

Am Anfang war...Der Webmaster Kurs für Jedermann! Die Wahl des CMS



Jeder fängt mal von vorne an – auch als Webmaster. Und jeder kennt sie, die kleinen und großen Probleme bei der Gestaltung einer Webseite. Ab sofort stellen wir euch regelmäßig unsere großen und kleinen Probleme mit unseren privaten (!) Webseiten vor – und wie wir sie gelöst haben.

Mit “wir” sind Attila und meine Wenigkeit gemeint. Er kommt aus dem Vertrieb, ich selber bin für die Content-Produktion der SEO Küche zuständig. Wir sind keine OnPage-ler, bei unseren Kollegen bedanken wir uns aber für ihren Rat und Expertise!

Natürlich gibt es für jedes Problem am CMS, am Server oder in der Datenbank eine Lösung im Netz. Uns geht es aber nicht darum einfach einen Link zur Lösung zu posten, sondern das *Was* (ist eigentlich das Problem), das *Warum* (ist es ein Problem) und das *Wie* (haben wir das Problem gelöst) im Detail vorzustellen

Jedermann fängt mal von vorne an, deswegen ohne Umschweife:

Die Wahl des CMS

Was ist das Problem: Wir brauchen ein CMS

Warum brauchen wir ein CMS?: Es spricht nichts dagegen, eine reine html-Seite aufzusetzen. Mit der

Hilfe von [Selfhtml](#) kann man sich die Grundlagen der Webseitengestaltung auch selbst beibringen. Aber ein CMS bietet von Anfang an mehr Möglichkeiten, ist schöner und einfach praktikabler.

Wie entscheiden wir uns für ein CMS?: Man legt zuerst ein Ziel für die Webseite fest. Wozu soll die Seite existieren?

Mein Ziel ist denkbar langweilig: Möglichst einfach Grundlagen der Webseitenerstellung und der Webseitenpflege, also html, php und java, erlernen. Inhaltlich haben wir keine Ansprüche, deswegen mache ich einen Blog mit (ausführlicheren) Buchbesprechungen.

Bei der Wahl des Blogsystems habe ich mich für [Anchor CMS](#) entschieden. Wieso?

Natürlich ist Wordpress das Non Plus Ultra: einfach zu bedienen, tausende Plugins, unendliche Themes, für kleine wie für gigantische Webseiten geeignet. Egal ob Shop, Blog, oder Unternehmenspräsenz: WordPress hat für jeden Wunsch die passenden Konfigurationen.

Anchor CMS ist 1.1 MB groß, Open Source und die neueste Version ist 0.9.2. Es gibt eine Handvoll [Themes](#), bisher keine Plugins und eine unfertige offizielle Webseite. (<http://anchorcms.com/docs/managing-content/metadata>). Die einzige Google Plus Community hat stolze 25 Mitglieder (Stand 6.8.2014). Achja, und keine Unterstützung für IE 6/7/8!

Praktisch keine Zeile ist hardcoded. Man kann alles am Design oder der Funktionalität selber ändern – sofern man die nötigen Kenntnisse in html, php und java mitbringt.

Und genau deswegen eignet sich Anchor CMS *perfekt* für eine Reihe über die Grundlagen der Webseitenerstellung und Webseitenpflege.

So viel dazu, Attila hat sich noch ein paar Gedanken mehr gemacht:

Zielsetzung: Es soll eine Informationsseite aus dem Bereich Tierwelt und Tierhaltung wieder in Betrieb genommen werden, die seit einem Jahr offline ist. In einem zweiten Schritt soll ein Online-Shop um das Thema gebaut werden, welches über die gleiche Domain erreichbar sein soll. Die ursprüngliche Contentseite soll größtmöglich nachgebildet werden; bei dem späteren einbinden des Online-Shops sollen Elemente wie der Warenkorb oder Produktvorschläge auf den Content-Seiten nicht auftauchen.

Online-Marketing-Sicht: die Seite selbst hatte sich bis zur Offline-Schaltung sehr gut entwickelt und

kurz danach die komplette Sichtbarkeit verloren. Die Seiten waren allerdings bis Mai diesen Jahres noch im Google-Index (Trust?)



Die dazu gehörigen Facebook-Seiten wurden seit dem weiter gepflegt und sind nach wie vor sehr erfolgreich.

