

Mobile Webseiten erstellen: Webseiten für Smartphones und Tablets

12.12.2012 01:14 von [Marco Dittmer](#) (Kommentare: 12)



Lange Zeit genügte es, als gewissenhafter Webseitenbetreiber, seine Internetseiten möglichst für alle gängigen Webbrowser zu optimieren. Doch immer mehr Besucher Ihrer Seiten surfen heute mobil. Egal ob mit dem Smartphone oder dem Tablet-PC - eine nicht für Smartphones optimierte Webseite lässt sich auf diesen mobilen Endgeräten nur sehr schlecht betrachten und meist ist auch die Navigation nur begrenzt möglich (*Probleme mit Dropdownmenüs*). In unserem neuen Blogbeitrag zeigen wir Ihnen in wenigen Schritten, wie einfach es ist, seine Webseite für Smartphones zu optimieren und erklären, worauf Sie unbedingt dabei achten sollten.

Die optimale Lösung für die mobile Webseitenansicht

Es gibt sehr viele verschiedene Möglichkeiten, die eigene Webseite für Smartphones zu optimieren. Jedoch möchten wir Ihnen hier, die aus unserer Sicht optimalste Variante dafür vorstellen. Zunächst gilt es, zu erkennen, mit was für einer Art von Gerät der Besucher auf die Internetseite zugreift. Um mobile Browser zu erkennen, benötigen wir ein PHP-Skript, das zuallererst den Browser erkennen und dann die entsprechend optimale Webseitenansicht ausgeben soll.

Browser per PHP ermitteln

Im ersten Schritt ist, wie bereits erwähnt, der aktuelle Browser des Seitenbesuchers zu ermitteln. Man findet im Internet bereits unzählige, verschiedene PHP-Skripte, die dies tun. Leider funktionieren **nur wenige wirklich zuverlässig** für alle gängigen Smartphones und Tablet-PC. Daher haben wir das folgende Skript auf die Kompatibilität auf allen gängigen, mobilen Endgeräten getestet.

Die Mobile-Browser-Funktion

Wir empfehlen den nachfolgenden Skriptcode in eine Funktionen-PHP-Datei (*mobile_browser_funktion.php*) zentral zu speichern und über den PHP-Befehl `include()` in die *index.php* zu implementieren. So können Sie auf jeder Ihrer Seiten immer wieder auf die *find_mobile_browser()* - Funktion zurückgreifen.

Quellcode: mobile_browser_funktion.php

```
1 <?php
2 function find_mobile_browser()
3 {
4     if(preg_match('/(alcatel|android|blackberry|benq|cell|elaine|htc|iemobile|iphone|ipad|ipaq|ipod|j2me|java|mi
5     {
6         return true;
7     }
8     else
9     {
10        return false;
11    }
12 }
13 ?>
```

Browsererkennung einfügen und abrufen

Im weiteren Schritt müssen Sie nun die *mobile_browser_funktion.php* in Ihre *index.php* einfügen und abrufen. Dazu müssen Sie einfach die PHP-Datei "includen". Dies sollten Sie am besten bereits ganz am Anfang der Datei vornehmen, also noch vor dem `<head>`-Tag. Danach rufen Sie, ebenfalls noch vor dem `<head>` die Browsererkennungsfunktion über `$mobile_browser = find_mobile_browser();` auf.

Ein weiterer, **sehr wichtiger Punkt**, den es zu beachten gilt, ist dem Besucher selbst entscheiden zu lassen, mit welcher Ansicht er Ihre Seite betrachten will. Daher sollten Sie zusätzlich einen Button einrichten, den der Besucher mit einem Smartphone klicken kann, um doch die Standardansicht Ihrer Webseite aufzurufen. Die Umsetzung dafür gelingt am besten durch das Setzen sogenannter Cookies (*Informationen, die auf dem Gerät des Besuchers zwischengespeichert werden*).

Die Umsetzung der Browsererkennung, die wir Ihnen unten im Skript zeigen, gibt je nach Browser entweder das mobile Stylesheet oder das Standardstylesheet aus. Dafür haben wir in unserem Beispiel zwei verschiedene CSS-Stylesheet-Dateien:

- *mobile.css* und
- *desktop.css*.

Im `<body>`-Teil wird dann auch jeweils ein anderer Inhalt ausgegeben. Wir gehen später noch auf die Besonderheiten für ein mobiles Layout ein, aber so viel sei an dieser Stelle schon einmal gesagt; wo normale Webseiten **zwei- oder dreispaltige Layouts** verwenden, ist bei einem mobilen Layout definitiv ein **einspaltiges Layout** zu empfehlen.

