

Designing for mobile devices – on the traditional web page

Aleksander Dye

NKI Distance Education

March 2006

Designing for mobile devices – on the traditional web page

By Aleksander Dye
March 2006

Abstract

This paper is written with the intent of clarifying some issues involving designing for small screens and mobile devices. First the term mobile device is defined, then the mobile devices difference from traditional computer screens are taken into consideration. We then decide on what to show the mobile users and how to display it on a small screen. The experiences and recommendations on these issues are based on extended work on the issue mobile learning as evidenced by previous EU projects (NKI)

What is a "mobile device"?

A mobile device in this paper is considered to have the following characteristics and are illustrated¹ by some varieties of mobile devices below:

- Can be used wireless
- Can be used standing with minimal effort
- Is small enough to be held in one hand



This definition rules out a laptop with wireless connectivity, not to say that is not a mobile device as such, but it is not mobile in the way that we have to design specifically for it. In our definition it is the mobility and the use of the device when mobile that is key. For this paper the definition and guidelines are for PDA's² or other mobile devices capable of displaying html³ or xhtml⁴ (from now on referred to as (x)html) on screen with the same approximate display size as a PDA.

What is the main differences on a mobile browser?

The main difference which makes designing for a mobile device a challenge is the size of the screen – or the lack of it. We need to think about how we utilize the valuable screen real estate⁵ that is given to us. We are faced with a menu that fills most of our screen and if a page has adverts, well that takes up most of the screen as well. This leaves little room for actual content, which is what the users are after.



What should we show our mobile users?

Keep in mind the fact that a mobile client is usually only used when the traditional desktop (or

- 1 Images are provided by [Idedata](#) in Norway who have aided us in some technical problems with PDA's
- 2 Personal Digital Assistant
- 3 HTML is the lingua franca for publishing hypertext on the World Wide Web. (W3C)
- 4 Extensible HyperText Markup Language (XHTML™) is a family of current and future document types and modules that reproduce, subset, and extend HTML, reformulated in XML.(W3C)
- 5 The amount of space available on a display for an application to provide output (UF)

laptop) computer is not accessible. The users that uses mobile devices are usually on the move, they are mobile and goes on the web to do specific tasks. They are not browsing the web, they are either searching for something specific or they are doing a defined task. Your mobile user is not at your site by chance (they could be, but rarely is this the case), they are at your site to do a specific task. Your mobile user is most likely to be a valuable user, one of your regular users who also wants to access your site from somewhere (s)he can not use a more comfortable sized screen. This is most likely a user who talks about your site and gives you credit for what you do... Why not give the user what (s)he wants as well.

If it is not needed – remove it. If you have adverts on your site, remove it from the mobile page, (s)he will come back later to see the adverts, (s)he is not in a browsing mode at this time anyway.

Keep in mind that the mobile user usually pays for the amount of data (s)he downloads (in both time and money), therefore it is wise not to use large or unnecessary graphics.

How to display the information on a mobile device

Remember the size of the screen at all time. Hide what is not useful and clean up the cluttered pages that you serve the traditional clients. The 200 pixel menu takes up a bit of the sidebar of a computer screen, but fills up most of the mobile clients screens and pushes the content the users are there for away or into a small column of unreadable text.

In our courses we have moved⁶ the menu to the top and removed the tables to give the content the full width of the mobile browser, but we keep the menu on the side for other clients. This way on a mobile client the menu is on top and when that is scrolled past the screen is clean to show only meaningful content to the user.

Design guidelines

Key to remember is the size of the screen which for most PDA's are 3,5" large displaying 240x320⁷ pixels. The most popular screen resolutions on the web in the world are (Onestat):

1.	1024 x 768	57.38%
2.	800 x 600	18.23%
3.	1280 x 1024	14.18%
4.	1152 x 864	4.95%
5.	1600 x 1200	1.67%
6.	1280 x 800	1.56%
7.	1600 x 1024	0.55%

This table gives us a pointer to the fact that screen sizes are expanding, therefore we should be designing for the larger screens, right? Wrong! Design for *every* screen size that *does not* compromise the most used screen size! If it is possible to create a design without fixed pixels and fixed positioning you are mostly there... But how do you do that?

