

# Guidelines For Creating Web Content For Mobile And PC Browsing

Version 1.0; September 27, 2004

# Browsing

**NOKIA**

Copyright © 2004 Nokia Corporation. All rights reserved.

Nokia and Nokia Connecting People are registered trademarks of Nokia Corporation. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

#### **Disclaimer**

The information in this document is provided "as is," with no warranties whatsoever, including any warranty of merchantability, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. Furthermore, information provided in this document is preliminary, and may be changed substantially prior to final release. This document is provided for informational purposes only.

Nokia Corporation disclaims all liability, including liability for infringement of any proprietary rights, relating to implementation of information presented in this document. Nokia Corporation does not warrant or represent that such use will not infringe such rights.

Nokia Corporation retains the right to make changes to this specification at any time, without notice.

#### **License**

A license is hereby granted to download and print a copy of this specification for personal use only. No other license to any other intellectual property rights is granted herein.

## Contents

<b>1</b>	<b>Overview .....</b>	<b>5</b>
1.1	Introduction.....	5
1.2	Mobile and PC Browsing.....	5
1.3	XHTML or HTML? .....	6
<b>2</b>	<b>Design Guidelines .....</b>	<b>7</b>
2.1	Styling Pages.....	7
2.2	Page Structure .....	8
2.2.1	Original and Narrow layout.....	11
2.3	Content .....	12
2.3.1	Text.....	12
2.3.2	Page titles.....	12
2.3.3	Page length.....	12
2.3.4	Optimizing code .....	12
2.4	Navigation .....	12
2.5	Access Keys and Anchors.....	13
2.5.1	Access keys.....	13
2.5.2	Anchors.....	14
2.6	Search.....	14
2.7	Frames.....	15
2.8	Pop-Up Windows.....	15
2.9	Rich Media.....	15
2.10	Images and Colors.....	15
2.10.1	Image optimization .....	15
2.10.2	Color depths.....	16
2.11	Tables .....	16
2.12	Forms.....	16
2.12.1	Input fields.....	17
<b>3</b>	<b>Summary .....</b>	<b>18</b>
<b>4</b>	<b>References .....</b>	<b>19</b>

## Change History

September 27, 2004	Version 1.0	Initial document release

# 1 Overview

## 1.1 Introduction

This document provides content development guidelines for HTML- and XHTML-based browsing using a mobile or PC browser. The guidelines discuss similarities and differences between browsing experiences in mobile devices and PCs, and what needs to be considered during Web page design to make Web content browsable on a mobile device. The focus is on studying aspects of the user experience and usability. The document also addresses the importance of using XHTML for Web development.

## 1.2 Mobile and PC Browsing

Most Web sites are written for, and tested exclusively on, desktop computers with large color monitors. Mobile devices typically have much smaller screens, and until recently it was a challenge to present Web pages in such a small area. Today there are various mobile browsers available that support traditional Web content markup languages like HTML and XHTML. For example, Nokia has introduced devices supporting these markup languages in Series 60 Developer Platform 2.0 and Series 80 Developer Platform 2.0. A subset of XHTML that is for mobile browsers — XHTML Mobile Profile — is also supported in various devices.

The most important differences between application development for PCs/Web and for mobile devices are:

- Mobile devices have smaller displays, as well as a wide variety of display sizes.
- Current mobile devices with color displays support either 4,096 (12-bit) or 65,536 (16-bit) colors while some PCs are able to display over 16 million colors (32-bit). Secondly, color displays have been available in mobile domain only for a couple of years now.
- Text inputting is slower in mobile devices than it is with a full PC keyboard.
- Mobile devices usually have no mouse for activating an object, which limits the possible user interface components and slows down object activation.
- Some mobile devices and/or browser view modes support only vertical scrolling.
- Soft keys are used for activating commands in mobile devices; the number and purpose of soft keys vary between devices from different manufacturers.
- Connection establishment and data transfer between the terminal and the server is slower than in a fixed domain.
- The amount of cookie data that can be stored in a mobile device is limited.
- The context of use cannot be predicted as easily as with, for example, an office PC application.
- Mobile users may have to pay for each piece of transferred data.

