Dokumentation Portfoliowebsite

# Einleitung:

Das Projekt begann mit der Sichtung der Kriterien. Dabei schaute ich mir vor allem die von Herr Erb verlinkten Beispiele genauer an. Da alle Beispiele singe-page-webapps waren, plante ich eine solche zu gestalten. Am Abgabetag wurde dann klar, dass es tatsächlich keine single-page App sein sollte, sondern aus mehreren Webseiten zu bestehen hatte. Glücklicher weise hatte ich die Webseite in größere Teilbereiche wie Info, Hobbie, Projekte und Kontakt unterteilt. Diese Sections wurden dann auf verschiedene Webseiten aufgeteilt, die alle das selbe Format und die selbe Navbar hatte.

# Struktur:

Home

Hobbies

Contact

Projects

CV

(Lebenslauf)

styles.css

index.js

Die einzelnen Webseiten sind, wie durch die blauen Pfeile dargestellt, mit Hyperlinks untereinander verlinkt. Es gibt nur eine einzelne CSS und ein JS Datei.

Jede Webseite hat dieselbe Navbar, nur fehlt der jeweilige link zur Seite, auf der man gerade ist. Außerdem haben alle die selbe Fußzeile.

Die section mit der Klasse navbar dient als Container für die a Elemente, die zu einer anderen Webseite linken.

Der Rest des bodys nur ein linear gradient background, der nicht vom Scrollen beeinflusst wird. Darüber liegt das div mit der id „main“, auf dem sich der Rest der Webseite abspielt. Es ist je nach Bildschirmbreite 50% bis 400px breit. Je schmaler der Bildschirm, desto schmaler.

Home:

Das „main“-div der Home Webseite enthält im Wesentlichen zwei div Elemente, die ja nach breite nebeneinander oder übereinander sind. Das eine Enthält das Profilbild, das andere eine Tabelle mit einem Steckbrief über mich. Dieser Steckbrief hat die Form einer Tabelle.

Projects:

Das „main“-div enthält hier vor Allem eine ul also ungeordnete Liste mit projekten. Diese stichpunkte enthalten meist einen Link zu dem Github repositorie.

CV:

Auch hier ist das „main“-div wieder von Interesse, denn es enthält einen Lebenslauf in Form einer Tabelle.

Hobbies:

Enthält eine ul von meinen Hobbies.

Contact:

Enthält zwei Bilder. Das eine linkt direkt zu meinem Github Profil, das andere kopiert meine E-Mail-Addresse, mit Hilfe von JavaScript in die Zwischenablage des Klienten.

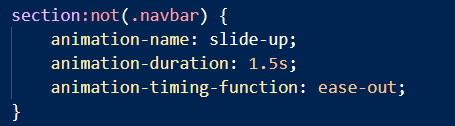
## Besondere Stellen:

Auf der kontaktwebseite wird eine E-Mail-Addresse in die Zwischenablage des Klienten kopiert. Das geschieht mit Hilfe einer simplen JavaScript fuction, die in index.js wie folgt lautet und funktioniert:

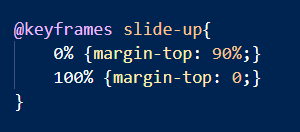


Damit sie auch beim Klicken des Bildes ausgeführt wird, hat dieses img-Element einen OnClick tag mit dem Namen dieser Funktion.

Jede Section (außer die Navbar) hat eine Slide in Tween-Animation beim Betreten der Webseite. Dies funktioniert indem man in CSS sog. Keyframes definiert und zu einem Element mit Hilfe eines Selektors hinzufügt. Außredem braucht man noch eine Dauer, über die die Animation ausgeführt wird und wahlweise eine function, wie die Animation im Verhältnis zur Zwi taubläuft (z.B. linear, ease-out). Das Ganze sieht dann folgendermaßen aus:



Die dazugehörigen keyframes sind die Folgenden:



Zu Beginn der Animation ist sie auf Grund der Margin top fast nicht zu sehen, da sie so weit unten ist. Im Verlaufe der Animation wird das immer weniger, bis es schließlich ganz oben angekommen ist.

# Reflexion:

Die Grundstruktur fiel mir relativ leicht, die Animationen waren für mich allerdings neu und es hat doch kurz gedauert das richtig hinzubekommen. Die generelle Anwendung von JavaScript war für mich nicht neu, ich musste nur die navigator.clipboard.WriteText-Funktion suchen. Außerdem kam die Erkenntnis, dass es keine Singlepage App sein sollte etwas spät, was das Ganze dann doch kurz stressig machte, sich doch auch leicht beheben lies. Das Projekt hat mich wieder zum Front-end-development geführt und gezeigt, dass es gar nicht so schlimm ist, wie ich es in Erinnerung hatte.