⁶ See Designing with CSS later on for how-to on this issue

⁷ Current display for PDA's sold in Norway from [idedata](#) 08. February 2006, there are some with greater display sizes but most have this resolution.

Designing with CSS

One of the most important features of style sheets is that they specify how a document is to be presented on different media: on the screen, on paper, with a speech synthesizer, with a braille device, etc. When you remove style definition from the (x)html⁸ and move it into css⁹ you can control the look and feel of multiple web pages, in one or more css files that are independent of the (x)html that is viewed in the browser. This way you loosely couple the data from the display rules and have more control over the layout in different browsers or devices.

The greatest advantage here is (in my opinion) that for the different devices you can have different layouts and design elements. The way we solved this is by the use of media types in css. This is done by the use of media types. The @media construct allows style sheet rules for various media in the same style sheet. *"A CSS media type names a set of CSS properties. A user agent that claims to support a media type by name must implement all of the properties that apply to that media type."* (W3C-m). An example from W3C is given below which shows different font layout for the different medias. Note that the last one sets line-height for both screen and print:

```
@media print {  
    BODY { font-size: 10pt }  
}  
  
@media screen {  
    BODY { font-size: 12pt }  
}  
  
@media screen, print {  
    BODY { line-height: 1.2 }  
}
```

Colors and Fonts

There are web safe colors as well as fonts(DftW), these are easily identified and well known to designers as well as most tools for creating web pages. The web safe colors and fonts are the ones that are not operating system dependent, but displayed in the same manner on every platform. The color management system currently used by Web browser software is based on an 8-bit, 216-color (not 256) palette. (BsC) Netscape fixed this problem with the 216-color palette for the web. Today most users have their screen set to thousands or millions of colors, so for the desktop market this is not the same problem (however, two screens does most of the time show colors slightly different based on settings). The problem is that for the mobile devices there is not as easy to obtain the same information, some devices

Browser-safe palette
216 colors

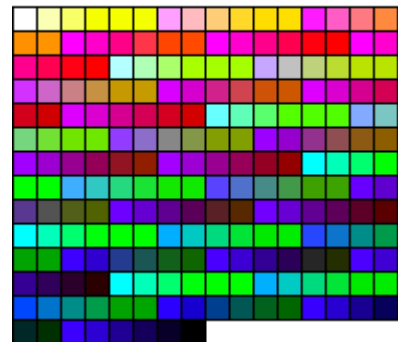


Illustration 1: "Browser-safe" colors

display one font, others another. If you can identify some commonly used fonts for the mobile devices you should use them for the css. This will limit the choice in fonts, but this does not have to impact the traditional design – one font set for the desktop browser and another for the mobile devices. The following is a list of the most common font combinations for traditional web design and should also work on most other devices:

⁸ I say (x)html because that is what is view in the browser, where the definition for the size is could of course be somewhere else in the code, but the end result is what I am talking about.

⁹ Cascading Style Sheets (CSS) is a simple mechanism for adding style (e.g. fonts, colors, spacing) to Web documents (W3C)

- Arial, Helvetica, sans-serif
- Times New Roman, Times, serif
- Courier New, Courier, mono
- Georgia, Times New Roman, Times, serif
- Verdana, Arial, Helvetica, sans-serif
- Geneva, Arial, Helvetica, sans-serif

The use of **bold** and/or *italic* is the same as with the traditional web pages. Do not use them exceedingly, but to emphasize the message. It is easier to read bold than italic so we recommend using bold if the word is of importance to the readers understanding. When we look at font-size definitions the best way to implement this is the same as the recommendations for web pages – use percentage and let the user set his or her preferences, do not set a fixed size, the screen of a mobile device is small and the web page might be compressed. If the font-size is set fixed by pixels it will be too small to read for some users.

Using Illustrations

The problem with illustrations are obvious; the screen of the device is too small, the download time too long, not all devices display color. You can either see the full image and scroll in every direction or the image is too small to understand. Keep in mind that download time and cost should be kept low for the users, and if the user has a device such as a smart phone the memory might be an issue, but this is rarely the case with a PDA.

