

MOBILES WEBDESIGN MIT MEDIA QUERIES

Passend gemacht

Viele kleine Geräte mit unterschiedlichen Bildschirmformaten stellen Web-Designer vor große Herausforderungen, wenn sie benutzerfreundliche Seiten gestalten wollen. Eine große Hilfe sind dabei neue Design-Konzepte wie die CSS-Erweiterung Media Queries. VON SEBASTIAN NIEMANN



Web-Designer hatten es nie leicht, denn sie mussten schon immer Seiten entwerfen, die in ganz verschiedenen Bildschirmformaten und -auflösungen einen guten Eindruck hinterlassen. Seit das Web auch durch die Augen der unzähligen Smartphones und Tablets gesehen wird, hat sich das Design-Problem potenziert. Die Lösungen dafür aber muss der Web-Designer nicht jedes Mal neu erfinden, sondern es gibt technische Rahmenwerke, die ihm viel Arbeit abnehmen. Eines davon heißt Media Queries, seit Juni ein W3C-Standard (www.w3.org/TR/css3-mediaqueries).

Mit Media Queries optimieren Web-Designer Seiten für mobile Endgeräte. Gleich zu Beginn werden viele Kenner einwenden, dass Smartphones und Tablet-PCs die Darstellung von

Haus aus richtig regeln. Mobile Geräte folgen einer einfache Strategie, Websites darzustellen: Sie skalieren die Website einfach auf die zur Verfügung stehende Breite der Bildschirmfläche herunter. Allerdings bleibt hier die Benutzerfreundlichkeit (Usability) der Website auf der Strecke. Bild- und Textelemente sowie Menüleisten sind so klein, dass man mit zwei Fingern auf dem Touchscreen in die Website hinein- und wieder herauszoomen muss, um z.B. Texte zu lesen oder durch die Website navigieren zu können.

Responsive Webdesign nimmt sich dieses Umstands an. Dem mobilen Internetnutzer wird auf seinem Gerät ein verändertes Design und Layout angeboten, am besten sogar maßgeschneiderte Inhalte. Wer wünscht sich

nicht entgegenkommende, ansprechende, also benutzerfreundliche Websites in seinem (mobilen) Browser? Genau genommen handelt es sich bei Media Queries um eine CSS-Erweiterung. Mit der erweiterten Funktionalität, insbesondere via CSS 3 und speziellen „@media“-Regeln, lässt sich eine Website für verschiedene Display-Größen, Seitenverhältnisse und vor allem unter dem Aspekt der „Usability für mobile Endgeräte“ optimieren.

Unterschiede Desktop vs. Mobile

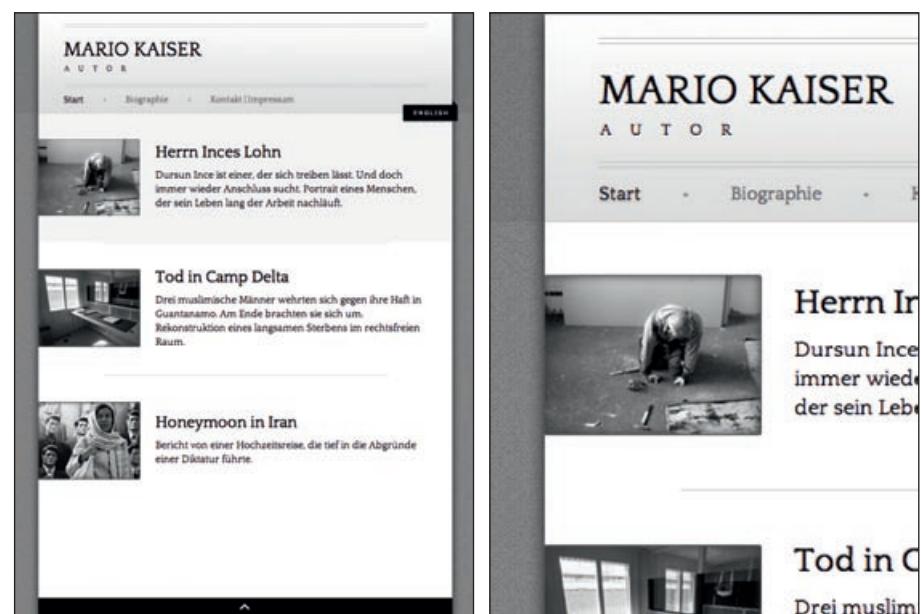
Websites sind in der Regel zunächst für Desktopcomputer oder Laptops mit einer Monitorauflösung von mindestens 1024 Pixel in der Breite ausgelegt. Genau genommen bestimmt die Größe des Browser-Fensters, wie viel wir

