

UNIVERSITY OF BERGEN

FACULTY OF SOCIAL SCIENCES

Department of Information Science and Media Studies

MASTER THESIS

Niche News and the Mobile Web: An Alternative Interface

Author: Bård Magnus Bachmann

Supervisor: Barbara Wasson

June 25, 2013

Abstract

The presentation of web content has become increasingly complex as the content is accessed on several different devices. When presenting content on mobile phones there are several options to choose between, such as *responsive web design*, *web applications*, or *native applications*. Each option has advantages and disadvantages that should be considered to achieve a positive user experience.

This thesis presents and discusses three possible interfaces for presenting niche news on mobile phone. The goal of the research was to understand which design elements and which functionality of niche news presentation is core to the user experience, as well as discern the technology on which to present these features. To this end, design science research was employed as the research method. A co-operation with *Serienytt.no*, a niche news blog, was arranged in order to gain access to existing content, and for readers to test and evaluate the prototypes.

The prototypes were created in three iterations, of analysis, development and testing. A web survey was used to gain an initial understanding of the user group. *Heuristic evaluation*, with a total of seven experts, was used to improve upon the first and second iterations, while *user testing* was used for the summative evaluation. Five readers from *Serienytt.no* participated in the testing.

An analysis of the collected data showed that all of the *technologies* tested can result in positive user experiences. Responsive Web Design is the cheapest to produce, while applications cost more, but can give a more varied user experience. As for the *design elements* in niche news presentation, it was shown that simple and clear interaction is the most appreciated. Elements such as single columns, tabs, search bars and share-functionality are all valid and easily understood by users. In *functionality*, it was shown that the expectations of functionality is different between a website and an app. A mobile website is expected to convert most of a website's content, while an app is not necessarily expected to have the same functionality. These results are based upon data collected in this study, as well as previous research.

The results presented in this thesis are based upon the analysis of the data collected in this study, as well as previous research. These results are compiled into a list of guidelines for

niche news on mobile phones, which highlights the main considerations that should be made when adapting niche news content to mobile phones.

Preface

This thesis concludes my Master in Information Science education at The University in Bergen (UiB). The thesis was carried out over two semesters, fall 2012 to spring 2013, at the Department of Information Science and Media Studies, in cooperation with Serienytt.no.

I would like to express my appreciation of the people who have helped during the course of this study. First, I would like to give a special thanks to my thesis supervisor, Barbara Wasson, whose guidance has helped me throughout this thesis. Her suggestions and encouragement have been instrumental to the project.

I would also like to thank the contributors of Serienytt.no, and especially Vidar Daatland, for the cooperation and the permission to use their content. It made the development process highly enjoyable and more relevant as research.

My thanks also go to the experts that participated in testing, for new point of views and discussions, and the readers of Serienytt, whom contributed with their opinion on the mobile web.

Contents

Abstract	ii
Preface	iii
1 Introduction	1
1.1 Motivation	2
1.2 Research Questions	5
1.3 Thesis Contents	5
2 Web Technologies	6
2.1 Mediums	6
2.2 HyperText Markup Language (HTML5)	10
2.3 Cascading Style Sheets 3 (CSS3)	11
2.3.1 Media Queries	12
2.3.2 Browser Support for CSS3	13
2.4 Responsive Web Design (RWD)	15
2.5 Adaptive Web Design (AWD)	17
2.6 Touch Gestures	17
2.7 Summary	18
3 Fields of Research and Related Work	20
3.1 Human-Computer Interaction and Interaction Design	20
3.2 HCI Research on Testing Usability	21
3.2.1 Usability Inspection and Heuristic Evaluation	21
3.2.2 Questions, Options and Criteria	23
3.2.3 Usability Testing	24
3.3 HCI Research on Web Usability	25
3.4 Research on Mobiles and Touch-Screen Interfaces	26
3.5 Design Guidelines for Mobile Phones	27
3.6 Media Studies and Content Personalisation	28
3.7 Summary	30
4 Research Methodology	31

4.1	Design Research	31
4.1.1	Design Research Guidelines	32
4.1.2	Multiple-Case Design	33
4.1.3	The Design Cycle	34
4.1.4	Design Rationale	35
4.2	Development	35
4.2.1	System Development	36
4.2.2	Sketching	36
4.3	Formative Data Gathering	37
4.3.1	Questionnaires	37
4.3.2	Usability Inspection and Heuristic Evaluation	38
4.4	Summative Data Gathering	41
4.4.1	Selecting Participants	41
4.4.2	User Testing	42
4.4.3	Qualitative Interviews	43
4.4.4	Handling the Data	43
4.5	Data Analysis	44
4.5.1	Usability Analysis	45
4.5.2	Validity	45
4.6	Summary	46
5	First Iteration - Design and Development	47
5.1	Current service	47
5.2	Web survey	48
5.2.1	Web Survey Questions	48
5.2.2	The Web Survey	49
5.2.3	Web Survey Results	49
5.2.4	System Requirements and Platform Choice	50
5.3	Designing the prototypes	51
5.3.1	Platform Choice	51
5.3.2	Sketching Designs	52
5.3.3	Prototype Description	52
5.4	First development cycle	55
6	Second Iteration - Tuning the prototypes	59

6.1	Heuristic Evaluation	59
6.2	Evaluation Results	59
6.3	Design Issues	61
7	Third Iteration - Finalise the prototypes	65
7.1	Expert Evaluation	65
7.2	Evaluation Results	66
7.3	Design Issues	66
8	Prototype Evaluation	71
8.1	User Testing	71
9	Discussion	76
9.1	Prototype Preferences	76
9.2	Mobile Technologies and Niche News	80
9.3	Design Elements and Niche News	82
9.4	Functionality Beyond the Norm in Niche News	87
9.5	Guidelines for Niche News on Mobile Phones	89
9.6	Summary	91
10	Conclusion	92
10.1	Thesis Summary	92
10.2	Research Contribution	92
10.3	Limitations and Weaknesses in the Research	93
10.4	Development for Serienytt	94
10.4.1	Future Research on Mobile Usability and News	94
	Appendices	101
	Appendix A Heuristic Schema	101
	Appendix B Questionnaire	103
	Appendix C Initial System Specification	108
	Appendix D Prototype Screenshots	110

List of Figures

1	Interaction with a mouse (left) and touch (right)	1
2	A m.db.no article accessed on a desktop computer (left) and mobile (right) .	2
3	271 Norwegian news and niche news providers mobile websites (as of May 2013)	3
4	Non-mobile version of Serienytt.no	4
5	Mobile Browsers in Norway by Market Share - September 2012 to February 2013 (StatCounterGlobalStats, 2013).	14
6	Boston Globe's Website: An example of responsive web design. Top-left: mobile phone. Top-right: tablet. Bottom: desktop computer.	16
7	Simple example of a QOC-diagram (for comparing development platform)	23
8	Overview of the development process	34
9	Serienytt's current desktop site and mobile site	48
10	Prototype 3 - From paper to digital sketch.	52
11	Prototype 3 - First version	55
12	Category toggle featured in prototype 2 and 3	56
13	First version of prototype 2's table layout	56
14	Second layer of prototype two's news story selection.	57
15	First version of the front page and menu of Prototype 1	58
16	Example of overlap in Serienytt's adaptive website	60
17	First version of the transparent menu	62
18	Example of web app menu bar with device buttons (left) and Android action bar without device buttons (right)	64
19	Example of dropdown menu	67
20	Article with video - the adaptive website (left) and responsive website (right)	67
21	Front page - the adaptive website (left) and responsive website (right) . . .	68
22	Category placement screen	68
23	Front page of prototype 3 with smaller font sizes	70
24	Options in column layouts that was tested	83
25	The three menu layouts that were tried in this study	84

List of Tables

- 1 Technical overview of mobile platforms 7
- 2 Web browsers for mobile phones and their rendering engines 8
- 3 Mobile Phone Browser’s compatibility with CSS3’s recommended and proposed recommended functionality. (CanIUse, 2013) 13
- 4 Phone Browser’s compatibility with CSS3’s candidate recommendations. (CanIUse, 2013) 14
- 5 List of gestures used in Android (AndroidDeveloper, 2012a) 18
- 6 Nielsen’s list of heuristics (Nielsen, 1994) 22
- 7 Design Guidelines (Hevner et al., 2004) 32
- 8 The aspects of the three prototypes 53
- 9 Summary of the results of user testing 73
- 10 Rating of the possible choices in technology for niche news 80
- 11 Guidelines for presenting niche news on mobile phones 90
- 12 App-specific guidelines for presenting niche news on mobile phones 91

1 Introduction

For many years Internet use has been dominated by desktop devices and interfaces. Websites have been accessed on computer monitors with high resolution and aspect ratios, such as 4:3 or 16:9. However, this era has passed and today people access the web on a plethora of devices in different shapes and sizes. Furthermore, the introduction of touch phones and tablets has changed how users interact with the web (Wroblewski, 2011). Traffic from mobile phones now accounts for 14% of the worldwide internet traffic ¹.

According to Google² (2012), 48% of mobile users become frustrated or annoyed when they access a site that is not tailored for mobile phones. A website that has not been adapted for mobile devices appears small and requires interaction, such as zooming, to read text or look at an image. This results in a tedious user experience. Another interaction issue is that a mouse-pointer is far more accurate than touch interaction. While small links or elements are easy to interact with on desktops, it becomes a hard task on touch phones. Figure 1 illustrates the difference in interaction area between a mouse and touch. The mouse pointer can easily click on the link, while the touch interaction (indicated by a red circle) hits several links at once.

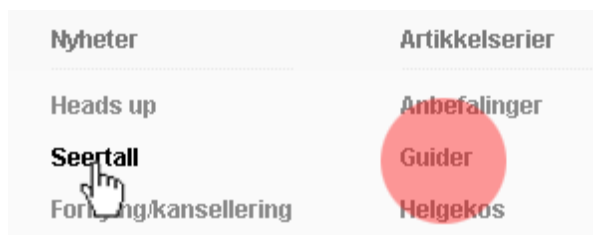


Figure 1: Interaction with a mouse (left) and touch (right)

Issues like these have encouraged many websites to create mobile-specific versions of their content, which are designed for smaller screens and portrait orientation. These mobile-specific websites usually contain larger fonts, simplified layouts, larger clickable elements, and is accessed through their own web address such as mobile.myaddress.com. This is the most common solution used among news media in Norway. The problem herein lies in that websites are often shared between different platforms, which can result in users ending up on the wrong version. For example, db.no, one of the largest news websites in Norway, has three versions: m.db.no (touch phones), ipad.db.no (tablets) and db.no (desktops).

¹http://gs.statcounter.com/#mobile_vs_desktop-ww-monthly-201204-201304

²<http://googlemobileads.blogspot.no/2012/09/mobile-friendly-sites-turn-visitors.html>

If a user, who is on a touch phone, finds an interesting article, see figure 2 (left), and wants to share it with his friend he links the article to his friend. His friend, who is using a desktop computer, sees the version in figure 2 (right). Since the user had linked the mobile version, it results in a poor experience on the desktop system. It is possible to avoid this, for example, by including code in the website application that redirects automatically to the correct version, but it is not a foolproof solution.



Figure 2: A m.db.no article accessed on a desktop computer (left) and mobile (right)

A better solution to the problem would be to use technologies such as responsive web design, through web standards such as HyperText Markup Language (HTML), Cascading Style Sheets (CSS) and JavaScript. This would allow the web content to be tailored to any resolution and screen size. Another solution would be create an "app", which is a new version of the content that is specifically designed for touch phones. While responsive web design and "apps" have their own problems, the goal is the same - to create a better user experience for users on mobile phones.

1.1 Motivation

Accessing news is a popular activity for users on mobile phones, and is gradually becoming more popular, according to Comscore³. Looking through 271 Norwegian news

³http://www.comscore.com/Insights/Press_Releases/2012/3/Number_of_European_Smartphone_Users_Accessing_News_Surges_74_Percent_Over_Past_Year

and niche news providers, I found that many news providers have begun adapting their content to mobile phones, as seen in figure 3.

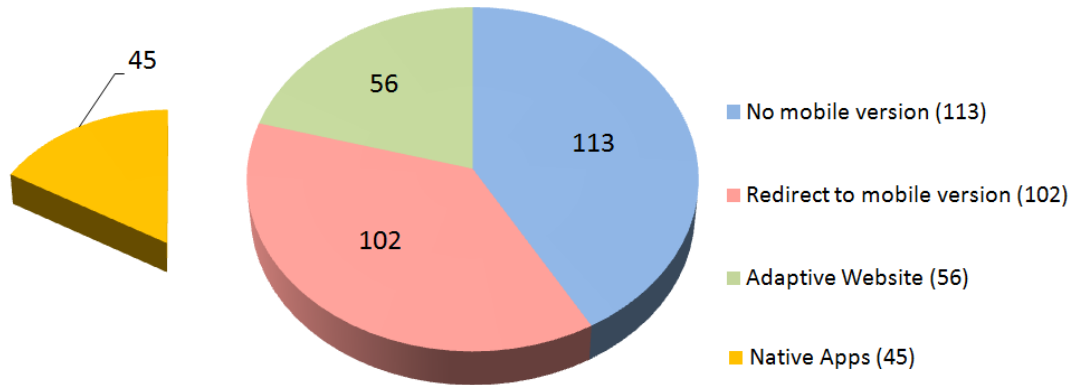


Figure 3: 271 Norwegian news and niche news providers mobile websites (as of May 2013)

There were several trends that were found through this survey. **Large budget news providers** present their content through a mobile website, and native apps for the most common mobile operating systems (iOS and Android). **News providers affiliated with large media groups** band together within the media groups to buy website templates and/or app templates. **Smaller budget news providers** use either adaptive web designs to alter their content to better suit mobile phone resolutions or do not have mobile versions. These trends led to a number of questions: *Is it really necessary to create native applications to achieve a positive user experience on mobile phones? Can a news provider with a small budget get as good a usability with cheaper technologies?* In this mindset, niche news providers are interesting as they often have limited budgets, and cannot afford to create several native applications.

Niche news providers are focused on a specific subject or theme. The news is often presented as a weblog, with social interaction between the writers and the readers. These niche news blogs attract like-minded users through topic-specific information. The social interaction results in a more personal relationship between the reader and the writer, as dialogues between them can occur in article comments or forums (Allan, 2006).

Serienytt.no is a niche news blog that specialises in news and reviews about tv-series (written in Norwegian). *Serienytt* is run non-profit through the volunteer work of a small group of writers. These writers are all amateur writers that are very passionate about tv-series and the news related to them. Financially the website is run on ad profits in addition to the financial contribution of *Serienytt*'s owner and editor. Finally, they claim that monetary profit is not a goal and they will not let advertisers affect their content ⁴.

The readers of *Serienytt* are spread all over Norway, with more than 5000 unique weekly readers. Statistics from *Serienytt*, from April 2013, show that their readers access the website from various platforms, with different screen resolutions and sizes. Twenty-three percent of the readers accessed the website from Apple's iOS and ten percent from Android, which illustrates that mobile phones are highly relevant for *Serienytt*, and thus their content needs to be presented properly on mobile devices. If they did not have a mobile version, their website would look like figure 4 which would result in a lot of zooming to click links and other elements.

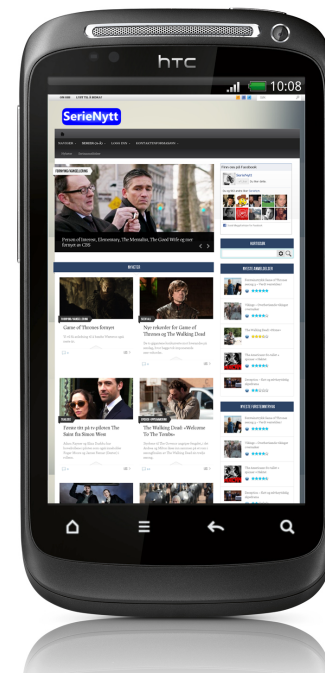


Figure 4: Non-mobile version of *Serienytt.no*

In this research I am interested in how niche news should be presented on mobile phones. *Serienytt.no* was therefore approached as to their interest in being a test case for this research, and they agreed. Co-operation is mutually beneficial for both parties, as *Serienytt.no* gets access to the results of the study and the prototypes for mobile interfaces. For the research, this real-life context provides news content, and actual users that can participate in user testing and the evaluation of the prototypes.

⁴<http://serienytt.no/om-oss/>

1.2 Research Questions

The main goal of this study is to explore how niche news should present its regular website content on modern touch phones. There are several facets which must be investigated in order to suggest how this content should be presented. These facets have been defined in three research questions:

1. Which technologies should be used to present niche news on modern smart phones?
2. Which design elements does the users prefer in niche news presentation?
3. Does a niche news "app" require more functionality than regular web content?

The research was focused on answering these questions because the *technology chosen* (1) affects the technical possibilities and specifications of the product. *Design elements* (2), which relates to the visuals of the product, and the *functionality in apps compared to web content* (3) helps developers understand the expectations of users and understand which design features are important. This will help to achieve better user experience on mobile phones. The research into these questions results in guidelines for niche news on mobile phones.

1.3 Thesis Contents

This thesis is organised into 10 chapters. Chapter 2 introduces the technologies that can be used to present niche news on touch phones. Theory and related works, within the field of interaction design, is presented in chapter 3, which also introduces some research on news on mobile phones from media studies. Research Methodology, chapter 4, explains the research, development, and data collection methods that were used in the study. Chapters 5-7 document development iterations, with data collection, analysis, and prototype design. These chapters are followed by the documentation of the summative evaluation of the prototypes. Chapter 9 discusses the evaluation results and related research, and presents a set of design guidelines for niche news on mobile phones. Finally, the last chapter summarizes the results and suggests future research.

2 Web Technologies

This chapter presents the different web technologies and mediums that are used to deliver web content on mobile phones. The chapter includes programming and scripting languages, design techniques, and, mobile phone norms.

2.1 Mediums

There are many ways to deliver content that is specifically customised for mobile phones, and several mediums that can be used to display content (Fling, 2009). Three of them are especially important for niche news: the mobile website; the mobile web application; and the native application.

A mobile website is a website that has been designed for mobile phones. Mobile websites are usually simplified versions of the desktop websites, which allows for easier touch input. To remove some functionality from a desktop version is quite common for mobile websites, which makes it easier for the user to find the core functionality and information of a website (Fling, 2009). These websites can either be a separate version of the desktop version, accessible through its own addresses such as `mobile.myaddress.com` or `m.myaddress.com`, or through CSS3's media queries. Media queries allow the browser to load a separate set of rules to build the web pages, based on device information. Both methods allow the web site to reduce the download on mobile phones, which makes for quicker interaction and downloads on slower internet connections. A mobile website can even go so far as to emulate the visuals of a native application.

The removal of information from mobile websites is, however, not supported by the World Wide Web Consortium (W3C). In their Mobile Web Best Practices, they write that they wish the Internet to remain as "One Web". The best practices further explain:

"One Web means making, as far as is reasonable, the same information and services available to users irrespective of the device they are using. However, it does not mean that exactly the same information is available in the exactly same representation across all devices. The context of mobile use, device capa-

bility variations, bandwidth issues and mobile network capabilities all affect the representation". (W3C, 2008)

To summarise, the W3C believe that web content can be visually altered to suit mobile phones, but should not exclude information. To meet this practice it is therefore common to have a "go to desktop version"-link if you are excluding information. This allows there to be two versions of a website, which suit the user's need in any given situation.

The alternatives to mobile websites are applications (or apps) customised for mobile phones. Table 1 shows the possible technologies for delivering niche news on mobile phones and some of the technical constraints. There are two core technologies to choose

Table 1: Technical overview of mobile platforms

	Works on a single OS	Works on several OS		
Platform:	Native apps	Hybrid apps	Web apps	Web Content
Runs in:	Native program	Native Shell	Web Browser	
Written with:	Native code	Web standards HTML, CSS and JavaScript		
Installed from:	App stores	Websites		

between: Native apps, and, web apps. A native app is an application that runs locally on the phone, in the native code, and are built specifically for a certain operating system. These applications can be installed from the app store for each platform: Apple App Store (iPhone); Play Store (Android); Windows Phone Marketplace (Windows Phone); and, Blackberry App World (Blackberry). Web apps on the other hand are run as web content, and can be opened in the device's web browser Through web standards, such as JavaScript, HTML5 and CSS3, web apps can be accessed through a web address (URL). Web apps can therefore be defined as "a website that offers app-like functionality" (Castledine et al., 2011). The app stores for each platform will, however, not allow you to publish web apps, so you have to choose a different payment system if you want to charge for the applications.

Native apps are locked to a specific operating system. This means you have to create an app for each operating system in several different programming languages (Charland and Leroux, 2011). Android uses a Java dialect, iPhone employs in Objective C, Windows Phone in C or C++, and so on. An Android app will not work on an iPhone and vice versa. Native apps are locked to a single platform, so several applications are required if you wish to avoid platform fragmentation. Web apps can avoid this fragmentation, since web browsers exist for most platforms and they all employ the same web standards.

Another issue is performance, where native apps have a definite advantage. A native app is built, optimised and compiled for a specific platform, which allows it to run more smoothly and with a better user experience than web apps. Web applications have to run through an additional layer of interpretation to present content. Each web browser has chosen a rendering engine that does this job, as shown in table 2 (Castledine et al., 2011).

Table 2: Web browsers for mobile phones and their rendering engines

Browser	Android Browser	iOS Safari	webOS	Google Chrome	Blackberry Browser	Opera	Firefox	Internet Explorer
Engine	Webkit					Presto (Webkit)	Gecko	Trident

Android Browser, Blackberry Browser, webOS, Safari and Chrome all use an open source-engine WebKit, Firefox uses Gecko, Internet Explorer uses Trident, and Opera uses Presto, but recently announced that they will change to WebKit⁵. The interpretation that these rendering engines do creates a natural limitation to the functionality of web apps. Native apps can access the entire technology stack of the operating system, while web apps can only access this stack through the browser. This means that a lot of methods to access the hardware, which native apps can access through the operating systems Source Development Kit (SDK), cannot be accessed by web apps. Castledine, Eftos and Wheeler (2011) offer an example:

“In iOS, for example, native applications have access to a while set of functionality that’s unavailable through Mobile Safari; for example, push notifications, the camera, the microphone, and the user’s contacts”. (Castledine et al., 2011, pg.3)

⁵http://en.wikipedia.org/wiki/Mobile_browser

JavaScript, HTML5 and CSS3 provide application programming interfaces (APIs) and built-in methods, which allow the browser to access some of the phone's hardware. For example, with JavaScript there are GeoLocation APIs that allows the browser to find users location; Web Storage APIs allows browsers to store data; and Touch Event APIs can notice touch interaction with the web browser. Video, Audio and Canvas-elements from HTML5 allows for insert media, while CSS3 offers transforms, transitions and animations of web elements (Castledine et al., 2011). A web app can access some of the mobile device's sensors and functionality, but not all.

In addition to the built-in functions of JavaScript, HTML5 and CSS3, frameworks that allow for universal app development and more advanced functionality have been developed. Sencha Touch, jQuery Mobile, jQT and many more developer frameworks offer varying functionality for web apps, and help the developer to create user interfaces designed for mobile phones. Still, there are set limits to how much of the hardware a web app can reach, and this is what services like Apache Flex and Phonegap address, by allowing for the creation of "hybrid apps".

Hybrid apps are web apps that are compiled into a native app shell. The contents of a web app are viewed through a "webview" from native code, which allows the JavaScript to interact with native methods and code. A JavaScript API can connect methods in JavaScript to the relevant methods in native code. This means it is possible to call native methods that usually would be exclusive to a native app, like the camera, microphone and contacts from JavaScript code. The gap between a web app and a native app is even smaller with this approach, but there are still issues. Performance is one of the issues that hybrid apps still struggle with, which comes from the way the code is initialised. For the JavaScript to be interpreted by the device, a hybrid app has to be parsed in the memory load, which native apps can do pre-emptively. This creates some latency in execution and therefore also in the user interface. The effects of heavy loading would be noticeably longer on a hybrid application compared to a native one (Charland and Leroux, 2011).

A final option is to use a regular desktop website. Today's smart phone browsers offer tools such as zooming and panning to allow for navigation on web sites, which enables a user to browse most web pages. The network speeds enabled by 3G make it possible to download the full versions, but it is slower than its desktop counterparts. This could lead to problems in a mobile environment, where a user is travelling while browsing.

2.2 HyperText Markup Language (HTML5)

HTML is the World Wide Web's markup language, which is used to structure and publish content on the Internet. Since its inception in 1990, HTML has been used to create websites through semantic markup, with HTML tags and text content. HTML is regularly updated by the Web Hypertext Application Technology Working Group (WHATWG). The latest generation, HTML5, has been in development since 2008 and is currently at the candidate recommendation stage.

