# Porting iPhone web app to WRT on Nokia devices

*Recommended additional reading*: Porting iPhone web app to WRT - a porting example, which goes through a real porting example (porting a horoscope web app from iPhone to WRT), with downloadable WRT widget.

27 Jun 2010

## Introduction

This article provides information that iPhone web app developers need to know when switching to Web Runtime (WRT) development on Nokia devices based on Symbian OS. In this article, WRT refers to WRT on the following Nokia devices: S60 5th Edition devices, S60 3rd Edition, Feature Pack 2 devices, and selected S60 3rd Edition, Feature Pack 1 devices.

Porting to WRT is a great way to add volume to the user base and expand markets, with minimal investment. It is beneficial to target as many vendors as possible, and WRT offers an economical way to do it. Lots of iPhone web apps in the market work on Nokia devices based on Symbian OS, with only minor adjustments and optimizations to the screen (adjustments may need to be made to resources on the UI front end; nothing may need to change on the back end). Small modifications to your existing iPhone web application can increase the number of users and open up additional markets.

Optimising an iPhone-specific web page to S60 5th Edition touch devices is a straightforward process because the browsers are based on the same technology. By checking the device-specific user-agent header information on the server side, the web page can be optimised for each target platform.

This article assumes the reader has a good basic knowledge of web technologies (HTML, CSS, JavaScript, Ajax, XML, etc.). The real porting example is described in details in the article Porting iPhone web app to WRT - a porting example.

## Nokia Web Runtime (WRT)

For those familiar with iPhone development, the concept of a Nokia WRT widget is similar to using UIWebView in iPhone. You get to run local HTML, JavaScript™ and Cascading Style Sheets (CSS), provided with a system installer and an application icon. Unlike UIWebView applications on iPhone, Nokia WRT widgets do not require any native code or compilers to be created. Zip your web files and add a widget descriptor file with an application icon. Rename the zip file extension to .wgz, and you're ready to go.

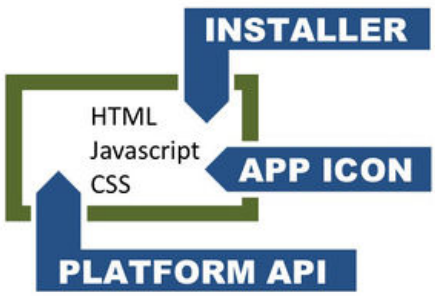
Widgets do not need Symbian signing either, so they are immediately ready to be published anywhere, including Ovi Store.

Click here for more information about WRT.

### Quick launcher to a web page

When your existing iPhone web application is running on a web server, Nokia WRT provides an easy way to add a quick launch icon on the user's mobile device. Simply create an HTML page to redirect the browser into your website. Add an application icon and widget descriptor file, package the files with ZIP-utility software, change the extension of the ZIP file to *.wgz and you have a WRT widget in its simplest form. Users can download the widget, install it, and add the shortcut on the application menu and idle screen.

### Stand-alone applications

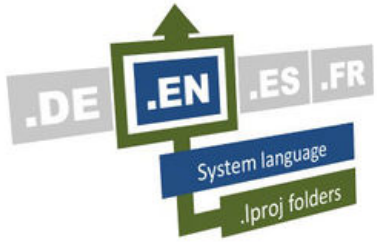



Instead of running web content from the server, you can embed all web files and resources into the widget package. Nokia WRT offers an excellent framework for running stand-alone web applications. As a WRT widget, your application can be used offline


(C) Copyright Nokia 2014. All rights reserved.

