

PSP HACKS™

*Tips & Tools for Your Mobile Gaming
and Entertainment Handheld*



**HACK
#46****Add a JavaScript Keyboard to Your PSP Web Portal**

Make your web surfing activities a whole lot easier by providing an alternate QWERTY keyboard in place of the PSP's internal keypad, making your PSP-based web surfing as simple as Up-X-Down-X-Down-Right-X-Down-Down-Down-X.

"Find Yourself a PSP Web Browser" [Hack #41] provides an extra dimension that allows even the most novice user the ability to surf the Internet from her PSP. However, getting online is the easy part! Once there, you quickly discover that web surfing is not as simple as typing in a URL or clicking on a link. Since the PSP is missing a standard keyboard, you are forced to use its internal keypad that has even less functionality than a cell phone. If you are planning on using the browser hack for more than just goofing around, then this form of data entry quickly gets old.

To help overcome this challenge, I offer you a JavaScript keyboard that not only provides you with an easy-to-use QWERTY keyboard with all your standard alphanumeric characters, but also gives you some extra features, such as one-button `http://` entry and a URL checker to make sure that you typed everything correctly before attempting to load the URL.

The Code

You'll have to add some JavaScript to your `/wipeout/index.html` page that is loaded when the browser hack is performed using *Wipeout Pure*.



This only works if you are using your own custom `index.html` page for the browser hack. To do this, you will need a personal web server that you control and that is accessible to your PSP when the browser hack is executed [Hack #42].

Open the `index.html` page in your favorite editor and enter the following code. You will need to replace everything between and including the `<body>` tags, or the script will not work as designed. Feel free to customize the script/html as you see fit. Once the code is pasted in, save the page and upload it to the web server.

```
<body bgcolor="black" onLoad="frmKeypad.http.focus()">
<script>
var theurl="";
function GO(){
    window.location=theurl;
}
</script>
<form name="frmKeypad">
    <table><tr><td align="center">
```

```
<input type="button" value="0" onClick="theurl=theurl+'0'">
<input type="button" value="1" onClick="theurl=theurl+'1'">
<input type="button" value="2" onClick="theurl=theurl+'2'">
<input type="button" value="3" onClick="theurl=theurl+'3'">
<input type="button" value="4" onClick="theurl=theurl+'4'">
<input type="button" value="5" onClick="theurl=theurl+'5'">
<input type="button" value="6" onClick="theurl=theurl+'6'">
<input type="button" value="7" onClick="theurl=theurl+'7'">
<input type="button" value="8" onClick="theurl=theurl+'8'">
<input type="button" value="9" onClick="theurl=theurl+'9'"><br>
<input type="button" value="q" onClick="theurl=theurl+'q'">
<input type="button" value="w" onClick="theurl=theurl+'w'">
<input type="button" value="e" onClick="theurl=theurl+'e'">
<input type="button" value="r" onClick="theurl=theurl+'r'">
<input type="button" value="t" onClick="theurl=theurl+'t'">
<input type="button" value="y" onClick="theurl=theurl+'y'">
<input type="button" value="u" onClick="theurl=theurl+'u'">
<input type="button" value="i" onClick="theurl=theurl+'i'">
<input type="button" value="o" onClick="theurl=theurl+'o'">
<input type="button" value="p" onClick="theurl=theurl+'p'">
<input type="button" value="q" onClick="theurl=theurl+'q'"><br>
<input type="button" value="a" onClick="theurl=theurl+'a'">
<input type="button" value="s" onClick="theurl=theurl+'s'">
<input type="button" value="d" onClick="theurl=theurl+'d'">
<input type="button" value="f" onClick="theurl=theurl+'f'">
<input type="button" value="g" onClick="theurl=theurl+'g'">
<input type="button" value="h" onClick="theurl=theurl+'h'">
<input type="button" value="j" onClick="theurl=theurl+'j'">
<input type="button" value="k" onClick="theurl=theurl+'k'">
<input type="button" value="l" onClick="theurl=theurl+'l'"><br>
<input type="button" value="z" onClick="theurl=theurl+'z'">
<input type="button" value="x" onClick="theurl=theurl+'x'">
<input type="button" value="c" onClick="theurl=theurl+'c'">
<input type="button" value="v" onClick="theurl=theurl+'v'">
<input type="button" value="b" onClick="theurl=theurl+'b'">
<input type="button" value="n" onClick="theurl=theurl+'n'">
<input type="button" value="m" onClick="theurl=theurl+'m'">
<input type="button" value="a" onClick="theurl=theurl+'a'"><br>
<input type="button"
  value="http://" name="http" onClick="theurl=theurl+'http://'">
<input type="button" value="." onClick="theurl=theurl+'.'">
<input type="button" value="/" onClick="theurl=theurl+'/'">
<input type="button" value=":" onClick="theurl=theurl+':'"><br>
<input type="button" value="?" onClick="alert('Type the theurls using
your arrows and the X button. Use the URL Check button to be sure you
have the right address. Once complete, hit the GO button.')">
<input type="button" value="Clear" onClick="theurl=''">
<input type="button" value="URL Check" onClick="alert(theurl)">
<input type="button" value="GO!" onClick="window.location=theurl"><br>
<br>
</td></tr></table>
</form>
```

```
</body>
```

