**Anforderungen – CMS**

**Zielgruppe (mögliche Verwendungen)**

* + Blogsysteme (Blogger aller Art zB. Kosmetikblog, Modelleisenbahnblog…), einfache Firmenwebseiten, ähnliche, einfach gehaltene Webseiten…

**Registrierung von Benutzer**

* + nicht registrierte Benutzer können nur die öffentlichen Artikel lesen
  + alle Benutzergruppen registrieren sich über das selbe Formular
    - Angaben im Formular:
      * Emailadresse
      * Vorname
      * Nachname
      * Nickname/Benutzername
      * Geburtsdatum (? 🡪 je nachdem welche Inhalten der Admin zur Verfügung stellt vielleicht für ein CMS notwendig)
  + Gäste sind in unserem System registrierte Benutzer
  + Gäste sind generell nur für die Nutzung des Gästebuchs freigeschaltet und können bei Missachtung der Höflichkeitsregeln o.ä. ggf. für immer oder einen bestimmten Zeitraum gesperrt werden
  + Redakteure und Designer müssen sich nach ihrer Registrierung mit dem Admin in Verbindung setzen und sich die für ihre Aufgaben notwendige Rechtefreigabe, z.B. zur Templateerstellung oder Inhaltsverwaltung einholen
  + Admin legt Benutzer im Backend an und gibt die Daten weiter an den Redakteur, Designer usw. => Benutzer kann nicht selber sagen, dass er Redakteur etc. werden will; er kann seine Rolle NICHT selber festlegen.
  + Administrator legt die Rolle fest.
* Wenn der Benutzer schon als Gast angelegt ist, dann kann der Admin ihn „hochstufen“. Admin muss das wissen. Austausch zwischen Admin und Gast

**Anmeldefunktion und Abmelden**

(für verschiedene Benutzer (Designer, Admin, Gast, Redakteur, …)

* + Frontend
  + Backend
  + Die Anmeldung im Backend erfolgt nicht über das Frontend.
    - Wenn ich auf dem Frontend bin und mich da anmelde, dann bin ich auf dem Frontend.
    - Wenn ich im Backend bin, dann kann ich mich im Backend anmelden => eigene Seite.
  + Benutzer sollen sich wieder löschen/entfernen können
  + Mehrfach falsche Eingabe von Passwort 🡪 Account wird gesperrt
    - Dreifach falsche Eingabe => PopUp
    - 5x falsche Eingabe => 0,5h sperren
  + Benutzer soll bei vergessenem Passwort ein neues per Email anfordern können

**Gästebuch**

* + Admin kann selber entscheiden ob er ein Gästebuch haben will oder nicht
    - so Art Plugin
  + es soll nicht möglich sein, Kommentare bei den einzelnen Artikeln (=Blogeintrag etc.) abzugeben, sondern nur im Gästebuch in Form von Gästebucheinträgen
  + Schimpfwörter sollen entfernt werden
  + Abmahnung an die User oder ggf. komplett oder für gewissen Zeitraum sperren
  + evtl. Nutzeranalyse anhand der Gästebuch-Einträge
  + Gästebuch – Schimpfwörter:
  + Version: Schimpfwörter von vornherein entfernen => d.h. der Check ist erst beim Abschicken und die Schimpfwörter werden durch \* oder andere Zeichen ersetzen. Der Check der Wörter ist nicht beim Eingeben direkt sondern beim Abschicken, d.h. bevor es in die Datenbank kommt wird der „Schimpfwortfilter“ darüber gelegt.

**Erstellung von Templates**

* + eigene Templates können direkt in unserem CMS erstellt werden
  + es ist NICHT möglich, fertige Templates aus dem Internet zu verwenden
  + zwei Beispieltemplates in dem CMS als Vorlage
  + die Erstellung der Templates soll so einfach wie möglich erfolgen
    - Über „Auswahlfelder“ 🡪 Nutzer braucht für Standart-Templates KEINE Programmierkenntnisse, wenn ihm die gegebenen Auswahlmöglichkeiten zu wenig sind, braucht er jedoch schon Programmierkenntnisse, da dann Veränderungen direkt im Code passieren müssen
* **Artikel einer bestimmten Page hinzufügen**
  + Text, Videos, Bilder, Tondateien …
  + Live-Texteditoren für die Erstellung der Artikel
  + neu hinzugefügter Artikel kommt immer ganz **oben** auf die Seite - alle anderen wandern je eine Position nach unten (wir haben bei den Templates die Möglichkeit, dass der Nutzer das aussuchen kann 🡪 Anforderung falsch oder falsch umgesetzt!)
  + jeder Autor weist selbstständig seinem Artikel ein oder mehrere freigewählte Tags zu
  + jeder Autor darf nur auf den Seiten Artikel veröffentlichen für die er die Rechte hat und nicht überall im ganzen System
  + Zeichenbegrenzung der Tags selber
    - ein Tag darf nicht aus mehr als 50(?) Zeichen bestehen
    - nicht: 1321wer97asd3f21w48r7a65sd1f321asd68ta3541321….36767
    - Tag müssen kurz und knapp sein;
    - Zahlen sind schon erlaubt;
    - evtl. mit #
    - für Artikel die Tags möglich; die muss er Redakteur vergeben
    - bei Gästebucheintrag noch nicht sicher => evtl. anders!