Already-existing projects indicate that end-user perspective and usability are very important in mobile service development. It is not enough to consider usability when developing the user interface — it needs to be a factor in content development as well. Mobile browsing is technically much more challenging than traditional PC browsing, which uses fixed connections, but users still want speedy access to content. This poses a huge challenge to site owners. When it comes to maximum waiting time, developers should look to the Web's seven- to ten-second time limit as a rough guide.

### 1.3 XHTML or HTML?

Because HTML has been available for wired Web content development much longer than XHTML, most Web sites use it. Even though the World Wide Web Consortium (W3C) and the Open Mobile Alliance (OMA), which develop markup language standards, are pushing XHTML and especially its subset XHTML Mobile Profile for mobile content development, it's obvious that today's browsers need to support traditional HTML, too. Device manufacturers are doing their best to get browsers to display original Web pages in the mobile browser so that users can access HTML content available on the Web. For the best mobile-service user experience, XHTML Mobile Profile is still the recommended optimal markup language. When content is targeted at both mobile and PC browsing, XHTML offers more capabilities and is therefore most suitable because the code itself can be the same, but optimized for different browsers using Cascading Style Sheets (CSS).

## 2 Design Guidelines

When designing a service aimed at both mobile devices and PCs, the design should begin with the mobile device's user interface. It is generally easier to extend a mobile-oriented service to a PC environment, rather than the other way around. However, if a PC-optimized Web service is the starting point, developers should break the service into small segments and choose only the core elements for the mobile service.

In order to create efficient services for mobile browsers, it is important to research existing information about the different mobile browsers on the market. Developers will need to know the maximum allowed size of documents, supported elements and image formats, screen space available, etc.

The introduction of XHTML and CSS to the mobile domain creates a variety of new UI constructions. XHTML contains more elements than WML, and the visual rendering of elements can be modified in numerous ways with CSS. Compared to WML, XHTML MP allows more possibilities for service providers to make their services attractive, while at the same time adding a layer of complexity and thus more usability challenges.

Both the information architecture and UI design must underline ease of use in mobile services, even more strongly than in Web design. This is a key factor in successfully ensuring customers' continual use of mobile services. Even though many features and possibilities are available thanks to evolving browsing technology, the focus should be on usability. Key factors in mobile usability are ease of use and learnability. Unfortunately, there are a lot of services that don't succeed here. This doesn't mean that the services are too complicated — rather, the method used to create the user experience was not user centric, and did not take into account the way end users would like or expect their mobile service to work.

### 2.1 Styling Pages

When styling XHTML/HTML content for small screens, it is important to find a good balance between minimizing scrolling and having a crowded layout. The float property offers a good way to use the screen size without making it crowded, and at the same time minimize the amount of scrolling needed to find the desired information. This is especially true when vertically large but horizontally small images are defined. The float property then offers a good way of using the white space next to the image.

Aligning content to the left is strongly recommended because it keeps the content readable. Be consistent when using alignment — the more different alignments (left, center, right) applied to an XHTML page, the more effort is required from the end user to get an overview of the content.

CSS give developers full control of the colors used in the browser. Colors are excellent for adding more "life" to a service, but at the same time they can decrease the usability of the service if used improperly. Limit the number of different colors on a page, and always use the same color for the same XHTML elements throughout the service. Not all mobile devices support colors, so make sure that the service content does not mention specific colors (for example, "Press the red link to proceed") because colors may be converted to black by devices that only support black and white. Be careful when using the color red because it can be associated with warnings or prohibitions. Red is used for alarms and alerts.

In general, try to apply the same styling on all pages in a service. Consistency increases learning — especially for users who return and use the service again and again.

When several pages share scripts or style sheets, add a link to external files instead of embedding the scripts/style sheets in the pages themselves.

Avoid having too many different text styles and text sizes on the same XHTML page. Two or three different text styles and sizes are the recommended maximum. Basically, this means separating headings (bold) and content (normal). Separate the main heading from subheadings and define one style for content. You may also use bold to emphasize some key words. Avoid using text emphasis properties such as italics and underlining (used ONLY for links) because it interferes with readability. text-decoration: underline property shouldn't be used for plain text. Only emphasize a few words on each XHTML page. Blinking text should be avoided because it decreases readability and makes it harder for the user to focus on the surrounding content.

Separate the link style from other content styles. Keep the length of each link minimal (for example, Calendar, Ring tones, etc). Avoid graphic-based navigation. Mobile users have to wait for graphics longer than text-based content. Enable navigation as soon as the user can see the content.

