Using jQuery Mobile in Nokia Asha Web Apps

This article explains how to use jQuery Mobile to create highly portable Nokia Asha Web Apps.

Note: This is an entry in the Asha Touch Competition 2012Q3

Warning: JQuery is not fully supported on the browser; developers must select components very carefully to ensure only supported elements are used. Note that an app using unsupported elements may run on *Local Preview* but will fail on *Cloud Preview* (and on a real device).

This article needs to be updated: If you found this article useful, please fix the problems below then delete the {{ArticleNeedsUpdate}} template from the article to remove this warning.

Reasons: hamishwillee (11 Sep 2012)
This code does not run in Cloud Preview due to unsupported elements in JQuery (e.g. `data_role` does not work). It should be updated to use supported elements.

Introduction

The jQuery library has become the standard tool for web developers when building new web apps. Now that large part of web usage is coming from mobile devices its great to see that jQuery is providing tools to create web sites or applications that work with majority of phones and tablets.

The major benefit of using jQuery Mobile is that you can have a common application engine, and change the app appearance simply by dropping in a new theme. As a result you'll need to write less code to make your web app to work with several devices.

In this tutorial I'll show you how you can create simple (or complex) application easily with jQuery Mobile and how this application can be easily be developed for multiple platforms (Series 40 and Windows Phone). The project goal is to create Pocket (Read it Later) client. The application's name is *Pouch* and it is available along with full source code here: Pouch.zip.

Benefits of using jQuery Mobile over plain HTML

There are many benefits on using jQuery Mobile, to name a few:

- Themes - You can easily theme your application with jQuery Mobile ThemeRoller. There are also good 3rd party themes like Windows Phone theme as shown later in this article.
- Plugins - There are probably hundreds of jQuery Mobile plugins available. There are for example implementations for different components like *date box* and for different gestures like *swipe to delete*.
- Fast prototyping - Codiga have created an online editor which you can use to build prototypes really quickly - you can check demo version of this on jQuery Mobile site.
- Large community - There is large community behind jQuery Mobile and therefore new plugins, tutorials, themes are continuously being created.

Requirements

We'll start by downloading and installing tools and libraries:

- Nokia Web Tools 2.0 - Great IDE and emulator for testing web apps on Nokia Asha devices
- jQuery Mobile - jQuery Mobile library (1.1.1 when writing this tutorial)
- Windows Phone Theme - Optional theme for Windows Phone

Create a new project

It really doesn't require much to use jQuery Mobile libraries in your Web Application. Set up a new project as usual:

1. Start Nokia WDE
2. Select **File > New > Series 40 web app**
3. Select **Basic Project**
4. Enter project name, version etc. information and click **Finish**

(C) Copyright Nokia 2014. All rights reserved.

5. New project is created

Next you can delete the **s40-theme** directory since the jQuery Mobile theme works well with Nokia Asha devices.

Add jQuery files to your project

Unzip your jQuery files and copy **jquery.mobile-1.1.1.min.js** and **jquery-1.7.2.min.js** to your project in folder **js**. Add the theme files to the **css** folder. Then add the following to your `<head>` tag:

```
...
 <link rel="stylesheet" href="css/jquery.mobile-1.1.1.min.css" />
 <link rel="stylesheet" href="css/jquery.mobile.theme-1.1.1.min.css" />
 <link rel="stylesheet" href="css/jquery.mobile.structure-1.1.1.min.css" />
 <script src="js/jquery-1.7.2.min.js"></script>
 <script src="js/jquery.mobile-1.1.1.min.js"></script>
...
```

Main Application Structure

The following code snippet shows a simple application structure that you might define in **index.html** (the code is explained afterwards):

```
<body>
<div data-role="page" id="loginpage">
 <div data-role="header"><h1>Pouch</h1></div>
 <div data-role="content">
  <div data-role="fieldcontain" class="ui-field-contain ui-body ui-br">
    Enter Pocket credentials. Visit
    <a href="http://getpocket.com">http://getpocket.com</a> if
    you don't have account yet.
  </div>
  <div data-role="fieldcontain">
    <label for="username" class="ui-input-text">Email:</label>
   <input type="text" name="username" id="username" value="">
    <label for="password" class="ui-input-text">Password:</label>
   <input type="password" name="password" id="password" value="">
    <button id="loginbutton" data-role="button">Login</button>
  </div>
  <div data-role="fieldcontain" class="ui-field-contain ui-body ui-br" style="text-
align:right;">
    &copy; 2012 Tommi Laukkanen
  </div>
 </div><!-- /content -->
</div><!-- /page -->

<div data-role="page" id="menu">
 <div data-role="header"><h1>Home</h1></div>
 <div data-role="content">
  <ul data-role="listview">
   <li><a id="addnewbutton">Add new</a></li>
   <li><a id="allitemsbutton">All items</a></li>
   <li><a id="unreaditemsbutton">Favorites</a></li>
   <li><a id="logout-button" href="#loginpage" data-transition="slide">Logout</a></li>
  </ul>
 </div>
</div> <!-- /menu page -->
```