HTML5 is built upon HTML 4.01 and XHTML 1.1 and adds more syntax features, such as `<video>`, `<audio>` and `<canvas>` elements. The addition of semantic elements such as `<header>`, `<footer>`, `<nav>` and `<section>` allow for higher readability of the HTML code. This helps developers understand the structure of a web page. Additional features such as Scalable Vector Graphics and Mathematical Markup give even more possibilities (Pilgrim, 2010).

The goals of HTML5 are summarised in the WHATWG's HTML Design Principles (W3C, 2009):

- **Compatibility:** Support the existing content of HTML. Do not add features that are already in widely used technology and support existing practices.
- **Utility:** Should solve real-world problems. Is secure by design and the Document Object Model (DOM) is consistent.
- **Interoperability:** Implementations in HTML should work well with other technologies.
- **Universal access:** Should be possible to access on different platforms and devices, able to publish in all world languages and be accessible to people with disabilities.

The goal of interoperability and compatibility can be seen in how HTML5 and JavaScript support each other.

The release of HTML5 with the new features eased the creation of web applications for developers, as it augmented the existing JavaScript libraries. JavaScript can use HTML5

APIs such as GeoLocation and Web Storage to connect with the device, and thus be interoperable through many devices.

JavaScript was developed by Netscape in 1995 as a client-side scripting language, which is dynamic, prototype-based, weakly typed and object-oriented. This allows JavaScript to interact with the user through the web browser. The document object model (DOM) elements in a HTML page can be manipulated and connected to code through selectors, which allows the JavaScript to alter the content of a web page. The web stack of HTML, DOM, XML and JavaScript, is commonly known as Asynchronous JavaScript and XML (Ajax). Ajax lets a web application communicate asynchronously with web servers, through XML or JavaScript Object Notation (JSON). This means a web application can retrieve data, while it is displaying web content in the browser. In addition, since Ajax is a web technology, it is supported by most browsers, on both desktops and smart phone devices. (Haverbeke, 2011).

Development of apps in HTML5 and JavaScript also gives access to plethora of existing plug-ins, libraries and frameworks for JavaScript development. Libraries such as jQuery, MooTools, EXT JS and Underscore.js all provide additional functionality and methods to expand upon the base of JavaScript (Haverbeke, 2011).

2.3 Cascading Style Sheets 3 (CSS3)

Cascading Style Sheets (CSS) is a styling language used to define the look and formatting of a web document, most often written in either HTML or Extensible Markup Language (XML). The style sheets created in CSS describe what layout, colors, fonts and other visual elements are presented on a web page. CSS-files can be used on any web page that links to it through HTML, which means a specific design can be reproduced over an infinite number of web pages. If the CSS-file is then modified, the changes will persist throughout all web pages that employ the CSS, which allows a web developer to easily change the design and look of a web site.

The current version of the CSS language, version 3 or CSS3, was developed by W3C's CSS Working Group. It has been in development since December 2005, and is continually improved upon. Most of the functionality in CSS3 is based upon the specification from

previous versions of CSS, but CSS3 separates them into several modules. CSS3 modules are developed by several authors with different deadlines, resulting in some modules being complete, while others are in various stages of development. W3C's official specifications gives an overview of the development status for the modules (W3C, 2013). Among the recommended modules added to CSS3 are Media Queries, Namespaces, Selectors, Colors, CSS 2.1 and CSS 1. Some work has also begun on CSS4, which are updates on modules from CSS3, but they are still not recommended for use.

2.3.1 Media Queries

Media Queries is an expansion of Media Types from CSS2. Media types are used to define different style sheets based upon device type and using media types such as "handheld", "print", "projection" or "screen". This means different designs can be defined for example: regular PC-browsers (screen) and mobile phones (handheld). The current state of media devices is, however, not as simple as this representation will have it. A computer screen or a mobile screen can have a wide variety of screen sizes (Powers, 2012).

In the new Media Queries-module, it is possible to separate by other categories than just media types. Some examples include: device width and height; orientation (landscape or portrait); and, colour (amount of bits). This means it is possible to create very specific rule sets, based upon these conditions (Powers, 2012). The example below sets the font-size for all paragraphs on screen resolutions that are between 601 and 768 pixels wide:

```
@media screen and (min-width: 601px) and (max-width: 768px){  
  p {font-size: 1.1em; }  
}
```

Thus by defining extensive device information, it is possible to create very specific designs for different devices. There are still issues with using this for mobile phones, as they have highly varying pixel densities⁶. Android devices have four set densities, independent of the physical screen size: Low Density (ldpi), Medium Density (mdpi), High Density (hdpi) and Extra High Density (xhdpi). The lowest setting, ldpi, has only 120 dots per

⁶A possible fix (min-device-pixel-ratio) for this is in CSS Values and Units Module Level 3, but it is currently in the candidate recommendation stage

inch (dpi), while xhdpi has 320 dots per inch. This means that a box set to 300 pixels wide will be less than an inch wide on xhdpi, while it is 2.5 inches wide on the ldpi screen. The difference show even more in square inches, as ldpi has 14400 pixels per square inch, where xhdpi has 102400 pixels (AndroidDeveloper, 2012c). The iPhone has similar issues with iPhone 4 and 5, and their retina displays. The retina display has a pixel density of 326 ppi, while the older versions of iPhone have 163 ppi (AppleInc., 2011a).

A wide variety of pixel densities available on mobile phones could create issues when designing for several screen sizes, but in general, devices with a high pixel density will have a higher screen resolution. If they did not then the physical screen size would simply be smaller.

2.3.2 Browser Support for CSS3

As there are many browsers and devices on mobile phones, there is also a varying degree of compatibility with CSS functionality. As can be seen in table 3, most of CSS3's *recommended* and *proposed recommended* functionality works on all browsers.

	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Blackberry Browser	Chrome for Android	Firefox for Android
			2.1: 96%				
			2.2: 96%				
	3.2: 91%		2.3: 96%				
	4.0-4.1: 91%		3.0: 98%	10.0: 98%			
	4.2-4.3: 91%		4.0: 100%	11.5: 100%			
	5.0-5.1: 98%		4.1: 100%	12.0: 100%			
Current	6.0: 98%	5.0-7.0: 84%	4.2: 100%	12.1: 100%	7.0: 98%	25.0: 100%	19.0: 100%
Near future					10.0: 100%		

Table 3: Mobile Phone Browser’s compatibility with CSS3’s recommended and proposed recommended functionality. (CanIUse, 2013)

The lowest compatibility rate is Opera Mini, which still has 84% compatibility. However, Opera has stopped all development on Mini to focus their efforts on Opera Mobile, which means Opera Mini will not receive future CSS updates. The compatibility rates means that it is completely safe to use selectors, media queries and the older CSS versions on mobile phone browsers. The state of the *candidate recommendations* is far less compatible with mobile browsers, as shown in table 4.

	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Blackberry Browser	Chrome for Android	Firefox for Android
			2.1: 38%				
			2.2: 40%				
	3.2: 36%		2.3: 40%				
	4.0-4.1: 45%		3.0: 55%	10.0: 19%			
	4.2-4.3: 46%		4.0: 69%	11.5: 50%			
	5.0-5.1: 64%		4.1: 69%	12.0: 57%			
Current	6.0: 73%	5.0-7.0: 18%	4.2: 69%	12.1: 65%	7.0: 56%	25.0: 76%	19.0: 78%
Near future					10.0: 81%		

Table 4: Phone Browser’s compatibility with CSS3’s candidate recommendations. (CanIUse, 2013)

Older versions of Opera, the Android Browser and Safari have very low compatibility with these functions, but as the browsers are updated, they support more and more candidate recommendation functionality. It is also relevant to see which browser users are currently using. Statistics from StatCounter Global Stats, seen in figure 5, show that the most common mobile browser in Norway is the iPhone’s Safari, and in close second is the Android Browser. Both of these are highly compatible with CSS.

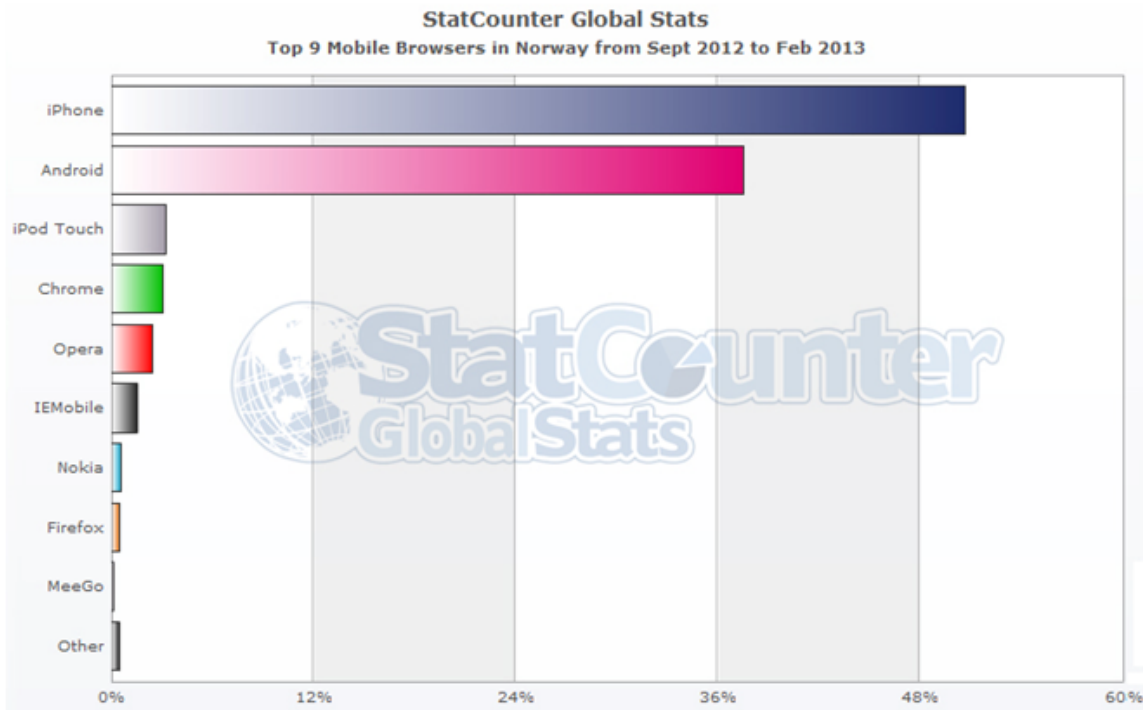


Figure 5: Mobile Browsers in Norway by Market Share - September 2012 to February 2013 (StatCounterGlobalStats, 2013).

Other browsers are not comparable in terms of market share, but iPod Touch, Chrome and Opera all have around 2-3%. Internet Explorer Mobile, Nokia and Firefox are all below 2% of Norway's market share. This means that most users will have a browser that is very compatible with CSS3, and it is a safe technology to use. The browsers that cannot handle the CSS-functionality will ignore those specific rules, but will still compile the rest. There will also be some difference in how the design looks on different browsers, but this is a natural part of how the rendering engine builds web pages.

2.4 Responsive Web Design (RWD)

Responsive Web Design (RWD) was introduced by Ethan Marcotte (2010) as a response to the ever-growing number of devices that could browse the web. Devices such as desktops, mobile phones, tablets and video game consoles all have browsers, with an equally wide variety of input modes, such as mice and keyboard, T9-keypads, touch interfaces, and game controllers. The goal of responsive web design is to alter how a web page looks depending upon the device. A RWD responds to the settings of a user's device, making it suitable for any screen resolution and screen orientation. Figure 6 shows a RWD on several devices. The design shows three key concepts of a RWD: flexible or fluid grids; flexible images; and, media queries. The fluid grids will remove columns depending on how much space is available on the device, thus making it easier to interact with for users, while flexible images will change in size, so they do not overlap the width of pages. Media queries are the technical solution that allows for this responsiveness (Marcotte, 2011).

RWD is achieved by altering the sizes of elements such as images and content-wrappers, using percentages in the CSS instead of fixed sizes. Image files, however, are static sizes, even though you change the visual size in CSS. This means it will either download images that are too small, then scale them up to larger sizes, or download large image files, then downscale them to smaller devices. On a mobile device, with low bandwidth, this could lead to an unnecessary toll on site download. There is a server-side solution, which is to have multiple versions of each image, in different sizes, enabling the site to automatically provide the one that is closest to the device size. While this takes the toll away from the client's device, it requires more disk space on the server (Bryant and Jones, 2012).

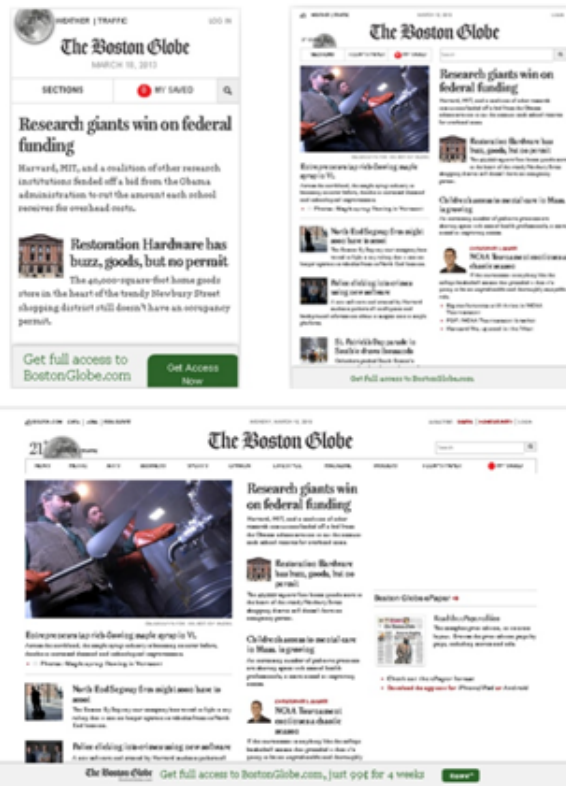


Figure 6: Boston Globe’s Website: An example of responsive web design. Top-left: mobile phone. Top-right: tablet. Bottom: desktop computer.

Font sizes are also possible to change to more device-relative sizes. This is achieved by defining font-size using em’s instead of pixels. Browsers set a default text size for HTML text, which is usually set to suit the specific device. If a browser sets its font-size to 16px, then 1.5em would be 1.5 times larger than the default (16px), namely 24px. A different browser could set its default font-size to 12px, which the relative 1.5em changes to 18px.

RWD, however, is not always the best solution. According to Bryant and Jones (2012) you can never expect pixel-perfect designs. The flexibility that allows responsive web design to suit all devices will naturally have some alterations in width and height between elements, and thus some differences are to be expected. In addition, a responsive web design requires more testing and development than a less adaptive web site (Bryant and Jones, 2012).

2.5 Adaptive Web Design (AWD)

Web pages that respond to screen resolutions and screen orientations through media queries that do not use flexible grids and flexible images are defined as "Adaptive Web Design" (AWD). Instead of using percentages to define sizes an adaptive web design will still use set pixel-sizes to adapt the web page to different devices (Gustafson, 2011).

This can be useful to modify existing web designs built for desktop browsers, so the designs work better on other devices. Creating a fully responsive web design is a large job, and basically requires that you rebuild the site from scratch, but with AWD, like in RWD, you can simply modify the existing design through media queries. Preset media types, sizes and orientations will give set designs, without flexible images and fluid or flexible grids, independent of the minor differences in devices. An AWD could, for example, set the rule: if the device width is below 600 pixels, then the image width is 600 pixels. On devices with a screen width of 600 pixels the image would completely fill the width of the screen, while on a lower screen width, the image would exceed the edges of the screen. (Gustafson, 2011).

AWD can be viewed less flexible version of responsive web design, but makes it easier to adapt from existing designs.

2.6 Touch Gestures

Touch gestures are preset interactions for touch interfaces that can be used to separate actions through different movements. Mobile devices are today equipped with multi-touch screens, thus allowing for more complex interactions. Android Developer (2012a) has defined a list of the different multi-touch gestures that work for their devices, which are connected to certain actions, as shown in table 5.

The interactions for other touch devices are quite similar. For example iOS includes all of these gestures, but also adds a shake-gesture to undo or redo actions. Native applications allow a developer to alter what these gestures do within an application. Apple Inc. (2011a), however, warns developers not to stray too far from the original actions registered to a gesture because they claim it can become confusing for the users. Once a

Table 5: List of gestures used in Android (AndroidDeveloper, 2012a)

Gesture	Action
Touch	Triggers the default functionality for a given item.
Long Press	Enters data selection mode. Allows you to select one or more items in a view and act upon the data using a contextual action bar. Avoid using long press for showing contextual menus.
Swipe	Scrolls overflowing content, or navigates between views in the same hierarchy.
Drag	Rearranges data within a view, or moves data into a container (e.g. folders on Home Screen).
Double touch	Zooms into content. Also used as a secondary gesture for text selection.
Pinch	Pinch open to zoom in or Pinch close to zoom out.
Pinch open	Zooms into content.
Pinch close	Zooms out of content.

gesture has been learned, users are comfortable in using them and inconsistency between apps would then become confusing and uncomfortable.

Web applications are basically locked to the gestures allowed by the browser application of the device, however, through JavaScript libraries and add-ons, such as jQuery, it has become possible to take advantage of touch gestures.

2.7 Summary

This chapter has discussed the different web technologies that can be used to deliver web content on mobile phones. Several mediums, such as native applications, hybrid applications, web applications, adaptive web sites and responsive web sites enable the customisation of content for mobile phones. By testing and reflecting on the possibilities of these technologies, it will be possible to answer the first sub question of the research question.

The techniques used to customise the content vary greatly between mediums, but they all share the ability to respond to device information. HTML5 and JavaScript can be used to

extract information specific to mobile phones, while CSS and Media Queries can be used to optimise web designs. Touch gestures and the interactions a user can do through a touch interface.

A choice among these technologies will be presented in the chapters 4 and 5. The restrictions and possibilities of the technology chosen will greatly impact how the final prototypes are presented.

3 Fields of Research and Related Work

This research is situated within the fields of Human-Computer Interaction and Interaction Design. This chapter reviews key works within each field that are relevant to this research.

3.1 Human-Computer Interaction and Interaction Design

Human Computer Interaction (HCI) is a research field that seeks to describe how humans interact with machines. HCI is usually described as a subset of *Interaction design*, which focuses on design, evaluation and implementation of interactive computer systems, and the phenomenons that surrounds it (Hewett et al., 1992).

Interaction design on the other hand, according to Preece, Sharp and Rogers (2007), is a broader field that includes design practices, theory, and research within all technologies. At the core of the discipline is *usability*, which describes how easily an interactive object is used and learned. The purpose of exploring the usability of systems is to enable positive *user experiences* with a product, through a clear interface, thus making a users interaction effective and easy. Unclear user interfaces will lead to errors in the interaction, which means the user has to focus on the interface instead of the task at hand. If a user can successfully interact with an interface, and focus entirely on the task, then a successful user experience has been enabled. A user experience can be defined as:

“... how people feel about a product and their pleasure and satisfaction when using it, looking at it, holding it and opening or closing it. It includes their overall impression of how good it is to use right down to the sensual effect small details have on them ...” (Preece et al., 2007, pg.15)

Forlizzi and Battarbee (2004) are among those who carry out research on user experiences. They have created a framework for understanding user experiences. This framework separates types of interactions into three categories: fluent, cognitive and expressive. *Fluent interaction* is described as when a user can automatically interact with a product without thinking about the product. *Cognitive interaction* is an interaction when the user thinks

specifically about how to use a product. This cognitive function can either result in new knowledge or errors. Finally *expressive interaction* is when the user can modify a product to express their relationship to it.

Designing interfaces is an integral part of HCI. This is shown by one of the core design methodologies in HCI, namely *user-centered design* (UCD). UCD is based around involving the users in the development of an artefact, in order to gain an understanding of what the users actually want. Through involving users at an earlier stage in the design process, UCD hopes to improve the usability of systems, as the designers can get an understanding of the users' tasks (Kramer et al., 2000).

3.2 HCI Research on Testing Usability

As mentioned in the previous section usability is related to how easily a system can be used and how easily it can be learned. Nielsen (2012) offers a more specific model of usability, with five components: 1) *Learnability*, 2) *Efficiency*, 3) *Memorability*, 4) *Errors* and 5) *Satisfaction*. *Learnability* relates to how easily a system is learned and its ease-of-use. *Efficiency* relates to how quickly users can perform their tasks. *Memorability* is how easily a user can relearn a system after a longer period without using it. *Errors* relate to how often errors occur and how severe they are. It also includes how easily users can recover from these errors. Finally, *satisfaction* is defined as if the system is pleasant to use. A high usability is key to an artefacts survival, as if the artefact is difficult to use, people will not use it (Nielsen, 2012).

There are several ways to test and improve usability. Some methods include testing with actual users, while other methods use experts to analyse interfaces.

3.2.1 Usability Inspection and Heuristic Evaluation

Usability inspection methods are often used to improve user interfaces in the early part in a development process, as usability inspection methods can be done at a low cost and does not require advance planning. The methods are therefore often used before usability testing (Nielsen and Molich, 1990).

Heuristic Evaluation is one type of usability inspection, where chosen experts evaluate a product, or an artefact, based upon certain heuristics and guidelines. While evaluating the artefact, they write down the different issues they find in the interface and then rate them by severity. The heuristics used for a heuristic evaluation are often based upon known usability principles that are relevant for user experience design (Nielsen and Molich, 1990).

One such list of usability principles, which can be used for heuristic evaluation of user interfaces, is Nielsen's (1994) ten heuristics, as shown in table 6. Nielsen's list of heuristics can also be further added upon, to fit a certain field of study (Nielsen, 1994).

Table 6: Nielsen's list of heuristics (Nielsen, 1994)

Heuristic	Description
Visibility of system status	The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
Match between system and the real world	The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
User control and freedom	Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
Consistency and standards	Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
Error prevention	Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
Recognition rather than recall	Minimise the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
Flexibility and efficiency of use	Accelerators – unseen by the novice user – may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
Aesthetic and minimalist design	Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Help users recognise, diagnose, and recover from errors	Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
Help and documentation	Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

The results of a heuristic evaluation can be analysed by using Questions, Options and Criteria or other design rationales.

3.2.2 Questions, Options and Criteria

Questions, Options and Criteria (QOC) is used to identify design problems, discover the different options one has for solving the problem, and support the decision between the different options based upon certain criteria. Figure 7 shows a simplified example of QOC reasoning. It asks the question "Which platforms should be used for a news interface for Serienytt.no" (MacLean et al., 1991).

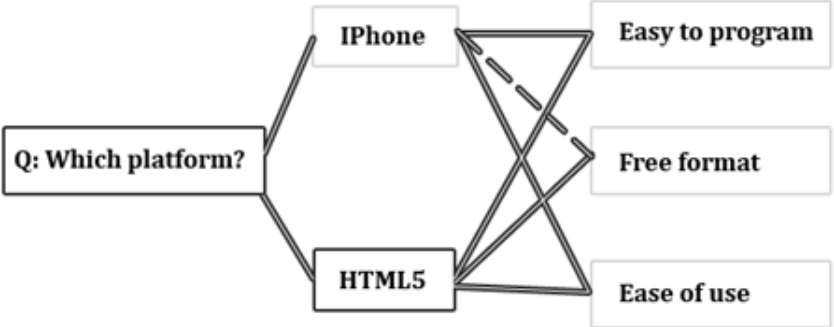


Figure 7: Simple example of a QOC-diagram (for comparing development platform)

Options, in this case, are limited to two: iPhone and HTML5. Solid lines are a positive evaluation, while stroked lines are a negative evaluation. While the iPhone is easy to

program and has ease of use, it is a proprietary format, which goes against the criteria set in the diagram. HTML5, however, gets a passing mark on all criteria and might therefore be a better fit for the example project.

3.2.3 Usability Testing

The goal of usability testing is to evaluate the quality of a product. Dumas and Redish (1999) claim that all usability testing shares five traits: improve the usability of a product with specific goals; the participants are real users; the users do real tasks; a researcher observes and records the actions of the participants; and, the researcher analyses the data, the real problems and recommend changes to fix the problems. Even though the main goal is to improve the usability of a product, it can also be used as a summative evaluation towards the end of a project. By analysing the actions of users, it allows the researcher to understand how a user experience originates and thus you can evaluate how "good" a product is (Dumas and Redish, 1999).

In usability testing it is common to give the user tasks to complete, which can be used to check the effectiveness and efficiency of the product. This is especially common in large prototypes, which are too broad to cover without direction. In smaller prototypes, it might not be necessary to use tasks, but this allows you to quantify how much time an assignment takes or how many steps it takes to complete (Dumas and Redish, 1999).

