

Hier findet sich eine Reflexion über die Entwicklung dieser Seite, in welcher hoffentlich auch etwaige Lernerfolge im Modul „Multimediale Webprogrammierung“ verdeutlicht und in einen Anwendungskontext eingesetzt werden.

Um diesen Text etwas zu strukturieren, würde ich erst einmal auf die allgemeine Umsetzung aller Unterseiten des Portfolios eingehen und danach jede Unterseite genauer analysieren, um eingesetzte Medienformate und Techniken genauer zu beleuchten.

Der gesamte Stil des Portfolios setzt auf das Framework Bootstrap, um eine vor allem responsive Erfahrung zu gewährleisten, ohne dass ich selbst auf sämtliche Bildschirmgrößen achten muss. Des Weiteren hilft solch ein Framework auch dabei, eine einheitliche Webseite erschaffen zu können, indem man sich auf einzelne Designelemente spezialisiert, dazu später auch noch einmal mehr. Ich habe mich speziell für Bootstrap entschieden, weil wir im Laufe des Moduls „Multimediale Webprogrammierung“ in diesem Semester schon einmal damit gearbeitet haben und ich damit sozusagen schon einmal „warm“ war und mich nicht extra zu einem anderen Framework neu informieren musste. Die gesamte Webseite besteht aus 7 Unterseiten, die alle durch eine uniforme Navigationsleiste erreichbar sind. Diese Navigationsleiste wurde direkt aus Bootstrap importiert und hat den Vorteil, bei kleineren Bildschirmbreiten in einen Button zusammenzufallen, der bei Betätigung eine vertikale Ansicht der Navigationselemente öffnet. Neben dieser Navigationsleiste befindet sich auf jeder Unterseite der gleiche Footer mit den gleichen Elementen, welcher jedoch nur einen vollständigen Effekt der Seiten erzeugen soll, was man vor allem unter „Rechtliches“ erkennen kann, da die Links nirgendwo hinführen. Weitere Inhalte auf den Seiten werden größtenteils entweder durch einfache Textblöcke (meist am Anfang der Seite) oder durch cards (wie bio-card oder project-cards), in welchen dann Text ausformuliert ist, dargestellt. Neben einfachen Texten verwenden die Seiten natürlich auch noch andere Medienformate, vorwiegend Bilder. Diese liegen als .webp Dateien vor, tragen also zumindest vom Dateiformat her erst einmal Positives zur Performance bei. Erstmals hatte ich für die „Skizzierung“ der Seite nur einfache .jpeg und .png Dateien verwendet, als ich jedoch die Performance getestet habe, wurde mir schnell klar, dass ich auf ein für Webanwendungen optimiertes Format umsteigen sollte. Was die Performance aber auch stark beeinflusst hat, waren die Größen der Bilder, auf die ich erstmals nicht geachtet habe, da manche tatsächlich in 4k Formaten vorlagen, was neben den Performance-Einbußen die Bilder selbst in den viel kleineren Darstellungen auf der Seite auch unnatürlich aussehen lassen hat. Ich habe die Bilder anschließend mit dem Werkzeug Squoosh auf eine gemeinsame Auflösung von 720 x 480px gebracht. Dies hätte natürlich auch noch weiter optimiert werden können, aber ich wollte auch gewährleisten, dass die Bilder auch in anderen Kontexten außerhalb der Webseite selbst zumindest in erkennbarer Auflösung angezeigt werden. Zur Ladezeitoptimierung wurden außerdem bei Bildern, die nicht direkt beim initialen Laden der Seite angezeigt werden müssen, „Lazy Loading“ implementiert, bedeutet, dass betroffene Medien erst geladen werden, wenn sie auch im Viewport zu sehen sind. Zur weiteren Performanceverbesserung wurden importierte .css und .js Dateien verkleinert, damit der Browser somit auch weniger laden muss, abgesehen von den Bootstrap Bibliotheken. Diese sorgen nun am Ende aller Optimierungen natürlich noch einmal für die größten Einbußen, woran jedoch auch nicht zu viel geändert werden kann.

Zusammengefasst erreichen alle Unterseiten bei Lighthouse eine Performance von 100 beim Desktop-Test und eine Performance von 98-99 beim Mobile-Test, was denke ich hier vor allem dem Import von Bootstrap geschuldet ist. Da Lighthouse des Weiteren auch Anforderungen der Accessibility, Best Practices und SEO mit den jeweiligen Seiten abgleicht, würde ich mich anhand dessen hier einmal weiterbewegen. Alle Unterseiten erreichen einen Accessibility-Score von 100, wohin jedoch auch erst einmal gearbeitet werden musste. Hierbei geht es vor allem um die Bezeichnung der Header in den .html-Dateien, da ich zuerst für die Überschriften immer nur die Header genommen hatte, die von der Größe her visuell am passendsten waren, ohne auf eventuelle Screen-Reader-Komplikationen zu achten. Dies habe ich entweder gelöst, indem ich die Header direkt angepasst habe oder diesen ein aria-level-Attribut hinzugefügt habe, damit sich am Stil der Überschrift nichts ändert, aber der Screen-Reader diese, wie gewünscht, interpretiert. Ein weiteres Problem gab es in der Mobile-Ansicht bei dem Button für die Navigation, welcher ebenfalls erstmals nicht vom Screen-Reader interpretiert werden konnte, da er kein Label hatte, welches ich nach dieser Erkenntnis natürlich hinzugefügt habe. Auch musste ich eventuelle Links