Remember common accessibility rules. Change the color of the visited links.

Avoid self-defined abbreviations because not all users will understand them. Use a clear, to-the-point writing style. Extra text is annoying on the Web, and even more so in mobile browsing.

Avoid disabling line wrapping of XHTML content. In other words, don't use the no-wrap value of the white-space property. Wrapping is strongly recommended because much more content is immediately visible without scrolling.

Do not remove underlining from links (anchors, <a> elements). In other words, do not use the text-decoration: none style on anchors. Users have no chance to see whether a nonunderlined anchor is an anchor or simply plain text before moving the focus bar to the anchor. Furthermore, an underlined anchor is a well-known UI object, and removing the underline will only confuse or slow down the navigation process of the service in use.

Do not remove borders on tables that only have a function of showing information in a tabular manner, meaning when not used for laying out content. Remove the border (border-style property) on tables that only have a function of laying out content.

Avoid too many empty lines between text sections on an XHTML page. Empty lines mean more scrolling, and some users stop scrolling when they see an empty line, believing they have reached the bottom of the page. Create pages that are visually pleasing and be consistent in how you separate content "elements" from each other. Also, keep descriptions as compact as possible.

## 2.2 Page Structure

Designing a Web page for multiple media is much simpler when you take document order into account. Document order is the order in which the XHTML/HTML source code appears. In most cases, a page is displayed according to this order. Content that appears early in the source code is displayed above content that comes later, and elements close to each other in the source code remain close to each other when displayed.

- Normally the **heading area** appears first, at the top, and generally includes the company logo, site name, helping site-level links (for example, FAQ and Index), and search field.
- The next component should include **navigation**.
- **Page name** ensures that the page name can be identified/recognized quickly and easily. It enables the user to define whether s/he would move on using navigation or look more deeply at the content below.
- In the **main content area**, the descriptive beginning helps the user figure out if this is the page s/he is looking for. More descriptive content is placed below the descriptive field.

- **Related**—links to related areas/products, etc.
- The **footer area** generally contains copyright information (link), but in some cases also offers navigation options on this level (for example, a page is one of five third-level pages; the footer will show all five as links in the footer, reducing the need for scrolling).

One exception to page order is tables. Tables are displayed cell by cell, left to right, for each row going down. If the cells are large, the distance between elements can be great, even if they are close in the page source. In the example below, cell 3 is far from cell 4, and cells 1 and 4 are visually close, even if they have the entire content of cells 2 and 3 between them.

```
1  2  3
4  5  6
```

While a PC screen may be large enough to display several columns of text, this is rarely feasible on a handheld device. For that reason, tables will usually be reformatted to the document order (that is, the text will be read 1-2-3-4-5-6).

Other exceptions to HTML being displayed in document order are CSS features floats and positioning. Float moves content to the far left or right of its containing box. Positioning can move content to any position on the page, stack content on top of one another, or hide some content behind other content.

It is better to have important content early, and to group additional/extra code (like navigation sections) so that it is easier to skip.

Figure 1 shows how the Opera Web browser displays the same content on a PC and a mobile browser.



Figure 1: Opera browser renders HTML content in the order in which it is defined

## 2.2.1 Original and Narrow layout

Several mobile browsers today provide two different views for displaying content: Original layout and Narrow layout. Original layout shows the page in the format in which it was originally designed, so the user can anticipate where the information is located on a page. However, Original layout might be problematic because reading long text rows can be laborious when the user needs to scroll back and forth horizontally. Also, if there are large areas with no content on the page, the user may get lost.

In a Narrow Layout, the 2-D content is reformatted to a 1-D view. All page areas are presented one after another in a single column, which means no horizontal scrolling. Simplicity is the main feature of this layout. However, there are some basic problems:

- The column can be really long, and require extensive scrolling.
- The top of the page is often exactly the same on all pages on a site, so the first screen looks exactly the same on a small screen. The user does not know if the page changed at all.
- Images that are wider than the display are squeezed to fit in the screen. Therefore, images may become blurry. This is a particular problem if there is text inside an image.

You can ease narrow layout usage by prioritizing Web layout elements that are optimal for narrow layout mobile browsing. In practice, this means that the user gets the most relevant content in the beginning, right after the heading (see Figure 2).