For the course designers the important factor to keep in mind at all times is that it is not the look of the text and images that are important, but the information that the design is to illustrate and emphasize, design comes afterwards. If an image is being used, there should also be a descriptive text for blind users (they can have the description read to them) and the same text could be displayed for the mobile devices.

The problem however is that if the text is not there in the first place, the cost involved in describing the illustrations is too large to defend for a small group of mobile users. The cost of doing so at course creation is much lower and not only gives the mobile users an advantage, but also increases the accessibility of the course to the general public.

Using flash on the PDA

Development work with flash on the PDA is interesting but it is useful to have focus on the readability on all flash applications. This is because low readability is much more noticeable on a small screen. Flash is authoring software developed by Macromedia, and is used for producing vector graphics-based animations. Flash makes it possible to develop various programs/features such as navigation interfaces, graphics illustrations, interactivity in a re-sizable file format that is small enough to stream across a normal modem connection. Flash is very suitable for use on the web due to the vector graphics ability to adapt and adjust to different display sizes and screen resolutions and it is designed for optimized delivery. The important thing to remember is not to use negative text¹⁰ and to be more focused on the readability of the color combinations of the background and text. The choice of font is also important.

It is possible to develop the courseware with the use of flash together with audio. This was done by NKI in a previous project and have given us insight to some of the challenges in developing for small screens. After experimenting with the course assignments developed in Flash, we adapted

¹⁰ White text on a black background is negative, black text on white background is positive text

them to the PDA. The assignments have been adapted and further developed from the original version intended for use on a large screen to be used on the PDA with the Internet Explorer and the Flash player 6 plugin for PDA installed.

Screen format on the PDA

We found that 240 x 270 (width x height) pixels was the optimal size and filled the screen as much as possible (see Illustration 2: Full size of the "palette" illustrated by the red color)



Using text in Flash

When we started the development we found that the text we used in the original assignments was too small and nearly impossible to read when presented on the PDA. We started to experiment with different font types and sizes. By default Flash uses anti-aliasing on the font and that makes the text a little blurry at the edges to soften the text. To get the text sharpened and easy to read on a small screen, we found that we could use an option in flash call "dynamic text". This proved better concerning readability and solved our problem.

In Illustration 1: "Browser-safe" colors, the font-size is 12 pixels, but the flash renderer shows different sizes using two different methods. We have no good answer why Flash does so, but that is not special to the PDA since this is how the program

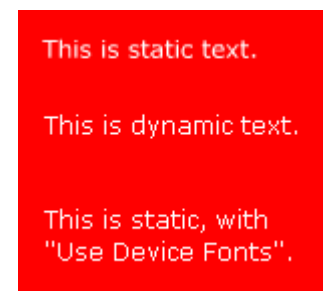


Illustration 3: Different methods with text, verdana 12 pix.

itself react. Later on, we also discovered we could keep the font static and use the options "Use Device Fonts" in flash. Flash uses device fonts to display certain text blocks, so that Flash does not embed the font for that text. This gave us the same output results as setting the text to "Dynamic" but we had more control over line breaking and text behavior and it might also increase the file size.

We experienced a lot of difference in behavior with the same flash file when using it on a PC versus the PDA. For instance we had textboxes¹¹ that showed three lines on a PDA, could only shows two lines on a PC. We solved this by adjusting the size of the textboxes to fit the PDA.

We have also made a couple of assignments that use "drag and drop"¹² functionality. This worked very well on the PDA, but we experienced the same issues here regarding text readability. One must also have in mind that some functions, like "onMouseOver"¹³, wouldn't work on the PDA since the

11 Text boxes are used for user input and to show output

12 In computer graphical user interfaces, drag-and-drop is the action of (or support for the action of) clicking on a virtual object and dragging it onto another virtual object. The basic sequence involved in drag-and-drop is: * Press, and hold down, the button on the mouse or other pointing device, to "grab" the object,* "Drag" the object/cursor/pointing device to the desired location,* "Drop" the object by releasing the button. ([wikipedia](http://en.wikipedia.org/wiki/Drag_and_drop))

13 OnMouseOver is used with JavaScript or other scripting language to "listen" to the users interfacing with the web page.