Für die getrennten Inhalte müssen Sie nur noch eine PHP-Datei mit dem mobilen Inhalt (*einspaltiges Layout*) und eine für den Desktop-Inhalt (*mehrspaltiges Layout*) erstellen. In unserem Beispiel sind dies die Dateien:

- *mobile_inhalt.php* und
- *desktop_inhalt.php*.

Quellcode: [index.php](#)



0 531 2097 8988

Sie haben Fragen? Zum Kontaktformular »»»

[Webseiten](#) [Onlineshops](#) [Wir für Sie](#) [Referenzen](#) [Blog](#) [Kontakt](#)

```
8 // Cookies setzen (Benutzer möchte zur Desktop-Ansicht wechseln, hier wird der Cookie für eine Stunde gespeichert)
9 if(isset($_REQUEST['sessionview_desktop']))
10 {
11     setcookie('sessionview', 'desktop', time() + 3600);
12     header('Location: http://ihre-domain.tld/');
13 }
14 // Cookies zerstören (Benutzer möchte doch wieder zur Mobile-Ansicht wechseln)
15 elseif(isset($_REQUEST['sessionview_mobile']))
16 {
17     setcookie('sessionview', 'mobile', time() - 3600);
18     header('Location: http://ihre-domain.tld/');
19 }
20 ?>
21 <!DOCTYPE...> <!-- gekürzte Fassung -->
22 <html...> <!-- gekürzte Fassung -->
23 <head>
24 <title>...</title>
25 <!-- Ihre META-Angaben -->
26 <!-- Stylesheets (Layout) für die Mobile-Version und Desktop-Version Ihrer Internetseite -->
27 <?php
28 // Mobile-Ansicht
29 if($mobile_browser && $_COOKIE['sessionview'] != 'desktop')
30 {
31     // Mitteilung an den Browser (speziell Opera Mobile), dass es sich um eine für Smartphone optimierte Seite handelt
32     echo '<meta name="viewport" content="width=device-width; initial-scale=1.0; maximum-scale=1.0; user-scalable=no">';
33     // CSS für Mobile-Ansicht
34     echo '<link rel="stylesheet" href="mobile.css" type="text/css" media="screen" />';
35 }
36 else
37 {
38     // CSS für Desktop-Ansicht
39     echo '<link rel="stylesheet" href="desktop.css" type="text/css" media="screen" />';
40 }
41 ?>
42 </head>
43 <body>
44 <?php
```

```

45 // Content Mobile-Ansicht
46 if($mobile_browser && $_COOKIE['sessionview'] != 'desktop')
47 {
48     include('mobile_inhalt.php');
49     // Button für das Wechseln der Ansicht durch den Benutzer
50     if($mobile_browser)
51     {
52         echo '<a href="'. $_GET['url'] . '?sessionview_desktop=1">Zur Desktop-Ansicht wechseln</a>';

```

nach oben

Xcreative | Friedrich-Wilhelm-Straße 2 | 38100 Braunschweig | Telefon: +49 (0) 531 - 2097 8988

Startseite | Impressum | Datenschutzerklärung

```

54 }
55 // Content Desktop-Ansicht
56 else
57 {
58     include('desktop_inhalt.php');
59     // Button für das Wechseln der Ansicht durch den Benutzer
60     if($mobile_browser && $_COOKIE['sessionview'] == 'desktop')
61     {
62         echo '<a href="'. $_GET['url'] . '?sessionview_mobile=1">Zur Mobile-Ansicht wechseln</a>';
63     }
64 }
65 ?>
66 </body>
67 </html>
68 <!-- Ebenfalls für die Sicherstellung der header-Location-Funktionalität -->
69 <?php ob_end_flush(); ?>

```

In unserem Codebeispiel, das Sie gerne auch für Ihre Webseite übernehmen dürfen, haben wir in den ersten Zeilen den PHP-Befehl `ob_start()`; und in den letzten Zeilen den Befehl `ob_end_flush()`; eingebaut. Damit stellen wir sicher, dass die PHP-Funktion `header('Location:http://ihre-domain.tld/')`; korrekt ausgeführt wird. Diese darf normalerweise nämlich nur direkt am Anfang des Quellcodes stehen; das bedeutet, es darf **nichts** davor ausgegeben werden. Deshalb empfehlen wir, diese beiden Codeteile immer zu verwenden, da es Ihnen somit offen steht, auch innerhalb Ihres Inhaltes, die *header-Location-Funktion* einzubauen (*beispielsweise für eine Login-Funktion*).

