FORUM NOKIA

Mobile Store Locator Case Study

Version 1.0; April 7, 2008

Web technologies

NOKIA

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. 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 document at any time, without notice.

License

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

Contents

Change history

April 7, 2008	Version 1.0	Initial document release.

1 Introduction

Internet browsing is becoming increasingly popular as mobile handsets improve their browsing capabilities and the number of mobile-optimized sites increases. To support this development the major mobile companies launched the .mobi initiative, where guidelines for developing mobile sites were also created and published. Mobile Store Locator stands as a useful example of .mobi implementation with regard to both the technical implementation and the interest a mobile-optimized site can arouse.

Mobile Store Locator, or simply Store Locator (http://www.apps.nokia.co.uk/clubnokiaservice/), is the mobile service for finding your nearest Nokia service center. Nokia service centers consist of the stores selling Nokia mobile devices and accessories, but they are also the places where you can leave your Nokia device for repair or recycling. In its first stage the mobile service was available only in the United Kingdom.

Based on the success of that first version of Store Locator, the go-ahead was given to develop the service further. The next version (http://83.145.232.81/storelocator) was offered in mid 2007. More countries are supported in this version, and the Arabic language has been taken into use (the layout was modified to support the reading of Arabic from right to left). Country and language are fetched from the browser information, but the user can also switch between countries during the session.

It is a natural progression to provide this search service for mobile browsers after having it available for PC Internet browsers. It's no surprise that users want to access useful information on the Internet, even though there may be no PC nearby. A mobile device is often the only way to do this.

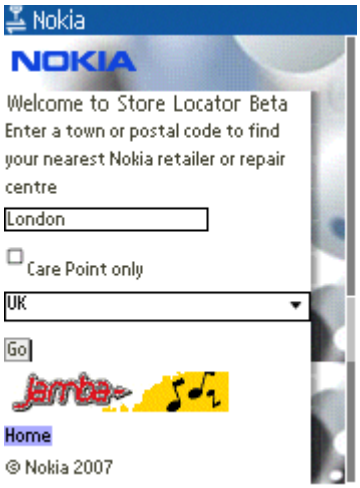

Figure 1: Nokia Store Locator

2 Technical implementation

2.1 Creating the design

Mobile-optimized Store Locator is essentially a simplified version of the service available for PC Internet browsers, not in its content, but in its layout. The service provides the same search capabilities, but the burden in data amounts is minimized and the layout fits within the smaller screen sizes.

The process of production began with the design of the mobile-optimized layout. Because the basic features and elements of the site were already predefined based on the existing site, the design was ready and preliminary tests performed with different screen sizes in two days.

There are differences between browsers with regard to the implementations, and between devices with regard to default fonts. Here are some general hints about what to do and what to avoid.

- Be aware of using bold text; the default bold text may be more attractive in some devices than in others.
- Tables are not well understood by mobile browsers; the same is true with div-tags outside the headers.
- Because the browsers in Nokia devices have a Back command, it is not necessary to have it separately in the pages themselves.
- Take advantage of CSS and the tester available from the Internet to test .mobi pages (http://mr.dev.mobi/).
- Pages should be identified as mobile pages with XHTML mobile profile. This way the mobile browser expects that the layout is in place and does not try to render the layout in its own way.

