, og knappen kan ikke altid ses
- Gå i ”Edit mode”
    - Edit mode er ikke synlig hvis brugeren scroller ned ad siden
- Fjerne ikke-relevante komponenter
    - Brugeren kan ikke finde krydset på widgets hvis vinduet bliver for lille
- Gå ud af ”Edit mode”
    - Edit mode er ikke synlig hvis brugeren scroller ned ad siden
- Eksportér layout
    - Export knappen fortæller ikke hvad der eksporteres, og knappen kan ikke ses hvis der scrolles ned

    - Svært at se, at resultatet (tekststrengen) fører én tættere på målet. For en uerfaren kunne det lige så godt ligne en fejlbesked
    - Får ikke at vide, at brugeren selv skal gemme tekst strengen til fremtidig import
    - Export knappen kan ikke nås hvis brugeren ikke er gået ud af “Edit mode”

Scenarie 3

Brugeren vil gerne importere layoutet fra Scenarie 2 til sin anden PC

Task list

- Tryk på ”Import”
    - Knappen er ikke synlig når man ruller ned, og fortæller ikke hvad der importeres
- Indsæt den gemte tekststreng
    - Hvis en bruger importerer en “invalid” tekststreng sker der ikke noget, og brugeren får ikke en fejl besked
    - Hvis tekststrengen er korrekt, men indeholder ugyldige værdier, får man en fejlbesked, og derefter slettes widget’en
    - Det er ikke klart hvad der skal indsættes i den pop-up der kommer op
- Tryk på ”Ok”
    - Hvis brugeren importerer det samme design der vises kan man ikke vide at importeringen var succesfuld

Scenarie 4

Brug af default layout

Task list

- Læs announcements
- Åbn announcement
- Se rettede afleveringer
- Gå til nuværende forelæsnings Zoom-link
- Check skema

Scenarie 5

Brugeren vil ændre indstillinger på afleverings-widget’en

Task list

- Gå i edit mode
- Skift til indstillingstilstand på afleverings-widget
- Skift indstillinger
- Skift ud af indstillingstilstand
- Gå ud af edit mode

11 Appendix B: Scenario in danish

Noah er en medicinstuderende på Aarhus Universitet der er på 4. semester sat til at prøvekøre det nye studerende.dk . Noah åbner først siden, hvor han er mødt med et simplistisk login system (wayf? nemID? andet).

Efter at logge ind, ser han et default dashboard som alle nye studerende ser når de først kommer ind på siden. Han kigger først på sit skema, og ser ugeplanen for hans uge er opdateret og korrekt. Han ser at der er et zoom link til de forelæsning han skal til i dag, og lokaler på hans fysiske steder (eks. lab timer).

Han syntes skemaet fylder for lidt i forhold til announcement-widget'en. Han sletter announcement-widget'en, da han ikke mener der er specielt meget nyttig information og gør størrelsen på skemaet større. Han åbner hamburger menuen og ser på de widgets den her. Noah kan godt lide "starting soon" widget'en og placerer den lige (over?) skemaet.

Han flytter afleverings-widget'en helt til højre og gør den så lang så han kan se alle hans afleveringer for nu. Han ændrer en indstilling i afleverings widget'en der gør, at "due soon" kommer op når afleveringen er til om 7 dage, istedet for hvad den var før (3?), da hans afleveringer generelt kun er store (4+ uger).

Han lukker fanen ned fra hans laptop, da han skal til med.lab. På vej til au åbner han (app? webside? nok webside) op på hans telefon, for at finde ud af hvilket laboratorie han skal til i dag.

Først skal logger han ind rimelig smertefrit. Han skal desværre igennem wayf, men han sørger for applikationen husker han er logget ind.

studerende.dk har husket hans ændringer til layout, og Noah klikker hurtigt på lokalet fra "starting soon"-widget'en, der automatisk åbner telefonens maps applikation op (google maps, apple maps, lignene).