



# Web Engineering 1

Labor zur Vorlesung

„Gestaltung von flexiblen Webseiten-Layouts (CSS)“

Prof. Dr. Schneider

## 1. Erstellen einer flexiblen Web-Seite mit Layout

Bisher haben Sie HTML verwendet, um Webseiten logisch zu strukturieren. Damit konnten Sie sehr einfache Webseiten generieren. Das Webseiten-Layout bestand im Wesentlichen aus einer linearen Anordnung der Blockelemente (Schichtung von Blockelementen).

In diesem Kapitel sollen Sie eine Webseite erstellen die neben der gezielten Auswahl von Schriftarten, Farben, usw. zusätzlich ein aus mehreren Spalten bestehendes und für unterschiedliche Fenster- bzw. Bildschirmgrößen angepasstes Design besitzt.

### 1.1. Edward Snowden – XML-Datei

1. Ergänzen Sie ihre Edward Snowden XML-Textformatierung (Kapitel2) mit einem den Formatierungen entsprechenden CSS-Sheet.
2. Überprüfung Sie ihr XML-Textdokument auf Korrektheit in einem Browser.

### 1.2. Flexibler Onepager

1. Ihre Aufgabe besteht darin eine [OnePager](#)-Webseite mit [css-Stylesheets](#) zu gestalten. Die Webseite soll ein responsives Verhalten besitzen, d.h. bei der Verkleinerung des Browserfenster oder beim Aufruf der Seite auf einem Gerät mit kleinerer Bildschirmbreite soll die Anzeige entsprechend angepasst werden.
2. Die minimalen Anforderungen an die Webseite sind
  - Webseite mit dem Namen `index.html` im Root-Verzeichnis.
    - `done`
  - Ein CSS-Stylesheet (im Unterverzeichnis `css`) mit dem Sie die Seite gestalten: Farbe, Schriften und Schriftgrößen, Media Queries, ...
    - `done`
  - Titelleiste über die gesamte Breite.
    - `done`

- Seitlich (links) ein Navigationsmenü mit min. 4 Links als Menüpunkte. Die Links können in Form von Ankern realisiert werden und vermitteln ein 3-dimensionelles Verhalten
  - done
- Der Inhaltsbereich besteht aus 3 Spalten. Pro Spalte befindet sich ein Textbereich und ein Bereich mit Multimedia-Dateien.
  - done
- Seitlich rechts ein Bereich, in dem Sie Fachbegriffe die Sie in ihrem Inhaltsbereich verwenden erklären können.
  - done
- Weitere Elemente die Ihnen Spaß machen.
  - done
- Bei maximaler Fensterbreite ordnen sich die Spalten im Hauptbereich der Seite nebeneinander an, bei minimaler Breite ordnen sich die Hauptbereiche untereinander an.
  - done

Beispielhaft können Sie sich an der folgenden Abbildung orientieren

# MEINE FLUIDE WEBSEITE

- Startseite
- Media Queries
- Flex Boxen
- Grid Boxen
- Impressum

## Was sind fluide Webseiten?

Diese Webseite erklärt die wesentlichen Bausteine zur Erstellung einer responsive Webseite.

### Media Queries

Weit hinten, hinter den Wortbergen, fern der Länder Vokalien und Konsonantien leben die Blindtexte. Abgeschieden wohnen sie in Buchstabenhausen an der Küste des Semantik, eines großen Sprachozeans. Ein kleines Bächlein namens Duden fließt durch ihren Ort und versorgt sie mit den nötigen Regelialien.

### Flex-Boxen

Es ist ein paradiesmatisches Land, in dem einem gebratene Satzteile in den Mund fliegen. Nicht einmal von der allmächtigen Interpunktion werden die Blindtexte beherrscht – ein geradezu unorthographisches Leben. Eines Tages aber beschloß eine kleine Zeile Blindtext, ihr Name war Lorem Ipsum, hinaus zu

### Grid-Boxen

Der große Oxmox riet ihr davon ab, da es dort wimmelte von bösen Kommata, wilden Fragezeichen und hinterhältigen Semikoli, doch das Blindtextchen ließ sich nicht beirren. Es packte seine sieben Versalien, schob sich sein Initial in den Gürtel und machte sich auf den Weg.