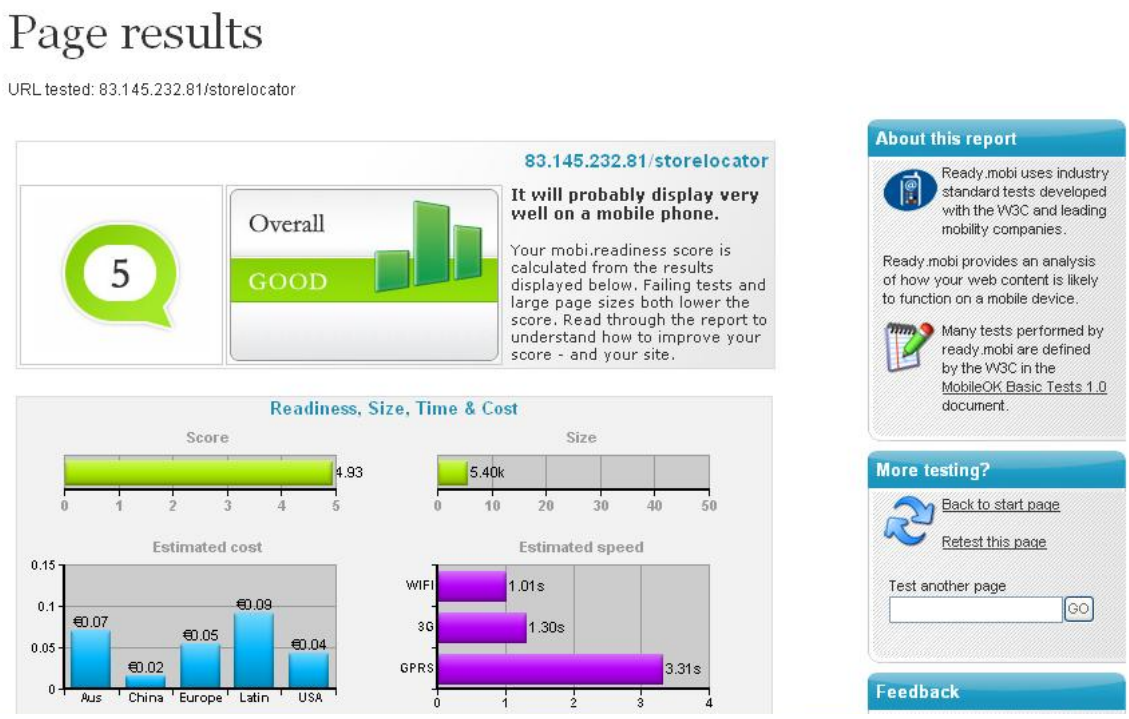

Figure 2: A view from the useful testing tool - http://mr.dev.mobi

2.2 Technical implementations

The site has a text field where users can write a city name or an area code. Store Locator then prints a list of stores according to the given search entry. When the user selects a store from the list, Store

Locator provides general information regarding the store and a map of its location. Data retrieval is done through a Web Services Description Language (WSDL) interface from a database.

If the store's phone number is available, the site provides a direct click-to-call link to the store.

Use the special URI scheme name wtai (wireless telephony application interface) and syntax, for example:

```
<a href="wtai://wp/mc;0208123456">Call tel. 0208 123 456</a><br />
```

It is also possible to save the GPS coordinates as a landmark; this feature is supported for Nokia N95 devices, which have GPS built in. By saving a landmark you actually save a landmark file (LMX file). LMS files are small and simple XML documents. The latest S60 3rd Edition devices understand the files as geographic information and enable their use in map and navigation software. is mapping software in free distribution capable of using this information. The software is preinstalled into certain Nokia devices; check Smart2go to find out whether it is available for download to your mobile device.

Because the browser, resolution, and screen size between different mobile phones change dramatically, the biggest challenge was to make the layout as flexible as possible. That's why it was important to be able to recognize the mobile phone type accessing the site. If the screen size is known, Store Locator defines the image sizes according to the screen size.

2.3 Code example of landmark implementation:

Example jsp page that creates the landmark XML:

```
<%@page  contentType="application/vnd.nokia.landmarkcollection+xml "
%><?xml version="1.0" encoding="UTF-8"?>
<lm:lmx xmlns:lm="http://www.nokia.com/schemas/location/landmarks/1/0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.nokia.com/schemas/location/landmarks/1/0
lmx.xsd">
<lm:landmarkCollection>
<lm:landmark>
<lm:name>${store.name}</lm:name>
<lm:coordinates>
<lm:latitude>${store.location.latitude}</lm:latitude>
<lm:longitude>${store.location.longitude}</lm:longitude>
</lm:coordinates>

<lm:addressInfo>
<lm:country>${store.country}</lm:country>
<lm:county>${store.county}</lm:county>
<lm:city>${store.city}</lm:city>
<lm:postalCode>${store.postcode}</lm:postalCode>
<lm:street>${store.address} ${store.address2}</lm:street>

<lm:phoneNumber>${store.phone}</lm:phoneNumber>
</lm:addressInfo>
<lm:category>
<lm:id>36000</lm:id>
<lm:name>Shopping</lm:name>
</lm:category>
</lm:landmark>
</lm:landmarkCollection>
</lm:lmx>
```

The maps were created in separate production runs for different screen sizes, based on the coordinates of the stores, the data provided by a map provider, and using specific software for creating map images. Many of the mobile devices on the market are still missing support for vector

images, such as SVG images. This requires the creation of several different image sizes in order to generate support for the majority of mobile devices.