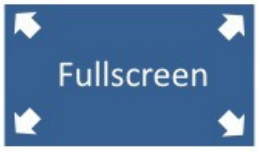
and connect to the web only for content updates. The resulting application starts faster and is not limited to bandwidth issues. And if you are interested in going a step further in WRT, a stand-alone app opens up great opportunities to extend your concept, as detailed below:

## Easy localisation



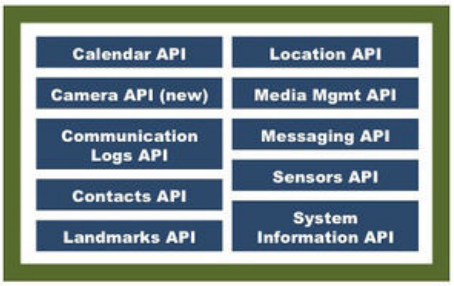

Nokia WRT widgets provide an easy, built-in solution for localising an application. Runtime selects a localisation file automatically according to the current device language.

## Application-style window




Nokia WRT widgets are always run in independent windows, in fullscreen mode. Instead of just hiding browser controls, the widget framework provides a way to take full control of familiar options menus and softkeys.

## Access to Platform Services API




Nokia Platform Services APIs open access to device features for JavaScript. Everything from address book contacts to inbox messages and GPS location are available through a simple JavaScript interface. Nokia Platform Services 2.0 now includes support for camera features as well.

## Ajax without limitations

To make server code unnecessary, WRT enables Ajax connections to all hosts. You can use JavaScript to fetch content from multiple domains without safety restrictions.

## Extended use cases

Nokia WRT provides events for recognising when the widget is entering to the background. You can use this to optimise the user experience and to perform updates and refreshes only when it is most suitable for the user. The latest S60 5th Edition devices, like Nokia N97, also support a homescreen view to enhance your application's visibility to the user. You can add a summary of the latest news, sports results, or anything else, to be displayed as a quick preview of fullscreen content.


(C) Copyright Nokia 2014. All rights reserved.

# Porting the user interface

Both iPhone and S60 5th Edition device browsers use WebKit engine. This is good news for developers, because rendering and processing of web content is similar on both platforms.

Screen resolution for S60 5th Edition devices is 640 x 360 pixels, whereas in iPhone the screen is 320 x 480 pixels. This should be taken into consideration when adjusting the size of the touch areas, such as input fields and buttons. For more information, see Nokia Developer.

Commonly supported features include:

- HTML 4.01, XHTML 1.0/1.1, and XHTML Mobile Profile (MP)
- CSS 2.1
- JavaScript 1.5
- XMLHttpRequest object for Ajax
- URL schemes for short message service (SMS), mail sending, phone call, Maps, and YouTube

Because iPhone is based on a newer version of WebKit, S60 5th Edition devices do not support the use of the following:

- CSS 3.0
- HTML5 features (that is, offline functionality)
- WebKit CSS visual effects (such as rounded corners for divs)

Note that client-side storage, introduced in the HTML 5 specification, can be replaced in Nokia devices based on Symbian OS by using the Platform Service API's setPreferenceForKey method.

## iPhone-specific features

When porting a web application developed for iPhone, certain features need to be focused on for Nokia devices, such as the following:

### Touch events

Current Symbian devices do not support multitouch as they have resistive touch screens (Except Nokia N8). Therefore, none of the iPhone-specific events that begin with 'ontouch*' and 'ongesture*' are supported. As a replacement, use the standard events, such as onclick, onmousedown, etc.

### Viewport metadata

iPhone web apps support the viewport metadata to give information about the zoom and content area that you want to use in your website. The browser for Nokia devices based on Symbian OS automatically adjusts the content to the screen, so there is no support for this feature.

### Canvas vector drawing

Canvas vector drawing is currently not supported on Nokia devices. As a replacement, consider using Flash Lite from Adobe or


(C) Copyright Nokia 2014. All rights reserved.

SVG Tiny (SVGT) images. Canvas tags can be exported to Flash ActionScript 2.0 with only minor modifications.

