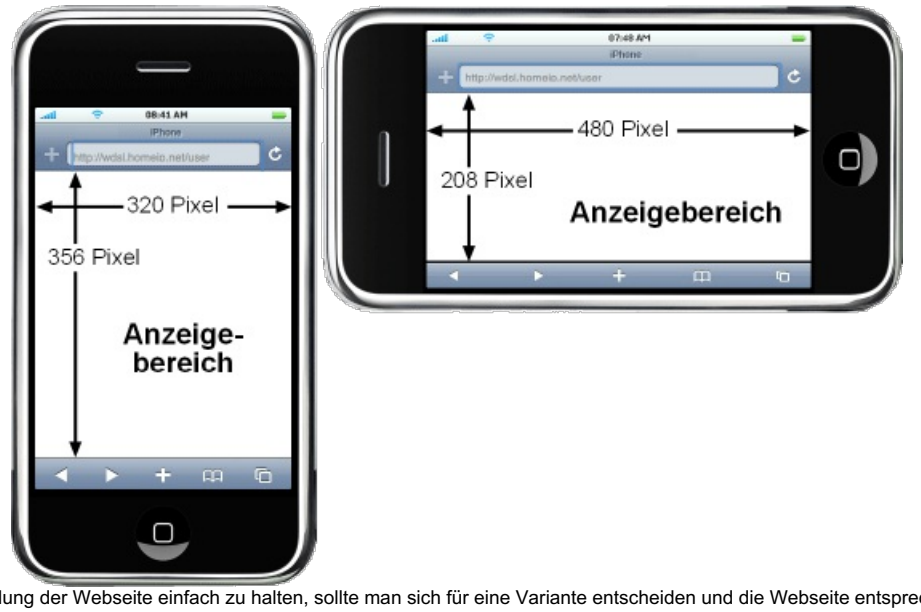
Applikation zum Web-IO Digital:

iPhone optimierte Webseiten für Web-IO Anwendungen

Produktübersicht
Applikationsübersicht

Die hier gezeigten Techniken nehmen zwar Bezug auf das Apple iPhone, können aber durch Anpassung der Anzeigegröße an die Displayauflösung auch für jedes andere, internetfähige Mobiltelefon umgesetzt werden. Einzige Voraussetzung ist die Unterstützung von JavaScript und im speziellen von HTTP-Request-Objekten (AJAX-Technik).

Was die Anzeige des iPhone betrifft, gibt es zwei Varianten:

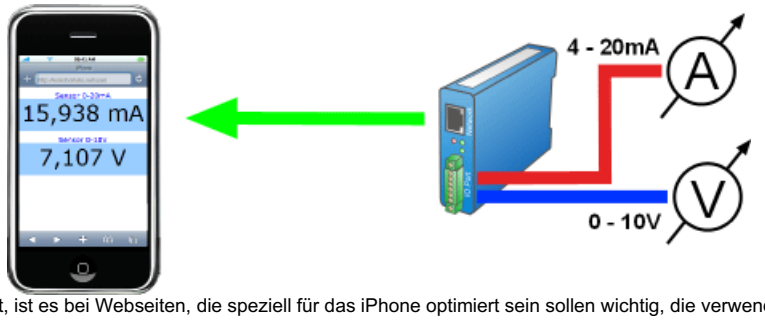


Um die Erstellung der Webseite einfach zu halten, sollte man sich für eine Variante entscheiden und die Webseite entsprechend aufbauen.