This script uses the variable `theurl` to hold the entered characters. Each time a button is selected, the associated letter is added to `theurl` via the `onClick` JavaScript command (e.g., `onClick="theurl=theurl+'a'"`).



There are many other JavaScript commands available to you when creating your own custom page. However, it is important to note that not every function is supported by the *Wipeout Pure* browser. If you intend to hack together something more advanced than this script, you will have to work out which functions will and will not work.

Once the URL is entered, you can use the URL Check button to view the entered URL in a pop-up window. Due to the very limited JavaScript support, this is about the only option to view what you have typed. If you entered the wrong characters, use the Clear button to reset `theurl` variable, or the BS button to remove the tail end characters of the URL until you reach the location of the typo. Once everything is typed correctly, the `window.location` function is called when you hit the GO! button, which sends the browser to the desired location. Figure 5-25 provides you with a screenshot of the keyboard, and Figure 5-26 illustrates the URL checker pop-up window.

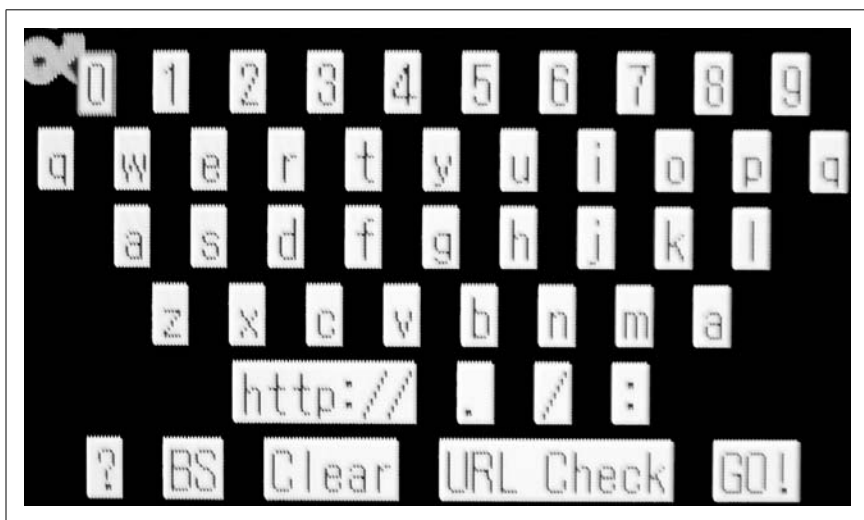


Figure 5-25. PSP JavaScript keyboard



Figure 5-26. The URL checker pop-up window

Hacking the Hack

Once you have the keyboard up and running, it is easy to customize it to your own needs. You can easily add a button-based bookmark menu on the bottom of the screen using the following format. For example, this button would load the Airscanner home page:

```
<input type="button" value="Airscanner!"  
onClick="window.location='http://www.airscanner.com'">
```

If you need capital letters, you can easily create an alternate keyboard that includes these ASCII characters. At this point, the options are up to you. Have fun!

—Seth Fogie