Archived:How to use MWS to provide local device functionality

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

The 'Mobile web server' was discontinued in February 2010. See here for more information.

Many Widset/Widget applications are limited by the fact that their APIs does not provide support for much of the functionality that is available to native applications. This could be functions like:

- Getting locations using the GPS module
- Taking photos with the camera
- Bluetooth connections
- Accessing Contacts, Inbox or External Applications
- Sending SMS
- and much more..

As for now, according to the Web Developers' Library- messaging, media and location are available since WRT 1.1. An easy fix to obtain these functions in Widgets is to use the Archived:Mobile Web Server to provide these functions.

```
This can be achieved in a few steps:
```

Connecting to MWS from a Widset/Widget

The examples below shows how to get information from MWS

* **Widget Example**

```
function queryMWS()
{
   // This code uses Prototype, but a standard javascript request will work just as well
   new Ajax.Request("http://localhost/hello.py",
   {
      method: 'get',
      asynchronous: 'false',
      onSuccess: function(transport, json)
      {
         return true;
      },
      onFailure: function()
      {
         return false;
      }
    });
}
```

* **Widset Example**

```
void queryMWS()
{
   // Prepare the URL.
   String URL = "https://MWS_username:MWS_password@MWS_username.mymobilesite.net/hello.py";

   // Fetch from the URL.
   call(null, "httpLocalhost", "get", ["url" => URL], success, failure);

   void success(Object state, Value ret)
   {
      setBubble(null, "The server successfully returned " + ret);
      return;
   }

   void failure(Object state, String error)
   {
      setBubble(null, "Local server is not responding properly.");
      return;
   }
}
```

Using Python modules with MWS to provide local device functionality

Two files is necessary to get python executing on MWS:

***ht.acl**

Invalid language.

You need to specify a language like this: `<source lang="html4strict">...</source>`

(C) Copyright Nokia 2013. All rights reserved.

Supported languages for syntax highlighting:

4cs, 6502acme, 6502kickass, 6502tasm, 68000devpac, abap, actionscript, actionscript3, ada, algol68, apache, applescript, apt_sources, asm, asp, autoconf, autohotkey, autoit, avisynth, awk, bascomavr, bash, basic4gl, bf, bibtex, blitzbasic, bnf, boo, c, c_loadrunner, c_mac, caddcl, cadlisp, cfdg, cfm, chaiscript, cil, clojure, cmake, cobol, coffeescript, cpp, cpp-qt, csharp, css, cuesheet, d, dcs, delphi, diff, div, dos, dot, e, ecmascript, eiffel, email, epc, erlang, euphoria, f1, falcon, fo, fortran, freebasic, fsharp, gambas, gdb, genero, genie, gettext, glsl, gml, gnuplot, go, groovy, gwbasic, haskell, hicest, hq9plus, html4strict, html5, icon, idl, ini, inno, intercal, io, j, java, java5, javascript, jquery, kixtart, klonec, klonecpp, latex, lb, lisp, llvm, locobasic, logtalk, lolcode, lotusformulas, lotusscript, lscript, lsl2, lua, m68k, magiksf, make, mapbasic, matlab, mirc, mmix, modula2, modula3, mpasm, mxml, mysql, newlisp, nsis, oberon2, objc, objeck, ocaml, ocaml-brief, oobas, oracle11, oracle8, oxygene, oz, pascal, pcre, per, perl, perl6, pf, php, php-brief, pic16, pike, pixelbender, pli, plsql, postgresql, povray, powerbuilder, powershell, proftpd, progress, prolog, properties, providex, purebasic, pycon, python, q, qbasic, rails, rebol, reg, robots, rpmspec, rsplus, ruby, sas, scala, scheme, scilab, sdlbasic, smalltalk, smarty, sql, systemverilog, tcl, teraterm, text, thinbasic, tsql, typoscript, unicon, uscript, vala, vb, vbnet, verilog, vhdl, vim, visualfoxpro, visualprolog, whitespace, whois, winbatch, xbasic, xml, xorg_conf, xpp, yaml, z80, zxbasic

```
AuthType Basic
AuthName 'Hello'
AuthBasicProvider file
AuthUserFile conf/passwords.txt
AuthGroupFile conf/groups.txt

# If you want to give access to all users
#Require valid-user

# If you want to give access to only yourself or some groups (add all needed groups delimited with space)
Require group admin

# If you want to give access to some users (add all needed users delimited with space)
#Require user quest

AddHandler mod_python .py
PythonHandler hello
PythonDebug On
Options None

Order Deny,Allow
Allow from all

<FilesMatch "\.(pyc)$">
   Deny from all
</FilesMatch>
```

***hello.py**

Invalid language.

You need to specify a language like this: `<source lang="html4strict">...</source>`

Supported languages for syntax highlighting:

4cs, 6502acme, 6502kickass, 6502tasm, 68000devpac, abap, actionscript, actionscript3, ada, algol68, apache, applescript, apt_sources, asm, asp, autoconf, autohotkey, autoit, avisynth, awk, bascomavr, bash, basic4gl, bf, bibtex, blitzbasic, bnf, boo, c, c_loadrunner, c_mac, caddcl, cadlisp, cfdg, cfm, chaiscript, cil, clojure, cmake, cobol, coffeescript, cpp, cpp-qt, csharp, css, cuesheet, d, dcs, delphi, diff, div, dos, dot, e, ecmascript, eiffel, email, epc, erlang, euphoria, f1, falcon, fo, fortran, freebasic, fsharp, gambas, gdb, genero, genie, gettext, glsl, gml, gnuplot, go, groovy, gwbasic, haskell, hicest, hq9plus, html4strict, html5, icon, idl, ini, inno, intercal, io, j, java, java5, javascript, jquery, kixtart, klonec, klonecpp, latex, lb, lisp, llvm, locobasic, logtalk, lolcode, lotusformulas, lotusscript, lscript, lsl2, lua, m68k, magiksf, make, mapbasic, matlab, mirc, mmix, modula2, modula3, mpasm, mxml, mysql, newlisp, nsis, oberon2, objc, objeck, ocaml, ocaml-brief, oobas, oracle11, oracle8, oxygene, oz, pascal, pcre, per, perl, perl6, pf, php, php-brief, pic16, pike, pixelbender, pli, plsql, postgresql, povray, powerbuilder, powershell, proftpd, progress, prolog, properties, providex, purebasic, pycon, python, q, qbasic, rails, rebol, reg, robots, rpmspec, rsplus, ruby, sas, scala, scheme, scilab, sdlbasic, smalltalk, smarty, sql, systemverilog, tcl, teraterm, text, thinbasic, tsql, typoscript, unicon, uscript, vala, vb, vbnet, verilog, vhdl, vim, visualfoxpro, visualprolog, whitespace, whois, winbatch, xbasic, xml, xorg_conf, xpp, yaml, z80, zxbasic

```
def handler(req):
   from mod_python import apache
   import httplib

   try:
      #
      # Execute any python code here
      #
      req.write("Hello World")
      req.status = 200
      return apache.OK

      except Exception, e:
         req.status = 404
  return apache.OK
```

Good examples of how to use python modules can be found at the Mobile Python Book and Python section.

* **Example of using python to get the current location**

Invalid language.

You need to specify a language like this: `<source lang="html4strict">...</source>`

(C) Copyright Nokia 2013. All rights reserved.

Supported languages for syntax highlighting:

4cs, 6502acme, 6502kickass, 6502tasm, 68000devpac, abap, actionscript, actionscript3, ada, algol68, apache, applescript, apt_sources, asm, asp, autoconf, autohotkey, autoit, avisynth, awk, bascomavr, bash, basic4gl, bf, bibtex, blitzbasic, bnf, boo, c, c_loadrunner, c_mac, caddcl, cadlisp, cfdg, cfm, chaiscript, cil, clojure, cmake, cobol, coffeescript, cpp, cpp-qt, csharp, css, cuesheet, d, dcs, delphi, diff, div, dos, dot, e, ecmascript, eiffel, email, epc, erlang, euphoria, f1, falcon, fo, fortran, freebasic, fsharp, gambas, gdb, genero, genie, gettext, glsl, gml, gnuplot, go, groovy, gwbasic, haskell, hicest, hq9plus, html4strict, html5, icon, idl, ini, inno, intercal, io, j, java, java5, javascript, jquery, kixtart, klonec, klonecpp, latex, lb, lisp, llvm, locobasic, logtalk, lolcode, lotusformulas, lotusscript, lscript, lsl2, lua, m68k, magiksf, make, mapbasic, matlab, mirc, mmix, modula2, modula3, mpasm, mxml, mysql, newlisp, nsis, oberon2, objc, objeck, ocaml, ocaml-brief, oobas, oracle11, oracle8, oxygene, oz, pascal, pcre, per, perl, perl6, pf, php, php-brief, pic16, pike, pixelbender, pli, plsql, postgresql, povray, powerbuilder, powershell, proftpd, progress, prolog, properties, providex, purebasic, pycon, python, q, qbasic, rails, rebol, reg, robots, rpmspec, rsplus, ruby, sas, scala, scheme, scilab, sdlbasic, smalltalk, smarty, sql, systemverilog, tcl, teraterm, text, thinbasic, tsql, typoscript, unicon, uscript, vala, vb, vbnet, verilog, vhdl, vim, visualfoxpro, visualprolog, whitespace, whois, winbatch, xbasic, xml, xorg_conf, xpp, yaml, z80, zxbasic

```
def handler(req):

   from mod_python import apache
   import location
   import positioning

   req.content_type = 'text/xml'

   req.write("<?xml version='1.0' encoding='utf-8'?>")
   req.write("<twitnflick>")
   req.write("<modules count='" + str(len(positioning.modules())) + "' default='"
    + str(positioning.default_module()) + "'>")
   for x in positioning.modules():
      req.write("<module id='" + str(x['id']) + "'>")
      req.write("<name>" + x['name'] + "</name>")
      req.write("<available>" + str(x['available']) + "</available>")
      req.write("</module>")

   req.write("</modules>")

   positioning.set_requestors([{"type":"service","format":"application","data":"test"}])
 gpspos = positioning.position()
   req.write("<position>")
   req.write("<latitude>" + str(gpspos['position']['latitude']) + "</latitude>")
   req.write("<longitude>" + str(gpspos['position']['longitude']) + "</longitude>")
   req.write("<altitude>" + str(gpspos['position']['altitude']) + "</altitude>")
   req.write("<horizontal_accuracy>" + str(gpspos['position']['horizontal_accuracy'])
    + "</horizontal_accuracy>")
   req.write("<vertical_accuracy>" + str(gpspos['position']['vertical_accuracy'])
    + "</vertical_accuracy>")
   req.write("</position>")

   req.write("</twitnflick>")

   return apache.OK
```


(C) Copyright Nokia 2013. All rights reserved.

 (C) Copyright Nokia 2013. All rights reserved.