

REACT-PORTFOLIO

Dokumentation Individuelles Abschlussprojekt

EXPOSEE

Dies ist die Dokumentation für mein individuelles Abschlussprojekt am ZLI 2023. Diese Dokumentation beschreibt meinen Ansatz und zeigt mein Produkt zur Erstellung eines persönlichen Portfolios mit React.

Mustafa Sagaaro

29. Juni 2023

Inhaltsverzeichnis

1. Einleitung.....	2
1.1 Meine Projektidee	2
1.2 Konzeption.....	4
2. Inhalt.....	5
2.1 Informieren.....	6
2.2 Planen.....	8
2.3 Entscheiden	9
2.3.1 Warum Gsap und Three js?	9
3. Realisieren	10
3.1 Animationen	10
3.1 AOS	11
3.1 3D Modelle	11
Kontrollieren.....	13
Auswertung	14

1. Einleitung

Diese Dokumentation beschreibt den Prozess der Erstellung eines individuellen Projekts im Rahmen des Abschlussprojekts beim Zürcher Lehrbetriebsverband ICT (ZLI) im Jahr 2023. Das Ziel des Projekts war es, klare Ziele und Anforderungen festzulegen und diese innerhalb der vorgegebenen Zeit zu erreichen und umzusetzen. Zusätzlich sollte das Erlernte aus dem vergangenen Jahr präsentiert werden.

Das Projekt begann damit, eine Projektidee zu entwickeln und einen Projektantrag auszufüllen, in dem die Idee, die Projektziele und die Ausgangslage erläutert werden.

Nachdem der Projektantrag vom Coach geprüft und bestätigt wurde, war der nächste Schritt die Erstellung eines Konzepts und der eigentliche Projektstart.

1.1 Meine Projektidee:

Mein Ziel ist es, ein professionelles und funktionsfähiges Portfolio mit React zu entwickeln. Dabei möchte ich meine Projekte, Fähigkeiten und Erfahrungen auf eine übersichtliche und ansprechende Weise präsentieren.

Mein Portfolio wird eine benutzerfreundliche Benutzeroberfläche bieten, die eine reibungslose Navigation und Interaktion ermöglicht.

Zusätzlich plane ich, Social-Media-Funktionen zu integrieren, um eine einfache Weitergabe und Vernetzung zu ermöglichen. Durch den Einsatz von Animationen und visuellen Effekten möchte ich meinem Portfolio eine dynamische und ansprechende Atmosphäre verleihen.

Projekttitel	
React Portfolio	
Projektleiter (Name, E-Mail)	Coach (Name, E-Mail)
Mustafa Sagaaro sagaaromustafa@gmail.com	Fabian Dennler
Projektstart	Projektende
07.06.2023	30.06.2023
Idee / Kurzbeschreibung (Um was geht es bei diesem Projekt?)	
Ich plane, ein persönliches Portfolio mit React zu erstellen, das sowohl funktional als auch nutzbar ist. Darüber hinaus möchte ich auch Erfahrungen mit React sammeln.	
Ausgangslage (Was ist los? Warum wird der Projektauftrag gestellt?)	
Am Ende des Projekts strebe ich ein funktionsfähiges und optisch ansprechendes Portfolio an, das auch meine Kenntnisse in der Informatik demonstriert. Ich möchte dieses Projekt als Gelegenheit nutzen, wertvolle Erfahrungen mit React zu sammeln.	