The real users that participate in the testing must be part of the potential end users of a product, therefore the test group should be taken from the potential customers. The number of users in the test group is an important question and Nielsen (2000) claims five participants is a good number. In a test with fewer participants you will not find enough of the problems with a user interface, while with more participants it is unlikely that the users will find more problems. It is therefore better to do several small tests, instead of one large one (Nielsen, 2000).

3.3 HCI Research on Web Usability

Web usability is the study of how usability affects web browsing. The context of the Internet affects how the usability is perceived by users, and this shapes the interaction with it. Websites and web content are often accessed through web browsers, which has its own set of characteristics and conventions.

Krug (2005) defined the most important usability goal of web content as: "*Don't make me think!*". The interaction with a web interface has to be self-evident or self-explanatory, with the user considering the task instead of the product itself. This is evidenced by knowledge that users scan pages, they do not read them. Users want to spend as little time as possible to complete a task on the web, and they know that on a web page most of the information will not be related to what they are searching for. Therefore, users scan the content for something interesting, possibly something clickable to find the information they are seeking. If they cannot find the information they are searching for on the website, then they can easily click away from the site and try somewhere else (Krug, 2005).

The scanning behaviour ensures that users will not see all the content, so therefore a website should be designed like a billboard. Remove all the unnecessary content, ensure that the content has a clear visual hierarchy, make clear which elements are clickable, and follow platform conventions. How close other webpages are is also noticeable in performance, as slow page download will ensure that a user leaves for a different website (Krug, 2005).

Agarwal and Venkatesh (2002) are among the researchers who have studied web usability through heuristic evaluation. They tested using the Microsoft Usability Guidelines (MUG). These guidelines separate into five major categories: content, ease-of-use, promotion, made-for-medium, and emotion. Perceived usefulness is the core measurement of *content*, while *ease-of-use* relates to how much cognitive effort is required to interact with the site. *Promotion* is how much the site brands itself to users, so they'll remember the site at a later time. *Made-for-medium* relates to subjects such as community features and personalisation. The web medium has a highly connective nature, which should be catered to. *Emotion*, the final category, has according to Agarwal and Venkatesh been shown to

be very important to computer use. The results of the heuristic evaluation showed that the content was the most important factor in website usability. Thus, it is necessary that users understand what kind of content they'll be served from the website. Their research also showed that promotion is mainly important for investors, but not so much for users (Agarwal and Venkatesh, 2002).

3.4 Research on Mobiles and Touch-Screen Interfaces

In "Mobile Phone Web Browsing – A Study on Usage and Usability Of The Mobile Web", Schmiedl et al. (2009) question the need for specific web pages for mobile web use. Through user testing, they found out that users prefer to read on specialised mobile interfaces, compared to desktop versions. Schmiedl et al. (2009) also found that the speed at which users can find information was improved by 30-40% with mobile-tailored layouts. Some users found the tailored versions limited, because some of the desktop-version's functionality had been removed.

This highlights one of the important design decisions for presenting mobile websites, which is: Should a mobile-specific version remove content to simplify interaction with a mobile website? McGrane (2012) is among those who claim users should not be limited when accessing websites on a mobile phone. *"People use every device in every location, in every context. They use mobile handsets in restaurants and on the sofa. They use tablets with a focused determination in meetings and in a lazy Sunday morning haze in bed."* (McGrane, 2012, pg. 18). Mobile use is not exclusive to a 'mobile context', so content should not be tailored for people who are 'on the move and only requires the most important functions'. The use is not exclusive to task-based interaction, as there is also information-seeking interaction.

On the other side of the argument is Luke Wroblewski, whom in his book 'Mobile First' explains that mobile interfaces should only focus on creating interfaces for mobiles, then expand to desktop designs. Creating interfaces for mobiles ensures that the most important content is placed in focus, as designers cannot afford to waste space. The important tasks are brought to focus, and thus increasing the ease-of-use for users (Wroblewski, 2011).

Wroblewski (2011) mentions several constraints that affect how a mobile interface should be designed. *Screen size* is one of these problems, but it also includes performance and contexts, such as time and place. *Performance* is related to the speed of the system, as mobile phones are often constrained in both download speed and computing speed. Therefore a mobile website should consider using image-sprites (several files combined into a single files, which reduces the amount of HTTP requests), minifying JavaScript and CSS files, avoid using large JavaScript libraries to only use a few functions, consider using CSS3-functions instead of images, when possible, and properly caching the system. These are some of the performance-improving options that can be used to improve the mobile user experience. The *context* of mobile use, according to Wroblewski (2011), can also affect how the mobile is used. Sometimes the mobile is used simply for proximity, as going to a desktop will take more time, while at other times phones are used in a mobile situation, with its own set of constraints. To design for a positive user experience, these constraints have to be considered (Wroblewski, 2011).

The issues in screen size have also been researched by Park and Han (2010). Physical constraints also affect touch interaction, as thumb interaction will result in lower accuracy than using the index finger, yet one-thumb interaction is still used. Park and Han (2010) researched one-thumb touch interaction, by letting users interact with squares sized at 4mm, 7mm and 10mm wide. The results showed that 7mm and 10mm are the fastest, while 10mm is the most accurate. In addition, they also tested button location, and found that squares on the left side results in less errors than squares on the right side. This affects how interfaces should be designed (Park and Han, 2010).

3.5 Design Guidelines for Mobile Phones

Several research groups have made suggestions for design guidelines for mobile phones. Nielsen and Budiu, from the Nielsen-Norman Group, made 85 design guidelines for customising web content for mobile phones (Nielsen and Budiu, 2009). Through usability testing and diary studies they discovered that customised content for mobile phones has a higher success rate than normal web content. In addition, they claim that websites that users enter to '*kill time or browse around*' are especially important to customise for mobile phones. Based upon this observation they created their list of design guidelines.

Among the guidelines that Nielsen and Budiu (2009) have created for Web content for mobile phones is advice such as:

- *For touch phones, widget target area (i.e., clickable area) should be at least 1cm x 1cm.*
- *For touch phones, leave generous amount of space around widgets.*
- *Expandable menus should be used sparingly. Menu labels should clearly indicate that they expand a set of options.*
- *On browsing sites new content should be given priority. Users should not have to scroll to get to new content.*

These and many more guidelines indicate many specific details about how web content should be presented on mobile websites. In general, most of them relate to making tasks as simple and effective as possible to carry out on a small-screen interface. The visibility and effectiveness of design elements must be high for an interface to be successful on touch phones.

Another framework was created by Heo et al. (2009). Their framework consists of five main usability indicators: (1) Visual support of task goals, (2) Support of cognitive interaction, (3) Support of efficient interaction (4) Functional support of user needs and (5) Ergonomic support. These indicators focus on visibility, effectiveness and efficiency of a mobile system. This further supports the importance of a clear and simple interface for mobile phone interaction (Heo et al., 2009).

3.6 Media Studies and Content Personalisation

The three core elements commonly placed on the front pages of online newspapers are headlines, images and bylines. These elements are also central for news presentation in niche news. When a user scans the page, as Krug (2005) explained, large elements and contrasts catches the eye of a user. The goal of the large headlines and images is therefore to grab the attention of the reader, like they would on a billboard.

Zillmann et al. (2001) explain how photos attract readers. The common phrase 'an image is worth a thousand words' shows how images can tell a story, and thus lures the reader towards reading the text related to the images. Zillmann et al. (2001) also asked readers to read news reports with no images, innocuous images and agonistic images. Readers spent less time reading the reports with no images, than reports with agonistic images, while innocuous images only had a minor improvement compared to no images. This study shows why images are a hallmark element of news presentation on the web, despite the fact that it can noticeably increase load times.

Another article related to digital news presentation is "*Open User Profiles for Adaptive News Systems: Help or Harm?*" by Ahn et al. (2007), in which they researched the personalisation of news, based on manual editing. Their system let users add keywords to a news presentation, which filtered the news. Through user testing, they found that the personalisation of news should be used with caution, as too many keywords lowered the precision and recall of the system. However, the users were positive to the system.

Research on news relevant to mobile phones, however, is scarce, especially related to specific niche content. One of the studies that has been carried out on usability for mobile web newspapers by Jeong and Han (2012), who tested the usability of 303 websites. They focused on saving space for the content, and thus suggested improvements such as hiding the browser URL-bar on load and hiding article timestamps. Features they claim are required include: the current time and date; time for the latest news update; and, links to the full page. From the results of their study, they also claim that a unique design is important to distinguish the content, but templates are cheaper and easier to implement. If a template is used, then it is important to place the logo easily visible (Jeong and Han, 2012).

In Jeong and Han's (2012) study, they also claim that most newspapers should present their content as web content, not as apps. Their reasoning is that installing several news apps is cumbersome for users, and that developing several apps for a website is inefficient (Jeong and Han, 2012).

3.7 Summary

This chapter has explained relevant Interaction Design research including usability testing, web and mobile usability, touch screen interfaces and design guidelines. In addition, related research on news on digital platforms has been summarised.

4 Research Methodology

The study reported in this thesis used a design research methodology. This chapter explains what design research is, and why this study is appropriate for design research. Then the different research methods that have been used are introduced, including the choice of design rationale, the design methods, the system development methods, the methods of data gathering and how data will be analysed.

To answer the research questions of *choosing technology*, *choosing design elements* and, *choosing functionality*, this study tests several issues and options for each question. The testing is used to explain the advantages and disadvantages of choosing a specific technology, including a specific design element or functionality.

4.1 Design Research

In "Design Research and Information Science", Hevner et al. (2004), describe two research paradigms for Information Science: *behavioural science* and *design research*. Behavioural science aims to describe human behaviour through developing or verifying theories of human and organisational behaviour. To the contrary, design research wishes to expand upon the human and organisational capabilities through creating novel artefacts. Design research also works well with UCD, as both methodologies seek to create artefacts and understand human behaviour. (Hevner et al., 2004).

Design Research seeks to solve problems in the application of information systems. Hevner et al. (2004) writes that design research:

"It [Design Research] seeks to create innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, management, and use of information systems can be effectively and efficiently accomplished". (Hevner et al., 2004, pg. 83)

The notion of effectiveness and efficiency are core values in the application of information systems, and thus it is important to explore the different factors that influence the design

of information systems. This study explores effectiveness and efficiency of mobile phone browsing through alternative interfaces, it can be described as the creation and evaluation of novel artefacts through a design research methodology (Hevner et al., 2004).

4.1.1 Design Research Guidelines

To rationalise the creation of a problem space and solutions, Hevner et al. (2004) suggested seven design research principles, see table 7, which they express as guidelines for conducting a design research project. These guidelines give clear requirements for design

Table 7: Design Guidelines (Hevner et al., 2004)

Guidelines	Description
Guideline 1: Design as an Artifact	Design-science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation.
Guideline 2: Problem Relevance	The objective of design-science research is to develop technology-based solutions to important and relevant business problems.
Guideline 3: Design Evaluation	The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods.
Guideline 4: Research Contributions	Effective design-science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies.
Guideline 5: Research Rigor	Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.
Guideline 6: Design as a Search Process	The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.
Guideline 7: Communication of Research	Design-science research must be presented effectively both to technology-oriented as well as management-oriented audiences.

research. Focus on themes such as quality, efficiency, and functionality makes it easier to satisfy the requirements of design research. Once a satisfactory argumentation based upon the artefacts can be made for each of these guidelines, the project is finished. Finally, it is important to emphasise that design research must be based upon a satisfactory theoretical base. An excessive focus on the technological artefacts could result in artefacts that are useless in real-world setting (Hevner et al., 2004).

Hevner et al.'s (2004) guidelines were used throughout the process of this study to ensure that the study was conducted properly. *Design as an artefact*, the first of Hevner's guidelines, is fulfilled through the creation of multiple artefacts. The comparison of these artefacts is the *research contribution* of this study. Through analysing the different elements and functionality required for niche news on mobile phones it helps to build a design foundation.

The cooperation with Serienytt.no shows that the study is highly relevant to the niche news-genre. The *problem relevance* of the study is mainly for the research area of niche news on mobile phones, but the results may also be relevant to a more general situation like mobile web usability. *Design evaluation*, as well as *research rigour*, are important to the presentation of results. Through solid research foundation and careful evaluation of the artefacts, the research is valid. *Design as a search process* is met through the search for issues and solutions with multiple-case design. The *Communication of the research* is done through this thesis and by a report on Serienytt.no.

4.1.2 Multiple-Case Design

One goal of this study is to understand what design elements should be part niche news presentation on mobile phones and to create guidelines for future development. For this purpose, a multiple-case method was chosen. Benbasat et.al. (1987) writes that: "*Multiple-case designs are desirable when the intent of the research is description, theory building, or theory testing. [...] Multiple-case designs allow for cross-case analysis and the extension of theory.*" (Benbasat et al., 1987). This clearly states that multiple cases, or multiple interfaces, is more suited for theory building compared to single-case designs. By comparing the prototypes to each other, then a further understanding of the problem field will be achieved (Benbasat et al., 1987).

For the purpose of comparing multiple options in web mediums and technologies, three different prototypes will be made. By creating three different prototypes, it will be possible to look at several angles and solutions to which design elements are required for mobile news presentation. The technologies chosen for these prototypes will be chosen among the technologies covered in chapter 2.

4.1.3 The Design Cycle

Takeda et al. (1990) separate a design research project into five steps, known as a design cycle: 1) awareness/identification of problem; 2) suggestions of problem solutions; 3) development of artefacts; 4) evaluation of artefacts and finally 5) the conclusion of the project. Awareness of a problem can come from many sources, such as new developments in technology or research. The suggestion of problem solutions should lead to tentative designs, which can be developed to artefacts. Evaluation is then carried out based upon the criteria set during the identification of the problem (Vaishnavi and Kuechler, 2004).

This study used these five steps, mentioned by Takeda et al. (1990), as a design research method, and has completed all of the steps. First the problem space was analysed and suggestions were made to solve the problems that were discovered. Then development was done through several iterations of design suggestions, development of prototypes and testing of the prototypes, as can be seen in figure 8. The first iteration began with the initial system specification and resulted in three prototypes. Iteration two started with a heuristic evaluation of the previous iteration. The results of this evaluation informed the redesign and redevelopment of the three prototypes. The third iteration began with further heuristic evaluations by experts to fine-tune the prototypes further. This iteration ended with the final version of the three prototypes. After the third iteration, a final summative evaluation was done, to conclude the study. Each of these iterations will be described in detail in later chapters.

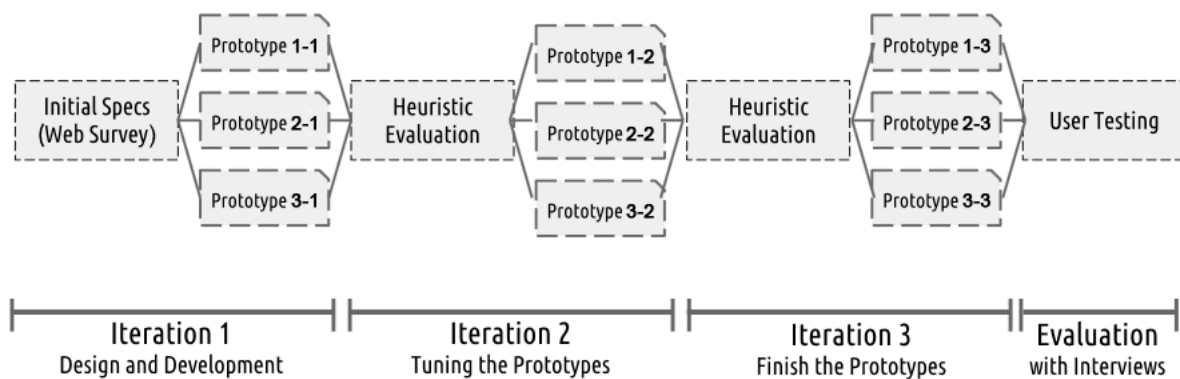


Figure 8: Overview of the development process

4.1.4 Design Rationale

Design rationales are documentation to decision-making processes that are made while designing an artefact. According to Lee and Lai (1991), design rationales are used in three different ways:

- A historical record of the reasons for the choice of an artefact
- A set of psychological claims embodied by an artefact
- A description of the design space

QOC is one such design rationale model, which will be used for suggestions of problem solutions. This method was also used to carry out a requirement analysis, which led to the system requirements of the applications. A clear understanding of the requirements of the systems needs is necessary, or as Philip Weaver declares: *"To develop a system or application without a clear idea of what it needs to do will inevitably lead to a system that is unfit for its intended purpose"* (Weaver, 2003, pg. 179). The reasoning behind a choice of problem solutions was therefore based upon QOC.

4.2 Development

The design process for the artefacts was an iterative process, with a separate iteration for each prototype. This required thorough planning and time management, to ensure that all three prototypes were of more or less equal quality. An iterative process, with testing, also allows external input on the quality of the prototypes to see if one prototype required more work than the others.

A project plan was created early in the project to ensure that no single iteration would receive more focus during development. Several testing sessions and several development cycles was an important part of evaluating different design elements, so an iterative process had to be ensured. The first step of the project plan was to gain an understanding of the content that Serienytt.no produces, the readers that enjoy their content and what functionality they had at the time. Once an understanding of the user group was reached,

some initial system requirements were made for each prototype and then a system development cycle began.

4.2.1 System Development

System development was done through agile methods, with interchanging development between prototypes. By swapping freely between the development of the three prototypes, it meant that the prototypes reached an equal quality at the end of each development cycle. Since the project specification was split into several "levels of importance", it was also possible to work on a prototype until the specification is met, and then go on to the next prototype. An agile methodology also allowed changes in the specification, so new literature or feedback from users or research peers led to a re-evaluation of the system specifications. The ability to change requirements at any stage of the development is a core principle of agile development (AgileManifesto, 2001).

4.2.2 Sketching

To design the interfaces, it was important to be able to test as many possible designs as possible. Ideas for interfaces had to be tested quickly and cheaply, to see if there was value to adding the idea to the prototype design. This process of finding ideas for the prototypes was done through sketching.

"The ability to rapidly sketch objects with uncertain types, sizes, shapes, and positions is important to the creative process. This uncertainty, or ambiguity, encourages the designer to explore more ideas without being burdened by concern for inappropriate details such as colors, fonts, and precise alignment". (Landay and Myers, 2001, pg. 57).

Sketching on paper was used for simplicity and flexibility. All of the prototypes received multiple designs, to explore the different options. The technology chosen for a specific application affected the options considered in the sketches, as some were impractical to

implement on a specific technology. Once several designs had been created for each prototype, some were chosen for a higher-fidelity digital sketching, which was used as templates for the first development cycle. Higher fidelity sketching is better at conveying the design possibilities and gives a more realistic interaction with the sketches (Landay and Myers, 2001).

4.3 Formative Data Gathering

Formative data gathering is part of an iterative development of prototypes to check that the components of an artefact meet the needs of users and to guide towards changes. Why does a component of the artefact work or why does it not work? Evaluation should occur throughout the entire development to achieve a "guided evolution" of an artefact (Nickerson and Landauer, 1997).

In order to reach awareness/identification of the user group and the problem space, and an initial suggestion of problem solutions, questionnaires were used. This meant that more informed design suggestions could be made. In the development iteration cycles, the development and evaluation of artefacts, heuristic evaluation was used to fine-tune of the prototypes.

4.3.1 Questionnaires

Questionnaires is a quantitative research method that allows a researcher to collect demographic data about the user group's opinions (Preece et al., 2007). The questions used for a questionnaire can either be open-ended or closed. Open-ended questions allow the test subjects to answer freely, while closed questions have a series of options to answer the question. According to Bryman (2012), closed questions are used more often in survey research, because they are easier to answer and easier to decode for the researcher. A questionnaire usually starts with simple questions with demographic information, like age, gender and topic proficiency, then goes on to specific questions related to the research topic. This allows the researcher to understand the context of the test subject's answers (Preece et al., 2007).

For the purpose of understanding what content Serienytt's readers are looking for on a mobile platform, a questionnaire was created. This questionnaire together with the collection of previous guidelines for mobile content, and, personal observations of Serienytt's current services was used to analyse what specifications should be set for preliminary designs of the prototypes. The data collection gave an initial hypothesis for the design elements and functionality that were required for the applications.

4.3.2 Usability Inspection and Heuristic Evaluation

Usability inspection is a method that uses evaluators who have the experience and knowledge to analyse a user interface. Through inspection of the artefact prototypes, the evaluators will evaluate the usability of an interface as well as specify the severity of the issues (Nielsen, 1994). Usability inspection does not involve real users, and is therefore also known as expert evaluation. Usability inspection is often easier and cheaper to execute than full-scale user testing, but real users' input is invaluable for a finished product (Preece et al., 2007). Therefore, in this study, usability inspection was used for formative evaluation of the prototypes, while user testing was used for the summative evaluation.

At the end of the first two iterations of the prototypes there will be a heuristic evaluation. The experts were selected among students with education within the subject of Human-Computer Interaction. These evaluations had three core goals: a) to discover unclear or buggy interaction, b) to ensure that the prototypes were of equal quality and c) to inquire the experts for new ideas for functionality and designs.

During the search for usability issues, the experts were left to their own devices, but for technical issues or questions, the researcher was present. Observation of the experts interaction with the prototypes could be interesting, but it is important to not distract them (Nielsen, 1994). Part of the heuristic evaluation included mock tasks, as per Nielsen's (1994) suggestion, in order to ensure thorough evaluation. It also suggested that each interface is looked through twice, so the experts were advised to do so.

After usability issues were found, they were ranked according to severity. This was carried out through a discussion between the researcher and experts and focused on analysing three principles. The *frequency* that the usability issue occurs, in terms of how

often the user will experience it. The *impact* the issue has. Is it something the users can easily overcome? How *persistent* or how often the issue appears for the user. Can it be avoided when the user knows the issue or will it persist throughout the product's lifetime? With these principles the issues will then be ranked between 0 and 4.

- 0: No issue.
- 1: Cosmetic problems
- 2: Minor usability issue
- 3: Major usability issue
- 4: Usability catastrophe

Usability catastrophes are important enough to delay release of the product, while major issues should be fixed before the next iteration. Minor issues are relevant once the all major issues are fixed, and cosmetic issues only need fixing if there is spare time on the project (Nielsen, 2000).

An important part of heuristic evaluation is the choice of heuristics for testing. The core heuristics chosen for this test were Jakob Nielsen's ten heuristics (Nielsen, 1994), introduced in chapter 3.2.1. See appendix A for the full heuristic evaluation form. These heuristics are general heuristics that are relevant for almost all human-computer-interfaces. To supplement Nielsen's ten heuristics, more mobile-specific heuristics were added, to ensure that the experts were looking at all possible usability issues. These additional heuristics are based upon earlier research in the field of heuristics and mobile heuristics.

Nielsen and Budiu (2009) claim that there are four main usability hurdles for mobile users. One of these hurdles explains that screen size is an important issue for mobile developers. The smaller screen of mobile phones leads to fewer visible elements at all times, thus requiring that users use short-term memory to build an understanding of the information flow. Kaikkonen and Roto (2003) also suggest similar designs from their study. They place extra attention on the fact that information should be structured so the most important elements grab the users' attention. For niche news, this means that

the actual news contents should be clear and invites further reading. This led to a new heuristic being added to Nielsen's original 10 heuristics:

- a) Small screens: Mobile screens have small screens, so it is important to remove unnecessary information. Is there information from the original mobile site that should be removed? Or is there information that should be added to the prototypes?

Heo et.al. (2009) have a similar goal in their list of usability factors: *Functional support of user needs*. This is related to the functionality and the way elements can be manipulated to suit the user needs (Heo et al., 2009).

A different usability hurdle, as defined by Nielsen and Budiu, explains that it is hard to operate an interface without a mouse, so interaction takes longer time and are more error-prone. Text-entry is a particularly slow process. Touch input can lead to errors in interaction and how the generally slow interaction is further delayed by small touchable elements. Thus, two additional heuristics were added based upon this hurdle:

- b) Low misclick-ratio: Are the clickable elements easy to under? Are the elements large enough to easily be tapped by the user? Do the clickable elements sometime register during other gestures than tapping/double tapping?
- c) Simple and clear design: Is it clear how the interaction should be done and where the interactions will lead the users? Is the language used in the application easy to understand? Is the structure of the content logical for the user?

These points are further supported by Park et.al. (2011) and their usability principles like *informativeness* – user interfaces should be easy to understand and *visibility* – the information should be visible and clear to a user.

From Park et.al.'s (2011) list of usability principles, two additional heuristics were selected:

- d) Personalisation: Will a user be pleased with the level of customisation in the applications? Should the users have greater control in how the application can be used?