von der Website sehen können oder wollen. Außerdem: Jeder vom Webdesigner definierte Pixel der Website wird 1:1 auch so im Browser-Fenster des Desktop-Monitors abgebildet. Wenn die Homepage nicht in ein Fenster passt, nutzen wir die sich einblendenden Scroll-Balken. Hier bestimmt und dominiert das Betriebssystem. Mittels Maus oder Touchpad wird ein Cursor zielgenau bewegt. Mit ihm wird gescrollt, geklickt und navigiert. Mobile Endgeräte unterscheiden sich durch eine deutlich andere – und gerade bei Smartphones – kleinere Monitorauflösung. Mobile Devices bieten darüber hinaus auch die Möglichkeit, durch Drehen des Geräts um 90 Grad die Ansicht vom Hoch- in ein Querformat zu ändern (Landscape-Modus). Der mobile Browser verkleinert einfach die Website auf die Breite des jeweiligen Geräts. Jedoch kann man sich bei der kleinen Größe lediglich einen Überblick über das generelle Layout der Website verschaffen. Würden die mobilen Browser die Website nicht automatisch skalieren, würden wir wie im Bildbeispiel (Bild rechts) nur einen recht kleinen Ausschnitt der Homepage sehen können.

Durch die Verwendung von Touchscreens wird die Navigation durch eine Website vergleichsweise grobmotorisch mit den Fingern erledigt. Einen Cursor gibt es nicht. Es folgt das übliche Heranzoomen eines gewünschten Ausschnitts mit zwei Fingern. Das erscheint anfangs interessant, neu, modern und intuitiv, ist aber prinzipiell umständlich. Spätestens hierbei benötigt man schon zwei Hände – Usability für Mobile Devices sollte anders funktionieren.

Das Design

„Don't shrink, rethink.“ Diesem Ansatz von Nokia folgend, sollte man sich bereits beim Entwurf einer Homepage die entsprechenden Gedanken machen. Wie und was möchte ich in welcher Größe und Reihenfolge in mobilen Browsern darstellen. Man beginnt mit der klassischen Desktop-Version der Homepage. Dann sollte die Darstellung auf Tablet-PCs berücksichtigt werden – sowohl für den so genannten hochkant Portrait-Modus als auch für den querformatigen Landscape-Modus. Abschließend wird, gewissermaßen als Substrat, eine mobile Version für Smartphones überlegt. Da im Design die Gestaltungsmöglichkeiten gegen Unendlich konvergieren, verschafft die folgende Website einen sehr guten Überblick über aktuelle, aber auch anspruchsvolle Trends und Möglichkeiten des Responsive Webdesign: **mediaqueri.es**. Praktisch ist beim Einsatz von Media Queries, dass eine Website durch einfaches Verkleinern des Desktop-Browsers-Fensters in der Breite,



So sieht eine Webseite im mobilen Browser mit und ohne automatischer Skalierung aus.

gleich seine Responsive-Qualitäten offenbart. Eine Überprüfung einer für Mobile Devices optimierten Website ist nativ auf einen Tablet-PC oder Smartphone nicht unbedingt notwendig, sollte aber bei kommerziellen Projekten zumindest einmal überprüft werden. Die zahlreichen Beispiele von mediaqueri.es sollten aktiv durch Verkleinerung des Browser-Fensters durchgesehen werden. Das Augenmerk liegt dabei auf verschiedenen, sich verändernden Aspekten: Größe der Typografie, der Bildelemente bzw. Grafiken und deren Verhältnisse zueinander, Art und Position der Menüleiste, ursprüngliches mehrspaltiges Desktop-Layout, Transformation schrittweise hin zu einem einspaltigen System für Smartphones.