Mit den Befehlszeilen, die mit den Teilen

- `if(isset($_REQUEST['sessionview_desktop']))` und
- `elseif(isset($_REQUEST['sessionview_mobile']))` beginnen,

prüfen wir, ob ein Benutzer mit mobilem Endgerät - denn nur dieser sieht den Link-Button "Zur Desktop-Ansicht wechseln" - den besagten Ansichtwechselbutton geklickt hat. Dabei wird in der Browser-Url `ihre-domain.tld/?sessionview_desktop=1` aufgerufen. Sobald dies geschieht, wird ein Cookie mit einer Haltwertzeit von einer Stunde (3600 Sekunden) gesetzt. Klickt der Besucher, der nun die normale Standardansicht Ihrer Seite betrachtet, auf den, wieder nur für ihn als mobiler Surfer sichtbaren Link-Button "Zur Mobile-Ansicht wechseln", so wird der Cookie, durch aufrufen der Browser-Url `ihre-domain.tld/?sessionview_mobile=1` wieder "zerstört" und die Haltwertzeit dessen um 3600 Sekunden reduziert.

Je nachdem, welcher Browser ermittelt wurde oder welche Ansicht der Besucher gewählt hat, werden dann die unterschiedlichen Inhalte "inkludiert". Natürlich kann man bereits ganz am Anfang eine Browserweiche einbauen und direkt auf eine **extra Domain weiterleiten**. Viele namenhafte Seitenbetreiber nutzen diese Möglichkeit. Dabei wird auf:

- `mobile.ihre-domain.tld` oder auf
- `m.ihre-domain.tld` weitergeleitet,

Dabei handelt es sich schlicht und ergreifend um eine Subdomain zu Ihrer Hauptdomain, die dann wiederum für Smartphones optimiert wird. Etabliert hat sich dabei die **Variante mit dem "m" als Subdomain**. Da sich der Inhalt Ihrer Webseite, egal ob Standardansicht oder mobile Ansicht, nicht ändern wird, ist aber mit dieser Methode nicht sichergestellt, dass Google und andere Suchmaschinen dies nicht als Duplikate Content interpretieren und somit Ihre Internetseite **suchmaschinentechnisch abstrafen**. Lesen Sie mehr dazu in unserem Blogbeitrag "[Suchmaschinenfreundliche Weiterleitung - der 301 Redirect](#)".

Quellcode-Variante: Weiterleitung auf eine Subdomain/ Browserweiche

```

1 <?php
2 // Wieder ganz am Anfang Ihrer index.php
3 include('mobile_browser_funktion.php');

```

```

4 | $mobile_browser = find_mobile_browser();
5 | if($mobile_browser &&$_SERVER['HTTP_HOST'] != 'm.ihre-domain.tld')
6 | {
7 |     header('Location: http://m.ihre-domain.tld/');
8 | }
9 | else
10 | {}
11 | ?>

```

Layout für Smartphone-optimierte Webseiten

Wie bereits erwähnt sollten Sie bei der Layout-Erstellung für mobile Endgeräte auf ein **einspaltiges Layout** beschränken. Bestenfalls haben Sie einen DIV-Container als Außenrahmen. Diesem geben Sie dann eine Breite von 99%, sodass er je nach Auflösung und Größe des Displays immer **fast** das ganze Display ausfüllt. Auch für die Einbindung von Bildern sollten Sie prozentuale Angaben machen. Außerdem sollten Sie für die mobile Webseitenansicht unbedingt **kleiner gerechnete Bilder** verwenden und **für das Layout dieser Ansicht ganz auf Grafiken verzichten**. Denn viele Mobilfunkverträge beschränken das Downloadvolumen auf gerade einmal 500 MB. Außerdem gibt es eine unerschöpfliche Bandbreite an CSS-Stylesheet-Mitteln, um ein wirklich ansprechendes Layout auch für Smartphones zu erstellen, deshalb schonen Sie das Volumen Ihrer Besucher.

Für das Menü empfehlen wir eine einfache *"Aufklapp-Variante"*, die ohne Java oder ohne auf andere Mittel zurückzugreifen funktioniert.

```

1 | <?php
2 | if(isset($_REQUEST['menue']))
3 | {
4 |     // Menü wird hier angezeigt
5 | }
6 | else
7 | {
8 |     echo '<a href="'. $_GET['url'] .'?menue=1">Menü aufklappen</a>';
9 | }
10 | ?>

```

Klickt der Besucher auf den Link-Button *"Menü aufklappen"*, wird die Browser-Url *ihre-domain.tld/?menue=1* aufgerufen. Sobald dies der Fall ist, wird der PHP-Befehl `if(isset($_REQUEST['menue']))` ausgeführt und das Menü, das Sie an den Teil mit der **Information** einfügen, angezeigt.