\*Tags = Label in DB

* + - * Groß- und Kleinschreibung bei der Eingabe ist egal weil unsere Eingabe sofort in Kleinbuchstaben umgewandelt wird. (So auch in der Datenbank gespeichert, um Duplikate zu vermeiden)
      * Die Suche der Tags erfolgt **innerhalb** unserer Webseite.

**Suchmaschinenoptimierung**

* z.B. Schema.Org =>
  + - <temp>20grad</temp>
  + Nice-to-have => nicht unbedingt das Relevanteste.
  + Suchmaschine für Suchmaschinenoptimierung ist Google

**Verwendung von folgenden Frameworks:**

* + Zend für vereinfachten Datenbankzugriff und Benutzerauthentifizierung
  + Frameworks NICHT verpflichtend, nur dort wo wirklich sinnvoll anwenden

**Bootstrap**

* + nur für das Backend
  + für das Frontend nicht verwenden
    - Grund:

Admin des CMS soll das Aussehen des Frontend selber festlegen können. Wenn er es unbedingt haben will, dann soll er Bootstrap verwenden aber der Admin soll nicht dazu gezwungen werden.

**Rechteverwaltung durch Rollen**

* + Admin kann eigene Rollen anlegen
    - er kann einer Rolle bestimmte Rechte zuweisen
    - Jede Rolle „Gast“ hat die gleichen Rechte. Wenn ein Gast andere Rechte haben soll, dann muss der Admin neue Rolle anlegen (z.B. Premiumgast).

**Kontoübersicht im Frontend und Backend**

* + Übersicht über das eigene Konto im Backend, dazu gehört
    - Benutzername, Passwort etc. ändern
    - die von mir erstellten Beiträge ansehen/bearbeiten => Übersicht
    - angemeldete Benutzer können sich über das CMS **KEINE** Mails schreiben.
    - Öffentlich sichtbar bei registrierten Benutzern NUR der Benutzername/Nickname

**Barrierefreie Webseiten**

* + Webseiten für Sehbehinderte, ...
  + Wird im Frontend nicht verwendet sondern nur Im Backend darauf achten
    - => Rot-Grün-Schwächen berücksichtigen
    - <nav> verwenden für Screen-Reader
    - Schöne, leicht lesbare Schriftart für das Backend
  + Wenn das Frontend barrierefrei sein soll, dann muss der Admin, dh. Seitenbetreiber darauf achten.

**Plugins**

* + Admin soll bei Bedarf ein Menü einfügen können (von uns zur Verfügung gestellt)
  + Admin soll bei Bedarf Suche einbinden können (von uns zur Verfügung gestellt)
  + Admin soll bei Bedarf Gästebuch einbinden können (wenn Zeit vorhanden von uns realisiert und zur Verfügung gestellt)
  + Admin soll die Möglichkeit haben, PlugIns von „externen“ Anbietern (egal ob kostenlos oder kostenpflichtig) einbinden können. (z.B. PlugIn für Shop-Webseiten, Temperaturanzeigen, Eine Uhr, einen Countdown … )
* Wenn Zeit vorhanden, dann das machen.

**Allgemeines/Weiteres**

* Backend-Größe => nicht für Handy ausgelegt.
* Admin kann die Benutzer für einen bestimmten Zeitraum oder für immer (sehr großer Endwert: 31.12.4897) sperren
* Pages anlegen / Seiten erzeugen (können von berechtigter Person erzeugt und über Seitenverwaltung verwaltet werden)
* Einbinden von sozialen Netzwerken falls noch Zeit ist