- e) Mobile Conventions: Does the applications follow the current conventions for touch phones and is the interaction consistent throughout the applications? Is there delay in the use of the applications?

Personalisation is based on the *adaptability* principle from Park et.al. (2011). Heo et.al. (2009) also claim that a lack of functions and styles has a negative impact on usability.

Mobile conventions can also be seen in Park et.al. (2011), in terms of *familiarity* and *predictability* – the interaction should be based on previous experiences and be consistent throughout the application. Similar traits are seen in Nielsen and Budiu (2009) whom state that it is generally a good idea to conform with the platform conventions.

Finally, a sixth heuristic was added based on Heo et al. (2009). Heo et al. (2009) suggest a heuristic on *ergonomics*. This heuristic is a measurement to ensure that the physical manipulation of the applications is understood by touch phone users:

- f) Ergonomics: Are the gestures practical and easily understood?

4.4 Summative Data Gathering

Summative data gathering is the evaluation of a project, in this case the final evaluation of the prototypes created in this study. A two-part summative evaluation, with user testing and observation followed by qualitative interviews, was used in this study.

4.4.1 Selecting Participants

The target audience of the mobile versions of Serienytt is all of their readers, so it was chosen to test with the actual user group. These are people, at all ages that are passionately interested in TV-series and news revolving around TV-series.

Regarding the number of participants in the summative evaluation, Nielsen (1994) and Preece, Sharp and Rogers (2007) suggest between 5-12 participants. After testing with five

users there is a lower chance of finding new usability issues. The goal of the evaluation was therefore at least five participants, whom are readers of Serienytt. Anyone from this user group would be a valid test participant, as long as they: 1) have experience with smart phones, 2) knows of Serienytt's content and 3) are 16 or above years old. The reasoning behind these criteria is to understand the interaction of people who know what they are searching for on Serienytt, they know how to search for content on a mobile, and are old enough to sign a consent form without parental guidance⁷.

4.4.2 User Testing

User Testing was done through observation with the think-aloud method. By observing how a user interacts with an artefact, it is possible to understand the users' context, goals and tasks. It is also possible to use observation as a type of evaluation, to see if the prototype can support these goals and tasks. The think-aloud technique can be used in observations, to understand what a participant thinks while interacting with a prototype. Plain observation might show tendencies in the interactions, but will not allow a researcher to understand the thought processes behind those actions. Think-aloud therefore asks the participants to orally explain their actions. This can be an obtrusive method, but if a controlled environment is used, then the participants are less likely to be disturbed by questions asked by the researcher. Through observation and think-aloud it is possible to understand a user's experience with a specific design element (Preece et al., 2007).

The user test was carried out as the final iteration, and serves as the final data gathering-session for the study. The reasoning for choosing observation as the final evaluation is that this is the last chance to test on the actual target audience. In a perfect scenario the target audience would have been included in more stages of the development, possibly by including more iterations of the prototypes. In addition, there was a shortage of possible test participants from the user group, which hindered more user testing and the time constraint of a masters project.

The context of the testing is an important part of the mobile experience. Mobile phone browsing is done almost everywhere, both in transit and stationary. It could be interesting to see the interaction in an actual real-life situation, like on the bus or at home, but this is

⁷http://www.nsd.uib.no/personvern/om/vanlige_sporsmal.html

not the main goal of this study. Testing was therefore done in a controlled environment, where the data gathering through video would be easier and of better quality. Since all of the testing was done under the same conditions the testing sessions were comparable to one another. Testing with a guide helps to systematise the data obtained from the user testing. This structure is called structured or systematic observation. A schedule of a predetermined period of time and actions allows an observation to be aggregated into results, since all of the participants followed the same routine (Bryman, 2008). The user testing will therefore be done with an observation guide. This guide will include free exploration, specific tasks and suggested questions as guidelines to ensure that a user tries the features of the prototypes.

4.4.3 Qualitative Interviews

While observation is critical to understanding a user experience, it can also be valuable to interview the users of a product. To fully understand the user experience it is necessary to ask the users about their experience and what parts of the user experience they like or dislike (Kuniavsky, 2003). Qualitative interviews will therefore supplement the observation and think-aloud sessions, so that a further understanding of a user's feelings towards a design element could be assessed.

In qualitative interviewing it is important to act as a neutral interviewer. Kuniavsky (2003) describes that an interviewer must: *"As the person writing and asking the questions in a non-directed interview, your job is to step outside everything you know and feel about your product"* (Kuniavsky, 2003, pg. 119). Acting completely neutral is an important part of interviewing, since a biased question could skew the answers to a question in a certain direction (Kuniavsky, 2003). Some questions were made as suggestions before the qualitative interviewing, but the interviews were to be based mainly on gaining an understanding of the situations and thoughts that came up during prototype testing.

4.4.4 Handling the Data

The data from the summative evaluation was video and audio recordings of the testing. To avoid users acting unnatural during the testing of the prototypes only the hands of the

participants and the device were filmed. Heath et.al. (2010) claims that there is a chance that this could still affect how the users act during testing, so it is therefore necessary to explain to the participants what type of information was gathered from the user testing and that the data would become anonymous. This leads to the effects of a camera's presence to be reduced. Therefore, only the hands-on interaction with the prototypes was recorded, in addition to the audio conversations. These recordings were then used as the main data source for analysing the final evaluation of the prototypes.

Analysis of the video content was done through two methods: 1) watching the recordings several times, and writing down general trends; and, 2) transcription of the interviews and question sessions of the user testing. Parts of the free exploration and prototype tasks were also transcribed, if the think-aloud revealed important information or insight related to the research questions.

4.5 Data Analysis

The tests and evaluations done during the course of the project produces data in different forms. This data needs to be analysed accordingly. Quantitative data, like the questionnaires, are analysed through the numbers and general trends that occur in the answers.

Qualitative data, gathered through heuristic evaluation and user testing, however, is not as simple to analyse. Preece, Sharp and Rogers (2007) write that identifying recurring patterns or themes is one possible type of qualitative analysis. In heuristic evaluation these patterns can be observed if one or more of the experts observe the same usability issue or related issues. If a problem was observed during the one of the prototype's heuristic evaluation, then usability problem could also be relevant for the other prototypes. The written notes from the heuristic evaluation can also be naturally categorised, as the experts write the usability issues into several heuristics. Looking for critical incidents, the last type stated by Preece, Sharp and Rogers (2007), could appear during testing and would be recorded. Think-aloud gives an additional advantage for this purpose, as it is possible to discern why the critical incidents occur through conversation.

4.5.1 Usability Analysis

Usability analysis has one specific goal – *“to find the real problems with the product – and with the process that was used to develop the product”* (Dumas and Redish, 1999, pg. 310). Which issues that are noticed in the prototypes will cause difficulties when the actual users try the application after release? (Dumas and Redish, 1999).

An additional tip that Dumas and Redish (1999) give for analysing usability problems, is to seriously approach any problem that any participant had during the user testing. Usability testing is a qualitative method, and therefore the problems discovered by a single user may be part of a larger subset of actual users. Thus, the problems found during a usability test should be organised by scope and severity (Dumas and Redish, 1999).

4.5.2 Validity

Validity, as defined by Golafshani (2003), is a measurement of how truthful research results actually are. Does the data actually measure what it is meant to measure? In short, is the research data accurate to answer the research problem?

Quantitative data are generally considered valid, depending on its internal and external validity. Internal validity describes the conditions of the research and if the research was done in a controlled environment. External validity is the measurement of how much the study can be generalised and if the results are valid in real life conditions. Some researchers have, however, argued that qualitative data cannot be considered valid data, based on the validity-term defined in quantitative research. Qualitative method is always seen through the eyes of a researcher, so there is no guarantee that the researcher really seeing what they think they are seeing. To meet these ends, some researchers redefined the criteria for qualitative research (Golafshani, 2003).

A set of criteria was made by Lincoln and Guba (1985). They felt the criteria of internal validity, external validity, reliability and objectivity were too positivist, and therefore established new criteria for qualitative data and its trustworthiness:

- **Credibility:** Is the data and the method used to gather the data believable?
- **Transferability:** Is the data applicable for other contexts?
- **Dependability:** Would the results of the data be the same with similar participants and a similar context?
- **Confirmability:** Are the findings neutral and the researcher unbiased?

The meaning behind these criteria is generally the same as its quantitative counterparts. Discussions about issues like validity and trustworthiness are used to create rigorous research, which in turn makes the research defensible. Following the criteria set by Lincoln and Guba (1985), ensured that the research in this study is rigorous, and the answers to the research questions of this study is relevant in the field of HCI.

4.6 Summary

This chapter has introduced the design research methodology. In addition, more specific methods such as Questions, Options and Criteria; heuristic evaluation; user testing and the validity of research have been described.

5 First Iteration - Design and Development

This chapter documents the design process and development process that led to the first edition of the prototypes. It describes carefully what choices were made during this development; how the specific test sessions are related to each prototype and the analysis of these test sessions.

To answer the research questions, defined in 1.2., it was necessary to: a) gain an understanding of the user groups needs, b) design how the content should be presented in the prototypes and c) test the designs.

5.1 Current service

The current service that Serienytt provides is niche news and articles about television series, both international and national. Serienytt's content is divided into several categories, such as news, reviews, trailers, recommendations and categories for each specific tv-series. This content is produced through WordPress, a widely popular content management system, based upon PHP and MySQL technologies. In addition to the article content, there is also some supporting functionality like related news, search the website, article discussion, and polls.

Serienytt currently presents their content as an adaptive web design, through media queries, that can be accessed through web browsers on mobile phones. As seen in figure 9, the adaptive version adapts to screen widths below 767 pixels, and changes from a two-column design to a single-column design. What is commonly named the sidebar, which is a smaller column in a two-column design, is placed below the content in the main column. It does not, however, meet the requirements for a responsive web design. The website does not use a fluid grid, nor are the images flexible. On all mobile phones the width of the page is 320px wide, which leads to unused space on the sides, if the phone has a screen width between 321px and 766px.

Every element of the desktop version is transferred to the mobile phones, which results in the sidebar being non-visible unless you scroll down on the mobile. Adding to the website

elements, the mobile version also has two additional main menus, which overlap with the regular menu. These menus have the exact same content, but are designed differently.



Figure 9: Serienytt's current desktop site and mobile site

5.2 Web survey

Serienytt had an established user group that could be used to gain insight into the preferences and needs of the user group. Obtaining an understanding of the user group meant that all stages of the QOC-process would be more relevant for the future iterations of the prototype, as the criteria could be based upon actual user feedback.

5.2.1 Web Survey Questions

The questionnaire had three core purposes: a) to profile the user base, b) to understand which functionality from the original website that they enjoyed, and c) to see if the user group was interested in mobile-specific applications. The questions were therefore directed towards this purpose, while avoiding leading questions. Reliable results were necessary for the entire project, as this would be the basis for the first iteration of the

prototype and the creation of their system requirements. In total there were 15 closed questions, with multiple options, and a final open-ended question that allowed the user to mention which functionality and design elements they thought were the most important. The reasoning behind only choosing to have 16 questions was to increase response rate. Bryman (2008) claims that short questionnaires tend to get higher response rates. The questionnaire is presented in appendix B.

5.2.2 The Web Survey

As the target group was spread all over Norway, it was decided that a questionnaire presented as an internet survey was warranted. The web survey was hosted through Lime Survey⁸, a free open-source survey tool. After the survey was prepared, the survey was posted as a news article on Serienytt's front page, with a short text explaining what the survey was for, how it would be used in research, Serienytt's participation, and the length of the survey.

The survey was first posted on Serienytt.no's front page in the beginning of August 2012. Later on, there was also added a banner to the top of the page urging people to contribute to the web survey. After two weeks a total of 105 users participated and answered the survey, with 86 of the participants completing the entire survey. Ideally, there should have been more participants as only 105 of about 3000 readers answered the survey. This means about 3,5% of the unique readers answered the survey. However, as a sample used as a basis for further testing, this was acceptable.

5.2.3 Web Survey Results

The web survey provided some interesting data. 50% of the users had Android phones, while 43% had iPhones. This meant that a mobile application would have to work on both platforms, or two different applications would have to be made, as excluding almost half of the user group is not a practical solution. The users are also interested in a Serienytt-app, as 75% of the participants replied that they would be interested in a Serienytt-app.

⁸<http://www.limesurvey.org/>

Questions related to research question two, news presentation, showed that the current way people find news on Serienytt was through the front page, with 75% of the users claiming that they used this method. This is an example of how users scan the web page to see if there is something interesting, without going too deep into the content. 70% of the questionnaire participants also recognised that the news images affect if they click themselves into a news article.

Further interesting statistics include that 56% of the participants read comments, but do not write comments themselves, while 23% both write and read comments. 71% do not care about how much data traffic they use while browsing on the mobile phone and 65% claim they watch videos on their mobile phone. This means that the issue of bandwidth is not the most important for the readers of Serienytt or the current bandwidth used by the website is satisfactory.

5.2.4 System Requirements and Platform Choice

The survey showed that on several issues, the user group was split. As an example of this, the question of "What medium will the prototypes be made in?". Since 75% were interested in a Serienytt app, then the criteria for making an app is strong. 25% of the user group are however not interested in an app and Nielsen and Budiu (2009) show that a mobile-specific website can be enough in many cases. Since the user group is spread between Android and iPhone, and to exclude users was seen as unreasonable, it was decided to make a web app, a hybrid app and a mobile website.

By choosing to create a responsive web design, a web application and a hybrid application it became possible to evaluate the advantages and disadvantages of each medium, thus answering one of the research questions. The core reason behind the choice of these three mediums was that they are based upon web technologies, and therefore compatible with most mobile devices. This allowed testing with both Android-users and iPhone-users, to see the differences within the user groups. In addition, testing with three versions, it became possible to answer the second sub-question of the research question, see section 1.2., as it inquired about what functionality is expected of digital niche news. Because the prototypes could be used on several different screen sizes, it was important to make responsive designs for all prototypes. The web and hybrid applications were based upon

the *mobile first*-principle, to create cleaner and simpler interfaces. This meant that some of the content would be removed for the sake of readability. The mobile website was, however, not to follow mobile first, but rather a total conversion of the content as McGrane (2012) suggests. To see the entire list of initial system requirements see appendix C.

5.3 Designing the prototypes

This section explains the design decisions that led to the initial versions of the prototypes. The results from this web survey and existing design guidelines were used to create system requirements for the prototypes.

All of the system requirements were results of the QOC-method. Since there were to be three prototypes, then sometimes options were chosen that was rated lower based on the criteria.

5.3.1 Platform Choice

As briefly mentioned in section 5.2. three different prototypes were created to further develop Serienytt.no's mobile services. For the mobile website, a responsive web design was considered better than the current adaptive design, as it would be more flexible. This meant that the CSS-files currently linked to the website would be changed, while the rest of Serienytt's content would remain.

The web application framework chosen for the web and hybrid applications is also an important choice. There were two main competitors for this choice: JQuery Mobile and Sencha Touch. Both frameworks are available for free, can service highly customisable design interfaces and are compatible with PhoneGap. This allows both services to make both web applications and hybrid applications. In the end, Sencha Touch was chosen because of its focus on documentation and that it uses a model-view-controller (MVC) pattern as the basis for programming. A MVC pattern separates the job of different classes into clear and well-defined tasks, which in turn allows for more flexibility in the communication between classes.

5.3.2 Sketching Designs

The initial system requirements were at this stage quite general goals. More specific goals or requirements were to be added through the process of sketching, which meant that new ideas could surface and be evaluated against the responses from the web survey. All the prototypes, including the responsive web design, were sketched, even if the first prototype was going to be based on Serienytt's current design. This was to see if other designs would offer an improved user experience and to spawn ideas for designs for the other prototypes. Once several designs had been created for each prototype, some were chosen for a higher-fidelity digital sketching. These were more detailed sketches, which gave an overall feel to each to design. Figure 10 gives an illustration of the process, from paper sketch to digital sketch.

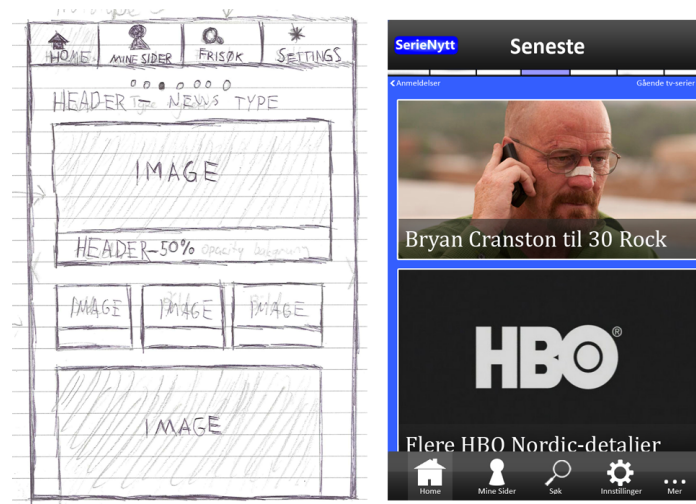


Figure 10: Prototype 3 - From paper to digital sketch.

Finally, a design was chosen for the second and third prototype through discussions with peers. These discussions also led to suggestions for improvements on each design, as a different point of view gave new opinions and ideas.

5.3.3 Prototype Description

The first prototype, a responsive website, was designed to be similar to contemporary news sites. The second prototype, the hybrid application, was decided to focus on keep-

ing the presentation clean and with as much information as possible. A minimal approach was decided to keep it as informational as possible, with a low data-traffic cost. The final prototype was based upon visual appreciation and that it would feel 'good' and 'exciting' to use the prototype. It had no restrictions on data use. By making three different prototypes, it was possible to query the user group from several angles. Even if 70% of the users do not care about data usage, then a prototype that still caters to the 30% that do could be made. Table 8 shows an overview of the aspects of the three prototypes.

Table 8: The aspects of the three prototypes

	Prototype 1	Prototype 2	Prototype 3
Medium	Responsive Web Design	Hybrid Application	Web Application
Data Use	Unrestricted	Restricted	Unrestricted
Front Page Customisation	None	Major customisation	Minor customisation
Main News Interface	Front page	Category tables	Carousel with news categories
Related Articles	Yes, many options	Yes, few options	No related articles

From the results of the questionnaire it was also decided that all of the prototypes would have a "front page", which allowed the users to get an overview of the news they enjoyed. Both the web app and the hybrid app, however, would let the user decide for themselves what type of news they get on the front page. Around 75% of all users claim that they are looking for news about specific television series, so the ability to "filter" what tv-series and article types that appeared on the front page, could make for a more pleasurable user experience.

The first prototype was, as mentioned, based upon the current design of Serienytt. Some small alterations were to be made to ensure that the adaptive web design turned into a responsive one, but otherwise the design would be kept. The reasoning behind this choice was to be able to compare the two alternative designs to something reminiscent of the current. It was to serve as the baseline for the other applications. None of the information was to be removed from adaptive version, to keep it as truthful to the original as possible. This included videos, images and Wordpress-widgets that the article authors had used. The widgets include for example a tab layouts and alternative text-boxes.

The second prototype was based upon a table-structure, which the user could fill with news categories. In each slot of the table, the user could enter a specific category from a pre-determined list of categories. These categories were based on Serienytt's current structure, and included, among others, latest news, specific tv-series (such as Game of Thrones or Breaking Bad), and specific news categories (such as reviews and new television series). This flexibility allows users with high competence to become more effective in their search for news. Issues could arise, however, as users with low competence could become confused and thus get errors. It was therefore important to create a user interface that clearly shows all the possible interactions (Goodwin, 1987). A low affordance of the elements that can be interacted with could lead to errors. In addition to the ability to customise the front page, it was also decided to add some additional utility-functions for users, such as the ability to turn images on/off. This allows users who want to spend a low amount of data traffic to do so.

Prototype three, the final prototype, is an interface that is based around a carousel. The carousel allows the user to swipe between different screens, which contain different types of news. The same categories that were available in prototype two was possible to add as a panel in the carousel with user interaction, but was presented as simply as possible, as advised by Goodwin (1987). This checked if a larger set of utilities will be preferred above a smaller utility set, which is easier to interact with.

Images have no restrictions in this prototype, and are a prominent feature of the applications. Large news images grab the attention of the users, as noted by Zillman (2001). Preece, Sharp and Rogers (2007) agree with this, as colour contrasts compared to other content and larger elements usually gets more attention. Normally, it is advised to be conscious about the data usage when developing for mobile phones (Nielsen and Budiu, 2009), but since 70% of the user group claim to not care about data traffic, this was tested in the prototype. This could lead to some minor delay in the system as the download of each news page will be noticeably slower.

5.4 First development cycle

The first edition of the prototypes was made with Sencha Touch 2 and CSS3 for prototype 2 and 3, and HTML and CSS3 for prototype 1. **Prototype 3** was the first to be made, as Sencha Touch has existing carousel and tab panel-components. This meant that the main interface components for this prototype were easy to implement. The front page of the third prototype is a simple single-column design, with the ability to swipe between news categories, as seen in figure 11. At the bottom of the interface, there are two tabs: the main view and the category selection screen. If the user taps one of these tabs, then the main screen will change the main view. The interaction can be seen at the bottom of figure 11, where the home-screen is selected. This minimal interaction, with high graphic fidelity, was the result of two core goals: 1) The interface had to be aesthetically pleasing, and 2) The interface must afford interaction.

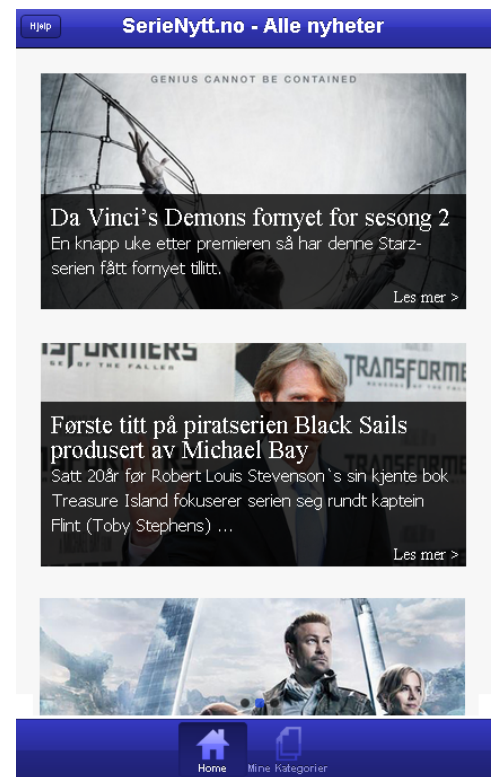


Figure 11: Prototype 3 - First version

The affordance of an interface is crucial for the success of a simple interface. Every news story must afford to be clicked, which is why every news story has a text field with "read more" written on it. This gives a clue about the interaction that is possible with a news story, as a written "read more" has become a perceived affordance in digital media. It therefore becomes easier to understand that the images can be interacted with (Norman, 1988). Ten news stories, all related to a single news category, were placed in a category. Images filled the entire width of the mobile phone, to maximise the visibility of the images (Preece et al., 2007). Headlines and excerpts were then overlaid on the images, to create a natural connection between the images and the content of the news story.

The only customisation that the user could interact with was a list of categories, which let the user turn categories on/off. An example of the category toggle can be seen in figure

12. This list was sorted into four categories: "all news", "main categories", "genres" and "tags", in that order. These are ordered by the perceived importance of Serienytt's editor. Within each of those four categories, the subcategories are sorted alphabetically. These categories are based upon the menu system of Serienytt.no's desktop version, so the readers who are used to their system do not have to learn a new one.



Figure 12: Category toggle featured in prototype 2 and 3

The category *All news* is a single category, which contains all the latest news. *Main categories* are all the categories defined in the current main menu, such as reviews, new series, trailers and recommendations. *Genres* are types of tv-series, such as action, comedy and so on, and *tags* are used for anything below that level, such as a specific tv-series. Each of these subcategories had a visual toggle showing if the category was currently on or off. As a technical constraint, it also limited the amount

of categories that could be chosen at any given time. This was because the application would have to load more content, and thus become slower in use.



Figure 13: First version of prototype 2's table layout

Prototype 2 had a table system, with six slots that could be filled with news categories. There were two columns of tables, with five news items per category, as seen in figure 13. Once a user clicked on a table, then they reached a larger table with more news stories within the same category. This can be seen in figure 14. The reasoning behind this choice was that each news story would have a smaller clickable area than 1x1 cm, which Nielsen and Budiu (2009) claim is a suitable surface area for touch input. Smaller elements are prone to errors in the interaction. The second prototype was made to give a broad overview of several news categories and with the small screen sizes of a mobile phone, it was decided that each news story itself should not be clickable.