Zurück

Einen Kommentar schreiben

<input type="text"/>	Name*
<input type="text"/>	E-Mail (wird nicht veröffentlicht)*
<input type="text"/>	Webseite
<input type="text"/>	Was ist die Summe aus 9 und 3?*

Über neue Kommentare per E-Mail benachrichtigen

Kommentar absenden

Kommentar von [Gis](#) | 24.05.2014

danke für die Antwort Marco,

aber wie ich schon sagte, sind es Antworten für Leute die die Grundkenntnisse haben und eben wissen was sie tun.

Kann es nicht umsetzen.

Danke dir für deine Bemühungen.

grüssli Gis

Antwort von Marco Dittmer

Hallo Gis,

der Artikel beschreibt auch lediglich die Einbindung der Browser-Weiche und der Verwendung von Cookies, um auch die Möglichkeit zu bieten, immer die Desktop-Variante anzeigen zu lassen. Schau Dir doch einfach mal meine letzte Antwort für Thomas an. Die dort beschriebene Möglichkeit ist eine super Alternative, auch mit nur grundlegendem Wissen. Contao in Verbindung mit den RockSolid Themes, die bereits Demo-Daten mitliefern, ermöglicht einem "relativ einfach" bereits gut ausschauende, für mobile Geräte optimierte Webseiten zu erstellen. Es bedarf eben ein wenig Fleißarbeit.

Viele Grüße

Marco Dittmer

Kommentar von Thomas | 23.05.2014

Hi,

bisher habe ich nur mit html und css gearbeitet und noch nie php angerührt, deswegen die vielleicht einfältige Frage:

Sind das völlig getrennten Welten, oder könnte ich mit einigermaßen einfachen Mitteln und einigen php-Schnipseln eine PC-Handy-Weiche basteln? Oder geht so was auch ohne php?

Vielen Dank und Grüße,

Thomas

Antwort von Marco Dittmer

Hi Thomas,

es ist nicht zwingend erforderlich eine PC-Handy-Weiche zu implementieren. Alternativ empfehle ich Dir die Umsetzung eines Responsive Webdesign. Dieses reagiert dabei auf die Ausgabegröße der Webseite. Das bedeutet, selbst wenn Du die Größe des Browser-Fensters verkleinerst oder wieder vergrößerst, passt sich die Webseite diesem an.

Jedoch ist es auch nicht gerade einfach, sich in das 12-Spalten-Grid-System einzuarbeiten. Ich weiß jetzt nicht, mit welchem System (CMS, sonstige Programme) Du Deine Webseite umgesetzt hast. Doch darf ich Dir eine Empfehlung aussprechen, bevor Du den Entschluss fasst, Dich in PHP, CSS (12-Grid, etc.) ausgiebig einzulernen:

- Installiere das Contao Open Source Web-CMS: [hier](#)
- Lade Dir ein ansprechendes Respsive-Theme bei RockSolid Themes herunter (ggf. kaufen): [hier](#)
- Als sehr gute Lektüre für Contao kann ich Dir "Websites erstellen mit Contao 3" von Peter Müller sehr empfehlen, außerdem die Hilfestellung direkt von Contao: [hier](#), die ausgezeichnete Contao Community: [hier](#) und natürlich das kostenlose online verfügbare Contao Handbuch für Anwender & Redakteure von Nina Gerling: [hier](#)
- Hilfestellungen zu den Themes findest Du direkt bei RockSolid Themes: [hier](#)

Natürlich ist es am Anfang immer schwer, sich in ein neues System einzuarbeiten. Doch hast Du mit den gezeigten Systemen und Tools die Möglichkeit, einen absolut modernen und wirklich frischen, für mobile Geräte optimierten Webauftritt zu erstellen. Und ganz am Rande,

Kommentar von [Gis Volkmann](#) | 13.05.2014

Halli Hallo,

da sich einige beklagt haben, dass sie auf dem handy zuviel scrollen müssen.... möchte ich nun gerne euer script einbauen.

