* 1. Technischer Hintergrund

Wie bereits im Vorherigen Abschnitt erwähnt dient Responsives Design dazu, nur eine Webseite zu schreiben, die sich auf die Bedürfnisse des Endgerätes auf dem sie angezeigt wird, anpasst.

Hierzu werden neuere Webstandards wie Html5, CSS3 und JavaScript genutzt. Besonders wichtig sind hierbei die sogenannten Media Queries, welche dazu dienen, die Eigenschaften des Endgerätes abzufragen. Diese abzufragenden Eigenschaften sind Breite und Höhe des Gerätes/Fenster, Bildschirmauflösung, Eingabemöglichkeiten oder auch die Orientierung, was besonders bei mobilen Endgeräten von wichtig ist.

Im Folgenden Beispiel werden Media Queries kurz dargestellt.

Hier wird der Inhalt der Seite mit einer Breite von 800px dargestellt, es sei denn der Bildschirm hat eine breite, welche 1025px unterschreitet. In diesem Fall wird der Inhalt auf eine maximale Breite von 600px beschränkt.

#content {

**width**: 800px;

}

@media screen and (max-width: 1024px) {

#content {

**width**: 600px;

}

aside {

**display**: none;

}

}

Eine direkte Einbindung in HTML ist auch möglich. Hierbei werden separate Stylesheets erstellt, welche nun abhängig von der Displaygröße geladen werden. Dies hat den Vorteil, dass nur der Stylesheet geladen werden muss, welcher für das entsprechende Endgerät benötigt wird. Hiermit sorgt man für schnellere Ladezeiten der Webseite und einen geringeren Traffic, was vor allem für Geräte mit mobilem Internetzugang einen großen Vorteil darstellt. Dies funktioniert natürlich auch für andere Faktoren, wie beispielsweise die Orientierung des Gerätes.

<**head**>

<**link** rel="stylesheet" type="text/css" href="css/all.css" />

<**link** rel="stylesheet" type="text/css" media="all and (max-device-width: 480px)" href="css/smartphones.css" />

</**head**>

Quelle: [WIK]

Die Unterteilung, ab welcher Gerätegröße oder Auflösung man das Layout der Seite ändert, nennt man Breakpoints.

An diesen erstellten bricht man nun das Design der Seite um, beispielsweise von einem mehrspaltigen Layout in ein ein- oder zweispaltiges

Das folgende Beispiel enthält 3 Breakpoints, welche üblicherweise für Responsive Webseiten genutzt werden:

Unter 760px: Der Inhalt wird linear untereinander gesetzt, besonders für Smartphones sinnvoll

760px bis 980px: Zwei Spalten, gedacht für Tablets und kleine Browserfenster

980px bis 1280px: Drei Spalten, gedacht für das klassische Browserfenster

über 1280px: Layout bleibt unverändert, nur Schrift und Bilder werden größer



Quelle: [VENT]

Ein Großes Hilfsmittel für das Arbeiten mit Responsivem Design sind Frameworks. Diese sollen dabei Helfen, das erstellen einer Responsiven Webseite zu vereinfachen und sind gerade für Programmiere Gedacht die neu in diesem Feld sind und nicht alles von Grund auf neu schreiben wollen.

All diese Frameworks enthalten bestimmte Standarddefinitionen zu den dynamischen Design- und Strukturelementen der Webseite, wie CSS-Basierte Gridsysteme mit mehreren Spalten und vordefinierten Breakpoints, standardisierte CSS-Klassen und vor allem Problemlösung bei der Nutzung von älteren Browsern wie dem Internet Explorer 8, die Features wie CSS3 nicht unterstützen.

Besonders zu erwähnende Frameworks hierfür sind beispielsweise Bootstrap, welches Open Source ist, auf einem 12-spaltigem CSS System basiert und mit Wordpress Templates arbeitet, oder auch Foundation, Version 4, welches ebenfals Open Source ist und besonders für Einsteiger empfohlen wird. [RDT]

2.4 Vor- und Nachteile von Responivem Design

Einer der Großen Vorteile von Responsivem Design ist der geringere Pflegeaufwand im Vergleich zu zwei oder gar mehreren Webseiten für verschiedene Endgeräte.

Da nur ein HTML Code geschrieben und betreut werden muss, können deutlich weniger Kosten für den Auftraggeber oder Betreiber der Webseite anfallen, weshalb sich immer öfter für Responsives Design entschieden wird. Außerdem müssen neue Inhalte nur ein mal eingefplegt werden, was dauerhaft für weniger Arbeit und Kosten sorgen kann.

Leider existieren auch viele Probleme mit Responsiven Webseiten.

Da der gleiche Inhalt der auf dem Desktop dargestellt wird umgestellt werden muss, sehen mobile Seiten meist sehr lang aus und der Nutzer muss wesentlich mehr Scrollen, um ein gewünschtes Ergebnis zu finden. Zudem ist die Navigation meist im Rahmen der CSS-Möglichkeiten angepasst, was für Potenzielle Usabilitykonflikte sorgen kann und oft dazu führt, dass mobile Seiten überladen wirken und der Nutzer sich schwieriger zurecht findet. Dies kann dazu führen, dass Kunden beispielsweise länger nach einem Gewünschten Produkt suchen müssen, was sich negativ auf Verkaufszahlen und die Reputation beispielsweise eines Onlineshops auswirken kann.

Außerdem stellt sich oft die Frage wie viele verschiede Designs tatsächlich benötigt werden.

Befürworter von Responsivem Webdesign Argumentieren oft, dass immer mehr Geräte mit wesentlich unterschiedlichen Displaygrößen Internetfähig werden und das alte Modell, in dem für jedes Gerät eine eigene Seite Designt wird, immer aufwendiger wird.

Hier widerspricht jedoch Patrick Voelcker, der wie viele andere ein Gegner von Responsivem Design ist mit dem Argument, dass obwohl immer mehr Displaygrößen ihren Weg ins Web finden, zwei Seiten völlig ausreichend sind da beispielsweise auf Tablets, besonders mit den stetig steigenden Auflösung, sich Desktopvarianten als die Beste Möglichkeit des Betrachtens erwiesen haben und das tatsächlich nur zwei Varianten einer Seite benötigt werden, eine Mobile und eine für alle anderen Geräte. [SCHROTT]

[WIK] http://de.wikipedia.org/wiki/Cascading\_Style\_Sheets#Eigenschaftsspezifische\_Stylesheets\_.28Media\_Queries.29

[VENT]

http://www.mediaevent.de/css/media-query.html

[RDT]

http://www.responsive-design-test.de/responsive-design-frameworks

[SCHROTT]

http://www.mobile-zeitgeist.com/2013/08/07/warum-responsive-webdesign-schrott-ist/

http://www.webmasterpro.de/coding/article/css-responsive-webdesign-media-queries-fuer-iphone-ipad-und-smartphones.html

http://responsivedesign.is/strategy/page-layout/defining-breakpoints

https://developers.google.com/web/fundamentals/layouts/rwd-fundamentals/how-to-choose-breakpoints