The figure illustrates two versions of the Nokia.com homepage. On the left is the 'Original layout' (desktop view), and on the right is the 'Narrow layout' (mobile view).

**Original Layout (Left):**

- Header: NOKIA CONNECTING PEOPLE, Select a country, Nokia.com, SITE INDEX, Search [ ] GO.
- Navigation: Home | Phones | Home Products | Business | Operators | Shop | Support | Investors | About Nokia.
- Main Content:
  - Banner: 'Ready, Set, Go!' with a woman running and a Nokia Mediamaster 260 phone.
  - Text: 'Ready, Set, Go! The Nokia Mediamaster 260 is now available for all three major transmission methods: cable, satellite, and terrestrial. Don't miss a minute of the action with the [Nokia Mediamaster 260 >>](#)
  - Grid of sections:
    - Phones and more... (Compare phones, Find a new phone, See All Our Phones, Support and Software for Nokia Phones, Software Market, Buy Tones, Graphics, and Games, Enhancements)
    - Business Solutions (Phones for Business, Operators)
    - Nokia Corporate (Careers at Nokia, Nokia News)
    - Nokia Phones (See all our phones and enhancements, tips on imaging, and fun stuff)
    - Support and software (Support, Software Market)
    - Need help deciding? (Compare phones, Find Your Phone, Phone models)
    - Digital TV (Compare Digital TV receivers)
    - 1001 Reasons... (To always have a Nokia imaging phone with you, What's yours ??)
    - Enhancements (See the latest headsets, car kits, batteries, and more in Enhancements)

**Narrow Layout (Right):**

- Header: NOKIA CONNECTING PEOPLE, Select a country, Nokia.com, SITE INDEX, Search [ ] GO.
- Navigation: Home | Phones | Home Products | Business | Operators | Shop | Support | Investors | About Nokia.
- Main Content:
  - Banner: 'Smart Moves' with a woman's face.
  - Text: 'Nokia Mobility Conference 2004 Smart ideas call for smart moves and smart business partners. Make yours at the Nokia Mobility Conference 2004, taking place November 3-4 in Monaco. [Register now >>](#)
  - Text: 'Ready, Set, Go! The Nokia Mediamaster 260 is now available for all three major transmission methods: cable, satellite, and terrestrial. Don't miss a minute of the action with the [Nokia Mediamaster 260 >>](#)
  - Text: 'N-gage on Tour The N-gage truck hits the streets this summer! Check out the new N-gage™ GD, try out the latest games and become the Crash Nitro Cart Champion! [Find out more >>](#)

Figure 2: An Original layout and a Narrow layout for www.nokia.com

## 2.3 Content

### 2.3.1 Text

The text shown in a mobile browser is basically the same that is seen in a PC browser. However, developers should be conscious of some key issues in order to ensure a quality mobile-browser experience:

- It is much better to use text, rather than graphics, for headlines.
- Content should be broken up by the use of subheads.
- Anchors can be used to enable easy page reading.
- Styling elements should be used reasonably (see Section 2.1).

### 2.3.2 Page titles

Good page titles are more important on mobile devices than on a PC. Designers should be consistent in naming the pages, and use brief, descriptive names for pages. The optimum page title should have one or two words. The main thing is to ensure that the user can see the link on one line in his/her device.

### 2.3.3 Page length

There is no one suitable recommendation for page length: the appropriate page length depends on the content and the browser. Series 80 Developer Platform and Series 90 Developer Platform browsers allow users to view pages closely like in a PC. Browsing with the Series 60 Developer Platform is more challenging with regard to the amount of content per page.

The recommended page length depends on the content of the page. If the user is really reading the content (for example, news), as many as 20 screenfuls of text (Series 60) may be okay. The key issue is to reduce scrolling by minimizing the header area and ensure that the most important content areas are clearly recognizable and placed at the beginning of the page.

The page length should be tested in the usability test. This is the only way to actually figure out what type of situations can/should be handled with a short page and which can include more content.

### 2.3.4 Optimizing code

It's recommended to use only meaningful code and remove all unnecessary comments. The code should be as compact as possible. For example, `<h3>A Headline</h3>` is not only shorter and more readable than `<p><font size="+2" face="Helvetica"><b>A Headline</b></font></p>`, it also tells the browser this is a headline. The Opera browser can use this information to the user's advantage. For instance, on devices with a full keyboard, the S and W keys are used to go to the next or previous headline.