The choice of categories was the same system as the one employed in prototype 3, but the second prototype also has a follow-up-screen. In addition to the category list, it also has a screen which lets the user place the news categories in a grid. This was originally meant to be a drag and drop-system, where the user could drag news categories from the list of chosen news categories into six preset tables. However, the plug-in that allowed for drag and drop-functionality did not work in Safari and Opera. One of the requirements for the web applications was that it would be compatible with most web browsers, so drag and drop-functionality was scrapped. Drag and drop was replaced with a self-made system, which let you click to select a category from the category-list, and then place it by clicking on the tables. It was expected that this system might create some confusion and errors, but a quick search for alternative drag and drop-libraries gave no suitable results.

There were small changes to **prototype 1** in this iteration. The main elements of the adaptive website were made responsive, and it also removed some of the current overlapping functions. The mobile-specific dropdown-menu, which had the same elements as the desktop dropdown-menu, was removed. Removing the mobile-specific menu resulted in more space for the news and no content became impossible to reach. Some optimisation was done to the desktop version, as it was currently not using the space available on the mobile screen. The optimisation can be seen in figure 15.

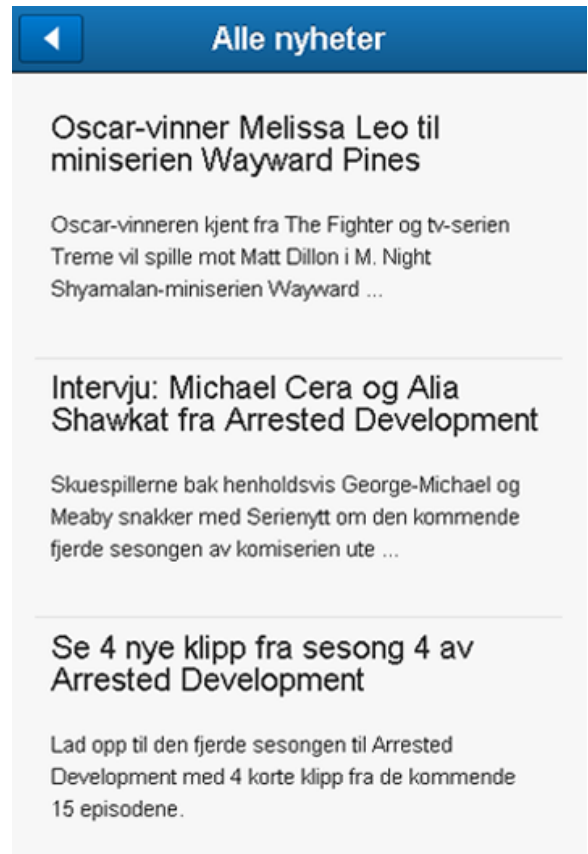


Figure 14: Second layer of prototype two's news story selection.

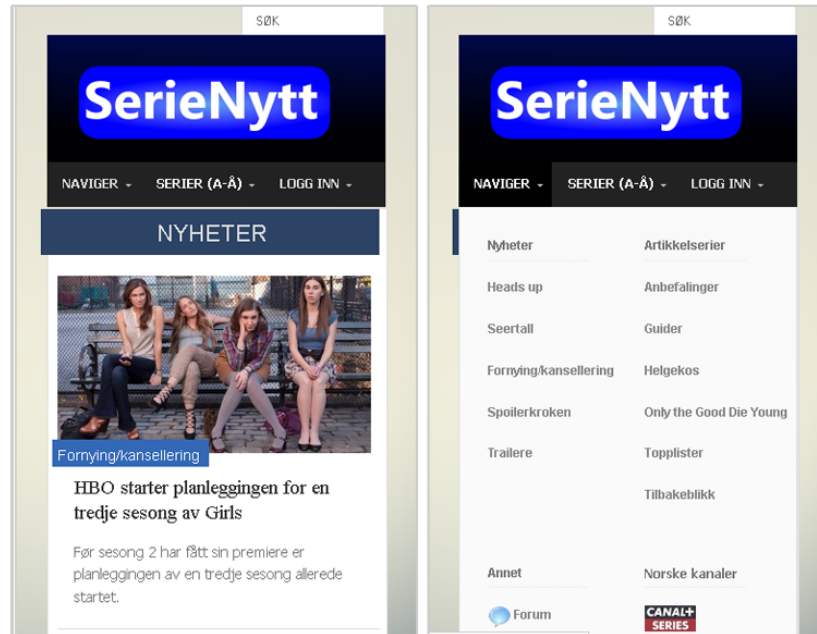


Figure 15: First version of the front page and menu of Prototype 1

The menu was changed to a double column-design, which used the space available. In addition to the menu changes, the current website also employs a slider for featured news. The slider is not optimised for mobile units and takes a long time to load on some mobile browsers. None of the content offered by this slider was unique and could be found in other locations on the front page. It was therefore also removed from the first prototype.

The first iteration's development ended as most of the initial system requirements were met.

6 Second Iteration - Tuning the prototypes

This iteration consisted of the first tuning of the prototypes. It began with an evaluation of the prototypes, then started a new design cycle with suggestion of problem solutions and development of artefacts.

6.1 Heuristic Evaluation

Heuristic evaluation was the next step in the process, to find out if the evaluation of implementation of the prototypes would reveal issues or lacking functionality. The options chosen in the QOC-sessions could also require re-evaluation. For this evaluation, it was chosen that prototype 1 would not be tested, but rather the original and adaptive website. The reasoning behind this choice was to get an affirmation that the usability experts agreed that changes were required for the website. If they did not agree with the features that had been altered in the first prototype, then maybe what was perceived by the researcher as a usability issue, was not an issue.

The heuristic evaluation was carried out, by four experts in a group setting, through the methods discussed in 4.3.2. Android experts were chosen as the evaluators, to see what mobile conventions they put emphasis on. At this point another heuristic evaluation was already planned with iPhone experts to see if they focused on different conventions. The experts sat down for 45 minutes to evaluate the prototypes, and a 15 minute discussion to rank the severity of the issues. The experts own phones were used for testing, as testing up to this point had only been done on a single phone. Testing on other phone models would show potential compatibility issues.

6.2 Evaluation Results

The results of the heuristic evaluation reaffirmed many of the changes that had been done to the adaptive website in the **first prototype**. Issues such as several menus with overlapping elements, as seen in figure 16, where the dropdown menu is overlapping the news

slider, the experts argued were usability catastrophes. Overlaps like these occurred in several places. They also noted that the adaptive website was not centred, which had also been fixed in the first iteration. A last catastrophe was found in *minimalistic design*, as the experts felt the article pages contained too much irrelevant information. In addition, they found several smaller problems regarding the *visibility of system status* and that it sometimes broke mobile phone *conventions*. An example of this is illustrated by the three different menus, which contain overlapping information, while others are completely unique to a single menu. They expected that a single menu is offered to the user, which contained all of the menu options.

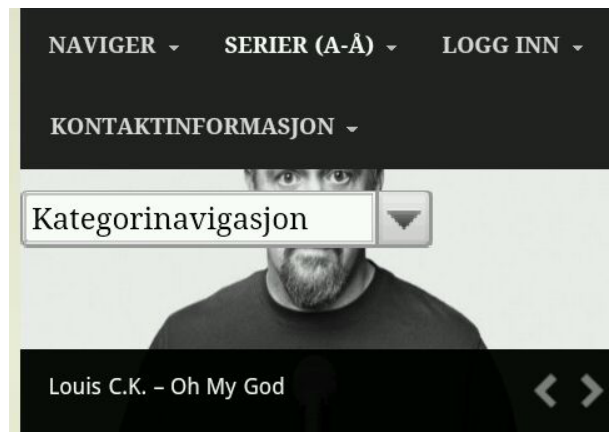


Figure 16: Example of overlap in Serienytt's adaptive website

Results of the heuristic evaluation showed that the choice of the dual-layered news system in the **second prototype** was confusing for the experts. When they pressed a news article on the original table, they expected to be directly sent to the news article, instead of the news category. This was categorised as a mismatch between the system and the real world. The experts, however, noted that this interaction was expected when they pressed the category header. Prototype 2 also suffered from the drag and drop-functionality being dropped, which confused the experts. Native app conventions had influenced them to expect drag and drop-interaction with the category boxes. They also noted that the category boxes were too small to accurately hit with touch interaction.

The experts also reacted to the news article interface, as the news articles had a two-column design. One of the columns was too small on a mobile portrait view, but was fine in landscape view. They suggested that this should either be changed to a single-column design or give the second column more space. Finally, the experts felt that the interface lacked the ability to zoom.

Prototype three was generally approved by the experts and had no usability catastrophes. The experts, however, found some bugs in the interface that needed to be fixed for error

prevention. Less important issues that were found included the swipe indicator being too small. This led to the swiping interaction being less obvious and the experts believed that less experienced users could miss the swiping interaction entirely. A connecting issue that was found was that it was unclear what happened when they selected categories on and off in the category list. The visibility that something happens on the main screen when interaction was done with the category list was considered low.

In addition to these issues specific to the prototypes, some of the experts felt the lack of the use of device buttons specific to Android in prototypes 2 and 3. Android phones have, for example, a back-button, which is not present on other devices. Instead of using the back-button that was added to the top-left corner of the designs, they felt that the use of the device-specific back button should also be added. When the back-button was pressed on the current prototype, the device exited the entire application.

6.3 Design Issues

The issues found in the heuristic evaluation were analysed with QOC to find possible options. Heuristic evaluation had already categorised the data, which meant that the questions that required analysis were already listed. For **prototype 1**, most of the questions were already answered. For example the overlapping menu had already been changed to a singular menu. The menu chosen was the regular website menu, because of the fact that it is known to the readers and a two-column menu design allowed for more of the menu options to be displayed at once.

The amount of information on each page was also ranked as a usability catastrophe in the heuristic evaluation. This meant that either: a) articles should only contain article-content; b) articles can contain some related information in addition to the article-content; or, c) the articles are fine as they are. In terms of usability, the experts meant that it was not user friendly to have large amount of information on a single page, which excludes option c. However, both of the other options were already being tested in the prototype 2 and 3. It was therefore chosen that the less desirable option c would remain in prototype 1, to see if the actual user group agreed with the experts. Keeping the design elements used by the regular website design allowed testing more elements with users, which was the goal of the study.

Prototype 2 was the prototype that the experts found the most issues with during testing. Interaction with news elements in tables, which was the highest ranked issue, had to be re-evaluated. The weight on different criteria that had been set in the first QOC-session was disagreed with by the experts, so the alternative option, which was to allow news stories to directly link to the news story, was chosen instead. This led to a new question: It is possible for each news story be accurately clicked, while still allowing for a larger overview of the news tables? The chosen option for this question was to give each news story a slightly larger clickable area. Apple Inc (2011a) claim that 44x44 pixels is the smallest clickable area for accurate touch input. This can result in smaller clickable elements than 1x1 cm. The clickable elements in prototype 2 was therefore chosen to be higher than Apple's advice of 44x44 points, but less than 1x1 cm (Nielsen and Budiu, 2009). This meant that fewer news stories could fit on the screen at any given time, but this was deemed less important than the usability of the interface.

To further support the visibility of the news in prototype 2, it was chosen to change to a different menu system than the previous iteration. This version received a menu button on the front page, which created a semi-transparent menu screen. All of the previous menu options were moved to this menu. The advantage of this menu is that changes to the front page can be seen instantly. If the user turns on news images, then there is an instant visual feedback because the front page now has images for the first news story in each category. This can be seen in figure 17, where news images has been turned on, and the image has become visible through the menu.

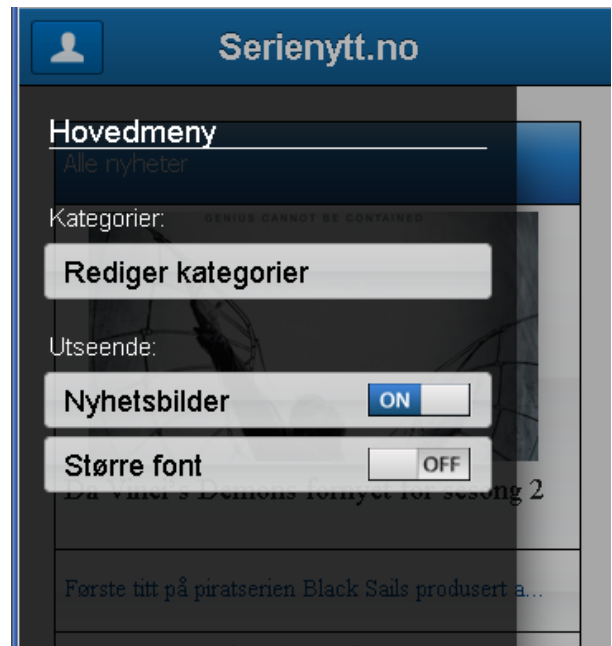


Figure 17: First version of the transparent menu

The news article-interface was also changed as a result of this session, as this was considered a usability issue. The original two-column design was altered to a single-column design, which placed the second column below the first. This was deemed a better solution

than giving the small column more space, because this would take attention away from the main news content. As the final large usability issue, drag and drop-functionality was discussed again. However, because of the lack of compatible libraries and the effort it would take to integrate a less compatible library, it was still chosen that the click-system would remain, but more emphasis was to be placed on clarifying the interaction with these buttons, through help-texts.

Prototype three, as mentioned, mainly had technical issues. Bug fixes did not really require to be analysed further, as they were already categorised in terms of importance. Another technical issue was that misclicks were occurring, because the system had issues with determining if the user was swiping between categories or if they were clicking on a news entry. Such issues just required a quick fix. In terms of more interaction-related issues, it was decided to enhance the visibility of the carousel indicators and the ability to swipe between categories.

In this systems development-process, it was decided that the **second prototype** was to receive Android-specific functionality, like the back-button to go backwards in the history and menu-button to open the menu. Phonegap, a web-api that wraps web applications into hybrid applications, was decided to be employed for the second prototype. By using Phonegap, it was possible to create code which can call platform-specific methods. This meant that the second prototype could overwrite the current action, closing the application, when the user pressed the device back-button. Instead it returned to the previous page the user was on within the application.

Android Developers (2012b) does however advise developer to stop making functionality for the menu button, because some of the devices made for the Android 4.0, or later version of the operating system, does not have dedicated menu buttons. Instead the best practices suggest that the functionality should be added to the "action bar", which works like device buttons on the top or bottom of the application. For this prototype however, final testing will be done on a phone with Android 2.3. In this version, the best practices still allowed the use of these buttons. The application however already has a menu bar with a menu button and a back-button. This menu bar has functionality similar to an action bar, so it was therefore decided to use the device buttons and the web app menu bar instead of the action bar. Figure 18 shows the similarity between these two interface elements. Both interfaces have tabs and menu buttons. The final decision

for this problem was therefore to let prototype 2 have functional device buttons, while device button functionality was not added to the third prototype. This meant that it was possible to see if the users would react if the device buttons do not respond.

The system development session also introduced many usability features, such as increasing font sizes and contextual help pages, to both prototype 2 and 3. Many of the usability issues classified as rank 2 or 3 in severity, which were found during expert evaluation, was categorised as "Help users recognise, diagnose, and recover from errors". Through alert notifications, with easily understood language, it became easier to understand what to do in the application and when something had gone wrong.

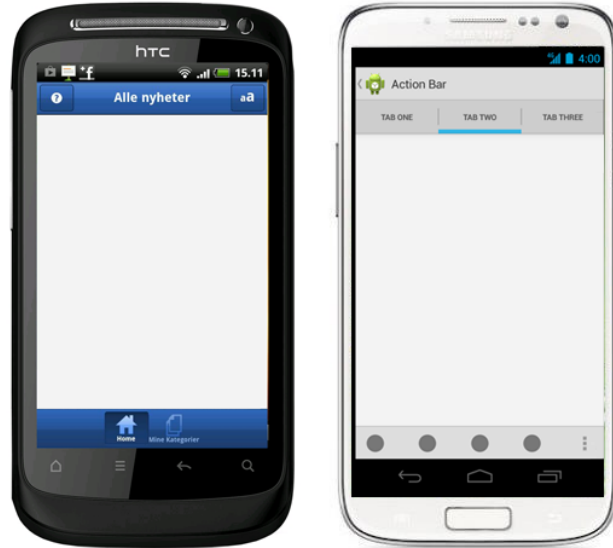


Figure 18: Example of web app menu bar with device buttons (left) and Android action bar without device buttons (right)

To make the interaction with the category-placement screen on prototype 2 clearer, it was decided to colour code the system. Each category was mapped with a specific colour, and when the category is selected, it becomes highlighted. In addition, when a category was placed into the placeholder tables, these tables got a written notification with the corresponding category. This meant that it became easier to see the location of each category and the fact that the category had been placed.

To enhance the visibility of the swiping functionality in prototype 3 it was chosen to do three countermeasures: 1) Each page within the carousel received arrows pointing towards the next page, if there was one; 2) A help-button was added which, if pressed, explained the swiping functionality; and, 3) Increase the size of the current indicator and give it a stronger colour contrast with the background. All of these measures point towards some horizontal interaction on the front page, which should result in high learnability and memorability of the functionality (Preece et al., 2007).

7 Third Iteration - Finalise the prototypes

The third iteration was the final iteration of the prototypes, and was aimed at finishing the prototypes. At this point, the core functionality of each prototype was already decided and a precursory evaluation of the designs had been done. Like the second iteration, the third iteration started with the evaluation of the issues and solutions found in the previous iteration. The iteration ended with three finished prototypes, ready for summative evaluation.

Development was done by going through the list of issues from the heuristic evaluation. The issues were fixed in order, ranked by the severity of the problems. Most of these issues had concrete solutions, like adding elements or simply fixing bugs, but other issues like the issue of platform-specific functionality had yet to get a specific solution.

7.1 Expert Evaluation

After the second system development-process, another heuristic evaluation with experts was done. The goal of this evaluation was to check if the changes that had been done to improve usability had worked, if there were still other issues that had yet to be found and to see if experts that normally use iPhones notice different usability problems.

The procedure for the expert evaluation was done similarly to the first heuristic evaluation. Three experts checked for usability issues, with one hour sessions. A change in this evaluation was that instead of group evaluation with discussion, it was done with experts one at a time. This meant that each of the experts could be queried more specifically about the interaction, but a larger discussion about the severity of the issues was dropped. In addition, there was a closer walkthrough of each heuristic. The first heuristic evaluation only found issues within some heuristics, while in other heuristics no issues were found. Through specifically going through each heuristic, it was ensured that the most important issues were found for each heuristic, instead of many issues for certain categories. The ranking of severity was done through discussions between the researcher and experts. If two or more experts found the same issue, then the severity rankings were later aggregated together to get a rating of the issues.

7.2 Evaluation Results

The second heuristic evaluation resulted in less bug-related usability issues than the first evaluation. Most of the additions that had been made between the first and second iteration was approved of and in general the issues were ranked with a lower severity. The issues were also spread over a wider range of heuristics, so the evaluation resulted in many new suggestions for improvements upon the second and third prototype. However, the experts found very few usability issues in the **first prototype**. The notes were mainly that the menu was not aesthetically pleasing, nor minimalistic design and that embedded videos should be responsive.

Prototype 2 still had some remnants of the issues that were found in the first heuristic evaluation. The interaction with the category placement still was not easily understood by users and the misclick-ratio of some elements was still too high. In addition to these notes, the experts suggested improvements such as "allow for more table templates, not just a two column design", "turn comments on/off and turn related stories on/off", and "when images is turned on, then every news story should get an image".

The **third prototype** got positive reactions from the experts, but there were still some performance issues. Slight loading delay occurred when experts went to the front page of the application. In addition, misclick-ratio issues were found, like some of the design elements designed to be interacted with were too small for users with large hands and the application sometimes got confused if the experts swiped slightly diagonally, not pure horizontal or vertical. Other notes from the experts were smaller such as, changing some icons, warning when the user has chosen no categories, and, changing the language used for some explanations.

7.3 Design Issues

For the **first prototype**, the issue of menus was ranked as the most important to be fixed. In the previous QOC-session, the option of a modified version of the desktop website's menu had been chosen, while the single dropdown-menu had been rejected. The criteria for this question was therefore redefined, based upon the comments from the experts

during heuristic evaluation. A dropdown system was chosen, because of the low misclick-ratio on dropdown menus and to create a simplistic design for the mobile phone system. The dropdown design uses the full width of the mobile screen to make each entry easy to tap, and is ranked by importance. Main categories are listed first, then channel-related categories and finally specific categories for tv-series. A section of the dropdown menu can be seen in figure 19. This choice resulted in a quite long list of categories, but the theory was that users were likely to find the content they were searching for by glancing through the menu options.

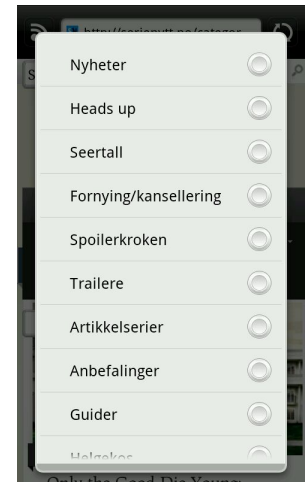


Figure 19: Example of dropdown menu

In addition, there was done changes to improve upon the optimisation of the headers, images and video on small-screen devices. Some remnants of the adaptive design still existed in the CSS files, so these had to be made fully responsive. Normally, videos on Serienytt.no were embedded in an "iframe" with a set width for the desktop and mobile versions. This was

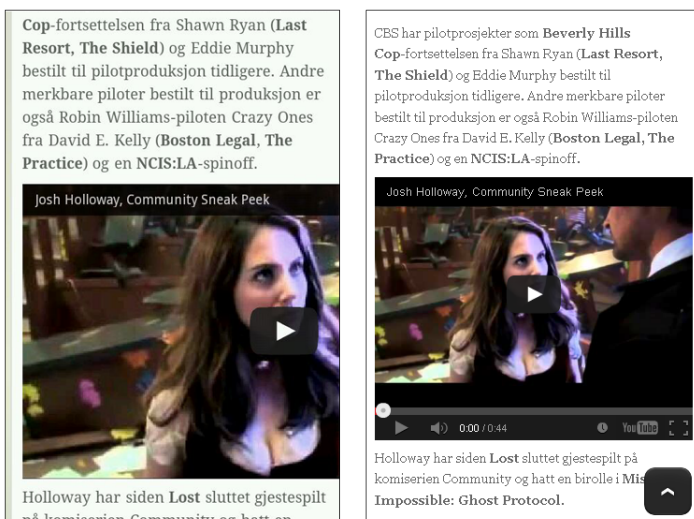


Figure 20: Article with video - the adaptive website (left) and responsive website (right)

changed to a responsive width and height, so it did not create horizontal scrolling on mobile phones. The effects of this can be seen in figure 20, where the adaptive website (left) does not adapt to the width of a video element, so parts of the video is placed outside the visible screen. The responsive version, however, has automatically changed the width of the video, thus allowing for easier interaction with the video player.

Similar results can be seen in other elements, such the first on the front page, as seen in figure 21. On the adaptive version, the first image goes beyond the full width of the

screen, while on the responsive version there is a small offset. This figure also highlights how prototype 1 overrides the browser's default look of a dropdown, while the adaptive version does not.

The changes to **prototype 2** were quite extensive in this iteration, to both fix the previous issues and increase the customisation of the application. First, all of the issues ranked 3 or higher were fixed, and then the suggestions from the experts were added. These changes resulted in major changes in the menu of the application, as several options were added. To lower the misclick ratio, thus lowering the amount of errors, and, to increase the visibility of the menu options, it was decided to increase the size of each menu option and to categorise them into three sub-genres: categories, appearance and help. Each menu item was now at least 1cm x 1cm large, as advised by Nielsen and Budiu's (2009) guidelines.