Eure Erklärungen hier sind toll, für Leute, die dann noch wissen was sie tun :D

ich möchte es auch gerne vorher wissen, und bitte um eine Erklärung für Dummies ;)

z.B. verstehe ich hier nur Bahnhof:

Die Mobile-Browser-Funktion

Wir empfehlen den nachfolgenden Skriptcode in eine Funktionen-PHP-Datei (mobile_browser_funktion.php) zentral zu speichern und über den PHP-Befehl include() in die index.php zu implementieren. So können Sie auf jeder Ihrer Seiten immer wieder auf die find_mobile_browser() - Funktion zurückgreifen.

zentral speichern ? PHP Befehl include ??? wie wo mach ich das ?

so muss nun wech...

freue mich aber schon auf Hilfe

LG Gis

Antwort von Marco Dittmer

Nun Gis,

um die Umsetzung durchführen zu können, benötigst Du ein paar wenige Grundkenntnisse in PHP-Programmierung. Wie beschrieben erstellst Du eine Datei, in der Du den Quellcode mit der Bezeichnung "mobile_browser_funktion.php" einfügst. Speichere diese dann als PHP-Datei ab. Füge dann in Deine index.php folgenden Quellcode für das "includen" ein:

```
include('mobile_browser_funktion.php');
```

Kommentar von Frank | 26.04.2014

Auf meiner Homepage habe ich die Weiterleitung die ich von www.mobilonso.com erhalten habe eingefügt. Die Weiterleitung funktioniert zwar, aber zurück nicht. Muss ich wie in dem Artikel beschrieben auf Cookies zurückgreifen? Vielen Dank

Antwort von Marco Dittmer

Hallo Frank,

nun, ich kann Dir jetzt nicht genau sagen, wie die Funktionsweise von mobilonso.com ist. Persönlich würde ich auch nicht auf einen externen Anbieter zurückgreifen, um eine mobile Version der Webseite zu erstellen.

Es gibt ja einerseits die Möglichkeit, die Webseite komplett mit der [Responsive Webdesign](#) Technik umzusetzen, oder aber eben mit einer Browserweiche. Diese bietet eben dann auch die Möglichkeit des Besuchers, selbst zu entscheiden, wie er die Webseite betrachten möchte.

Meine Empfehlung aus heutiger Sicht: die Webseite definitiv responsive zu gestalten, mit der Option, die Desktop-Variante anzeigen zu lassen.

Kommentar von [Deniz Bayar](#) | 12.09.2013

Ich möchte gerne eine Lösung mit m.domain.tld Subdomain haben und da hat mir dieses Tutorial schon sehr geholfen. Vielen Dank dafür.

Die Browsererkennung und Weiterleitung funktioniert hervorragend.

Problem: Der "zum Desktop wechseln" Link funktioniert leider noch nicht.

Code ganz oben (mobile Version):

```
<?php
ob_start();
include('mobile_browser_funktion.php');
$mobile_browser = find_mobile_browser();

if(isset($_REQUEST['sessionview_desktop']))
{
setcookie('sessionview', 'desktop', time() + 3600);
header("Location: http://domain.tld);
}
elseif(isset($_REQUEST['sessionview_mobile']))
{
setcookie('sessionview', 'mobile', time() - 3600);
header("Location: http://m.domain.tld);
}
?>
```

Code im Header:

```
<?php
// Mobile-Ansicht
if($mobile_browser && $_COOKIE['sessionview'] != 'desktop')
{}
else
{
header("Location: http://domain.tld);
}
?>
```

Code im Body:

```
<?php
if($mobile_browser)
{
echo '<a href="'. $_GET['url'] .'?sessionview_desktop=1">Zur Desktop-Ansicht wechseln</a>';
}
if($mobile_browser && $_COOKIE['sessionview'] == 'desktop')
{
echo '<a href="'. $_GET['url'] .'?sessionview_mobile=1">Zur Mobile-Ansicht wechseln</a>';
}
?>
```

In der Desktop Version:

Ganz oben:

```
<?php
ob_start();
include('mobile_browser_funktion.php');
```

```

$mobile_browser = find_mobile_browser();
if(isset($_REQUEST['sessionview_desktop']))
{
setcookie('sessionview', 'desktop', time() + 3600);
header("Location: http://domain.tld");
}
elseif(isset($_REQUEST['sessionview_mobile']))
{
setcookie('sessionview', 'mobile', time() - 3600);
header("Location: http://m.domain.tld");
}
?>

Im Header:
<?php
// Mobile-Ansicht
if($mobile_browser && $_COOKIE['sessionview'] != 'desktop')
{
header("Location: http://m.domain.tld");
}
else
{}
?>

Im Body:
<?php
if($mobile_browser)
{
echo '<a href="'. $_GET['url'] .'?sessionview_desktop=1">Zur Desktop-Ansicht wechseln</a>';
}
if($mobile_browser && $_COOKIE['sessionview'] == 'desktop')
{
echo '<a href="'. $_GET['url'] .'?sessionview_mobile=1">Zur Mobile-Ansicht wechseln</a>';
}
?>

```

Antwort von Marco Dittmer

Hallo Deniz,

das Beispiel von mir zeigt nicht direkt die Lösung mit Cookies für die Weiterleitung von der www-Seite zur m-Seite und zurück. Daher musst Du hier ein paar Anpassungen vornehmen.