Projektziele (Erwartete Resultate. Forderungen und Wünsche)	Prior.¹
Entwickeln eines ansprechenden und modernes Frontends, das die Aufmerksamkeit der Besucher auf sich zieht.	A
Schaffen einer benutzerfreundlichen Benutzeroberfläche, die eine reibungslose Navigation und Interaktion ermöglicht.	A
Präsentation von Projekten, Fähigkeiten und Erfahrungen auf eine übersichtliche und professionelle Art und Weise.	A
Integration von Social-Media-Funktionen, um eine einfache Weitergabe und Vernetzung zu ermöglichen.	B
Integration von Animationen und visuellen Effekten, um dem Portfolio eine dynamische und ansprechende Atmosphäre zu verleihen.	B
Einfache Regelmäßige Aktualisierung und Erweiterung des Portfolios, um neue Projekte und Erfahrungen hinzuzufügen machen können.	C
Kontakt-Möglichkeiten gewährleisten.	C
Vorgehensziele (Forderungen und Rahmenbedingungen)	Prior.¹
Ich möchte meine Arbeit strukturiert und kontrolliert angehen, indem ich mir klare Meilensteine setze und diese systematisch abarbeite.	A
Ich möchte regelmäßig meine Fortschritte beim Coden in ein Git-Repository commiten, um meine Arbeit zu sichern und den Überblick über meine Entwicklung zu behalten.	A
Die Dokumentation sollte regelmäßig aktualisiert und nicht vernachlässigt werden.	B
Lernziele (Welche Kompetenzen werden angestrebt?)	Prior.¹
Der Code sollte äußerst übersichtlich sein, und ich setze mir selbst das Ziel, in der Lage zu sein, jede einzelne Zeile erklären zu können.	A
Am Ende des Projekts möchte ich in der Lage sein, React zu verstehen und erfolgreich anzuwenden.	A
Ressourcen (Personal, Sachmittel, Material?)	Menge
Ich verwende eine Vielzahl von Ressourcen wie YouTube-Videos, Google(w3schools, codeacademy ...) und sogar ChatGPT, um ein Verständnis für React zu entwickeln und meine Kenntnisse zu erweitern. Allerdings nutze ich diese Quellen ausschließlich zum Lernen und Verstehen, nicht zum Abschreiben von kompletten Code-Blöcken oder zum Stehlen von Arbeit.	
Ich verwende Visual Studio Code als meinen Editor und GitHub, um meinen Fortschritt zu speichern.	
Projektentscheidung (Freigabeunterschrift)	Datum

1.2 Konzeption

Use Cases:

1. Das Portfolio ist übersichtlich und ist Benutzerfreundlich.
2. Ich kann problemlos Änderungen wie neue Projekte zum Portfolio hinzufügen.
3. Sollte gut genug Zeit, um sich damit zu bewerben.

Risiken:

1. Schwierigkeiten beim Erlernen und Anwenden von React.
2. Zu langsame Fortschritte wegen zu wenig Kenntnisse mit React.
3. Komplexität der Integration von Animationen und visuellen Effekten

Verwendete Technologien und Tools:

React: Verwendung von React als Hauptframework für die Entwicklung des Portfolios.

HTML/CSS: Erstellung der UI-Komponenten und des Layouts.

JavaScript: Programmierung der Logik und Interaktionen.

Three.js und Parallax: Für die Animationen und visuellen Effekte der Seite

Git und GitHub: Versionskontrolle und Speicherung des Codes.

Visual Studio Code: Verwenden Sie Ihren bevorzugten Code-Editor für die Entwicklung.

Neues Lernen:

React: Erlernen der Grundlagen von React.

Animationen und visuelle Effekte in React: Erforschung von Bibliotheken oder Techniken zur Integration von Animationen und visuellen Effekten. Diese auch erlernen.

Arbeitspakete (Beispiel):

Projektplanung und Konzepterstellung (2 Stunden)

Erstellung der Projektkomponente und der Detailansicht (2 Stunden)

Lernen, recherchieren und Anwenden von React (16 Stunden (2 Werktagen))

Styling und Anpassung des Designs (8 Stunden)

Integration von Animationen und visuellen Effekten (16 Stunden)

Testen und Fehlerbehebung (4 Stunden)

Dokumentation und Projektabschluss (8h)

Total = ca. 8 Werktagen

2. Inhalt

Im Laufe des Projekts habe ich strukturiert und geplant mit der IPERKA Projektmethode gearbeitet, um einen klaren Kopf zu behalten und effizient vorzugehen.

Dies ist der Ablauf:

Informieren: Die nötigen Informationen suchen und verstehen

Planen: Vorgehen planen und vorbereiten

Entscheiden: für Technologien entscheiden und Wireframe der Seite erstellen.

Realisieren: Umsetzen der Ziele

Kontrollieren: Testprotokoll erstellen und durchführen.