The default headline look may not be what you're looking for, and for that reason many designers started using the font tag years ago and have stuck with it ever since. However, this is no longer necessary because CSS gives full control over how the headlines should look, and every browser, including Netscape 4, reliably supports this part of CSS.

## 2.4 Navigation

In mobile services it often seems that a shallow structure is easier to understand than a deep structure. Links and pages should feature descriptive names to help the user find the information s/he needs.

It is difficult to say how many links should be provided on a link list page. If the links clearly belong together and are easy to browse (one line per link, alphabetical or otherwise logical order so that the user does not have to read through each link), it is better to provide 30 links on a single page rather than five links on six pages. If there are tens of links, it may be a good idea to provide sorting options before showing them. If the link can fit on one row, it makes the selection clear and the page look better.

The user must keep the structure of the application in his/ her mind, because there is no room to show the site structure on each page (like, for example, the navigation bars of an ordinary Web page). Most users seem to be able to keep four levels of hierarchy in their mind; after that, the structure is no longer clear. Accordingly, the most important and frequent tasks should be able to be accomplished with a maximum of four steps. Also, keep in mind the following:

- Understanding the structure and content helps users feel they are in control of the application. A feeling of control appears to be one of the most important factors in service satisfaction.
- Navigation within the service and the user interface element should be consistent. This means that placement should be the same on each page.
- On each page, except for the form and confirmation pages, there should be a link to the main page of the service and the main page of the current subsection of the service.
- Pages should be provided where the user can see the overview of the content of the application or subsection.

## 2.5 Access Keys and Anchors

### 2.5.1 Access keys

To help users understand the concept of access keys, developers should make sure that they are visible on the screen and in a form that resembles phone keys.

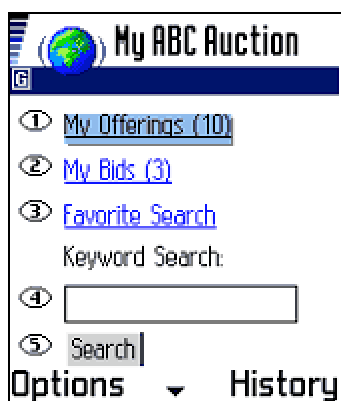


Figure 3: Access keys (numbers 1 to 5) enable faster movement

Links in a device display are handy, even essential, for navigation, especially when a document is long and links are used to jump to sections of the document. However, even with links, it is necessary to scroll to the link, which may require that the user make multiple key presses to reach the link.

An access key enables the user to activate any visible link with one key press, without needing to navigate to the link. Including the access key attribute within the <a> element creates shortcuts. The value of the attribute specifies the key that the user must press to activate the link.

However, in order for the user to know what key to press, it is necessary to specify as text (within a <p> element just prior to the <a> element) the key number you have chosen for the access key attribute. Entering the value in text ensures that the user will see the value of the access key together

with the text describing the target of the link. So, for example, specifying the following line in your content:

```
1 <a href="offerings.html" accesskey="1">My Offerings (10)</a>
```

results in the following display:

```
1 My Offerings (10)
```

## 2.5.2 Anchors

Anchored links are widely used on the Web and are equally efficient for mobile devices. Anchors are recommended as the best way to control service navigation. They enable “jumping” inside the page, which takes less time than downloading an entirely new page. Remember to use “Back to top” links when using anchors.

```
<p>
Part 1 start here. Jump directly to
<a href="#sec2">Part 2 </a>
This page... <br />
is... <br />
full of... <br />
text. <br />
<a id="sec2"></a>Part 2 starts here.
</p>
```

Example 1: Code example for using anchor to jump from one part to another inside an XHTML page.

## 2.6 Search

Users want access to information as quickly as possible, for reasons of both efficiency and cost savings. If users use several different services, it may not be handy for them to bookmark every page of your service; they may, for example, use search instead. Here are a few important issues to remember when designing search functionalities:

- Novice users need well-structured navigation, and they appear to use search less often than experts.
- Knowing the area where the search will be done helps in finding the right keywords for the search.
- Users can often define the criteria that are important to their task; providing the possibility to sort makes usage more fluent.
- Knowing where they are in result pages helps users navigate and know where they are in the application.
- You should inform users where the search will fetch the information from: the current section of the site, the whole site, or the entire Web.
- Expert users should be offered an option to sort the search result (for example, by date/name).
- You should provide a possibility to narrow the search if the results list is too long.