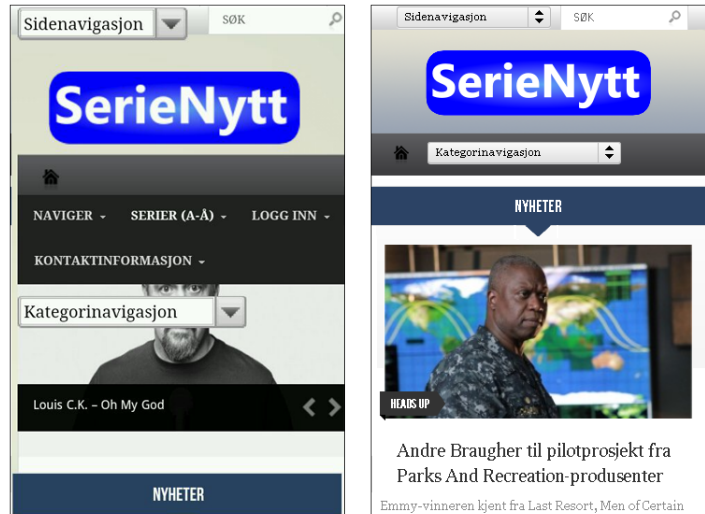


Figure 21: Front page - the adaptive website (left) and responsive website (right)

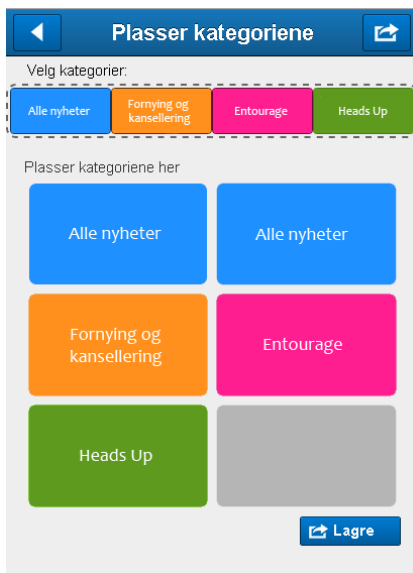


Figure 22: Category placement screen

For the clarification of the category placement-screen, it was decided to make the category tables become coloured when a category was placed in it. If a single category was placed in several tables, then both of these tables had the same colour. Through a system like this, the placement of each category became more easily understood. The next iteration was also decided to allow for a table to use the full width of the device. If a user places either the same category in the two tables next to each other or if one category is placed, then the application will join the two tables together into a single table. Figure 22 shows an example of this system in work, with the category "Alle nyheter" in both two tables, which will then join up and become one column

on the front page. "Heads Up" will also use the full width of the device, since nothing has been placed in the final grey square.

However, the experts disliked the addition of the "save"-buttons that had been added as a result of the first prototype iteration. This meant that there was some inconsistency in between the platform expectations between Android and iPhone-experts. The Android experts had in the previous iteration stated that a save-button should be added below both the category selection and the category placement-screen in prototype 2. iPhone users would however expect that when a category was set in a screen, then the application would not require that users tap a save button to store the information. They would just press the back-button and assume that the changes had already been made. This was an interesting divergence in the platform conventions, and an important question for an application that was meant to be used on both platforms. In the end, it was decided to keep the save-confirmation in prototype 2, while in prototype 3 it automatically stores the changes. However, since the save-interaction might be confusing for iPhone-users, it was decided to create a warning-screen if the user tried to press back without storing the information. This meant that the user had to confirm that they truly wished to return without saving the changes they had made.

The data from the heuristic evaluation of **prototype 3** pointed towards passable usability, and it seemed that the measures from the second iteration had worked as intended. However, the performance issues made the visibility of system status low, since the application sometimes has delay on button presses. This also affected the error prevention and efficiency of the system. The reason behind this problem was believed to be the processing power of the mobile phones, as only the older phones had noticed this issue. An analysis the application could require less processing power than it currently did. To improve performance, it was decided to decrease the amount of data in the application at any given time. In the previous iterations the application had loaded all the news in the categories chosen by the user, which meant a lot of data was cached. To reduce the delay, it was therefore chosen to only load the specific category which was selected. This meant that when users swipe between categories, there is less interaction delay in the system, but a slight loading delay instead. Information load was, however, easier for the system to notice, so a loading message could appear on the screen. The theory was that this would be less confusing for the users and result in fewer errors.

A less technical issue that was found by the experts was that one of the experts felt the headline and the lead paragraph took too much attention away from the images in prototype 3. However, the other experts did not take note of this. Still, this was considered as a usability question in QOC, to see if there would be negative side effects to altering the text elements for each news article. The options that were considered for this question was to either: a) lower the font-size of the header and lead paragraph; b) remove the lead paragraph from each news story; or, c) let the image be covered by the text. Lowering font size could have negative effects, as it would become harder to read the headlines, while removing the lead paragraph gives the user less information about the news article. The result of this analysis was to lower the font-size slightly, because the user has the ability to increase the font size by a single button-tap. This can be seen in figure 23, where the font size is smaller, but the user can still press the icon in the top right corner. Removing the lead paragraph could have worked, but this method was already tested by the second prototype, where only the headline remains on the front page.

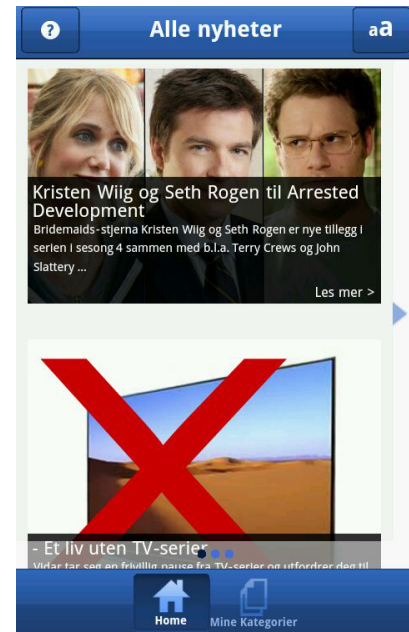


Figure 23: Front page of prototype 3 with smaller font sizes

In addition to these changes, the experts noted that the loading icon upon category selection was unclear, because they did not have time to read what it said before it disappeared. To fix this, it meant that either the loading had to be artificially lengthened or no loading symbol would appear at all. In the end, a compromise was made, where as the message would appear if the loading took more than 0.5 seconds. This meant that normally the loading icon would not appear at all, while when the system was acting slowly, it would appear to notify the user of the loading status.

The final versions of the prototypes can be seen in appendix D.

8 Prototype Evaluation

This section describes the user testing and qualitative interviews that was done as the final evaluation of the project. In Takeda et al.'s (1990) design cycle this was the evaluation of prototypes and the conclusion of the project.

The goal of this session was to compare the prototypes and the individual components that the prototypes were composed of, to answer what design elements and functionality was important for niche news. In addition, the user testing would give a confirmation on the choices made in the presentation medium.

8.1 User Testing

Recruitment for user testing was done through Serienytt.no, with an article asking for participation in user testing. After two weeks, five participants were found in Bergen.

The five participants for user testing were spread between the ages of 20 and 50 years old, with a broad spectre of mobile competence. Different patterns in the use of Serienytt's current products were also present in the user group, as some were new readers, while others were experienced. This variation was also reflected in their use of Serienytt's current mobile website, as some had never even browsed Serienytt.no on their mobile phones, while other participants read weekly on their mobile phones.

Each user tested all three prototypes. This was possible because of the relatively low amount of functionality in each prototype. A user could therefore easily compare the prototypes to each other and reflect upon which prototype they preferred. The problem with this choice was that the actual news stories remained the same on all prototypes, and it was therefore likely that they would ignore news stories that they had read on the other prototypes. Nielsen (1990) suggests that a user test should not last for longer than 60 minutes with a single user, because the participant might become tired. Each prototype was therefore tested for 15 minutes. Because of the small amount of information in each prototype, this was enough to get an understanding of the user experience. Another advantage of short test duration, which affected the choice of test duration, was that the

user would not have time to look at every news article in the first or second prototype. News articles that the user found interesting could therefore be unseen when the user reached the third prototype.

Testing of the prototypes was separated into three categories: Free exploration, tasks and follow-up questions. Free exploration was placed at the start of each prototype, to evaluate the learnability of the interfaces. The tasks were premade, and aimed to ensure that the user performed all the main tasks of the prototype. Some of the tasks were altered slightly in between users, such as adding a certain category to the front page. If a user had already tried this feature, then the task would be faster than if the user had yet to find the feature. Therefore a harder task was given, to see if the user had fully understood the feature. After the tasks a couple of follow-up-questions were asked to the user about the functionality, or additional thoughts about the prototype. The focus of these questions was on the expectations the participant had of the functionality. Some of the prototypes lack features that are part of the original website. This was to see which features were essential to niche news presentation on mobile phones. If they felt a feature was missing, then there is a high likelihood that it is a feature that should be added to the guidelines. This process would then be repeated for the other prototypes, with the same tasks, but with different questions at the end.

After the prototypes had been tested, a short qualitative interview followed. The questions for these sessions were both premade and based upon the observations that had occurred during the testing. Premade questions ensured that topics that were deemed relevant to the research questions, based upon the results of the questionnaires, related research and heuristic evaluations, were brought up during user testing. This served as a backup in case the topics felt unnatural to bring up during the prototype observation.

In addition, the interview session let the users reflect on their use of news applications on the mobile phone. As the observation was done under controlled conditions, there could be some problems that were not observed that could occur during everyday use. Comparison between the prototypes and other news applications was also added as a topic for these discussions, to get an additional idea of the functionality and user interfaces that the participants enjoy to use.

A short summary of the testing of each prototype can be seen below, in table 9.

Table 9: Summary of the results of user testing

Prototype 1:	
Task 1: Find all news in the category "Heads Up"	<ul style="list-style-type: none"> - 3 of 5 participants completed the task without mistakes. - 1 found the category after 2 misclicks - 1 unable to complete the task without help
Task 2: Find all news about the tv-series "Game of Thrones".	<ul style="list-style-type: none"> - 3 of 5 participants completed the task without mistakes. - 2 participants found the category after 1 misclick - 1 participant unable to complete the task without help
Task 3: Find and play a video from the trailers category.	<ul style="list-style-type: none"> - 5 of 5 participants were able to find and play video - 3 of 5 played in full-screen - 5 of 5 watched in landscape orientation
Question: Is the length of the pages a problem?	<ul style="list-style-type: none"> - 4 of 5 claims it is not a problem. 1 of 5 says it can be annoying.
Comments	<ul style="list-style-type: none"> - No one used the search field. - Browsing is done through watching the front page, and through related articles. - The two dropdown menus are confusing. - The main category menu is too long.

Prototype 2:	
Task 1: Find all news in the category "Heads Up"	<ul style="list-style-type: none"> - 5 of 5 participants had errors in adding Heads Up to the current front page. The most common issue was the lack of drag and drop-interaction. - 2 of 5 participants tried to press the back button once they had filled the tables, instead of pressing the save button. - 1 of 5 participants had troubles finding the edit screen.
Task 2: Find all news about the tv-series "Mad Men".	<ul style="list-style-type: none"> - 4 of 5 participants had no errors in the second try interacting with the edit-categories screen. - 1 of 5 participants tried pressing the back-button to get back to an earlier state, where he remembered Mad Men was active, before using the edit categories-screen.
Task 3: Can you find more news about Mad Men?	<ul style="list-style-type: none"> - 4 of 5 participants found the longer list of Mad Men news without issues, by tapping the category header. - 1 of 5 participants was unable to find the list. Instead of pressing the header, he looked through related articles and tried to fill all the tables in the edit category-screen with Mad Men.

Question: Would the ability to save your favourite news categories help you?	3 of 5 participants were positive to the ability to save their own custom front page, but 2 of these 3 thought the interface was too hard to interact with. 2 of 5 participants thought it was not worth the time to save their own preferences.
Comments	<ul style="list-style-type: none"> - Browsing was mainly done through clicking on specific news items, then looking at related articles. - Several participants found the design to be boring, but practical - It is positive that you can turn on and off images, because of data traffic. - 1 of 5 participants noticed a lack of a search field. - The responsiveness of images is useful. - No misclicks from small elements. - Most of the participants were negative to the fact that they were unable to comment on the page, but they could still read the comments.

Prototype 3:	
Task 1: Find all news in the category "Heads Up"	<ul style="list-style-type: none"> - 4 of 5 participants had no issues in finding news about Heads Up. - The last participant found the menu and added Heads Up, but did not find the ability to swipe. Required the help screen to find this functionality.
Task 2: Find all news about the tv-series "Homeland".	<ul style="list-style-type: none"> - All of the participants found the Homeland-news without errors.
Task 3: Find and play a video from the trailers category.	<ul style="list-style-type: none"> - No usability errors were found in this task. - 4 of 5 participants played the video in landscape with full-screen, while 1 played it in portrait.
Question: Do you react negatively on being sent to a different site to watch videos?	<ul style="list-style-type: none"> - 4 of 5 users did not react negatively to links instead of embedded videos. They do however agree that this makes them less likely to watch the video. - The last participant prefers embedded videos, but notes that it makes him feel more obligated to watch the videos.

Comments	<ul style="list-style-type: none">- This prototype was the most aesthetically beautiful.- Errors occurred because the application confused swipes with taps. However, there were no issues because of elements being too small.- Participants felt there was a lack of zoom functionality.- All of the Android users tried pressing the back-button and got confused when nothing happened.- None of the iPhone users pressed the devices back button.- 2 participants noticed that 'related articles' is missing.
----------	---

The results of the interviews were categorised and patterns in the answers were identified, as they are two of the methods Preece, Sharp and Rogers (2007) claim are valid analysis methods for user testing.

9 Discussion

This chapter discusses the analysis of the data collected through user testing and several observations made throughout the project are presented. As a reminder the user testing involved both free and task-based testing of the prototypes.

The chapter begins with a discussion of the user participant's preferences regarding various aspects of the three prototypes. Then each of the research questions are addressed. The chapter concludes by presenting a set of design guidelines for presenting niche news on mobile phones.

9.1 Prototype Preferences

This section discusses the prototype preferences that arose from the user testing and heuristic evaluation.

Browsing

Four out of five participants claimed that their main use of Serienytt was to browse the front page for interesting news, while one claimed to search for news about specific tv-series when she used Serienytt. During the free exploration of the observation, however, all five of the participants showed general purpose browsing instead of specific search browsing (Cove and Walsh, 1988). All of the participants browsed the front page of the prototypes, found an article they liked, and then went on to related stories. This loop could then go on for a couple of articles of related stories, before they went back to the front page for more stories. For prototype 3, which did not have any "related articles", however, resulted in the users spending more time on the front page.

In the free exploration, none of the participants used the search functionality available in prototype 1, nor did they note that the ability to search was missing from prototype 2 or 3. When asked if they feel that search functionality is necessary for niche news, all of the participants replied that a search function is useful, but they would rarely use it.

Orientation

The core browsing of all the test participants during testing was done in portrait orientation. Some of the users, however, sporadically changed to landscape mode to view images and video. Apple Inc. (2011a) claims that some tasks feel more “natural” in portrait orientations or the users “feel they can see more” in landscape orientation. The images and video, for which the users changed to landscape mode, were all wider than they were tall, so in this case they actually could see more of the content. Users also responded positively to the responsive width of images and video in the applications. This was especially true for prototype 2 and 3, which lacked zoom functionality. Three out of five users noted that the ability to zoom was important, even if they could increase the font-size and turn the device orientation for larger images.

Clickable elements and the sizes chosen for them resulted in very few “miss” clicks throughout all the prototypes, despite the lack of zoom. The “miss” clicks happened when the participants interacted with their thumbs. Apple Inc (2011b) claims that people tend to interact with mobile devices through three patterns:

1. In their non-dominant hand—or laying it on a surface—and gesturing with a finger of the dominant hand
2. In one hand, and gesturing with the thumb of the same hand
3. Between their hands, and gesturing with both thumbs

Interaction in pattern 2 or 3 resulted in a few misclicks on buttons that were smaller than 1x1 cm, while this was not a problem with pattern 1 interaction. Throughout testing all three patterns were used and swapped between. Pattern 1 was the most used, followed by pattern 3 and then pattern 2. Changes between these patterns could be a result the user noticing a misclick. If they get errors while using their thumbs, then they might naturally swap to a smaller finger. Nielsen and Budiu’s 1x1cm recommendation resulted in no “miss” clicks during any of the interaction patterns, while “miss” clicks occurred in interaction with clickable elements around 44x44 points, when the user was interacting with their thumbs.

Performance

There were, however, errors that occurred during the testing of the prototypes. Prototype 3 had issues with delay, which sometimes confused the users. The participants managed to quickly analyse the issue and discover that there was some delay themselves, so the confusion ended quickly. Three out of five users later explained that they did not react negatively to an application's slow feedback on occasion. One of them, however, stated that "This is a HTML5-app. So I assume that it is slow", while another assumed it was the Internet speed that had caused the slow interaction. The issue in performance of HTML5 applications is known among experienced users of mobile phones, so the expectations are different from a native application. This was most noticeable in prototype 3, with issues in execution time, which is generally assumed to hurt the user experience (Charland and Leroux, 2011).

Performance issues were also noticed in prototype 1, but not in execution time, rather related to latency. This is due to the web site loading more information than the two other applications. This leads to a slight delay during page change in prototype 1, but this is visually shown through the loading bar (Charland and Leroux, 2011). The usability experts in heuristic evaluation had claimed that the length of the pages was a usability issue, because of loading speeds and unnecessary elements. Users, however, claimed that the length of pages was a non-issue because the top of the page loaded in quickly enough for them to read the article before the bottom of the page had loaded.

Interface

In the free exploration of prototype 2, all of the users found the menu-button in the top-left corner and changed it to turn news images on. Without images the news felt 'boring' or 'non-attractive'. One of the participants noted that the ability to turn images on and off was 'cool' because sometimes you get slow internet speeds, in a high-mobility situation. The rest of the participants also agreed that the ability to toggle images was useful, but they would keep images on. The look of prototype 2's interface in general, was rated as less 'attractive' than the other two interfaces by four of the participants. It was also claimed that it looked 'less professional'. The last participant, however, liked this interface the most, because it was easy to read compared to the other prototypes. Prototype 3 was rated as visually more impressive than the other two prototypes, since it was more 'beautiful' and easier to interact with. In particular the ability to swipe between different categories was enjoyable. Four of five users found the swipe-functionality within the first

10-15 seconds of testing prototype 3, while the last did not find it until he pressed the help-button. Then he understood the functionality instantly. It was also noted that the ability to swipe between categories and the tab-panel made it "feel more like an app", compared to prototype 2.

Social functionality

Social functionalities, like the ability to comment and Facebook/Twitter-integration was missing from prototype 2 and 3, which some of the test participants claimed was a problem. The fact that you can read the comments, but you are unable to comment yourself is a negative. A mobile app should not disallow the user to comment on an article, just because of the platform. In addition, one of the participants felt that "Share on Facebook/Twitter"-buttons should be added to the prototypes. The participant also noted that it is important to have a link that sends the user to the article on the original website, so they can both read it there and use the comments section. On the other hand another participant claimed that being sent to the website to comment there instead of inside the app was just a "lazy solution".

It was, however, acceptable to annotate videos by clicking a link that sent the user to the video source for watching videos. In prototype 1, some of the users had issues with the size and loading of the embedded videos when they tried to get the video to fill the entire screen. In prototypes 2 and 3, which has the video annotated as links, they were quickly able to press the link and open the videos in YouTube's Android application, and get it full-screen from the new application.

Favourite Prototype

At the end of the user testing, the test participants were asked for which prototype they liked the most and why. Three answered prototype 3, because it was the easiest to use and it had very simple and clear interaction. However, two of them wished that the "All news'-category had more than 5 news stories. The other two test participants both picked prototype 1, because they felt it was the easiest to find information on, and it had a long front page with all the latest news.

9.2 Mobile Technologies and Niche News

This section addresses the research question: *Which technologies should be used to present niche news on modern smart phones?*

In chapter 2 the different technologies that could be chosen to present news content on mobile phones was discussed. The web survey and final evaluation of the prototypes showed that it is entirely possible to create a positive user experience as web content, web applications, or hybrid applications. However, in the final evaluation it was shown that users were impressed when they interacted with novel functionality.

In choosing a technology, there are several facets to consider, such as: the cost of development - in both time and money; the user experience; the availability of the end product; the marketing of the app; and, the possibilities to create creative solutions. These factors are highly varied depending on the choice, because of the restrictions set by the various technologies. Table 10 presents an analysis of each potential technology, according to these facets, based upon the results of this study. Native apps were not tested in this study, but from related research and the technological potential it is still possible to rate this against the other technologies. All the technologies satisfy each facet, but to varying degrees, as indicated by the number of check marks.

Technology \ Dimension	Responsive Web Site	Web Application	Hybrid Application	Native Application
User Experience	✓	✓ ✓	✓ ✓	✓ ✓ ✓
Development Cost	✓ ✓ ✓	✓ ✓ ✓	✓ ✓	✓
Availability	✓ ✓ ✓	✓ ✓	✓	✓
Distribution	✓ ✓	✓ ✓	✓ ✓ ✓	✓ ✓ ✓
Platform Freedom	✓ ✓	✓ ✓	✓ ✓ ✓	✓ ✓ ✓

Table 10: Rating of the possible choices in technology for niche news

The potential of **user experiences** is, as Charland and Leroux (2011) explains, largest on a native application. Since a native application can communicate directly with the operating system, through no additional layers, it will always have the most options. In addition, all of the platform conventions can be followed, since an app is designed for a

specific platform. Hybrid applications can also use some of a native applications functions, but performance issues will appear as the application becomes more advanced, while web applications have the same performance issues as hybrid applications, but can only emulate the native functionality. Web Content has even fewer functionality options.

For **development costs**, however, it is the opposite, as a responsive web design is the cheapest to produce. An existing website can easily be modified, through HTML, CSS3 media queries and JavaScript, to fit any mobile screen resolution. A web application is also cheaper to produce, but takes more time and effort to create. Web apps usually require to be built from the ground up, but because of the use of web technologies the code to a single web application works on most mobile phones. Hybrid applications require as much work as a web application, but more work is required if the hybrid app is to use native functions. These native functions will then only work for a single platform, which means more work has to be done to modify the content for every platform. The development cost of native applications is by far the worst, as a complete system has to be made for each platform. This results in a higher development cost and requires more time to produce.

Availability for the users is related to how many users a single application can reach. Native and hybrid applications are directly linked to a specific operating system, which means the user group is limited to that platform. Web applications will work on newer mobile phones, which have a compatible web browser. However, if the phone is using a web browser that is incompatible with the web application framework, then a different browser will have to be used. Finally, a responsive web site can work on any touch-screen mobile with a web browser.

In terms of **distribution** of the applications, then both native and hybrid applications can be marketed on the platform's app store. A regular web application and a responsive web site have to create their own distribution solutions. This gives a distinct advantage to the app stores, since they already store credit data. Simply pressing "buy" in an app store is enough to acquire a native app, while a web app it requires its own method payment method. **Platform freedom** and the amount of functionality that is possible on applications gives, as mentioned, an advantage to native application and hybrid apps. However, web applications are evolving and can emulate a native functionality, so this advantage may become smaller over time (Nielsen, 2012).

In summary, **Niche news** will, at the current time, have the best user experience on several native applications or hybrid applications. As shown by the user testing, the current platform conventions of Android phones and iPhones affect how the users interact with web applications, which can lead to some confusion. A RWD does not suffer from this problem, as the web browser itself imposes some own conventions. The problems of web applications are minor, and with some experience with the system, users can interact with high efficiency, safety and effectiveness. The question of platform choice therefore led to a few basic questions:

- *"Is the niche news site willing to pay for the best possible user experience?"*
 - If so, then native applications is the best choice.
- *"Is the niche news site on a low budget, but they still want a acceptable user experience?"*
 - If so, then a responsive web design is the best choice.

This is not to say that web apps and hybrid apps are bad solutions. They provide the middle ground in terms of cost and user experience. However, as the user testing showed, the design elements chosen for the product is at least as important, or of even greater importance. This is supported by Nielsen (2012) who claims that eventually mobile sites will be superior to native applications, but currently native applications give better user experiences. Nielsen notes that he is unsure when mobile sites will overtake native applications, but it will happen.

9.3 Design Elements and Niche News

This section addresses the research question: *Which design elements does the users prefer in niche news presentation?*

Images and layout

Heuristic evaluation and user testing gave insight into the design elements portrayed in the three prototypes. From the user testing, it became clear that the most important element of niche news is the front page and the news stories. Simplicity in presenting the latest news stories, with eye-catching images and interesting headers, is required for

positive user experiences. Zillman et al.'s (2001) suggestion, see chapter 3.6., that images attract attention to news headers and content seems to transfer to niche news. This importance of images was shown well by prototype 2, where all of the test participants included images. Prototype 1 and Prototype 3, both have a focus on large news images, which resulted in the users believing the designs looked better.

This can, however, not be contributed completely to the effects of images, as the layout and headlines are important aspects of communicating the content. To sum up the user experiences of the different layouts exhibited in this study, it was shown that:

- The current *desktop layout* with a single column for the most important news and two columns for older articles is generally preferred. Users have learned to use this interface through earlier experience and it gives them a suitable browsing experience.
- A *table layout* is considered organized and easy to read, but boring.
- The *image-focused and category-separated* layout of prototype 3 was visually pretty and the ability to swipe between categories is a welcome addition.