### Holy-Gray-Layout

Der Heilige Gral ist ein Webseitenlayout das aus mehreren Spalten gleicher Höhe besteht. Die Definition des Layouts erfolgt mit Stylesheets.

Grid Boxen ermöglichen die einfache Umsetzung eines Holy Gray Layouts. Dabei erhalten die einzelnen die einzelnen Spalten und Reihen zum Teil feste, zum Teil flexible Höhen und Breiten.

### Mobile First

Bei einem Mobile First Design startet man mit dem Webseiten-Layout auf dem kleinsten

3. Erstellen Sie zuerst **Sketche** ihrer Web-Seite, die diese in Abhängigkeit von der Bildschirmbreite darstellen. Überlegen Sie sich bei welcher Bildschirmbreite (**Breakpoints**) Sie den Übergang zwischen den verschiedenen Prototypen initiieren.

**Breakpoints:**

Gedacht für	Breite Min	Breite Max
Große Bildschirme (Notebook, Monitor, ...)	1024px	-
Mittlere Bildschirme (gedacht für Tablets)	600px	1023px
Kleine Bildschirme (gedacht für Handys)	-	599px

**Sketche sind als separate PDFs im Abgabeordner hinterlegt.**

4. Definieren Sie jetzt zusätzlich anhand der Skizze Schriftarten, Schriftgrößen und die farbliche Gestaltung ihrer Webseite?

- **Header**
  - Farbe: weiß
  - Größe: 3em
  - Schriftart: Helvetica Neue (bold)
- **Navigation**
  - Farbe: weiß
  - Größe: 1em
  - Schriftart: Helvetica Neue (bold)
- **Aside**
  - Farbe: schwarz
  - Größe: 1em
  - Schriftart: Helvetica Neue
- **Elementübergreifen:**
  - <h1>
    - Font Family: Verdana, Geneva, Tahoma, sans-serif
    - Größe: 26pt

- `<h2>`
    - Font Family: Verdana, Geneva, Tahoma, sans-serif
    - Größe: 16pt
  - `<h3>`
    - Font Family: Verdana, Geneva, Tahoma, sans-serif
    - Größe: 13pt
  - `<p>`
    - Font Family: Verdana, Geneva, Tahoma, sans-serif
    - Größe: 12pt
5. Erstellen Sie ein Kompositionsdiagramm und spezifizieren Sie das Grundgerüst ihrer Webseite. Wählen Sie ein geeignetes Verfahren für das responsive Design (Flex-Box, Grid-Box, ...). Definieren Sie je nach ihrer Layout-Gestaltung `id`-Attribute und `class`-Attribute über die sie anschließend gezielt per CSS ihre Seite gestalten können.
  6. Kompositionsdiagramm ist als separate PDFs im Abgabeordner hinterlegt.
  7. Nach Fertigstellung ihre HTML-Seite testen Sie diese auf verschiedenen Browsern.
    - Getestet auf Edge, Chrome, Firefox, Safari
  8. Erstellen Sie jetzt die zugehörige CSS-Stylesheet-Datei (`layout-basic.css`), in der die gewünschten Farben und Schriften der Webseite hinzufügen. Testen Sie das Aussehen der Webseite. Stimmt ihr Ergebnis mit dem ursprünglichen Entwurf überein?
    - Ja Ergebnis stimmt mit dem Entwurf überein
  9. Setzen Sie jetzt das Layout für das responsive Verhalten um. Legen Sie dazu eine weitere CSS-Stylesheet-Datei (`layout-responsive.css`) an und definieren in dieser das Verhalten an und zwischen den Breakpoints.
    - Layout-responsive.css wurde angelegt und das Verhalten mittels `@Media` festgelegt. Einmal Design für kleine Displays wie Handys, dann mittlere wie Tablets und ebenfalls für große wie Notebook oder Monitore.
  10. Testen Sie das Verhalten der Webseite bei verschiedenen Fensterbreiten.
    - Wurde getestet
  11. Geben Sie ihre Webseite zum Testen einem anderen Kursteilnehmer. Besteht dieser den Test? Wie gefällt ihre Webseite dem Tester?

- Test war erfolgreich und wurde von den Gruppenmitgliedern abgenommen und für gut befunden.