auf Unterseiten oder andere Webseiten farblich noch einmal anpassen, damit sich diese auch für visuell eingeschränkte Personen ausreichend vom Hintergrund abheben. Anderen Probleme, die vielleicht hätten entstehen können, konnten durch den weitgehend konsequenten Einsatz von Bootstrap, einem sehr optimiertes Framework in diesem Kontext, vorgebeugt werden. Gleiches gilt auch für die Best Practices (Score: 100), bei welchem entweder das Framework oder die umfassende Vorbereitung auf dieses Projekt dazu geführt hat, dass hier kaum noch Luft nach oben geblieben ist. Da SEO (Search Engine Optimization) auch ein integraler Part dieses Projekts hier ist, wird durch die eingesetzten Methoden auch ein Score von 100 erreicht, hier vor allem die Meta-Tags, wie name, description usw., Open-Graph-Einbindungen, wie og:url, og:image usw. und strukturierte Daten durch Schema.org., um so den Suchmaschinen zusätzliche Informationen der einzelnen Unterseiten bereitzustellen.

Das nun erst einmal allgemein zu der gesamten Portfolio-Webseite. Ich würde nun etwas genauer auf die einzelnen Seiten eingehen, da manche Techniken, die ich verwendet habe, einzigartig für jeweilige Unterseiten und einer näheren Betrachtung würdig sind. Beginnen wir also bei der Startseite, die erst einmal einen groben Überblick über meine Person geben soll. Ganz oben auf der Seite habe ich mein Profilbild mit Namen und derzeitige Tätigkeit platziert. Hier ist technisch zu erwähnen, dass die Bootstrap-Klasse profile-details verwendet wird, sowie eine Flexbox, um dieses Element responsive zu halten. Bei geringerer Bildschirmbreite fällt der Text unter das Bild, was durch eine media-query implementiert ist. Der Rest der Seite deckt sich mit dem, was schon weiter oben beschrieben wurde: Wir haben eine formatierte bio-card, in welcher jegliche weiteren Informationen enthalten sind. Interessant ist hier noch das Aufteilen von „Kernkompetenzen“ und „Kontakt“ in eine Zeile (row), um diese Elemente im Desktop-Modus nebeneinander darzustellen und in einer mobilen Ansicht untereinander.

Auf den darauf folgenden Seiten erwartet den Nutzer mein Lebenslauf, welcher durch den „Bildungsweg“ und den „Fähigkeiten & Hobbys“ auf verschiedenen Art und Weisen dargestellt wird, ohne aber zu aufdringlich zu sein (meines Erachtens nach). Beginnen wir bei der Unterseite Bildungsweg.

Nach der Überschrift springt dem Nutzer hier eine kurze Übersicht in die Augen, wo meine schulische Laufbahn zeitlich sortiert vorliegt. Wie im Code zu sehen, sollte das erst wie eine Art Roadmap dargestellt werden, woran ich einige Stunden beschäftigt war, bis ich zu dem Entschluss gekommen bin, diesbezüglich vielleicht doch eine einfachere Darstellung zu verwenden. Von dieser Roadmap sind jetzt mehr oder weniger nur noch die zwei grauen Striche da, welche den Text links und rechts visuell begrenzen. Um dies auszugleichen, habe ich zumindest noch von Bootstrap importierte nummerierte Icons vor dem jeweiligen Abschnitt gesetzt. Um darzustellen, dass ich mich gerade noch in der Hochschule befinde, habe ich mit einer anderen Schriftart den Schriftzug „aktuell:“ davor geschrieben. Im weiteren Verlauf der Seite habe ich meine „Abschlussnoten und Erfahrungen“ innerhalb einer bio-card visuell ansprechend durch Bootstrap-Cards implementiert, wobei zu bestimmten Bildungsabschnitten gehörige Erfahrungen responsive immer unter dem jeweiligen Abschnitt stehen bleiben. Um diese Unterseite abzurunden, habe ich mit dem carousel-Element von Bootstrap Bilder der jeweiligen Bildungseinrichtung abgelegt, durch welche durchgescrollt werden kann. Diese Bilder, sind auch wie die noch folgenden Bilder auf der Webseite mit dem Attribut <figure> implementiert, da ich so mit <figcaption> die Quellen der Bilder einfach und anschaulich verlinken konnte. Wenn man nun auf die Unterseite „Fähigkeiten & Hobbys“ navigiert, sollte direkt eine Sache auffallen, und das ist die erste selbst implementierte Animation auf der Webseite. Ich habe mich dazu entschieden, den Interessengrad meiner Fähigkeiten mit einem Fortschrittsbalken darzustellen, was erstmals ganz einfach mit Hilfe von Bootstrap möglich war (progress, progress-bar), jedoch war mir dies etwas zu statisch. Deshalb habe ich ein Skript geschrieben (skills.js), mit welchem sich die Balken nach dem Laden der Seite von 0% bis zu einem gewünschten Wert (data-target) innerhalb von 2 Sekunden „aufladen“. Dazu zählt neben den Balken selbst auch die enthaltene Prozentzahl. Wenn man die .js Datei genauer betrachtet, fällt außerdem auf, dass auch eine weitere Sache animiert worden ist, und zwar die hobby-cards im unteren Teil der Seite, welche, nachdem sie durch Herunterscrollen theoretisch gesehen werden können, mit einem Fade-In Effekt von unten eingeblendet werden. Die Bilder in den hobby-cards sind im Übrigen durch das Attribut „card-img-top rounded-top“ so implementiert,