A *layout* designed for mobile niche news has to enable browsing behaviour and give easy-to-use options to explore several news articles. From the user testing, it would also seem that it is important to have a lengthy front page with several news items. The necessity of creating a separate interface for mobile phones, as described by Wroblemeski (2011) is dire. Desktop layouts do not fare well in terms of usability for mobile phones. The small space of mobile phones means the amount of news columns is limited, which affected the layouts that were tried in this study. The study tried three distinct news layouts, single column, mixed column, and dual column, as seen in figure 24:

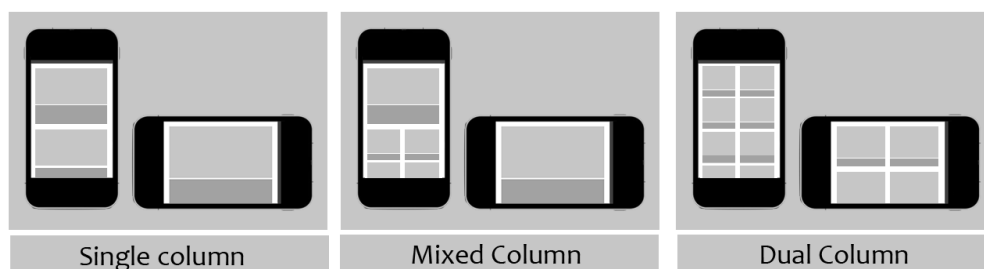


Figure 24: Options in column layouts that was tested

Among these, it was shown in the user testing that users appreciate full-width news images for the first news articles, but the size of each article becomes gradually less important as the user swipes down the page. This means that a niche news layout should either choose a single column or a mixed layout. The simplicity of these layouts is key to the interaction, and that it focuses the most important elements (Krug, 2005). From the user testing, it can also be assumed that users prefer to read news in a portrait format, so a layout should focus mainly on this view. However, it was also shown that responsive width of images and containers was positively received by the users.

Menus

Several different *menu layouts* have also been tested during the course of this study, as seen in table 25. The left phone indicates how the menu is placed when it is not interacted with, while the right phone indicates when the menu has been interacted with.

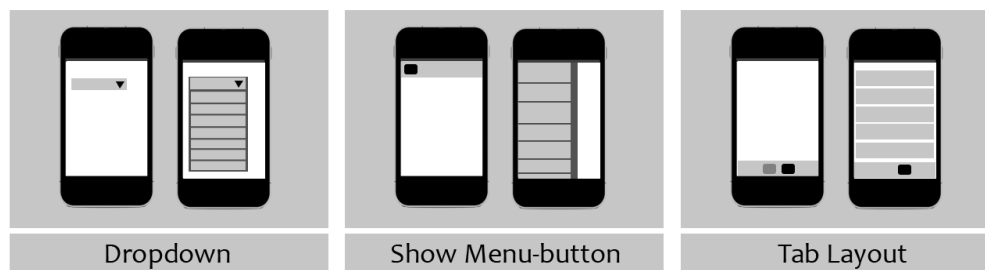


Figure 25: The three menu layouts that were tried in this study

Most of the users found prototype 1's dropdown menus confusing, because of the length and the fact that it is split into two. Prototype 2's menu also had very few usability errors, but the fact that the menu is hidden unless a user presses the menu button means it has lower visibility than a menu on the front page. This was evident by the fact that most of the test participants did not try to change the categories, and some never even pressed the menu button during free exploration of prototype 2, while for prototype 3 most of them did so. In addition, the low visibility of prototype 2's menu means it will have a lower memorability. Prototype 3's *tab layout* was well-received by the users, as the users claimed this was easy to use, compared to the menu system of prototype 2 (Preece et al., 2007).

The reason tabs were well-received could, however, be because of the limited amount of options on that system. This in itself makes the system easier to understand, and

the amount of options was the reason to why prototype 1's menu had lower usability. Nonetheless, some of the users requested the menu-system from the desktop website, which the heuristic experts had rejected. Serienytt.no's users have already learned a menu system from the desktop website, so they naturally sought categories in the same locations. From the user testing, it would seem that the most important features for a menu system are: 1) *ease of use*, 2) *high visibility*, and 3) *high memorability*.

Searching

Nielsen and Budiu (2009) claim *search functionality* is required for all mobile web sites, but it does not have to be in focus. In user testing of the three prototypes, however, no search behaviour was shown in the usability testing.

In the interviews, some of the users replied that the lack of a search field in prototype 2 and prototype 3 was negative. One user explained that they would not have used the search field even if one existed, because if they wanted to search they would use Google, while three participants claimed to only use the search field sporadically. The reason behind this ambiguity between the behaviour during user testing and their responses during interviews could be because the user testing situation elicited browsing behaviour. Niche news produces stagnant content, that remains relevant for weeks, months and possibly years. This means it is likely that some users will exhibit search behaviour, because in Serienytt's case, a user could become interested in a tv-series that started several years earlier and thus search Serienytt.no for their opinion on the series.

Social Functionality

Elements such as comments and social functionality are also important when presenting niche news. Serienytt.no is based upon a select group of authors sharing their opinions on tv-series and informing people of news relevant to those series. Connecting with other users' opinions is therefore a natural addition to this service (Allan, 2006). This was also the case in user testing, as three users claimed they were as likely to comment on a phone as they were on a desktop. The ability to share their own opinion about the articles is therefore important to enable on all platforms. However, as one of the users explained, the quality of a post from a mobile phone is likely to be lower than a desktop-written one, because of the awkward input of mobile phones. It could therefore be just as valuable to implement a 'Share'-functionality, which allows the user to simply share an article on Facebook or Twitter, while adding their own little comment. Social functionality on

mobile phones can therefore be said to be important, but if the editors want a comments section with high quality of posts, they can avoid implementing it, and rather support sharing.

Video

Serienytt.no is a natural platform for articles with video content, as the content revolves around the video medium. The interviews showed that videos are not central to mobile phone news, as the context of use is often in groups of people or while the users are in a high-mobility situation. This means that the users 'have to' use a headset or earplugs to listen to the sound, and they may not have the bandwidth to watch video.

In the web survey, however, around 65% claimed to see videos on the mobile, so the interviews may not be representative. A comparison between embedded or linked videos was done during the user testing and it showed that the users prefer linked content to embedded videos. The reasoning behind this, is that an embedded video is harder to interact with and more difficult to get on full-screen. Inline video will often be misaligned as the users swap between portrait and landscape modes. This results in fiddling around with the embedded video player, to try and hit the play-button, which was too small for the users to hit. Linked content does have cons, however, as linking a user away from your product can result in them browsing the linked site instead of the niche news-site.

For mobile phones, it is therefore better to link videos to services such as YouTube, which has an easy-to-use interface. In the future, however, mobile phones and embedded video players might get better usability, which could change the situation.

Suggested elements

The elements in niche news presentation should focus on *simple and clear interaction*. There are several options to reach a clear interaction, but some options seem favourable based upon the results of this study.

- **Layout:**

A *single-column* or *mixed-column* layout would be the best choice for niche news, so the first news stories get attention and are easily clickable.

- **Menus:**

The visibility of the options is key to the interaction. *Tabs* were the best option with

few elements, but it is not scalable to the large amount of options. A *"show menu-button"* is also a viable option, but it is important to have a clear hierarchy of the elements in this menu.

- **Searching:**

A *search field* is required, but it may be rarely used. Searching can therefore be less prominent in the design.

- **Social functionality:**

Ensure that users can *share* to popular social sites on mobile phones. If the site has a comment-system, then this should be included on mobile phones.

- **Video:**

Videos can be *embedded* if the video can be easily full-screened. If the embedded video player is not easy to interact with, then a **link** to the original source is preferable.

9.4 Functionality Beyond the Norm in Niche News

This section addresses the research question: *Does a niche news "app" require more functionality than news web sites?*

The difference in expectations between website and apps was shown throughout the process of this study. The experts suggested different types of functionality for the website and apps, and they advised based on specific platform conventions. This was also reflected in the user testing, as illustrated by a participant's explanation *"... an app does not require extra functionality. I'm more of a fan of apps containing less. It should be simpler, in every way. [...] A website is bulky. There is a lot of advertising, and too many bits and bobs."* This is a stark contrast to the website guidelines of W3C (2008) that mobile users should not be excluded from content. A simple and clean interface, that contains all the information of a niche news website, but still looks like *'an app'* is hard to produce, because of the lack of screen space. An alternative to adding all of this functionality in the app is to link the articles in an app up against the corresponding website article. This would allow an app to keep a clean and simple design, like the interface of prototype 3, while still giving the option to easily access the full content of the website.

While not required, the users explained that extra functionality was an asset. Storing news categories was approved of as implemented in prototype 3, because it was simple to use, while in prototype 2, the same ability to store news was disapproved of and noted as cumbersome. This highlights the necessity of a tight and lean interface. The two-stage approach of prototype 2 made people conscious about the interface, thus breaking Krug's (2005) 'Don't make me think'-rule.

Based upon current digital news outlets, it is unexpected to see functionality beyond simple vertical scrolling, hyperlinks and social functionality. It is therefore very important to have high visibility, a match between system and real world, and help and documentation of the functionality. A low value in these heuristics made the category-placement screen from prototype 2 confusing for the users. There were several causes for this confusion, of which, the disconnect between the layout in the category screen and the final front-page layout was the largest. Even if the categories were colour schemed, it was unclear what the users were actually doing and why they were doing it. Forcing the users to use new functionality was not an optimal solution. If the category placement-screen was optional, then prototype 2 might have resulted in a more positive user experience. The situation was also worsened by the implementation of a click-based system instead of drag and drop, which goes against platform conventions. There was little to no visible indicators that showed how to interact with this screen, and when users wanted documentation, there was little to find. In total, this led to a negative impression of the functionality.

The visibility, system and real world match, and help and documentation requirements were met for the swiping functionality of prototype 3, since it has high visibility because of the arrows pointing towards the next panel in the carousel. Swiping between pages, or in this case categories, might remind users of turning pages in newspapers or magazines. At the very least there is a match between the swiping motion of the finger and the content panning horizontally. Prototype 3 also has help and documentation readily available, if the user presses the contextual help button. In addition, the ability to swipe horizontally is part of the platform conventions for many touch phone operating systems. All of these factors together contribute to the ease-of-use of the functionality, which makes the user focus on the task instead of the technical solution (Nielsen and Molich, 1990).

Allowing users to toggle images, comments, and related articles on or off can also be seen as a valuable addition to current news presentation. In some situations, the data use of

a mobile phone is highly restricted, which means the ability to reduce the download is an advantage, as shown in the user testing. However, this is based upon the interviews where users were asked if they see the functionality as an advantage. In a real-life situation, users might not search or browse for niche news, when data use is restricted. During the interviews, some of the participants replied that they use Serienytt mainly at home or at work, which are fixed locations. From the results, it is therefore not possible to claim that the ability to restrict data usage is a necessary function, as the context makes the replies inconclusive.

Suggested functionality

The question of elements in niche news comes back to the discussion of: "*Should the entire content of a website be transferred to a mobile version?*". For a website version Wroblewski's (2011) **Mobile first**-mindset of focusing on the most important content, which in niche news is the news stories, is valid. From the user testing, it would, however, seem that all of the content should be placed in a responsive website version of a niche news site. The main elements must still be highly visible, like suggested by mobile first, but it should not eliminate functionality for the sake of simplicity.

This can, however, be said to not be the case for a Serienytt app. The expectations in functionality for apps is different from a website, which in turns affects the content. Altering or simplifying the content could be viable for an app, as users do not necessarily expect the same as they would from the website. A niche news app, based on user testing, is expected to have *a simple interface, news stories, and some form of social functions*. Functionality beyond this is optional.

9.5 Guidelines for Niche News on Mobile Phones

An analysis of the results of this project accumulates in a list of guidelines for niche news on mobile phones. These guidelines are based upon some of the guidelines suggested in the theory and related research, chapter 3, and an analysis of the data that has been done through surveys, expert evaluation and user testing. As there is coherence between the results of testing and earlier research, the guidelines should be valid for designs of niche news. Some of the guidelines are relevant to usability on mobile phones in general, while others are more specific to niche news as a genre. Table 11 and 12 summarises

the guidelines to be taken in consideration when developing a niche news interface for mobile phones.

Table 11: Guidelines for presenting niche news on mobile phones

Technology	
1	First create a responsive or adaptive web design of the desktop website. Do not create a web, hybrid or native app until the website adapts to mobile phone screen sizes.
2	Design for the portrait view first. Horizontal views are mainly used to magnify specific content.
Design*	
3	Ensure that all objects are clickable and readable without zooming.
4	A clickable element should be at least 0.75cm x 0.75cm wide, with enough space around it to avoid misclicks.
5	Allow users to return to the desktop version. Each article should have an option to read in the desktop version.
6	A search field is necessary. However, it does not have to be prominent.
7	Create layered menus instead of long menus. Ensure that the top-level items are descriptive of the sub-items.
Content	
8	Ensure that you have a front page with at least 15 news stories.
9	Use images for the most important news stories. Less important or older news can use small images or none at all.
10	The top news stories should use the full width of the screen. Later on, the content can split into more columns.
11	A news article-page should have related articles or other interaction that sparks interest about the subject matter.
12	Let the users easily share articles with others through social sites.
13	Videos that are important to the news story should be embedded in the article. Less relevant videos should be linked.

*The design guidelines have been rewritten from other design guidelines, mainly Nielsen and Budiu (2009). Guideline 4 was modified from Nielsen and Budiu's suggestion of 1cm x 1cm to 0.75cm x 0.75cm.

These guidelines in table 11 are relevant for responsive web design and apps in general. However, if the niche news site decides to create a native app, the additional guidelines from table 12 should be followed.

Table 12: App-specific guidelines for presenting niche news on mobile phones

App-specific	
14	Follow platform conventions. Web apps should imitate these conventions as well as possible
15	Avoid 'save'-buttons. Once a setting has been set, it should save automatically.
16	Tabs should be placed at the bottom of the screen. This ensures that users can easily see the changes on the main screen.

Through considering these guidelines when creating a mobile interface, the user experience should improve compared to the desktop interface.

9.6 Summary

This chapter has discussed the results of the user testing, discussed each of the research questions, and presented a suggestion for design guidelines for presenting niche news on mobile phones.

In *choosing a technology*, it has been shown that the developers must weigh the options, based on their specific requirements. A responsive web design is, however, the safest solution in most use-cases. The *elements* in niche news should focus on the core task, which is to present the news articles. It should also elicit social interaction between users. The expectations of niche news *functionality* is related to the technology chosen and its platform conventions. A mobile website should include all the content of the desktop version, while an app has less expectations.

10 Conclusion

The chapter is a summary of the thesis, discusses the research contribution and suggests future research.

10.1 Thesis Summary

This thesis describes a design process with the goal of understanding how to achieve a positive user experience for niche news on mobile phones. It also discusses which technologies one should choose to present this content. Through design research and user-centered design, Serienytt.no's content was presented on three prototypes that were tested and evaluated.

The study was carried out in three iterations comprising of design, user testing and analysis, followed by a final evaluation with real users of Serienytt.no. These iterations have been presented in a chronological order, and the thought process behind the choices made during development have been explained. Designing the initial prototypes based on earlier research and a web survey was the goal of the first iteration. The second iteration began with a heuristic evaluation, with Android experts. The iteration continued with analysis of the heuristic evaluation and a new design process focused on tuning the prototypes towards better usability. The third iteration also began with heuristic evaluation, but with iPhone experts. An analysis and a final design process followed this heuristic evaluation. The final evaluation with actual users comprised of user testing followed by qualitative interviews.

An analysis of the final evaluation led to sixteen design guidelines for presenting niche news on mobile phones.

10.2 Research Contribution

The research contribution of this study is mainly relevant to the research field of Human-Computer Interaction, and in particular for touch-screen usability research. Research on

touch-screen usability related to news content, or niche news, is scarce, so the methods and results of this study can be interesting to others carrying out related work.

The main results of the study is:

- A technical analysis of the advantages and disadvantages for choosing technologies (see section 9.2).
- An analysis of some of possible design elements and functions for niche news on mobile phones (see sections 9.3 and 9.4).
- 16 guidelines for delivering niche news content on mobile phones (see section 9.5).

Furthermore, the research resulted in:

- Three prototypes for Serienytt.no's content on mobile phones (see appendix D).

While niche news as a genre is the core focus of the thesis, some of these results are also relevant for other content genres. For example, the design-related and app-specific guidelines, which are relevant for mobile usability in general.

10.3 Limitations and Weaknesses in the Research

There are a number of limitations imposed on this study including:

- Time constraints (i.e., a Masters study over two semesters)
- Cost (i.e., a limited budget)
- Users for testing (i.e., a lack of willing test participants from Serienytt.no)

These limitations resulted in a number of weaknesses:

- User testing with real users was carried out on Android only

- Only a select number of user testing methods were employed
- Only a few participants in the final user testing

Further testing with different methods, such as diary studies, user testing in a mobile context or quantitative testing could give a different perspective than found in this study. Similarly, testing on a different phone and operating system could give a different perspective. Nevertheless, three prototypes were developed and tested, but these should be improved upon before being released to the public.

10.4 Development for Serienytt

For Serienytt, it would be most beneficial to create a new responsive design from the ground up. As a non-profit organization, with a moderately large user group, they should not fragment their content onto several platforms, as this would require more maintenance. This new design should focus on having a subtle menu and the placement of news should be easily visible, and content guidelines mentioned in section 9.5 should be followed.

An easier solution would be to further develop prototype 1, through fixing the issues found in this study.

10.4.1 Future Research on Mobile Usability and News

Mobile Usability in itself is a field that is constantly in development, as new devices continue to be developed. It could be interesting to see what possibilities several native apps could give to news presentation, to see if the advantages it gives are worth the cost of platform fragmentation. This could also be targeted towards larger news services, as they have more resources.

Another possibility is a more content-focused approach. This study is focused towards the design of niche news, but the content itself is an important element, if not the *most* important element. Thus a relevant question for future research is: *How does the difference*

between niche news presented as a weblog and news presented as an online newspaper affect the user experience.

The expectations between websites and apps is also an interesting field of study. *Se-rienytt's* readers assumed most of the desktop website's features would be transferred to a mobile website, but not to an app. It would therefore be interesting to see if this trend is persistent for all mobile content.

References

- Agarwal, R. and Venkatesh, V. (2002). Assessing a firm's web presence: A heuristic evaluation procedure for the measurement of usability. *Info. Sys. Research*, 13(2):168–186.
- AgileManifesto (2001). Principles behind the agile manifesto. <http://www.agilemanifesto.org>.
- Ahn, J.-w., Brusilovsky, P., Grady, J., He, D., and Syn, S. Y. (2007). Open user profiles for adaptive news systems: help or harm? In *Proceedings of the 16th international conference on World Wide Web, WWW '07*, pages 11–20, New York, NY, USA. ACM.
- Allan, S. (2006). *Online News: Journalism and the Internet*. Open University Press.
- AndroidDeveloper (2012a). Gestures. <http://developer.android.com/design/patterns/gestures.html>. [Online: accessed 17.03.2013].
- AndroidDeveloper (2012b). Say goodbye to the menu button. <http://android-developers.blogspot.no/2012/01/say-goodbye-to-menu-button.html>. [Online: accessed 17.03.2013].
- AndroidDeveloper (2012c). Supporting multiple screens. http://developer.android.com/guide/practices/screens_support.html. [Online: accessed 17.03.2013].
- AppleInc. (2011a). Platform characteristics. <http://developer.apple.com/library/ios/#documentation/userexperience/conceptual/mobilehig/Characteristics/Characteristics.html>. [Online: accessed 17.03.2013].
- AppleInc. (2011b). User experience guidelines. <http://developer.apple.com/library/ios/#documentation/userexperience/conceptual/mobilehig/UEBestPractices/UEBestPractices.html>. [Online: accessed 22.04.2013].
- Benbasat, I., Goldstein, D. K., and Mead, M. (1987). The case research strategy in studies of information systems. *MIS Quarterly*, 11(3):369–386.
- Bryant, J. and Jones, M. (2012). *Responsive Web Design*. Springer Science, United States.
- Bryman, A. (2008). *Social Research Methods*. Oxford University Press, United States, 3rd edition.

- CanIUse (2013). Can i use... compatibility tables for support of html5, css3, svg and more in desktop and mobile browsers. <http://caniuse.com/#agents=mobile&cats=CSS>. [Online: accessed 20.03.2013].
- Castledine, E., Eftos, M., and Wheeler, M. (2011). *Build Mobile: Websites and Apps for Smart Devices*. Site Point Ltd., Collingwood, Australia, 1st edition.
- Charland, A. and Leroux, B. (2011). Mobile application development: web vs. native. *Communications of the ACM*, Volume 54(Issue 5):49–53.
- Cove, J. and Walsh, B. (1988). Online text retrieval via browsing. *Information Processing and Management*, 24(1):31–37.
- Dumas, J. S. and Redish, J. C. (1999). *A practical guide to usability testing*. Intellect Books.
- Fling, D. (2009). *Mobile Design and Development: Practical concepts and techniques for creating mobile sites and web apps*. O'Reilly Media, United States, 1st edition.
- Forlizzi, J. and Battarbee, K. (2004). Understanding experience in interactive systems. In *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques*, DIS '04, pages 261–268, New York, NY, USA. ACM.
- Golafshani, N. (2003). Understanding reliability and validity in qualitative research. *The Qualitative Report*, 8(4):597–607.
- Goodwin, N. C. (1987). Functionality and usability. *Communications of the ACM*, 30(3):229–233.
- Gustafson, A. (2011). Adaptive webdesign – crafting rich experiences with progressive enhancement.
- Haverbeke, M. (2011). *Eloquent JavaScript: A Modern Introduction to Programming*. No Starch Press, United States.
- Heath, C., Hindmarsh, J., and Luff, P. (2010). *Video in Qualitative Research: Analysing Social Interaction in Everyday Life*. Sage Publications, London, United Kingdom.
- Heo, J., Ham, D.-H., Park, S., Song, C., and Yoon, W. C. (2009). A framework for evaluating the usability of mobile phones based on multi-level, hierarchical model of usability factors. *Interacting with Computers*, 21(4):263–275.

- Hevner, A. R., March, S. T., Park, J., and Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1):75–105.
- Hewett, T. T., Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., Perlman, G., Strong, G., and Verplank, W. (1992). *ACM SIGCHI curricula for human-computer interaction*. Association for Computing Machinery, New York.
- Jeong, W. and Han, H. J. (2012). Usability study on newspaper mobile websites. *OCLC Systems & Services*, 28(4):180–198.
- Kaikkonen, A. and Roto, V. (2003). Navigating in a mobile xhtml application. In *CHI 2003*, pages 329–336.
- Kramer, J., Noronha, S., and Vergo, J. (2000). A user-centered design approach to personalization. *Commun. ACM*, 43(8):44–48.
- Krug, S. (2005). *Don't Make Me Think: A Common Sense Approach to the Web (2nd Edition)*. New Riders Publishing, Thousand Oaks, CA, USA.
- Kuniavsky, M. (2003). *Observing the User Experience: A Practitioner's Guide for User Research*. Morgan Kaufmann.
- Landay, J. and Myers, B. (2001). Sketching interfaces: Toward more human interface design. *IEEE Computer*, 34(3):56–64.
- Lee, J. and Lai, K.-Y. (1991). What's in design rationale? *Human-Computer Interaction*, 6(3):251–280.
- Lincoln, Y. S. and Guba, E. G. (1985). *Naturalistic Inquiry*. Sage Publications.
- MacLean, A., Young, R. M., Bellotti, V. M. E., and Moran, T. P. (1991). Questions, options and criteria: Elements of design space analysis. *Human-Computer Interaction*, 6(3 and 4):251–280.
- Marcotte, E. (2010). Responsive web design. <http://alistapart.com/article/responsive-web-design>. [Online: accessed 10.04.2013].
- Marcotte, E. (2011). *Responsive Web Design*. A Book Apart, United States.
- McGrane, K. (2012). *Content strategy for mobile*. A Book Apart, New York.