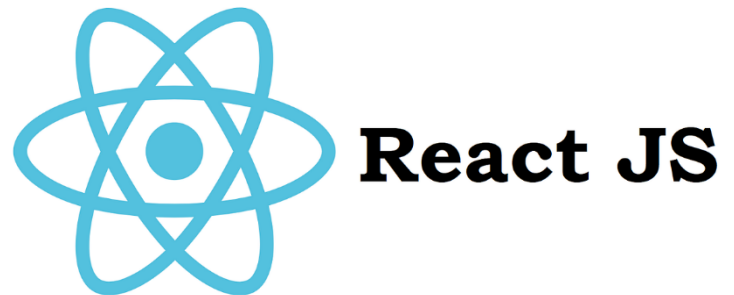
Auswerten: Fazit zum Projekt, Produkt finalisieren

2.1 Informieren

Nach eingehender Recherche über geeignete Programmiersprachen für ein Portfolio bin ich zu dem Schluss gekommen, dass React sich am besten dafür eignet. Darüber hinaus habe ich intensiv untersucht, wie Animationen auf Webseiten am besten umgesetzt werden können und bin auf Gsap gestoßen. Bei meiner Suche nach einer Lösung zur Integration von 3D-Modellen bin ich auf Three.js gestoßen.

React js:

React.js ist eine JavaScript-Bibliothek zur Entwicklung von Benutzeroberflächen. Sie ermöglicht die Erstellung von interaktiven und wiederverwendbaren UI-Komponenten. React.js basiert auf dem Konzept der Komponenten, die den Zustand und das Aussehen der Benutzeroberfläche verwalten. Durch die Verwendung von virtuellem DOM und einer effizienten Aktualisierungsstrategie ermöglicht React.js eine schnelle und reaktionsschnelle Benutzeroberfläche. Die Bibliothek wird von Facebook entwickelt und ist aufgrund ihrer Flexibilität und Leistung bei der Entwicklung moderner Webanwendungen weit verbreitet. React.js bietet auch eine aktive Entwicklergemeinschaft und eine umfangreiche Sammlung von Erweiterungen und Tools, die die Entwicklung erleichtern.



Gsap:



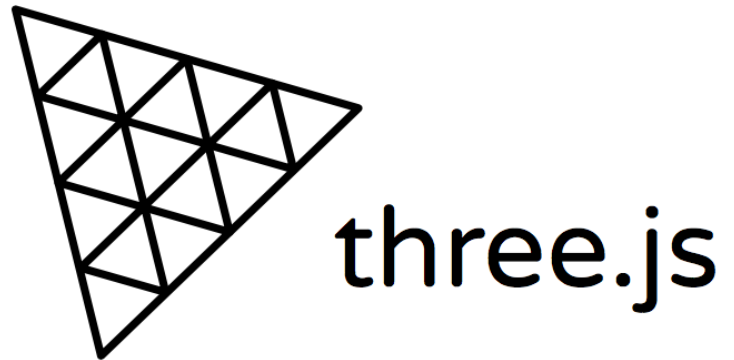
GSAP (GreenSock Animation Platform) ist eine leistungsstarke JavaScript-Bibliothek für Animationen. Sie ermöglicht die Erstellung von flüssigen und beeindruckenden Animationen in Webprojekten. Mit GSAP können verschiedene Eigenschaften von HTML-Elementen wie Position, Größe, Farbe und Opazität animiert werden. Die Bibliothek bietet eine einfache Syntax und umfangreiche Funktionen zur Steuerung von Animationen, einschließlich der Möglichkeit, zeitliche Verzögerungen, Wiederholungen und

Easing-Effekte festzulegen. GSAP unterstützt verschiedene Animationstypen wie Tweening, Timeline und ScrollTrigger, um komplexe und synchronisierte Animationen zu realisieren. Die Bibliothek ist in der Webentwickler-Community weit verbreitet und wird aufgrund ihrer hohen Leistung, Flexibilität und gut dokumentierten API geschätzt.

Three.js

Three.js ist eine JavaScript-Bibliothek für die Erstellung von 3D-Grafiken und interaktiven 3D-Anwendungen im Web. Sie bietet eine Vielzahl von Funktionen und Werkzeugen, um komplexe 3D-Szenen, Modelle und Animationen zu erstellen. Mit Three.js können Entwickler 3D-Objekte erstellen, Materialien und Texturen zuweisen, Beleuchtung und Schattierung einrichten sowie Kameraeinstellungen und Blickwinkel konfigurieren. Die Bibliothek

unterstützt verschiedene Rendering-Techniken wie WebGL, CSS3D und SVG, um die 3D-Darstellung in den Webbrowsern zu ermöglichen. Mit Three.js können Benutzer interaktive Steuerungen, Bewegungen und Effekte in ihren 3D-Anwendungen implementieren. Sie bietet auch Integrationen mit anderen Bibliotheken wie GSAP für Animationen und Physijs für Physiksimulationen. Three.js wird von einer aktiven Entwicklergemeinschaft unterstützt und bietet eine umfangreiche Dokumentation sowie zahlreiche Beispiele und Ressourcen, um die Erstellung von 3D-Anwendungen zu erleichtern.