### Custom icon

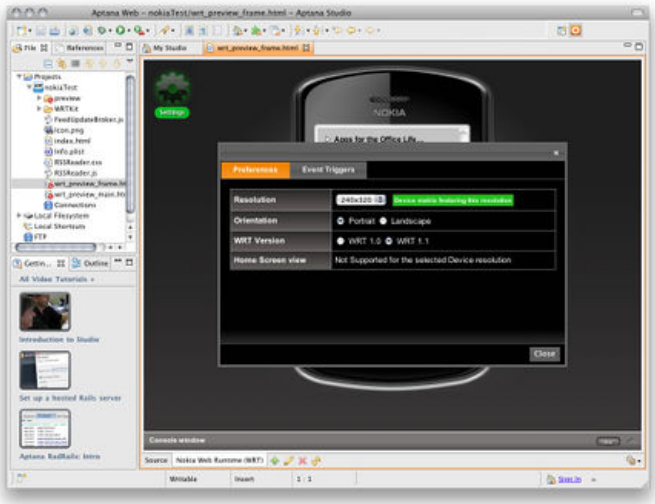
Homescreen icons for iPhone are defined by a meta tag. To use the same icon in your WRT widget, rename the icon to icon.png and copy it into the root folder of the widget. Nokia devices can use directly the default iPhone PNG icon size of 57 x 57 pixels. Information on how to create a perfect icon.png image for Nokia Web apps, see How to create the perfect WRT Icon

Hint: If you are just making a website with no WRT widget, try adding the following meta tag for Nokia devices to define a browser bookmark icon: <link rel="shortcut icon" href="/path_to_icon.png">

Both iPhone and Nokia-specific meta tags can be used in the same HTML page. iPhone will use the first one and Nokia devices will use the second. With Nokia devices, you can also use a 16 x 16 ICO file to reduce the download size.

## Developer tools

Nokia has developed Nokia Web Tools] (based on Aptana) that let developers easily preview the projects on an emulator. If you created the original iPhone app using Dashcode, the project can be imported.



Nokia Web Tools are available at no cost for both PC. The Nokia WRT plug-in is also available for Adobe Dreamweaver and Microsoft Visual Studio.

## Example

See Porting iPhone web app to WRT - a porting example.

## Resources


(C) Copyright Nokia 2014. All rights reserved.

## Nokia Developer's remote device access services

Want to make sure your application works with the latest Nokia devices? Nokia Developer's Remote Device Access services enable access to a live device over the internet. This makes it possible for developers to reduce the amount of effort spent on testing, demoing, and content and application development.

## Nokia Mobile Web Templates

Nokia Mobile Web Templates provide a series of custom, styled, and tested mobile web components. Match the components or use the full series to completely customise your mobile website or application.

## Guarana UI Library

The Guarana UI Library is a jQuery-based set of customisable UI elements, application frameworks, and templates that can be used to create user interfaces in Nokia's WRT platform.

## ApiBridge

Camera features are not part of Nokia Platform Service prior to version 2.0. Meanwhile, ApiBridge implements the same functionality to take and upload photos on S60 devices based on Symbian OS.

# Web browsing on Nokia devices based on Symbian OS

In addition to offering WRT widgets, Nokia devices based on Symbian platform provide a rich mobile web browsing experience through a WebKit-based browser. Optimizing an iPhone-specific web page to S60 5th Edition, for example, is straightforward because both platforms support standard web technologies such as HTML, CSS, and JavaScript.

However, there are some differences between devices, so when designing web pages, the supported browser and device features should be taken into account and serve the optimized content to target devices. This can be done easily by checking the UAProfile or user-agent header information that is unique to each device model. When the functionality is built in the server side, all that is needed is to optimize the web page to be target platform/device specific, by changing the supported resolution and UI elements. To make mobile website or application customization faster and easier, Nokia Developer offers Nokia Mobile Web Templates, a series of custom, tested mobile web styles and components targeted at mobile web developers and designers.

Copyright © 2010 Nokia Corporation. All rights reserved. Nokia, Nokia Developer, and Maemo are registered trademarks of Nokia Corporation. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. Apple, iPhone, iPod and MacBook are trademarks or registered trademarks of Apple Inc. Android is a trademark of Google Inc. Other product and company names mentioned herein may be trademarks or trade names of their respective owners.


(C) Copyright Nokia 2014. All rights reserved.