- Nickerson, R. S. and Landauer, T. K. (1997). Human computer interaction: Background and issues. In Helander, M. G., Landauer, T. K., and Pribhu, P. V., editors, *Handbook of Human-Computer Interaction*. Elsevier Science, Amsterdam, Netherlands.
- Nielsen, J. (1994). Heuristic evaluation. In Mack, R. L. and Nielsen, J., editors, *Usability Inspection Methods*. John Wiley and Sons, New York, United States.
- Nielsen, J. (2000). Why you only need to test with 5 users. <http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users>. [Online: accessed 10.04.2013].
- Nielsen, J. (2012). Mobile sites vs. apps: The coming strategy shift. <http://www.nngroup.com/articles/mobile-sites-vs-apps-strategy-shift>. [Online: accessed 10.04.2013].
- Nielsen, J. and Budiu, R. (2009). Usability of mobile websites: 85 design guidelines for improving acces to web-based content and services through mobile devices.
- Nielsen, J. and Molich, R. (1990). Heuristic evaluation of user interfaces. In *CHI 1990*, pages 249–256.
- Norman, D. (1988). *The Design of Everyday Things*. Doubleday, New York, United States.
- Park, W., Han, S. H., Kang, S., Park, Y. S., and Chun, J. (2011). A factor combination approach to developing style guides for mobile phone user interface. *International Journal of Industrial Ergonomics*, 41(5):536–545.
- Park, Y. S. and Han, S. H. (2010). Touch key design for one-handed thumb interaction with a mobile phone: Effects of touch key size and touch key location. *International journal of industrial ergonomics*, 40(1):68–76.
- Pilgrim, M. (2010). *At home and with computer access: why and where people use cell phones to access the internet*. O'Reilly Media, United States.
- Powers, D. (2012). *Beginning CSS3*. Springer Science, United States.
- Preece, J., Sharp, H., and Rogers, Y. (2007). *Interaction Design - Beyond Human-Computer Interaction*. John Wiley and Sons Ltd, West Sussex, 2nd edition.
- Schmiedl, G., Seidl, M., and Temper, K. (2009). Mobile phone web browsing: a study on usage and usability of the mobile web. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI '09*, pages 70:1–70:2, New York, NY, USA. ACM.

- StatCounterGlobalStats (2013). Top 9 mobile browsers in norway from sept 2012 to feb 2013. http://gs.statcounter.com/#mobile_browser-NO-monthly-201209-201302. [Online: accessed 17.03.2013].
- Takeda, H., Veerkamp, P., Tomiyama, T., and Yoshikawa, H. (1990). Modeling design processes. *AI Magazine*, 11(4):37–48.
- Vaishnavi, V. and Kuechler, B. (2004). Design science research in information systems.
- W3C (2008). Mobile web best practices 1.0. <http://www.w3.org/TR/mobile-bp>. [Online: accessed 20.03.2013].
- W3C (2009). Html design principles. <http://dev.w3.org/html5/html-design-principles>. [Online: accessed 15.03.2013].
- W3C (2013). Css current work and how to participate. <http://www.w3.org/Style/CSS/current-work>. [Online: accessed 20.03.2013].
- Weaver, P. (2003). *Success in Your Project: A Guide to Student System Development Projects*. Financial Times, United Kingdom.
- Wroblewski, L. (2011). *Mobile First*. A Book Apart, New York.
- Zillman, D., Knobloch, S., and Yu, H.-s. (2001). Effects of photographs on the selective reading of news reports. *Media Psychology*, 3(4):301–324.

Appendix A Heuristic Schema

Nyheter for Mobil – Heuristisk Evaluering:

Nielsens 10 heurstikker (1994):

Visibility of system status

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

Match between system and the real world

The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

User control and freedom

Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

Consistency and standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

Error prevention

Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

Recognition rather than recall

Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Flexibility and efficiency of use

Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

Aesthetic and minimalist design

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Help users recognize, diagnose, and recover from errors

Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

Help and documentation

Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Underpunkter/tilleggsspørsmål:

- Enkelt og klart design: Er det tydelig hvordan applikasjonenes interaksjon skal gjennomføres og hvor de ulike knappene vil føre brukeren? Er overskrifter, veiledning og menyer enkle å forstå? Er innholdets struktur i samsvar med brukerens forventning? Kan de vanligste oppgavene til brukeren gjennomføres med mindre enn 3 klikk?
- Ergonomikk: Er gesturene man gjennomfører praktiske og enkelt forståelige?
- Personalization og effektivitet: Vil brukeren være fornøyd med nivået av personalisering i applikasjonene? Bør brukeren ha større kontroll over utseende til applikasjonen?
- Lav feilklikk-ratio: Er det enkelt å trykke på de trykkbare elementene? Er elementene store nok til at de blir enkle å klikke på og dermed ender ikke brukeren opp på steder de ikke ønsker.

- Mobilfunksjonalitet: Er det funksjonalitet eller informasjon fra det originale nettstedet (Serienytt.no) som mangler i applikasjonene? Er det informasjon eller funksjonalitet som er unødvendig for applikasjonene.

- Mobilkonvensjoner: Følger applikasjonen konvensjonene for touchtelefoner og er interaksjonen konsistent gjennom hele applikasjonen? Føles applikasjonene gode å bruke/er det trege?

Framgangsmetode:

1. Forklare heurestikkene og hvordan testen skal gjennomføres. Dette sendes på mail til ekspertene, slik at de kan forberede seg på forhånd.
2. La hver enkelt ekspert gå gjennom applikasjonene (med "mock tasks") og skrive ned problemene de finner. Hver enkelt ekspert får utdelt en egen smarttelefon og penn/papir. Dette er valgt for å korte ned tiden som trengs til å gjennomføre evalueringen, siden deltakerne har lite tid tilgjengelig.

Om det oppstår problemer med bruken av applikasjonen, så er jeg tilgjengelig for som teknisk ansvarlig. Det vil også vurderes om det er passende at jeg observerer ekspertene mens de evaluerer prototypene, eller om det vil være en distraksjon. I utgangspunktet, så skal det ikke være nødvendig å studere ekspertenes handling i en heuristisk evaluering, men det kan oppstå interessante observasjoner. Dermed vil jeg sitte og observere ekspertene, men vil ikke forstyrre dem. '(Nielsen, UseIt)

Samtidig vil jeg, om det er nødvendig, gi hint til ekspertene om hvordan applikasjonene er ment til å brukes og hvilke muligheter som finnes. Dette vil gjøre at testerne både får gått gjennom hele innholdet til applikasjonene, samt at man ikke bruker unødvendig tid på at testerne leter seg gjennom applikasjonen.

Note: De bør gå gjennom hver enkelt interface minst 2 ganger.

3. Diskusjon om hvilke problemer de fant, mulige løsninger og prioritert liste av hvor viktige problemene er å fikse. Den prioriterte listen blir basert på systemet:

Rate by severity based up three principles :

The **frequency** with which the problem occurs: Is it common or rare?

The **impact** of the problem if it occurs: Will it be easy or difficult for the users to overcome?

The **persistence** of the problem: Is it a one-time problem that users can overcome once they know about it or will users repeatedly be bothered by the problem?

Rating system:

0 = I don't agree that this is a usability problem at all

1 = Cosmetic problem only: need not be fixed unless extra time is available on project

2 = Minor usability problem: fixing this should be given low priority

3 = Major usability problem: important to fix, so should be given high priority

4 = Usability catastrophe: imperative to fix this before product can be released

Appendix B Questionnaire

Serienytt på Mobil

Brukerundersøkelse om Serienytts brukere og deres vaner på mobil.

Denne brukerundersøkelsen er et samarbeid mellom Serienytt.no og Bård Bachmann, masterstudent i Informasjonsvitenskap ved Universitetet i Bergen. I masteroppgaven så ønsker jeg se på deres vaner ved bruk av Serienytt og tjenesten på mobil.

Undersøkelsen består av 15 spørsmål i avkryssingsformat, med mulighet for egne kommentarer. Dette bør ikke ta mer en noen få minutter.

Deltagelsen på denne undersøkelsen vil anonymiseres. Statistikken fra svarene deres vil hjelpe oss med å forstå hvilken funksjonalitet dere ønsker/bruker på mobil,

På forhånd takk,
Bård Bachmann og Serienytt.no

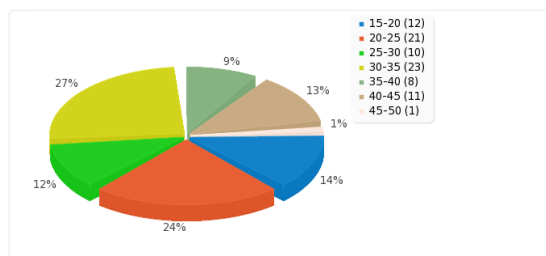
1. Personlig informasjon

Dette er for å kartlegge brukergruppen til Serienytt.no

1 [1-1]Alder?

Please choose **only one** of the following:

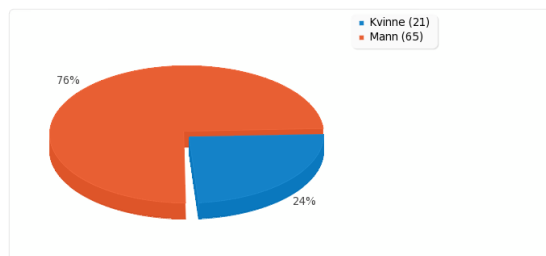
- 7-15
- 15-20
- 20-25
- 25-30
- 30-35
- 35-40
- 40-45
- 45-50
- 50-55
- 55-60
- 60+



2 [1-2]Kjønn?

Please choose **only one** of the following:

- Female
- Male



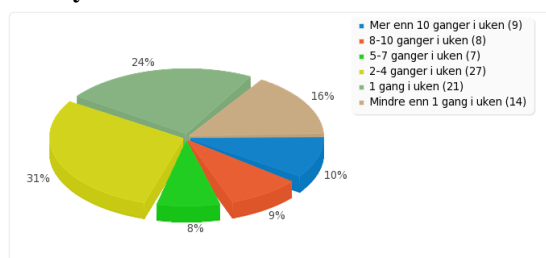
2. Dine nyhetsvaner

Brukes til å forstå dine nåværende vaner med nyheter generelt og med Serienytt.no

3 [2-1]Hvor mange ganger i uken leser du Serienytt?

Please choose **only one** of the following:

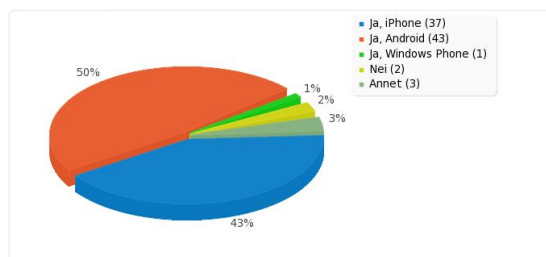
- Mer enn 10 ganger i uken
- 8-10 ganger i uken
- 5-7 ganger i uken
- 2-4 ganger i uken
- 1 gang i uken
- Mindre enn 1 gang i uken
- Aldri



4 [2-2]Har du en smarttelefon?

Please choose **only one** of the following:

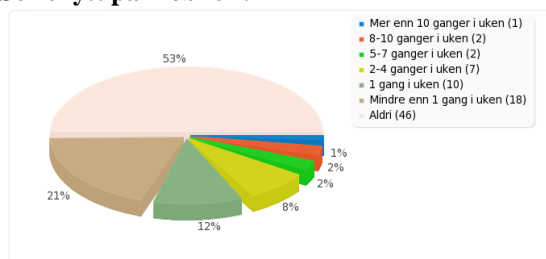
- Ja, iPhone
- Ja, Android
- Ja, Windows Phone
- Nei
- Other



5 [2-3]Hvor mange ganger i uken leser du Serienytt på mobilen?

Please choose **only one** of the following:

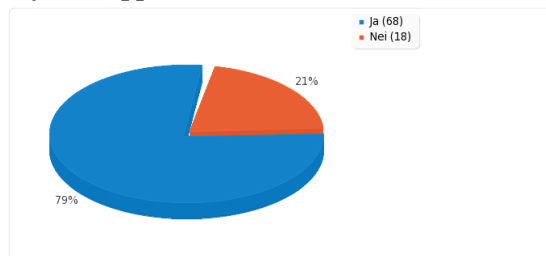
- Mer enn 10 ganger i uken
- 8-10 ganger i uken
- 5-7 ganger i uken
- 2-4 ganger i uken
- 1 gang i uken
- Mindre enn 1 gang i uken
- Aldri



6 [2-4]Kunne du tenkt deg å bruke en SerieNytt.no-app?

Please choose **only one** of the following:

- Yes
- No



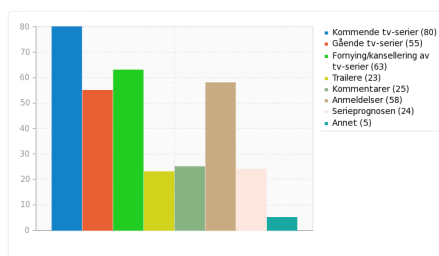
3. Innholdet fra Serienytt

Hvilket innhold synes du er mest interessant?

7 [3-1] Hva slags nyheter fra Serienytt er du mest interessert i å lese på mobilen?

Please choose **all that apply**:

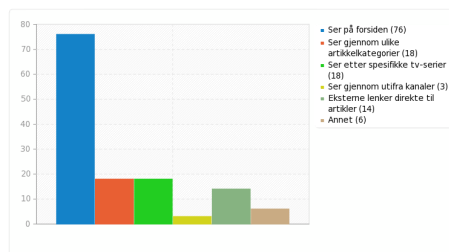
- Kommende tv-serier
- Gående tv-serier
- Fornying/kansellering av tv-serier
- Trailere
- Kommentarer
- Anmeldelser
- Serieprognosen
- Other:



8 [3-2] Hvordan leter du etter nyheter på SerieNytt.no?

Please choose **all that apply**:

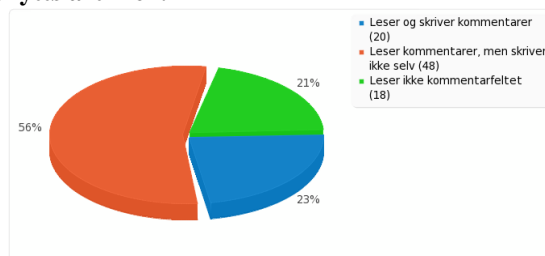
- Ser på forsiden
- Ser gjennom ulike artikkelkategorier
- Ser etter spesifikke tv-serier
- Ser gjennom utifra kanaler
- Eksterne lenker direkte til artikler
- Other:



9 [3-3] Bruker du kommentarfeltet til Serienytts artikler?

Please choose **only one** of the following:

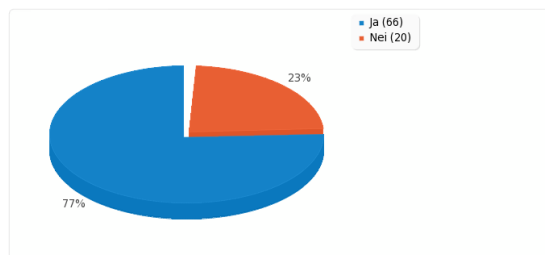
- Leser og skriver kommentarer
- Leser kommentarer, men skriver ikke selv
- Leser ikke kommentarfeltet



10 [3-4] Ser du etter nyheter om spesifikke tv-serier?

Please choose **only one** of the following:

- Yes
- No



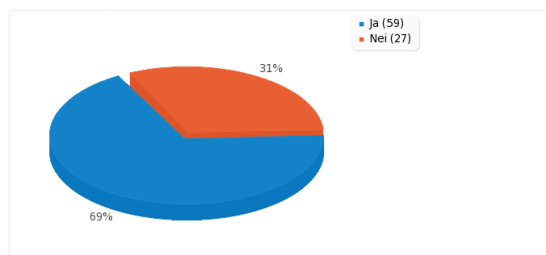
4. Utseende på mobil

Hva slags design/utseende liker du på mobil?

11 [4-1] Tror du bilder har noe å si for om du klikker deg inn på nyheter?

Please choose **only one** of the following:

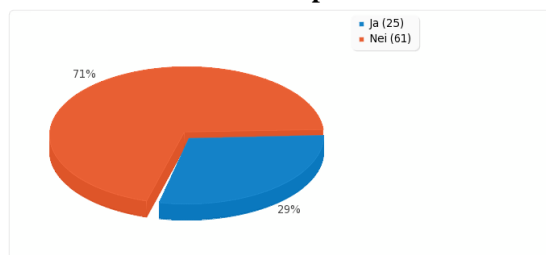
- Yes
- No



12 [4-2] Tenker du på hvor mye datatrafikk du bruker når du surfer på mobilen?

Please choose **only one** of the following:

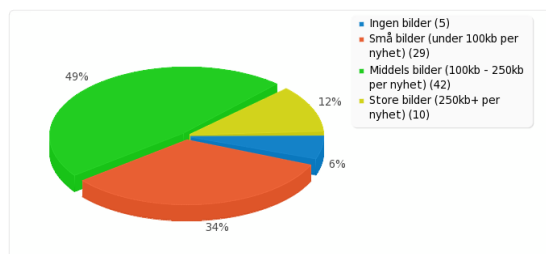
- Yes
- No



13 [4-3] Er du villig til å bruke datatrafikk til flere/større bilder om nyhetene?

Please choose **only one** of the following:

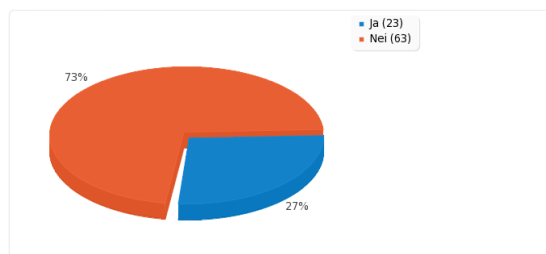
- Ingen bilder
- Små bilder (under 100kb per nyhet)
- Middels bilder (100kb - 250kb per nyhet)
- Store bilder (250kb+ per nyhet)



14 [4-4] Synes du det er ordinære nettartikler blir for lange til å lese på mobil?

Please choose **only one** of the following:

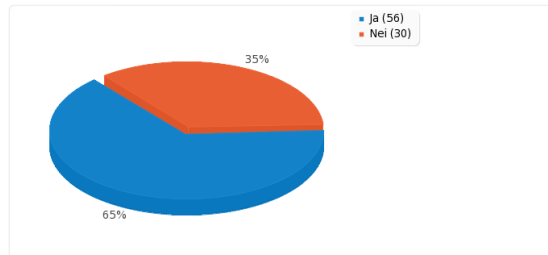
- Yes
- No



15 [4-5] Bruker du mobilen til å se videoer?

Please choose **only one** of the following:

- Yes
- No



5. Annet

16 [5-1]

Valgfritt: Har du noen ønsker/ideer om funksjonalitet du ønsker Serienytt skal legge til på mobilutgaven?

Please write your answer here:

Takk for din deltagelse.

Bård Bachmann og Serienytt.no

Appendix C Initial System Specification

Master 2012/2013 – Bård Bachmann

Kravspesifikasjon: Serienytt.no

Generelt for alle prototyper:

-	Må fungere på HTC Desire S
-	Applikasjonen krever internett-tilkobling for å finne nyheter, men ikke for å starte opp
-	Må inneholde flere Serienytt.no-artikler
-	Dette gjøres enten ved å konvertere fra HTML til TextViews eller eksport fra Wordpress.
-	Må kunne hente Serienytt-artikler i tilnærmet ekte-tid.
-	YouTube-filmer i artiklene bli gjenkjent gjennom tags. Taggene gjøres deretter om til lenker til appen.
-	Må tillate lesing av kommentarer fra nettsiden
-	Om teknisk mulig: tillate å skrive kommentarer selv

Spesielle krav for hver prototype:

PROTOTYPE 1 (nettavis-simulasjon) :	PROTOTYPE 2 (overskrift-fokusert design) :	PROTOTYPE 3 (bilde-fokusert design):
- Skal være i HTML og CSS-kode	- Skal programmeres med mobil rammeverk	- Skal programmeres med mobil rammeverk
- Kan leses fra telefonens nettleesere.	- Må være i 'hybrid app-format' - Dette betyr at den kan finnes i listen over mobilapplikasjoner	- Må være i 'web app-format' - Dette betyr at den kan finnes i listen over mobilapplikasjoner
- Datamengden på forsiden og menyer kan være på 250kb+, men saker bør være mindre	- Hver sak og meny skal ha minimalt med data - Under 200kb per side – Ca 40% ønsker 'små bilder' eller mindre. Derfor bør dette være et tilbud	- Hver sak må ha minst et stort bilde, relevant til saken. (Zillman)
- Forsiden er viktig for informasjon (Krug), mens når de har klikket seg inn kan man spare på datatrafikk.		
- De "viktigste" sakene får små bilder, mens andre er ren tekst	- Bilder bør unngås, med mindre det er små og vitale bilder	- Har ingen restriksjoner på bildestørrelser (71% svarer Nei på spørsmålet – "Tenker du

		<p>på hvor mye datatrafikk du bruker når du surfer på mobilen?)</p>
<ul style="list-style-type: none"> - Overskrifter skal skape interesse hos brukeren – (Krug) - Ren tekstlenke til ekstern video. 	<ul style="list-style-type: none"> - Overskrifter skal være informative for brukeren - Video noteres i tekstlenke, med advarsel om databruk 	<ul style="list-style-type: none"> - Overskrifter skal hovedsakelig underbygge bildene. - Videoer vises med bilde, som lenker til ekstern video – 64% ser video på mobilen, dermed er det nødvendig å tillate dette.
<ul style="list-style-type: none"> - Nyheter kan sorteres utifra forsiden og artikkelkategorier – (Web survey) 	<ul style="list-style-type: none"> - Samme sorteringssystem som Prototype 3? Eventuelt et system som finner "lignende" artikler. - Samme tv-serie - Samme regissør/skuespillere - Samme artikkeltype 	<ul style="list-style-type: none"> - Nyheter sorteres gjennom RSS-kategorier: <ul style="list-style-type: none"> - Nyeste - Utifra tv-serier - Artikkelkategori - Ytterligere sorteringsmåter blir lagt til om man får tid – som: seriesjanger, forfatter og egendefinerte søk.
<ul style="list-style-type: none"> - Kategoriene er fastsatt og kan ikke endres 	<ul style="list-style-type: none"> - Brukeren skal få bestemme layout og kategorier. 	<ul style="list-style-type: none"> - Brukeren skal kunne bestemme selv hvilke kategorier som vises på deres app.
<ul style="list-style-type: none"> - Menyknapper for hver kategori 	<ul style="list-style-type: none"> - En større meny 	<ul style="list-style-type: none"> - Disse kategoriene kan skiftes mellom ved å "swipe" til venstre og høyre.
<ul style="list-style-type: none"> - Bilde eller overskrift fungerer som lenker on-click til en side for hver artikkel 	<ul style="list-style-type: none"> - Tekstlenker/bilder som lenker til sider. 	<ul style="list-style-type: none"> - Hver sak får et bilde med overskrift, som on-click sender deg til en side for artikkelen.
<ul style="list-style-type: none"> - Ingen personalisering 	<ul style="list-style-type: none"> - Kan skru av og på designvalg, som for eksempel: Ingen bilder, laster bare inn deler av teksten. 	<ul style="list-style-type: none"> - Du velger "kategoriene" som kan swipes mellom, men designet er fastsatt.
<ul style="list-style-type: none"> - Bør ha tekstsøk 	<ul style="list-style-type: none"> - Bør ha tekstsøk 	<ul style="list-style-type: none"> - Tekstsøk kan bare legges til som en ny "kategori" (om det er teknisk mulig)
<ul style="list-style-type: none"> - Kommer rett inn på forsiden med nyheter, men siden har også en meny. 	<ul style="list-style-type: none"> - Har en hovedmeny med ulike valgmuligheter 	<ul style="list-style-type: none"> - Menyvalg finnes bare ved hjelp av "menyknappen" på telefonen.
<ul style="list-style-type: none"> - Full tekst fra artikkelene fra SerieNytt 	<ul style="list-style-type: none"> - Laster inn teksten i seksjoner? 	<ul style="list-style-type: none"> - Laster inn hele teksten
		<ul style="list-style-type: none"> - Undersøkelse: "Synes du det er ordinære nettartikler blir for lange til å lese på mobil?" – 76% = nei og 24% = ja.

Appendix D Prototype Screenshots

Prototype 1:

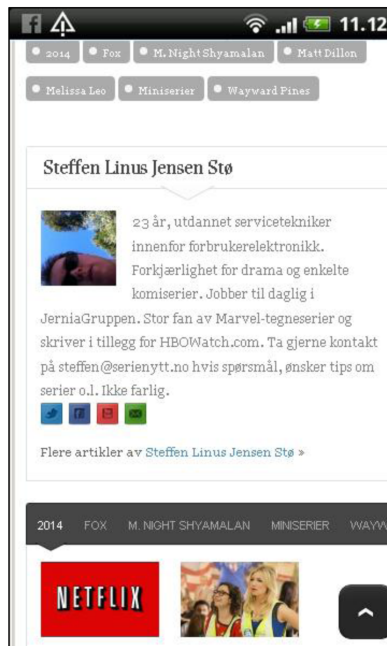
Front Page



News Article



News Article - Metacontent



News Article - Comments



Prototype 2:

Front Page (Images off)



News Article



News Article - Comments and Related



Front Page - Main Menu



Prototype 3:

Front Page (carousel)



News Article



Category Selection Screen