Auffällig ist auch die Veränderung bzw. Komprimierung der Menüleisten. Gerade auf einem Smartphone sollten die einzelnen Menüpunkte groß genug, also leicht mit einem Finger zu bedienen sein. Jedoch möchte man den Smartphone-User nicht ausschließlich mit einer Menüleiste plus Logo begrüßen. Etwas vom eigentlichen Content der Homepage sollte schließlich auch gleich beim ersten Aufruf der Website im Display sichtbar sein. Eine Methode wäre, das Menü komplett als Select oder DropDown (Ausklapptenü) darzustellen. Das spart Platz für Weiteres. Gegebenenfalls werden auch großformatige Aufmacher-Bilder im Kopf einer Homepage zugunsten aussagekräftiger und lesbarer Informationen ganz weggelassen. Diese Vorgehensweise folgt der Überlegung, welche Bedürfnisse der Anwender unterwegs

**Don't shrink,
rethink – gilt
als Prinzip des
mobilen Ent-
wurfs**

mit einem Smartphone in einem Bus, im Zug oder generell sich fortbewegend, haben könnte. Vermutlich ist hier eine kurze Vermittlung textbasierter Information hilfreicher als die Präsentation opulenten Bildmaterials. Letztendlich hängen solche Entscheidungen aber von der Erwartungshaltung der anvisierten Zielgruppe ab. Form und Inhalt müssen also auch im Rahmen eines Responsive Webdesigns neu aufeinander abgestimmt werden.

Der Content

Es lassen sich keine pauschalen Aussagen zur sinnvollen und gesonderten Aufbereitung des Contents für Mobile Devices treffen. Neben der einigermaßen noch eingrenzbaren Zielgruppdefinition ist es unmöglich, die Vorlieben des Anwenders im Umgang mit dem mobilen Internet vorherzusehen. Der Nutzer eines Desktop Computers oder Laptops wird in der Regel eher bequem an einem Schreibtisch sitzen und beide Hände für den Computer freihaben. Der spontane und flüchtige Einsatz eines Mobile Device an jedem denkbaren Ort, zu jeder Zeit und in jeder Situation macht hier den Unterschied. Es gilt die Herangehensweise für die inhaltliche Aufbereitung von Webseiten im Allgemeinen: Prioritäten setzen. Welche Merkmale und Angebote möchte ich dem (mobilen) Besucher der eigenen Homepage zuerst anbieten und welche dürfen eher sekundärer Natur sein? Und: Welchen Wert hat die dargestellte Information eigentlich für den Anwender? Für die letztendlich einspaltigen Layouts bei Smart-

Icon Fonts – eine typografische Alternative

• Mit sorgfältig platzierten Icons wirken Homepages ansprechender, beispielsweise für Twitter, Facebook & Co. Statt Bitmaps können Sie Icon Fonts verwenden. Der Browser rendert diese wie Typographie. Damit sind sie damit unendlich skalierbar und werden auch auf hochauflösenden Retina-Displays scharf dargestellt.



stellt. Zunächst müssen die Icons als Font-Sammlung vorliegen und via CSS-Import (@font-face ...) dem Browser zugänglich gemacht werden. Anschließend kann das gewünschte Icon in HTML aufgerufen werden. Die CSS-Definition für ein Facebook-F lautet: `facebook { font-family: Mein Icon Font; }`. In HTML schreibt man `f`. Es gibt zahlreiche lizenzfreie Icon Fonts. Jeder Anbieter liefert allerdings eigene verbindliche Tabellen für die richtige Einbindung. (Beispiel: fortawesome.github.com/Font-Awesome)

phones kommt der Aspekt ganz besonders zum Tragen. Hier ist nicht viel Platz für zweideutige Experimente. Die Homepage eines Fotografen etwa bietet gleich zu Beginn ein paar hervorragende Fotografien aus dem Portfolio an. Die Homepage eines Journalisten kommt direkt zur Sache, indem sie einige aktuelle Publikationen zum Weiterlesen anbietet. Eine gut lesbare Typografie ist hier wesentlich. Bilder könnten wiederum auf der Desktopversion der Homepage Eingang finden. Hier ist mehr Platz (vergleiche Abbildungen unten).