Wenn Du auf der mobilen Seite bist (sprich mit einem Smartphone oder ähnlichem surfst), also m.domain.tld, kannst Du die Cookies nicht über `Zur Desktop-Ansicht wechseln` aufrufen. Dies bedeutet doch, dass Du weiterhin auf der mobilen Domain bist. Natürlich musst Du aber dennoch die Lösung mit den Cookies verwenden, da ein einfaches Verlinken auf die www-Seite (Desktop-Seite) natürlich zur automatischen Weiterleitung zurück zur mobilen Seite zur Folge hätte.

Probiere es doch einfach mal mit `Zur Desktop-Ansicht wechseln` aus. Somit speicherst Du den Cookie und leitest gleichzeitig zurück zur Desktop-Seite. Alle anderen Links musst Du natürlich entsprechend anpassen.

Kommentar von [Maximilian Borm](#) | 20.08.2013

Danke. Da müsste ich mich dann erst einmal reinarbeiten. Mit Cookies habe ich bisher noch nie etwas gemacht...

Aber es gibt ja (glücklicherweise) im Internet genug Seiten!

LG

Maxi

P.S.: Gerade bemerkt, dass Ihr auch aus BS seid ;)

Kommentar von [Maximilian Borm](#) | 20.08.2013

Danke!!!

An dem Menü arbeite ich noch, das wird demnächst dann auch laufen. Ich werde es wohl so machen, dass die Startseite so bleibt, aber auf den Unterseiten eine kleine Navi-Leiste oben erscheinen wird.

Die Möglichkeit zum Wechseln in die Desktop-Variante hatte ich auch schon bedacht, aber da (noch) nicht wusste, wie genau ich das realisiere, habe ich das erstmal nach hinten gestellt.

LG

Maxi

Antwort von Marco Dittmer

Die einfachste Lösung für den Wechselbutton wäre die Verwendung von Cookies.

PHP-Quellcode:

```
<?php
if(isset($_POST['sessionview-desktop']))
{
    setcookie('sessionview', 'desktop', time() + 43200);
    header('Location: DEIN ZIEL');
}
elseif(isset($_POST['sessionview-mobile']))
{
    setcookie('sessionview', 'mobile', time() - 43200);
    header('Location: DEIN ZIEL');
}
?>
```

Auf der mobilen Seite einbinden:

```
<?php
if($mobile_browser && $_COOKIE['sessionview'] != 'desktop')
{
    echo '<input type="submit" name="sessionview-desktop" value="Zur Desktop-Version wechseln" />'. "\n";
}
elseif($mobile_browser && $_COOKIE['sessionview'] == 'desktop')
```

```

{
    echo '<input type="submit" name="sessionview-mobile" value="Zur Mobile-Version wechseln" />'. "\n";
}
else {}
?>

```

Natürlich muss Du dann noch den form-tag <form action="" method="post" id=""> anpassen und einbinden...

Kommentar von [Maximilian Borm](#) | 20.08.2013

Ich habe mir heute auch mal eine mobile Website gebastelt. Die Tipps hier ware ganz hilfreich, und am Ende hatte ich dan eine funktionierende PHP Browserweiche ... hier erstmal die Seite (ist heute erstellt worden und wird noch verbessert) <http://mobile.mborm.net> . Wie findet ihr die bisher?