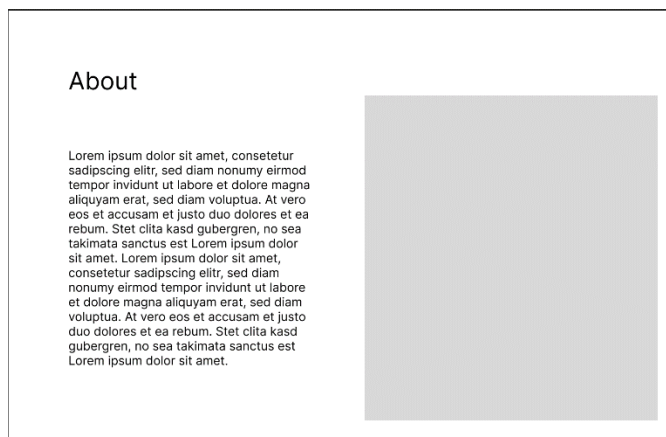
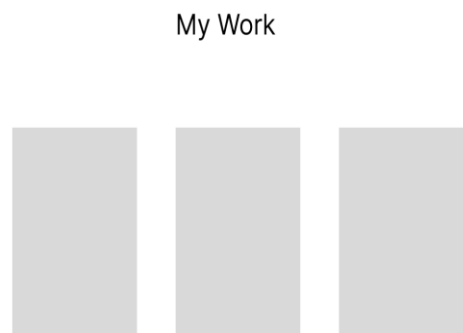


Ausserdem habe ich mir

2.2 Planen

Für die Planung meines Projekts habe ich zunächst ein Wireframe mit Figma erstellt. Wireframes sind eine äußerst nützliche Methode, um Ideen visuell darzustellen, noch bevor sie in Code umgesetzt werden. Sie ermöglichen es, einen groben Überblick über die Struktur und das Layout einer Webseite oder Anwendung zu erhalten. Mit Hilfe des Wireframes konnte ich meine Vorstellungen konkretisieren und die Benutzeroberfläche im Voraus planen, bevor ich mit der eigentlichen Umsetzung begann. Dieser Ansatz des visuellen Planens ermöglichte es mir, eine solide Grundlage zu schaffen und während des Entwicklungsprozesses effizienter vorzugehen.

2.2.1 Meine Wireframes



Contact

Name

Mail

Message

2.3 Entscheiden

Warum habe ich mich für dieses Projekt entschieden?

Ich habe mich bewusst für dieses Projekt entschieden, da ich bereits viel über React gehört hatte und dadurch ein großes Interesse daran entwickelt habe. Während meiner Suche nach einer geeigneten Projektidee stieß ich auf beeindruckende React-Portfolios, die mich regelrecht zum Staunen brachten. Insbesondere die animierten Elemente auf diesen Webseiten fesselten mich und weckten den Wunsch, selbst solche eindrucksvollen Webseiten erstellen zu können.

Ein weiterer ausschlaggebender Grund für meine Entscheidung war der Wunsch, mich weiterzuentwickeln und mein Wissen zu erweitern. Ich war motiviert, mich neuen Herausforderungen zu stellen und meine Fähigkeiten im Bereich der Webentwicklung auszubauen. Das Projekt mit React schien mir die perfekte Gelegenheit zu bieten, um neues zu lernen und mir selbst Herausforderungen zusetzen.

2.3.1 Warum Gsap und Three.js?

Ich habe mich dafür entschieden, GSAP und Three.js in diesem Projekt zu verwenden, aus mehreren Gründen.

Zunächst einmal habe ich bei meiner Recherche nach den besten Möglichkeiten, Animationen auf einer Webseite umzusetzen, auf GSAP gestoßen. GSAP (GreenSock Animation Platform) ist eine leistungsstarke JavaScript-Bibliothek, die eine Vielzahl von Animationseffekten und -funktionen bietet. Ich war beeindruckt von den zahlreichen Beispielen und Tutorials, die ich gefunden habe, und war überzeugt, dass GSAP die richtige Wahl ist, um meine gewünschten Animationen auf meiner Webseite umzusetzen.