Die Technik

CSS 2 bot bereits die Möglichkeit, durch „media types“ bestimmte Medientypen anzusprechen. Bekanntestes Beispiel hierfür ist eine gesonderte Druckversion der Homepage, die überflüssige Hintergrundgrafiken, Menüleisten usw. beim Drucken einfach ausblendet. Media Queries greifen das Konzept auf. Jedoch wird hier nicht ein Stylesheet einem bestimmten Ausgabemedium zugeordnet, sondern es werden Eigenschaften und Fähigkeiten des Ausgabemediums automatisch erkannt und bedient. Funktional relevant ist hier die Abfrage

ge der Breite und ggf. der Höhe des mobilen Browser-Fensters bzw. des Ausgabegeräts, auch dessen Lage bzw. Position (Portrait oder Landscape) kann berücksichtigt werden. Relevant für Responsive Web-Layouts ist vor allem aber immer die Breite des so genannten Viewports. Die Abfrage der Bildschirmauflösung bzw. der Pixeldichte wäre seit der marktreifen Einführung von Retina-Displays ein weiteres Kriterium. Dazu aber später mehr. Browser-edingte Darstellungsunterschiede, wie sie Entwickler vor allem vom Internet Explorer, Versionen 6 bis 9, her kennen, sind bei mobilen Browsern eher als marginal zu betrachten. Die Hard- und Software von Mobile Devices wird seitens der Nutzer vergleichsweise sehr viel häufiger erneuert, als beim heimischen, geschweige denn firmeneigenen Desktop-PC. Insofern kann zunehmend so gut wie ohne Einschränkung das Potenzial von Media Queries ausgeschöpft werden. Als CSS-Erweiterung bearbeitet der Webdesigner das bereits vorhandene Stylesheet für Desktop-Browser um zusätzliche Media-Queries-Anweisungen. Wird nun eine Eigenschaft eines Devices erkannt, überschreiben die Me-

dia Queries die voranstehenden Anweisungen für Desktop-Browser im Stylesheet.

Die Abfrage beginnt einfach mit:

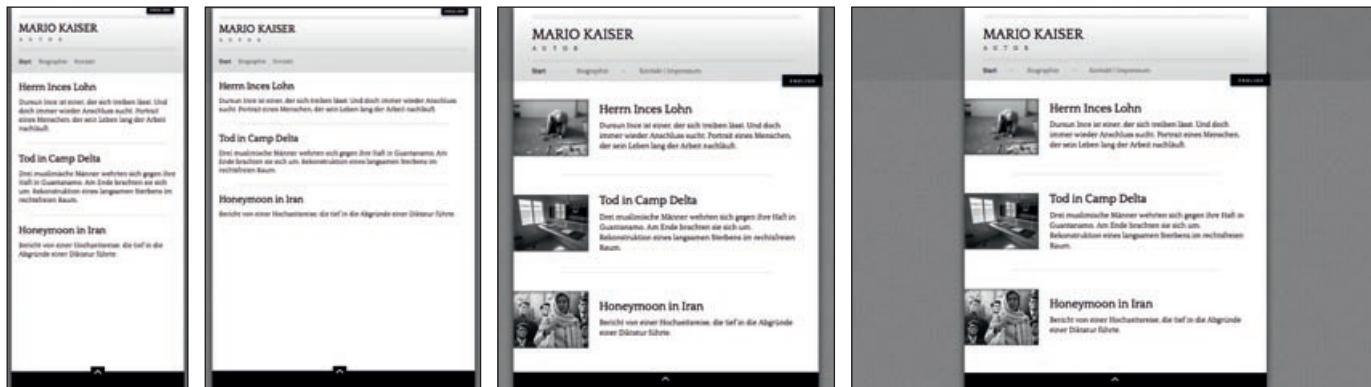
```
@media only screen and (min-width: 176px) and (max-width: 600px) { ... }
```

In diesem Beispiel wird festgelegt, welche Viewports, also Bildschirmgrößen, gemessen in Pixel von-bis, bedient werden sollen. Zwischen den geschweiften Klammern fährt man nun in klassischer CSS-Schreibweise fort, die neuen Spezifikationen für vorhandene Container und Inhalte anzugeben. Für eine geringere Breite des Gesamtlayouts ab einer Viewport-Breite von 600px bis runter zu 176px sind alle mobilen Endgeräte, aber auch Desktop-Browser bedient. Die Gesamtbreite der Homepage, die Überschrift `<h1>` und in der gleichen Eigenschaft das Aufmacherbild, skalieren sich automatisch mit. Ein Beispiel:

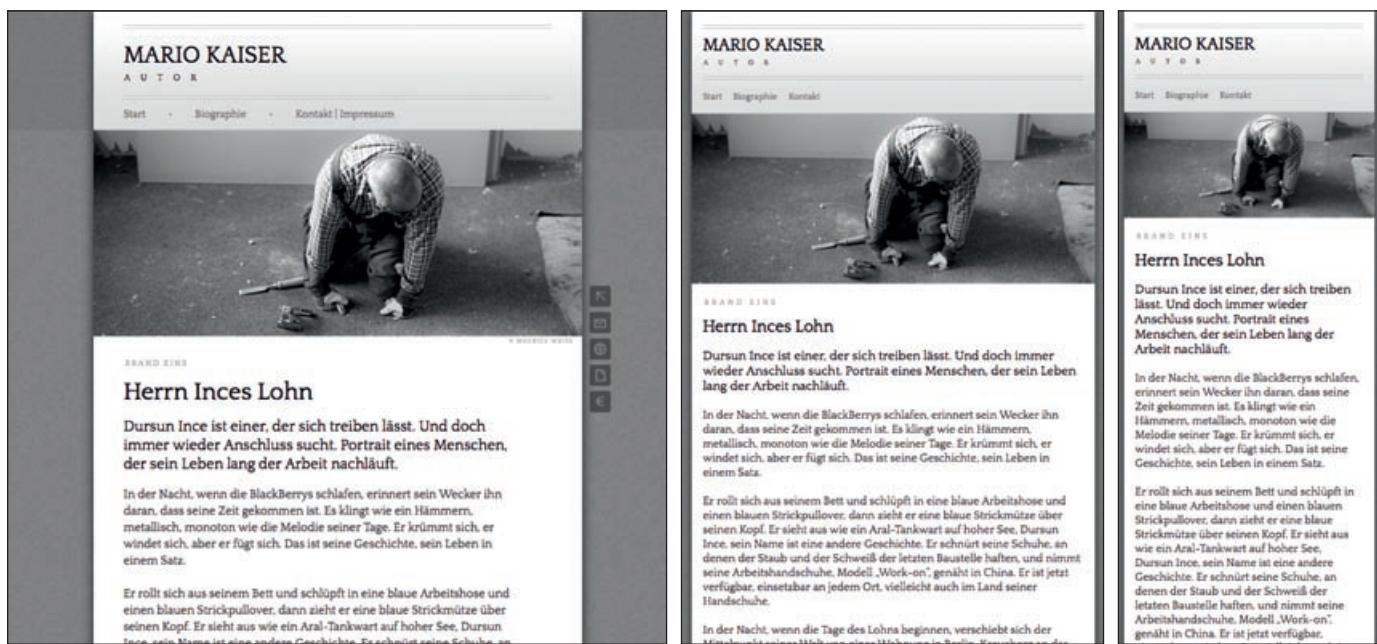
```
@media only screen and (min-width:176px) and (max-width:600px) {  
#wrapper {width:95%;} <!-- Rahmen um  
die Homepage -->  
  
#header h1 {font-size:120%} <!-- eine  
Überschrift -->  
  
#content img {max-width:100%;  
height:auto;} <!-- ein Bild --> }
```

Es zeichnet sich bereits ab, dass je nach Einsatzgebiet eine prozentuale Angabe vorteilhaft sein kann. Die Breite der Homepage, die Größe der Überschrift und das Aufmacherbild werden sich im Code-Beispiel immer automatisch dem Viewport-Size anpassen – unabhängig davon, welche Ausgabegröße der Browser zur Verfügung stellen kann Bild rechts. Sollen nur mobile Browser bedacht werden nutzt man statt `max-width:600px` einfach `max-device-width:600px`. Die Desktop-Variante der Homepage bleibt so von den Media Queries-CSS gesteuerten Designänderungen in jedem Fall unberührt.

Mehrspaltige Websites sollten auf einem Smartphone aus Platzgründen nur noch eine Spalte haben. Nebeneinander liegende Spalten werden üblicherweise durch die CSS-



Design und Content passen sich dank Media Queries an. V.l.n.r.: Smartphone, Tablet-PC und Desktop-Browser.



Prozentwerte halten das Layout flexibel, sodass sich dieses automatisch an die verschiedenen Geräte anpasst.

Anweisung `float:left;` und eine Breitenangabe positioniert. Media Queries lösen dieses Verhältnis auf, indem man schreibt:

```
#spalte1{float:none; width:auto;}  
#spalte2{float:none; width:auto;}
```

Nun werden die zwei Spalten untereinander dargestellt und sie sind auch gleich breit.