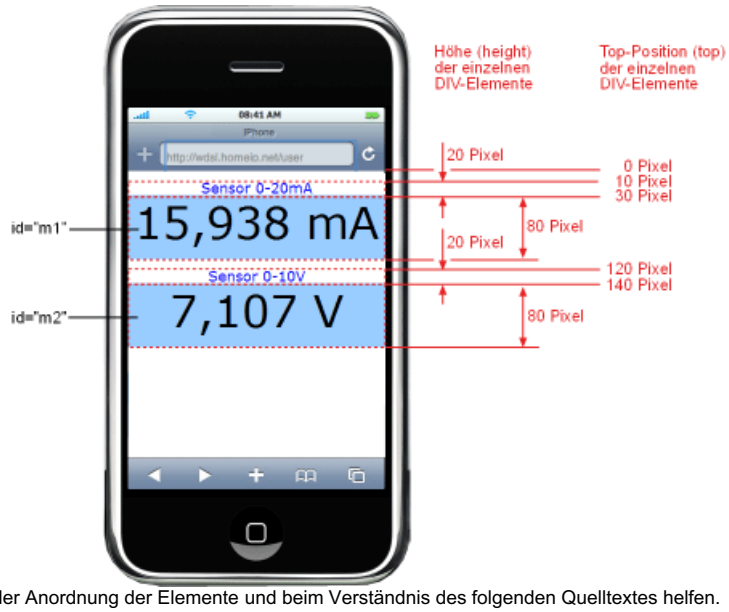
Bei vorhandenem WLAN-Zugang funktioniert das gezeigte Beispiel auch auf allen iPod Touch Modellen!

Beispiel - Messwerte aus Industrie- und Gebäudetechnik mit dem iPhone Anzeigen

In diesem Beispiel sollen Ströme oder Spannungen einem Web-IO 2xAnalog In gemessen und als dynamische Webseite auf dem iPhone angezeigt werden. Durch Vorschalten entsprechender Messumformer können so nahezu alle physikalischen Größen gemessen und auf dem iPhone bzw. Mobiltelephon dynamisch angezeigt werden.



Wie bereits angedeutet, ist es bei Webseiten, die speziell für das iPhone optimiert sein sollen wichtig, die verwendeten HTML-Elemente von Größe und Position pixelgenau zu definieren.



Die Zeichnung soll bei der Anordnung der Elemente und beim Verständnis des folgenden Quelltextes helfen.

```
<user.htm>
<html>
 <head>
  <title>Aktuelles Klima</title>
  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
  <meta name="viewport" content="width=320" />
  <style type="text/css">
    <!--
    * { font-family:Verdana, Helvetica; }
    .description{ position:absolute;
       text-align:center;
       font-size:18px;
       left:0px;
       height:20px;
       width:320px;
       }
    .value  { position:absolute;
       background-color:#99CCFF;
       text-align:center;
       font-size:55px;
       left:0px;
       height:80px;
       width:320px;
       }
    -->
  </style>
 </head>
```

Der Quelltext beginnt mit dem Web-IO spezifischen Tag<user.htm>, welches das Web-IO für die interne Zuordnung ins File-System benötigt. Das <user.htm> Tag wird vom Web-IO nicht an den Browser weitergegeben.

Dann folgt der normale <Head>-Bereich der Webseite. Wichtig ist die Angabe "viewport". Hiermit wird das normalerweise vom iPhone praktizierte Auszoomen der Webseite auf 320 Pixel begrenzt, so dass die Webseite immer komplett und in der gewünschten Skalierung angezeigt wird.

Im weiteren Verlauf des Quelltextes werden DIV-Elemente zur Beschreibung und Anzeige der Werte angelegt. Für diese Elemente finden sich entsprechende CSS-Style Informationen im Kopf der Seite. Unter anderem um Höhe und linke Position zu definieren.

```
  <body>
  <div class="description" style="top:10px; ">
   <w&t_tags=sensor1>
  </div>
  <div id="m1" class="value" style="top:30px;">
   <w&t_tags=m1>
  </div>
  <div class="description" style="top:120px; ">
   <w&t_tags=sensor2>
  </div>
  <div id="m2" class="value" style="top:140px; ">
   <w&t_tags=m2>
  </div>
  </body>
 </html>
```

Es folgt der <body>-Bereich der Webseite. Hier werden die im Kopf definierten CSS-Klassen so wie die top-Angaben genutzt, um die einzelnen DIV-Elemente in Position zu bringen. Den DIV-Elementen, in den später die akualisierten Werte angezeigt werden sollen, wird jeweils eine ID zugeteilt. Als Anzeige Text werden spezielle W&T-Tags eingetragen. Diese Tags werden vom Web-IO beim Laden der Seite in den Browser durch den gerade gültigen Wert ersetzt.