Ich habe das so gemacht:

```

function mobile_browser($useragent) {

if(preg_match('/android.+mobile|avantgo|bada-v|blackberry|blazer|compal|elaine|fennec|hiptop|iemobile|ip(hone|od)|iris|kindle|lge
|maemo|midp|mmp|opera m(ob|in)i|palm( os)?|phone|p(ixi|re)|v|plucker|pocket|psp|symbian|treo|up\.(browser|link)|vodafone|wap|windows
(ce|phone)|xda|xiino/i',$useragent)||preg_match('/1207|6310|6590|3gso|4thp|50[1-6]|770s|802s|a-wa|abac|ac(er|oo|s-
)|ai(ko|rn)|al(av|ca|co)|amoi|an(ex|ny|yw)|aptu|ar(ch|go)|as(te|us)|attw|au(di|-m|r |s )|avan|be(ck|ll|nq)|bi(lb|rd)|bl(ac|az)|br(e|v)w|bumb|bw-
(n|u)|c55v|capi|ccwa|cdm|-|cell|chtm|clcd|cmd|-|co(mp|nd)|craw|da(it|||ng)|dbte|dc|-s|devi|dica|dmob|do(c|p)|ds(12|-
d)|el(49|ai)|em(l2|ul)|er(ic|k0)|esl8|ez([4-7]0|os|wa|ze)|fetc|fly(-|_)|g1 u|g560|gene|gf-5|g|-mol|go(\.w|od)|gr(ad|un)|haie|hci|hd(-m|p|t)|hei-
|hi(pt|ta)|hp( i|ip)|hs(-c|ht(c(-| _|a|g|p|s|t)|tp)|hu(aw|tc)|i(-20|go|ma)|i230|iac( |-
|v)|ibro|idea|ig01|ikom|im1k|inno|ipaq|iris|ja(tv|al|bro|jemul|jigs|kddi|keji|kgt( |v)|klon|kpt |kwc|-|kyo(c|k)|le(no|xi)|lg( g|v(k|l|u)|50|54|e|-|e|v|-|a-
w)|libw|lynx|m1-w|m3ga|m50v|ma(te|ui|xo)|mc(01|21|ca)|m(-cr|me(d|irc|ri)|mi(o8|oa|ts)|mmef|mo(01|02|bi|de|do|t(-| |o|v)|zz)|mt(50|p1|v
)|mwbp|mywa|n10[0-2]|n20[2-3]|n30(0|2)|n50(0|2|5)|n7(0(0|1)|10)|ne((c|m)-
|on|tf|wf|wg|wt)|nok(6|i)|nzph|o2im|op(ti|wv)|oran|owg1|p800|pan(a|d|t)|pdxg|pg(13|-([1-8])c))|phil|pire|pl(ay|uc)|pn|-2|po(ck|rt|se)|prox|psio|pt(-g|qa|-
a|qc(07|12|21|32|60|-|2-7)|i(-)|qtek|r380|r600|raks|rim9|ro(ve|zo)|s55v|sa(ge|ma|mm|ms|ny|va)|sc(01|h-|oo|p-)|sdkv|se(c(-|0|1)|47|mc|nd|ri)|sgh-
|shar|sie(-|m)|skl-0|sl(45|id)|sm(al|ar|b3|it|t5)|so(ft|ny)|sp(01|h-|v-|v )|sy(01|mb)|t2(18|50)|t6(00|10|18)|ta(gt|lk)|tcl(-|tdg)-|tel(i|m)|tim(-|t-
mo|to(pl|sh)|ts(70|m-|m3|m5)|tx-9|up(\.b|g1|si)|utst|v400|v750|veri|vi(rg|te)|vk(40|5[0-3]|-v)|vm40|voda|vulc|vx(52|53|60|61|70|80|81|83|85|98)|w3c(-|
)|webc|whit|wi(g |nc|nw)|wmlb|wonu|x700|xda(-|2|g)|yas-|your|zeto|zte-/i',substr($useragent,0,4)) { return true; }

if (mobile_browser($_SERVER['HTTP_USER_AGENT'])) { header("Location: http://mobile.mborm.net"); ;}

```

LG - Maxi

Antwort von Marco Dittmer

Hallo Maxi,

ich freue mich, dass unser Skript und Tutorial Dir bei der Erstellung einer mobilen Webseite geholfen hat. Du solltest das Menü jedoch so gestalten, dass dieses auch auf jeder Unterseite weiterhin verfügbar ist. Auch solltest Du eine Möglichkeit einbauen, die es dem Besucher ermöglicht, trotz Nutzung eines Smartphones auf die Desktop-Variante der Webseite zu wechseln.

Viele Grüße

Marco

Kommentar von [Sascha Nickel](#) | 19.03.2013

Hallo zusammen.

Da ich mit Templates arbeite und die php seiten nicht immer neu erstellen möchte habe ich folgende Lösung übernommen:

```
function find_mobile_browser()
{
    if(check_mobile_browser())
    {
        if ($_GET['mobile_off'] == 'true') {
            cookie('PrintVersionDesktop', 'true');
            return false;
        } else if ($_GET['mobile_off'] == 'false') {
            cookie('PrintVersionDesktop', '');
            return true;
        }
    }
}

if (($SESSION['PrintVersionDesktop'] == true) && (check_mobile_browser())) {
    return false;
} else if ((empty($SESSION['PrintVersionDesktop'])) && (check_mobile_browser())) {
    return true;
}
}

function check_mobile_browser()
{
    if(preg_match('/(alcatel|android|blackberry|benq|cell|elaine|htc|iemobile|iphone|ipad|ipaq|ipod|j2me|java|midp|mini|mobi|motorola|nokia|palm|panasonic|philips|phone|sagem|sharp|smartphone|sony|symbian|t-mobile|up\\.browser|up\\.link|vodafone|wap|wireless|xda|zte)/i',
    $_SERVER['HTTP_USER_AGENT']))
    {
        return true;
    } else {
        return false;
    }
}
?>
```

in der \$template->load funktion überprüfe ich dann mittels find_mobile_browser() welcher Seitentyp angezeigt werden soll und öffne dann entweder /templates/mobile/home.tpl wenn es ein Mobiles gerät ist, oder die Normales Templates /templates/home.tpl

das hat den Vorteil das man die PHP seite nicht neu programmieren muss die Variablen evtl anpassen muss und nur noch das Design mittels css (auch im css Verzeichnis habe ich ein mobile verzeichnis erstellt) anpassen und evtl die Templates geringfügig anpassen.