(C) Copyright Nokia 2014. All rights reserved.

```
<div data-role="page" id="itemlistpage">
 <div data-role="header"><h1 style="white-space:normal;">Items</h1></div>
 <div data-role="content">
  <ul id="itemlist" data-role="listview">
   <li>Loading...</li>
  </ul>
  <button data-role="button" id="loadmorebutton">More</button>
  <a data-role="button" href="#menu">Back</a>
 </div>
</div> <!-- /item list page -->

<div data-role="page" id="newitempage">
 <div data-role="content">
  <div data-role="fieldcontain">
    <label for="newurl" class="ui-input-text">New URL:</label>
   <input type="text" name="newurl" id="newurl" value="">
    <button id="addbutton" data-role="button">Add</button>
  </div>
  <a id="backtolistbutton" data-role="button" href="#menu" data-
transition="slide">Back</a>
 </div>
</div> <!-- /item details page -->

</body>
```

First thing you'll notice is that application markup is divided into different pages. The `data-role` attribute are used to define semantics for the markup and in jQuery Mobile we can define which sections contain application pages, page content, page header, page footer and so on. You might have also noticed that I didn't use many *class* attributes, since jQuery Mobile uses the data attributes for styling different components.

Here are common data-role types that I have used in this sample application:

- `page` - Defines that section is contains components for one page - - Sample:

```
<div data-role="page" id="homepage">...</div>
```

- `content` - Defines that section contains main content for the page - Sample:

```
<div data-role="content">...</div>
```

- `header` - Header of page - Sample:

```
<div data-role="header"><h1>Home</h1></div>
```

- `footer` - Footer of page - Sample:

```
<div data-role="footer">...</div>
```

- `button` - Button that does something - Sample:

```
<button id="addbutton" data-role="button">Add</button>
```

- `listview` - List of items - Sample:

(C) Copyright Nokia 2014. All rights reserved.

```
<ul data-role="listview">
 <li>Item1</li>
 <li>Item2</li>
</ul>
```

You can read more information about the data attributes and possibility of applying options to each data attribute from jQuery Mobile documentation: data-attribute reference

Navigation between pages

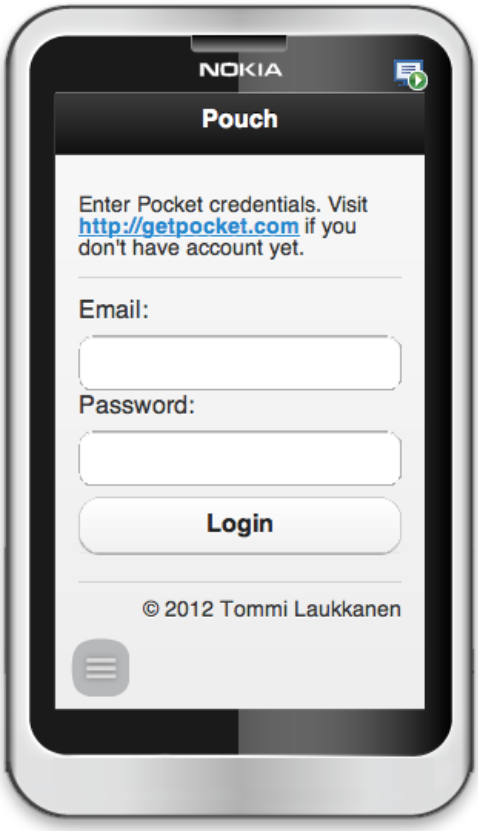
You can build basic navigation between your pages by binding the following script to your action:

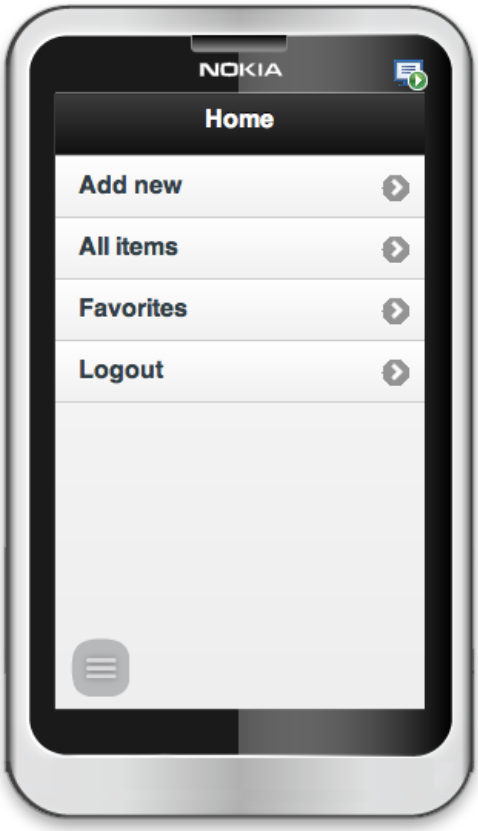
```
// Switch to page newitempage with sliding transition
$.mobile.changePage($("#newitempage"), {transition:"slide"});
```