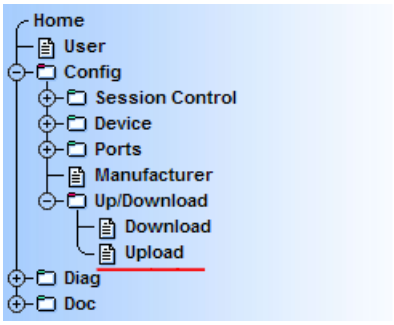
```
 <script language="JavaScript" type="text/javascript">
function DataRequest(sendstring)
 {
 var xmlHttp = window.ActiveXObject ? new ActiveXObject("Microsoft.XMLHTTP") : xmlHttp = new XMl
 if (xmlHttp)
  {
  xmlHttp.onreadystatechange = function()
   {
   if (xmlHttp.readyState == 4)
    {
    if (xmlHttp.status == 200)
     {
     var ReceiveData = xmlHttp.responseText.split(";");
     document.getElementById('m1').innerHTML = ReceiveData[ReceiveData.length-2];
     document.getElementById('m2').innerHTML =ReceiveData[ReceiveData.length-1];
     xmlHttp=null;
     }
    }
   }
   xmlHttp.open("GET", sendstring, true);
   xmlHttp.setRequestHeader("Connection", "close");
   xmlHttp.setRequestHeader("If-Modified-Since", "Thu, 1 Jan 1970 00:00:00 GMT");
   xmlHttp.send(null);
   maintimer = setTimeout("DataRequest('single')", 5000);
  }
 }
DataRequest('single');
</script>
```

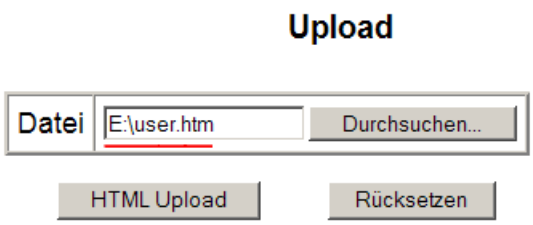
Kernstück der Webseite ist der JavaScript Teil, der im <head>-Bereich der Webseite untergebracht wird. Hier wird mittels der DataRequest Funktion das Kommando *single* zum Web-IO gesendet. Das Web-IO sendet Semikolon-getrennt die Werte für Temperatur, Luftfeuchte und Luftdruck zurück. Über das Split-Statement werden die Werte separiert und in ein Variablenarray geschrieben.

Mit der Methode `document.getElementById()` werden die entsprechenden DIV-Elemente zur Anzeige ausgewählt und die aktuellen Werte eingetragen.

Über einen Timer geschieht das zyklisch alle 5000ms Sekunden.

Die Webseite ist damit fertig und muss nur noch in das Web-IO hochgeladen werden.





Sie haben noch kein Web-IO und möchten das vorgestellte Beispiel einfach mal ausprobieren?
Kein Problem: Wir stellen Ihnen das Web-IO Analog gerne kostenlos für 30 Tage zur Verfügung. Einfach Musterbestellung ausfüllen, wir liefern das Web-IO zum Test auf offene Rechnung. Wenn Sie das Gerät innerhalb von 30 Tagen zurück schicken, schreiben wir die Rechnung komplett gut.

Zur Musterbestellung

Programmbeispiel herunterladen

W&T
www.WuT.de

Wir sind gerne persönlich für Sie da:

Wiesemann & Theis
GmbH
Porschestr. 12
42279 Wuppertal
Tel.: 0202/2680-110 (Mo-Fr. 8-17 Uhr)
Fax: 0202/2680-265
info@wut.de