im klartext die Links bleiben erhalten und die seiten müssen nicht neu programmiert werden ;)

Antwort von Marco Dittmer

Hallo Sascha,

wir freuen uns, dass Du unser Skript für mobile Websites verwendest. Natürlich ist es schön zu erfahren, welche Möglichkeiten der

Optimierung und Anpassung unser Skript noch bietet.

Deine Lösung lässt sich doch aber auch mit unserem Beispiel von oben umsetzen:

```
// Mobile-Ansicht
if($mobile_browser && $_COOKIE['sessionview'] != 'desktop')
{
    // CSS für Mobile-Ansicht
    ../templates/mobile/home.tpl
}
else
{
    // CSS für Desktop-Ansicht
    ../templates/home.tpl
}
```

Dabei wird die index.php aufgerufen und je nach Zugriffsmethode dann das entsprechende Template geladen. Dabei sollte der Besucher immer die Option erhalten, zu wählen, ob er die mobile Website oder doch lieber die Desktop-Variante besuchen möchte, bzw. angezeigt bekommen möchte. Die Inhalte sind davon in keiner Weise betroffen und müssen auch nicht parallel zu einander erstellt werden.

Sollte ich mich irren, so lasse ich mich sehr gerne eines besseren belehren.

Viele Grüße nach Speyer

Marco

Kommentar von [Ultima](#) | 20.12.2012

Das würde nun das Problem für die Mobile Version lösen aber bei der www Variante würde der else Teil der Schleife ausgeführt werden. Und dort wird durch den Header Location immer wieder eine Umleitung auf <http://www.ihre-domain.tld/> gesetzt. Dort sollte auch eine Überprüfung gesetzt werden zB.:

```
if($_SERVER['HTTP_HOST'] === 'm.ihre-domain.tld' && !find_mobile_browser())
header("Location: http://www.ihre-domain.tld/");
elseif($_SERVER['HTTP_HOST'] === 'www.ihre-domain.tld' && find_mobile_browser())
header("Location: http://m.ihre-domain.tld/");
```

Schöne Grüße

Ultima

Antwort von Marco Dittmer

Hallo Ultima,

ich habe jetzt das Skript noch einmal angepasst und hoffe, dass jetzt alles passt. Vielen Dank für Deine kritische Anmerkung und somit Deinem Beitrag zur Verbesserung dieser Seite.

Viele Grüße nach Markt Schwaben

Marco

Kommentar von [Ultima](#) | 20.12.2012

In der if-else condition wird immer eine Weiterleitung gesetzt, selbst wenn ich mich schon auf der richtigen Domain befinde. Das müsste vorher noch geprüft werden, bzw. aufgeteilt werden.

Antwort von Marco Dittmer

Hallo Ultima,

ich habe den if-Teil nochmal angepasst. So müsste es doch eigentlich funktionieren:

```
if($mobile_browser && $_SERVER['HTTP_HOST'] != 'm.ihre-domain.tld')
```

Wenn ein mobiler Browser erkannt wird und sich der Besucher noch nicht auf der Zieldomain befindet, soll er weitergeleitet werden oder eben auf der www-Domain bleiben.

Kommentar von [Ultima](#) | 19.12.2012

Das erzeugt eine Endlose Weiterleitung :)

```
<?php
// Wieder ganz am Anfang Ihrer index.php
include('mobile_browser_funktion.php');
$mobile_browser = find_mobile_browser();
if($mobile_browser)
{
    header("Location: http://m.ihre-domain.tld/");
}
else
{
    header("Location: http://www.ihre-domain.tld/");
}
?>
```

Antwort von Marco Dittmer

Hallo Ultima,

ich habe den angesprochenen Code-Teil ausprobiert und dieser funktioniert einwandfrei. Sei es mit dem PC (auf <http://www.ihre-domain.tld/>) oder mit dem Smartphone (auf <http://m.ihre-domain.tld/>).

Warum sollte dieser eine endlose Weiterleitung erzeugen?