2.4 Recognizing the device model and screen size

An important part of the implementation is the detection of devices based on their user agent profiles. The Internet browsers in Nokia devices give out user agent profiles. The profiles include a link to a full description of device capabilities. These descriptions also contain the size of the device screen.

You can use two complementary HTTP headers to recognize the device model.

- User-agent is an HTTP standard header. Depending on the device and browser, the user-agent may include the device model, or not. (http://en.wikipedia.org/wiki/User_agent)
- The UAProf header is a link to a UAProf file that describes in detail the capabilities of the device.

user-agent: Mozilla/5.0 (SymbianOS/9.2; U; Series60/3.1 NokiaN95/10.0.010; Profile/MIDP-2.0 Configuration/CLDC-1.1) AppleWebKit/413 (KHTML, like Gecko) Safari/413

uaprof: http://nds.nokia.com/uaprof/NN95-1r100.xml

One of the challenges is that a number of S60 3rd Edition browsers do not tell the exact device model (UAProf), but give only a general description of the device group to which the device belongs. To support these devices, a general solution has to be found with regard to the most important device capabilities. In Store Locator, and probably in most of the other sites as well, the defining feature was the screen size. The map images were created based on the smallest screen size of these devices, 176x208, but used in all of them. The result is imperfect, but it was appropriate at the time of the implementation. Beginning with Nokia N95 models, the exact model is revealed.

You can retrieve he screen size from a UAProf XML or an RDF file.

An example of an rdf file:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:prf="http://www.openmobilealliance.org/tech/profiles/UAPROF/ccppsc
hema-20021212#"
xmlns:mms="http://www.openmobilealliance.org/tech/profiles/MMS/ccppschem
a-20050301-MMS1.2#"
xmlns:pss6="http://www.3gpp.org/profiles/PSS/ccppschema-PSS6#">
   <rdf:Description rdf:ID="Profile">
   <prf:component>
      <rdf:Description rdf:ID="HardwarePlatform">
         ...
         <prf:ScreenSize>240x320</prf:ScreenSize>
         ...
      </rdf:Description>
   </prf:component>
   ...
   </rdf:Description>
</rdf:RDF>
```

Code example of fetching the screen size and using it in the implementation:

```
private int getScreenWidth(String xmlFile) {

String xmlParams[] = new String[1];
   xmlParams[0] = "screensize";
   Map<String, String> paramMap;

   Element element = cache.get("map-" + xmlFile);
 if (element != null) {
```

```
paramMap = (Map<String, String>) element.getValue();
 } else {
paramMap = fillMapFromXML(xmlParams, xmlFile);
element = new Element("map-" + xmlFile, paramMap);
cache.put(element);
 }

 return getWidth(paramMap.get("screensize"));
}

private int getWidth(String param) {
   int width = 0;
   if (param != null) {
      int end = (param.toLowerCase().indexOf("x"));
      String s = param.substring(0, end);
      width = Integer.parseInt(s);
   }
   return width;
}
```

Once the screen width is known, a corresponding map image is provided—not, perhaps, the most elegant solution, but useful at the time of the implementation.

```
public int getMapWidth(int screenWidth) {

 if (screenWidth < SCREEN_SIZE_SMALL) {
      return PICT_SIZE_SMALL;
   } else if (screenWidth < SCREEN_SIZE_MEDIUM) {
      return PICT_SIZE_MEDIUM;
   } else if (screenWidth < SCREEN_SIZE_LARGE) {
      return PICT_SIZE_LARGE;
   } else {
      return PICT_SIZE_EXTRA_LARGE;
   }
}
```

3 Statistics

The first version of the Store Locator service was published in early 2007 and was only available for locating stores in the U.K. Use increased rapidly after the service was launched, and the page got around 6,000 hits per day during the first part of 2007. Close to 100,000 queries were made each month. A couple of new countries were added in the middle of 2007, and interest has remained steady. These are impressive numbers for a service advertised with only a few banner links and limited to certain device models from Nokia .mobi pages. It demonstrates that if simple services that serve a clear need are provided as mobile sites, people will start to use them.

4 Future

GPS is becoming available in more device models. The maps and the landmarks will grow in importance to consumers and there might be efforts at some point to bring in more features for these devices. It seems that there is a bright future for Store Locator and for other mobile-optimized Internet sites.

Evaluate this resource

Please spare a moment to help us improve documentation quality and recognize the resources you find most valuable, by rating this resource.