Für die Integration von 3D-Modellen stieß ich auf Three.js. Three.js ist eine JavaScript-Bibliothek, die es ermöglicht, beeindruckende 3D-Grafiken und -Animationen in Webprojekte einzubinden. Ich war fasziniert von den Möglichkeiten, die Three.js bietet, um immersive und interaktive 3D-Erfahrungen zu schaffen. Die Vielseitigkeit und das breite Spektrum an Funktionen, das Three.js bietet, überzeugten mich davon, dass es die perfekte Lösung für meine Anforderungen ist, 3D-Modelle in meine Webseite einzufügen.

Darüber hinaus war ich auch daran interessiert, neue Technologien und Frameworks kennenzulernen, um mein Wissen und meine Fähigkeiten in der Webentwicklung zu erweitern. GSAP und Three.js sind beliebte und etablierte Tools in der Webentwicklung, und ihre Verwendung in diesem Projekt würde es mir ermöglichen, mich mit neuen Konzepten und Technologien vertraut zu machen. Ich war motiviert, meine Kenntnisse in der Animation und der 3D-Grafikprogrammierung zu vertiefen und neue Möglichkeiten der Webentwicklung zu erkunden.

3. Realisieren

Beim Realisieren des Projekts bin ich systematisch vorgegangen, um meine Ziele effizient zu erreichen. Hier ist eine detaillierte Beschreibung meiner Vorgehensweise:

Zunächst habe ich ein GitHub-Repository erstellt, um den gesamten Code zu versionieren und Änderungen nachverfolgen zu können. Dies ermöglichte es mir, meine Fortschritte sicher zu speichern und bei Bedarf auf frühere Versionen zurückzugreifen.

Als nächstes habe ich das Gelernte aus dem Schritt "Informieren" angewandt. Basierend auf meinen Recherchen habe ich React als Hauptframework für die Entwicklung meines Portfolios ausgewählt. React bietet eine effiziente und modulare Art der Entwicklung von Benutzeroberflächen. Ich habe die notwendigen Abhängigkeiten installiert und begonnen, meine Komponenten zu erstellen.

3.1 Animationen

In Bezug auf die Animationen und visuellen Effekte habe ich GSAP (GreenSock Animation Platform) verwendet. GSAP ist eine leistungsstarke JavaScript-Bibliothek, die umfangreiche Funktionen zur Erstellung von flüssigen Animationen bietet. Ich habe GSAP in mein Projekt integriert und begonnen, Animationen für bestimmte Elemente meiner Webseite zu erstellen. Dabei habe ich die verschiedenen Funktionen von GSAP genutzt, um beispielsweise die Opazität, Position oder Skalierung von Elementen zu animieren.

```
tl.fromTo(hero, 0.85, { height: "0%" }, { height: "70%", ease: Power2.easeInOut })  
.fromTo(hero, 1.2, { width: "100%" }, { width: "70%", ease: Power2.easeInOut })  
.fromTo(slides, 1.2, { x: "-100%" }, { x: "0%", ease: Power2.easeInOut }, "-=1.2")  
.fromTo(logo, 0.5, { opacity: 0, x: 30 }, { opacity: 1, x: 0 }, "-=0.5")  
.fromTo(introduction, 0.5, { opacity: 0, x: 30 }, { opacity: 1, x: 0 }, "-=0.5");
```

Das Element hero wird von einer Höhe von "0%" auf eine Höhe von "70%" animiert und dauert 0,85 Sekunden.

Das Element hero wird von einer Breite von "100%" auf eine Breite von "70%" animiert und dauert 1,2 Sekunden.

Das Element slides wird von einer horizontalen Position von "-100%" auf eine Position von "0%" animiert und dauert 1,2 Sekunden. Die Animation startet gleichzeitig mit der vorherigen Animation.

Das Element logo wird von einer Deckkraft von 0 und einer horizontalen Position von 30 Pixeln auf eine Deckkraft von 1 und eine horizontale Position von 0 Pixeln animiert und dauert 0,5 Sekunden. Die Animation startet verzögert um 0,5 Sekunden nach der vorherigen Animation.

Das Element introduction wird von einer Deckkraft von 0 und einer horizontalen Position von 30 Pixeln auf eine Deckkraft von 1 und eine horizontale Position von 0 Pixeln animiert und dauert 0,5 Sekunden. Die Animation startet verzögert um 0,5 Sekunden nach der vorherigen Animation.