When focusing on Series 40 deployment you could optimise the performance by leaving out the transition animation. This is because on Series 40 browser engine the JavaScript is executed on server side.

Further reading

That is all you need to start using jQuery and jQuery Mobile in your web app. I don't go into details since great documentation for jQuery Mobile can be found at jquerymobile.com.





Switching to the Windows Phone theme

One of the greatest benefits of using jQuery Mobile is the theming support so that you'll only need to write the application logic once and then just change the theme if you'd like to run your application in different platforms and look more native.

Assuming you first downloaded the Windows Phone theme from here, then:

1. Copy metro theme files to **metro** folder
2. Change your application to use new theme:

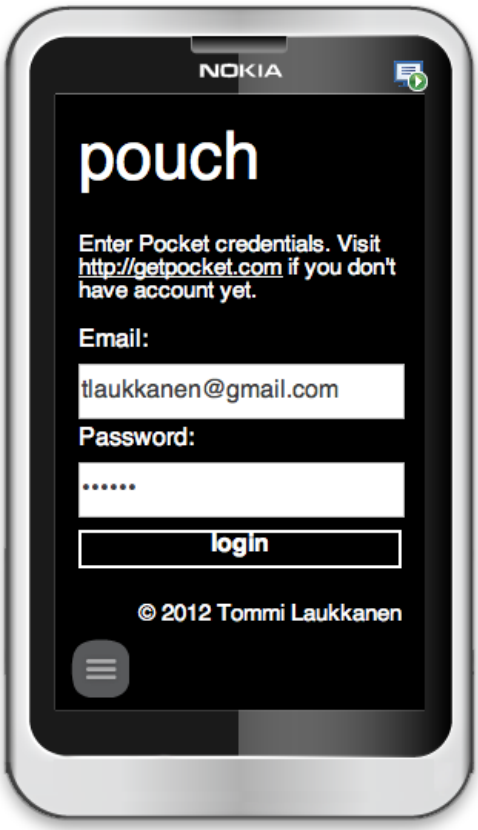
```
<link rel="stylesheet" href="metro/jquery.mobile.metro.theme.css" />
```

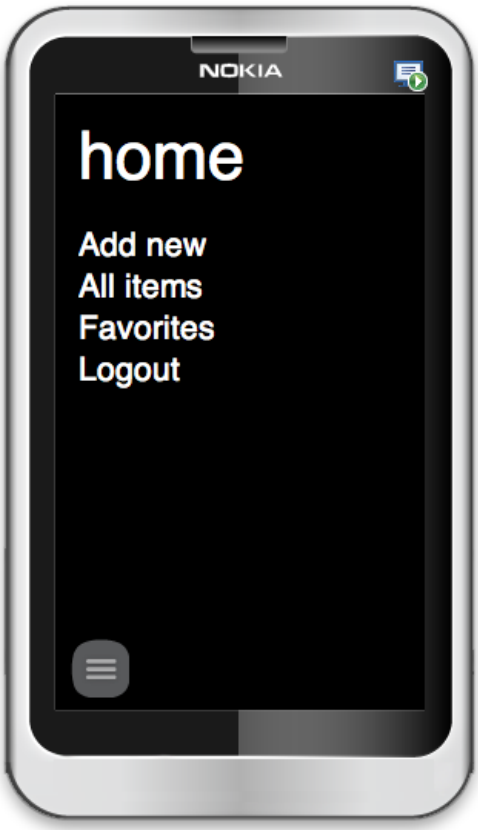

(C) Copyright Nokia 2014. All rights reserved.

```
<script src="metro/jquery.mobile.metro.theme.init.js"></script>
```

And then you are ready! Then you'll need to use Windows Phone SDK and for example PhoneGap to compile your Web App to "native" Windows Phone application.

You can also review the Windows Phone theme in your Series 40 web app emulator:





Last Words

jQuery Mobile is powerful framework but still lacks functionality which could be handy in Series 40 applications. In the future we,

(C) Copyright Nokia 2014. All rights reserved.

Series 40 community, could create a Series 40 theme to jQuery Mobile which would be as easy to plug-in as current Windows Phone theme.

(C) Copyright Nokia 2014. All rights reserved.