Das iPhone benötigt noch etwas Nachhilfe. In der HTML-Datei sollte im Kopf ein zusätzlicher Meta-Tag stehen:

```
<meta name="viewport"  
content="width=device-width, minimum-  
scale=1.0, maximum-scale=1.0" /><!--  
Responsive Design iPhone -->
```

Es ist auch erlaubt, ein separates Stylesheet anzulegen. Die Einbindung in HTML lautet entsprechend:

```
<link rel="stylesheet" type="text/  
css" media="only screen and (min-  
width:176px) and (max-width:600px)"  
href="style-device.css" />
```

Somit lassen sich übersichtlich für jede beliebige Viewport-Größe eigene, rein CSS-gesteuerte Designs erstellen. Ratsam ist das nicht. Schließlich sollte der zusätzliche Aufwand möglichst gering gehalten werden und dennoch ein möglichst breites Spektrum der Viewports gängiger und zukünftiger Mobile Devices abgedeckt werden. Welche Kombinations- und Ausdrucksmöglichkeiten Media Queries „@media“-Regeln“ sonst noch anbieten, kann hier detailliert erforscht werden: www.w3.org/TR/css3-media_queries.

Für einen einfachen Test der ersten Media-Queries-Gehversuche bieten sich die folgenden Webseiten an – die eigene, optimierte Website sollte dazu bereits per HTTP-Pfad

erreicht sein: www.responsinator.com und quirktools.com/screenfly. Auf diese Weise lässt sich schnell ein Überblick verschaffen, wie die Website auf Blackberrys, Androids und iPhones, bis hin zu diversen Tablet-PCs aussehen wird – Portrait- und Landscape-Ansichten eingeschlossen.

Ausblicke

Gerade unterwegs mit einem Mobile Devices sind die Bandbreiten durch den Provider begrenzt, der Datenstrom fließt langsamer. Warum muss sich der User ein in die Homepage integriertes, dekoratives Bild herunterladen, das in dieser Größe eigentlich nur für das Layout des Desktop-Browsers optimal ist? Schlimmer noch: Womöglich wird das Bild durch Media Queries einfach ausgeblendet, um ein für das Smartphone optimiertes, reduziertes Design anzubieten. Geladen wird das Bild trotzdem, weil es in die

HTML-Datei eingebettet ist. Es gibt hierzu verschiedene Ansätze, dem Problem entgegen zu wirken. Eine Lösung wäre, jedes Bild zunächst durch ein serverseitiges On-Demand-Skalierungsscript durchrechnen zu lassen und es dann erst auf den mobilen Browser zu schicken. Allerdings benötigen solche automatischen Routinen Zeit, in der das eigentlich für einen mobilen Browser zu große Bild letztendlich vielleicht doch schon heruntergeladen worden ist. Jeder zusätzliche Request, also eine Anforderung seitens des Browsers an den Server ihm eine besondere Datei zu

schicken oder für ihn eine Rechenoperation auszuführen, kostet außerdem ebenfalls Zeit. Eine weitere neue Baustelle stellen die neuen Retina-Displays dar. Deren Bildschirmauflösung ist doppelt so hoch. So ist das für ein Desktop-Browser optimierte Bild auf einem Retina-Smartphone qualitativ ausreichend. Auf einem Retina-Tablet-PC ließe sich das Bild gegebenenfalls auch noch optimal darstellen. Einem Desktop-Browser mit einem Retina-Display fehlt umgekehrt nun eine doppelt so hohe Bildauflösung. Er rendert und skaliert das Bild größer, wodurch es etwas unschärfer wird und seine darstellerische Qualität verliert. Patentlösungen sind hierfür derzeit nicht in Sicht. (Tipp für Icons auf RetinaDisplays siehe Kasten links)

Die Entwicklergemeinde zeigt sich dennoch erleichtert, dass neuere Webstandards in aktuellen Browsern zunehmend standardkonform implementiert werden. Es ist mittlerweile

so gut wie obsolet geworden, eine Homepage aufwändig für verschiedene Desktop-Browser zu optimieren. Ein ungeliebter Umstand, der Jahre lang den egoistischen Sonderwegen der Browser-Hersteller geschuldet war. Die Erweiterung einer Website durch Media Queries erscheint im Vergleich sinnvoll, da Mobile Devices einfach eine ganz andere Gerätekasse darstellen. Außerdem ist das Internet im mobilen Alltag angekommen. Der Mehraufwand für die Entwicklung von Responsive Websites ist hier gerechtfertigt und Media Queries das durchdachte Werkzeug dazu.

**Die Media
Queries sind
ein gut
durchdachtes
Werkzeug**

whs