- Users should be informed how many matches were found, but should see only part of the results per page (not over 15).
- The user should be informed where s/he is in the results (1/3; 2/3; 3/3) in the page title.

## 2.7 Frames

Frames are not useful in mobile browsers because they can be extremely difficult to handle on a small screen. Although frames are supported in some of the latest browser versions, it is not recommended to use them. If frames are used in the service, it is a good idea to preview the site using both Original and Narrow layouts to see how frames work in the browser.

## 2.8 Pop-Up Windows

Pop-up windows should be avoided on the Web page since they are supported only in a limited number of devices and are difficult to use in mobile browsers. In the Series 60 Platform, a pop-up window is displayed as a new page, not as a smaller page above another (like pop-ups on a PC browser), and this might confuse the user.

## 2.9 Rich Media

Series 80 Developer Platform 2.0 and Series 90 Developer Platform 2.0 support Macromedia Flash 5 as a plug-in. Audio and video files can be launched in their appropriate players. Since Flash content is not currently supported in various devices, developers should avoid embedded rich media in order to ensure successful mobile browsing. However, if Flash content is used on the Web page, the components should be small enough to be suitable for a mobile browser.

## 2.10 Images and Colors

Images are valuable in mobile services because they often "say" more than words in the same amount of space (just as with PC Web pages), while breaking up monotonous text blocks. However, images must be used with caution because they can reduce the usability of a service if employed poorly.

In many cases, users either turn off images to save time and money, or proceed to the next page without waiting for images to download. It is important to allow users to continue navigation before all images are downloaded. It is recommended to create a color scheme for the service that is in line with your brand.

### 2.10.1 Image optimization

The images used in applications designed for small devices should be small, simple, and clear. It is difficult to produce images that function well on all display sizes, and on both monochrome and color displays. Automatic resizing of black-and-white images makes them in most cases impossible to interpret. Images with text should be avoided since text might become unreadable in the Narrow layout.

Image height and width should be specified in the source code (either using CSS or HTML). This has several important advantages. Since the browser will know how much space an image uses beforehand, the page can get its final layout before the images finish downloading. This will give the impression that the page loads faster. If the browser is set not to download graphics at all, it still can keep the original layout. Finally, and perhaps most importantly, many images that are not fit for handheld devices can be avoided before they are downloaded. This makes a huge difference in download time and cost, ultimately determining whether the user will want to revisit your site, or visit it with graphics.

Avoid using splash screens. A splash screen is a service entrance page that typically shows a service or company logo. A small logo on your first page is sufficient to provide branding and familiarity with your service.

Analyze the target group of the service and use images accordingly. For example, animations and rich use of images may be appropriate for the youth market but not for a business application. Business-oriented users prefer highly optimized download times, but graphics that support the usage might give extra value to the service.

Make a few pixels of white space around images to prevent other images or surrounding text from "sticking" to image borders. This can be done with the CSS margin property.

To optimize the mobile experience, you may want to provide a colorful UI, even when using a low bandwidth. You may also choose to provide a more colorful/richer version for users who have a wideband connection. Remember that some proxies might reduce the number of colors in images.

Avoid using too many colors on the same page. Although colors can add "life" to a service, using too many colors defeats this purpose, and results in visual overload. Use colors consistently — for example, always use the same color for the same XHTML elements throughout the service.

Use colors that work for colorblind users. For example, using green and red next to each other might create difficulties for some users. Keep the color set simple in order to keep it clear.

#### 2.10.2 Color depths

Supported color depths:

The UI styles for Series 60 Developer Platform devices offer a 4,096 (12-bit in Series 60 Developer Platform 1.0) or 65,536 (16-bit in Series 60 Developer Platform 2.0) color screen.

The UI style for Series 80 Developer Platform 2.0 devices offers two 65,536 (16-bit) color screens; one on the cover side and one on the PDA side. However, the cover side display cannot be used for browsing.