PDA is equipped with a touch screen and a stylus and no mouse like on the PC.
 Below are some screens shots from assignments made in Flash (in Norwegian):



Illustration 4: Drag and Drop assignment



Illustration 6: Drag and Drop assignment



Illustration 5: Boat Operators Certificate "Game".- If you make a bad decision, you'll get an informative feedback telling you what you should have done to avoid an unpleasant encounter with shallow waters.



Illustration 7: Boat Operators Certificate "Game".- Take your boat trough rough waters guided by the navigation marks and beacons

Video on the PDA

At NKI we tried the use of video on the PDA with small video clips showing a discussion between two persons in a learning situation. This has worked very well using the Windows Media Player and we had no problems with viewing the files with high quality both on picture and audio. This is of course also due to high capacity of the wireless broadband solution. We encountered problems when trying to stream video directly from the web browser. Unlike Internet Explorer (IE) for PC, the old¹⁴ pocket version of IE was not capable of streaming video directly from the browser or start the Windows Media Player. This meant that one have to copy the URL into the Media Player and watch the video from the Media Player. This worked okay, but was a rather cumbersome method to watch a video.



Illustration 8: Screen shot from a video clip

Conclusion

The mobile devices you are developing for has a small screen and most of the time the bandwidth is small and cost is high. Keep the information as easily accessible as you can and strip away unnecessary illustrations. Always keep a textual description of an image for the users who can not (vision impaired users) or choose not to (users on a high cost access point who does not display images) show the images. If you have separation of content and presentation in mind at course creation, the cost of adaption to a mobile device, screen reader, converted to PDF¹⁵ or other format is both easier and comes at a lower cost. If the courses are already created, the cost is much higher. If there is only one course to be made available to the mobile device it will be easier to adapt that course, that to adapt a technical framework¹⁶ or LMS¹⁷ to do the job. The problem is to know when is the cost of adapting a course or framework/LMS is worth the time and effort and will give a positive return of investment. This depends of the size and complexity of the course and framework/LMS and the numbers of paying costumers as well as the price being payed. This is why it is hard to make a general recommendations to adapt a course or the framework/LMS. The best recommendation I can give is to have accessibility and separation of content and presentation in mind when designing courses. This ease the adaption to other medias in the future.

14 This is fixed in versions from windows mobile 2003

15 A file format created by Adobe Systems based on its PostScript® page description language. PDF files are platform- and device-independent, and are much easier to “port” from one user’s system to another without errors arising. PDF files are created and supported using Adobe’s “Acrobat” software tools; also, many other vendors have announced development of tools that support and expand the use of PDF in graphic arts production.
<http://www.adobe.com/products/vdp/glossary.html>

16 the underlying structure used to display the content in a context

17 LMS – Learning Management System

References

Design, Development and Evaluation of Mobile Learning at NKI Distance Education 2000-2005 (NKI)

Rekkedal, T., Dye, A., Fagerberg, T., Bredal, S., Midtsveen, B. & Russell, J
ISBN 82 562 6509 4

Mobile Web Best Practices 1.0

<http://www.w3.org/TR/2005/WD-mobile-bp-20051220/>

06. february 2006

Onestat.com (Onestat)

http://www.onestat.com/html/aboutus_pressbox38.html

08. february 2006

The World Wide Web Consortium (W3C)

<http://www.w3.org/>

09. february 2006

Usability first (UF)

<http://www.usabilityfirst.com/>

15. february 2006

Designing for the Web (DftW)

http://www.digital-web.com/articles/designing_for_the_web/

15. february 2006

Idedata

<http://www.idedata.no/>

16. february 2006

Media types on W3C (W3C-m)

<http://www.w3.org/TR/REC-CSS2/media.html>

16. february 2006

"Browser-safe" colors (BsC)

<http://www.webstyleguide.com/graphics/safe.html>

16. february 2006