1. Schritt: Domain

Da wir die alte Seite neu bauen wollten ist die Domainwahl nicht sehr schwer gefallen, vor allem weil wir dadurch die Wiedererkennung des Brands gewährleisten können und ggf. noch Trust seitens Google vorhanden ist. Die wirkliche Frage ist, wie wir die Domain des späteren Shops handhaben sollen. Dazu gibt es drei Möglichkeiten:

- Neue Domain: aus den oben genannten Gründen würde das nicht in Frage kommen, zumal es ein großer Vorteil für jeden Shop ist guten Content auf der eigenen Domain bereit zu stellen
- Subdomain (z.B: shop.meinehomepage.de): die Wiedererkennung des Brands wird hierdurch gewährleistet, leider sieht Google Subdomains als eigenständige Webseiten. Daher: No-Go
- Unterkategorie (z.B: meinehomepage.de/shop): wir haben uns für diese Variante entschieden. Es

bietet Usern einerseits sehr gute und Informative Inhalte, andererseits können sie auf der Seite bleiben, wenn es um eine Kaufabsicht geht. Google hat dementsprechend viel Content zum bewerten und dadurch werden auch den Shopseiten gute Rankings ermöglicht

2. Schritt: System

Bei der Systemwahl stehen verschiedene namenhafte (und weniger namenhafte) CMS-e und Shops zur Verfügung. Hier muss man sich einfach im Klaren sein wohin die Reise hinget und welche Ansprüche man an die zukünftige Homepage stellt. Unsere Anforderungen waren ganz klar: es sollten Contentseiten ohne weiteres erstellt werden können und diese sollten leicht auf das alte Design angepasst werden können.

In solchen Fällen denkt man ganz schnell an einfach zu bedienende und gleichzeitig sehr wandelbare CMS-e (Content Management Systeme) wie WordPress oder Typo3. Beides sehr gute Möglichkeiten, wobei hier eher WordPress punkten kann, da es sehr beliebt ist und entsprechend viele Plugins dafür programmiert werden. Wichtig für den späteren Shop: beide Systeme verfügen über Shop-Plugins.

Hierbei eine kleine Anmerkung: CMS-e sind im Normalfall Informationsseiten und sind nicht dafür ausgelegt Käufe abzuwickeln. Es fehlen oftmals grundlegende Features wie Cross-Selling und man hat eine bestimmte Begrenzung an Kategorien und Produkten. An unserem Beispiel: wir haben nur 5-7 verschiedene Produkte, die allerdings mit jeweils 3-4 Variablen hantieren. Eines dieser Variablen hat ca. 30 Auswahlmöglichkeiten. Hochgerechnet sind wir auf ungefähr 200 verschiedene „Produkte“ gekommen. Die Shopplugins der Content Systeme sind für eine solche Anzahl nicht konzipiert.

Dann steht die Frage im Raum: wäre es nicht besser einen Shop zu bauen, der zunächst nur Content-Seiten hat und später um den Shop erweitert werden kann? Gängige Lösungen in dem Bereich sind Magento, Gambio oder Shopware. Da ich bis dato nur mit Magento gearbeitet habe, haben wir uns dafür entschieden, es sei hier aber gesagt, dass Gambio als verständliches und einfach zu bedienendes Shopsystem gilt, was den Aufbau der Seite erheblich leichter machen könnte.



3. Schritt: Hoster

Sobald man sich für ein System entschieden hat, fällt die Hosterwahl etwas leichter. Man muss hier unbedingt auf die Anforderungen der CMS oder des Shops schauen und sich entsprechende Anbieter und Server suchen!

Ein Tipp am Rande: Kundenrezensionen helfen bei der Wahl enorm. Hier zählen vor allem viele positive UND negative Erfahrungen, damit man weiß worauf man sich einlässt.

Man sollte auch unbedingt darauf achten, dass der Hoster nicht nur Zugänge zum Adminpanel ausgibt, sondern auch direkte Zugänge zum Server. Es gibt auch in dieser Branche schwarze Schafe die nur selbst Änderungen vornehmen und das gern mal in Rechnung stellen.

Was haben wir heute gelernt?

- Immer konkrete Ziele setzen. Nicht nur fragen „was will ich bauen?“ sondern „wo will ich hin? Was will ich erreichen?“
- Man sollte sich immer gut vorbereiten und zukunftsorientiert denken, sonst baut man nach einem Jahr Betrieb gleich mal die Seite komplett neu
- Die Systemwahl richtet sich nach den Anforderungen Ihrer Seite, aber nicht vergessen: man sollte nicht vergessen, das eine CMS kein Shopsystem ist und umgekehrt das auch nicht ganz Reibungslos klappt (dazu später mehr!)

[Auf zum nächsten Beitrag.](#)