dass sie sich der Karte anpassen und oben abgerundet sind, währenddessen die Ecken unten gleich bleiben, um so den Effekt zu erzeugen, dass Bild und Text in der Karte verschwimmen. Hier lohnt es sich vielleicht auch nochmal zu erwähnen, dass ich für die Abstände der einzelnen Elemente zumeist die Bootstrap-Utility-Klassen „mt“ (margin-top) oder „pt“ (padding-top) bzw. ähnliche Attribute verwendet habe, damit Abstände einheitlich und natürlich aussehen. Nach der „Fähigkeiten & Hobbys“-Seite habe ich eine Seite in Angriff genommen, die abgeschlossene Studienprojekte beschreibt. Diese befindet sich direkt in Tandem mit der Unterseite „Referenzen“, aber dazu gleich mehr. Für die Seite selbst gibt es gar nicht so viel neue verwendete Methoden zu untersuchen. Als einzig neues Element tritt hier ein aus Bootstrap importierter Button auf, der auf das jeweilige Projekt unter den Referenzen verweist. Wichtig ist hierbei zu erkennen, dass der Button nicht nur auf die „Referenzen“-Unterseite führt, sondern auch auf die jeweilige ID des gewünschten Projekts. Drückt man nun beispielsweise auf den Button „Mehr erfahren“ unter dem Projekt „SQL-Datenbank“ wird man auf die „Referenzen“-Unterseite geschickt und zusätzlich leuchtet der Link kurz gelb auf, welcher auf das Git-Repository vom gewünschten Projekt führt, damit der Nutzer nicht noch unnötig danach suchen muss. Dies ist auch durch ein eigen geschriebenes Skript und CSS-Animation realisiert, indem der Hash der URL ausgelesen, das gewünschte Element gefunden wird und dieses dann das css-Attribut zugeschrieben bekommt, welches für eine 4-sekündige Highlight-Animation verantwortlich ist. Die beiden letzten Unterseiten „Übungsaufgaben“ und „Reflexion“ haben im Rahmen dieser Prüfungsleistung natürlich auch noch Relevanz, jedoch werden hier keine neuen Techniken mehr verwendet, die nicht schon erklärt wurden. Ich würde jedoch gerne noch einmal kurz die Implementierung der Übungsaufgaben ansprechen. Jede Übungsserie liegt geordnet in einzelnen bio-cards vor, in welchen man dann mit Buttons auf die gewünschte index.html der jeweiligen Unteraufgabe geschickt werden kann. Es ist auch möglich, sich die .pdf-Datei mit Aufgabenstellung anzuschauen. Alle mit Quellcode zu lösenden Übungsaufgaben liegen auch noch einmal zur genaueren Betrachtung im Projektordner vor.

Am Ende kommt man nun also auf dieser Seite hier an und wird von einem Textblock erschlagen. Zur abschließenden Reflexion kann ich sagen, dass ich mit diesem Projekt nicht nur das gelernte Wissen aus dem Fach „Multimediale Webprogrammierung“ anwenden, sondern auch Vertiefungsarbeit vor allem im Rahmen des Bootstrap-Frameworks leisten konnte. Hiermit konnte ich mir nicht nur die Frage beantworten, warum ich etwas in der Frontend-Entwicklung umsetzen sollte, sondern auch wie ich das tun kann. Hierbei waren vor allem tiefere Einblicke nicht nur in die Gestaltung, aber auch in der Optimierung (Performance, Accessibility etc.) von großer Bedeutung, da ich darauf in der Vergangenheit nur im theoretischen Sinne Wert gelegt, aber dies nie wirklich in einer realen Webanwendung priorisiert habe. Ich hoffe, ich konnte hiermit den Entstehungsprozess dieser Seite sowie die Lernerfolge im Modul umfassend visualisieren und nicht nur schon Gelerntes wiedergeben, sondern auch zeigen können, wie ich es angewendet und vertieft habe.