The UI style for Series 90 Developer Platform 2.0 devices offers a 65,536 (16-bit) color screen.

#### 2.11 Tables

Avoid making tables wider than the phone display and avoid long tables.

If the service is used with a small display, the tables may not display properly and reading tables will be difficult. This might happen when using a mode that generates the content as one long column.

Even if there is a way to scroll sideways on a wide table, there may be problems knowing the current location in the table, and it is hard to predict what is in the cells that are not visible.

If a data column is very long, users may forget the headers and have problems interpreting the data.

#### 2.12 Forms

Forms are used to handle the input given to input elements or text area elements and also to handle selections made in select/option lists.

Forms can use CSS to provide positioning of elements using padding and margin properties. CSS borders are also supported, but these appear outside the already-existing black border on input, text area, and select elements. Some important issues to remember when designing forms are:

- Users seem to lose their feeling of control and forget their selection if the form is longer than two screenfuls or if it is on more than one page. Forms should be as compact as possible and extra empty space inside a form should be avoided.
- Users trust that the input they give is immediately saved. If there are several ways to proceed from the form page, the input may not be saved even though the user meant to save it.
- Consistency in form and the possibility of canceling actions give users a feeling of control.
- Users may not remember what they have entered or selected, or may have made unwanted selections.
- Users should not have several ways to proceed from a form page — there should be a single submit (OK) button. If multipage form is used, the submit button should be on the last page.
- Users should have a way to cancel the form filling and go back.
- Users should be able to review on one page all the information they have entered, especially if they have entered data on a multipage form.

### 2.12.1 Input fields

Inputting data is more challenging and time consuming on a mobile device than on a PC, so your site should be constructed with a minimum need for user input, especially text input. Links are still the preferable way to navigate in mobile browsers. Some important issues when designing input fields are:

- Use selection lists instead of input fields. Text input can be used when a selection list would be either too long or even impossible. Text entry on a mobile device may be difficult for novice users.
- Entering special characters may be difficult with some phone models; avoid any special characters, such as @, \*, etc. In many mobile phone type devices, number or alphabet inputting is faster than mixed alphanumeric input.
- Avoid requiring letters and numbers in the same input field (especially in a password field). When the password contains both numbers and letters, users in tests have entered the wrong password without noticing it.
- Avoid requiring case sensitivity (especially in password fields). In password fields, when input characters turn to asterisks, novice users may have difficulties remembering what they have input.
- Inform users what they should be typing in the field and in what format. It is possible to specify in the code whether numeric or alphabetic characters are accepted in the input field. Information about the format is important; for example, a date can be entered in one of several ways.
- Try to interpret user input as much as possible, and make error correction easy and quick. If input can be interpreted in several ways, provide users with alternatives instead of an empty field. Entering information and waiting for a response is inconvenient with mobile devices, and users get frustrated if error correction is too complicated. It is preferable to keep users in the same view, informing them visually about missing fields, etc.

### 3 Summary

This document discussed browsing in mobile and PC browsers and introduced the differences in these two domains. Even though, in general, mobile devices have some limitations in capabilities when compared to PCs, browsing services can be useful in both ends, when designed well enough. The key in making a browsing service succeed is taking the user into account already in the design phase and guaranteeing that the service provides the best user experience.

## 4 References

"Navigating In A Mobile XHTML Application." Kaikkonen, A., Roto, V., proceedings of CHI2003 conference

*Series 60 UI Style Guide*, [www.forum.nokia.com](http://www.forum.nokia.com)

*Series 80 UI Style Guide*, [www.forum.nokia.com](http://www.forum.nokia.com)

*Series 90 UI Style Guide*, [www.forum.nokia.com](http://www.forum.nokia.com)

*Optimizing Content For Nokia UIs And Developer Platforms*, [www.forum.nokia.com](http://www.forum.nokia.com)

*Quick Guide To Optimizing XHTML Mobile Profile Services*, [www.forum.nokia.com](http://www.forum.nokia.com)

"The Total User Experience: How To Make It Positive In Future Wireless Systems And Services." Mikko Kerttula, Timo Tokkonen, WWRF Annual Workshop in Paris, December 6-7, 2001

*XHTML Guidelines*, [www.forum.nokia.com](http://www.forum.nokia.com)

XHTML and HTML specifications, [www.w3c.org](http://www.w3c.org)