3.1 AOS

Ausserdem habe noch AOS(Animate on Scroll) benutzt um einfache Animationen für meine Webseite zu gewährleisten:

AOS (Animate On Scroll) ist eine JavaScript-Bibliothek, die es ermöglicht, Animationen auf Elementen einer Webseite basierend auf dem Scrollverhalten des Benutzers zu aktivieren.

Hier ist ein Beispielcode:

```
<div class="box" data-aos="fade-up">  
  Dieser Inhalt wird beim Scrollen nach oben animiert.  
</div>
```

Im obigen Code wird das Element mit der Klasse "box" mit der Animation "fade-up" versehen. Wenn der Benutzer beim Scrollen das Element erreicht, wird es mit einer animierten Einblendung von unten nach oben angezeigt.

Die AOS-Bibliothek bietet verschiedene Animationsoptionen wie Fade, Slide, Zoom, Rotate und vieles mehr, die durch das Hinzufügen entsprechender Data-Attribute zu den HTML-Elementen definiert werden können. AOS erleichtert die Implementierung von Scroll-Animationen und verleiht Webseiten eine interaktive und dynamische Benutzererfahrung.

3.1 3D Modelle

Meine 3D Modelle habe ich von Sketchfab.

Für die Integration von 3D-Modellen habe ich Three.js verwendet. Three.js ist eine JavaScript-Bibliothek, die die Erstellung von 3D-Grafiken und interaktiven 3D-Anwendungen im Web ermöglicht. Ich habe Three.js in mein Projekt eingebunden und begonnen, meine 3D-Modelle zu erstellen und in meine Webseite zu integrieren. Dabei habe ich verschiedene Funktionen von Three.js genutzt, um Materialien, Beleuchtung und Kameraeinstellungen anzupassen.

Hier ein Beispiel wie ich 3d Modelle eingefügt habe:

```
<Canvas  
  id="contactcanvas"  
  style={{ width: "100vw", height: "95vh", zIndex: "1" }}  
>  
  <ambientLight />  
  <pointLight position={[100, 100, 100]} />  
  <CameraPosition />  
  <Camping />  
</Canvas>  
</div>
```

<ambientLight>: Dieses Element fügt der Szene eine Umgebungsbeleuchtung hinzu, die in alle Richtungen gleichmäßig strahlt. Dadurch werden Schatten erzeugt und die Objekte in der Szene sichtbar gemacht.

`<pointLight position=[100, 100, 100]>`: Dieses Element fügt der Szene eine Punktlichtquelle hinzu. Die `position`-Eigenschaft definiert die Position des Lichts in der Szene. In diesem Fall wird das Licht an der Position (100, 100, 100) platziert.

`<CameraPosition />`: Dieses Element ist eine eigene Komponente, die die Kameraeinstellungen definiert. Es positioniert die Kamera in der Szene, um den gewünschten Blickwinkel zu erhalten.

`<Camping />`: Dieses Element ist eine eigene Komponente, die die 3D-Grafiken und Modelle für die Szene enthält. Es wird durch Three.js gerendert und zeigt die gewünschten 3D-Elemente an.

Um den Fortschritt meiner Animationen und Effekte zu überprüfen, habe ich den Animationsverlauf in meiner Entwicklungsumgebung visualisiert. Dadurch konnte ich sicherstellen, dass die Animationen flüssig ablaufen und die gewünschten Effekte erzielt werden.

Email js

Damit mein Kontaktformular auch wie gewünscht Emails an mich versendet, habe ich Email js genutzt.

EmailJS ist ein Dienst, der die Integration von E-Mail-Funktionen in Webanwendungen vereinfacht.

Es ermöglicht Entwicklern, E-Mails über SMTP direkt von ihrem Frontend aus zu versenden, ohne einen eigenen Backend-Server einrichten zu müssen.

EmailJS bietet eine benutzerfreundliche API und Vorlagen, um das Erstellen und Senden von E-Mails über eine Vielzahl von E-Mail-Anbietern zu vereinfachen.



Design

Beim Design meines Portfolios habe ich versucht, mich weitgehend an mein ursprüngliches Wireframe zu halten. Das Wireframe diente als grundlegende Struktur und Layout-Vorlage für meine Webseite. Ich habe die verschiedenen Komponenten entsprechend angeordnet und das Styling anhand meiner Designvorstellungen umgesetzt. Dabei habe ich CSS-Regeln angewendet, um das gewünschte Erscheinungsbild zu erreichen.

Tailwind css

Tailwind CSS ist ein Utility-First CSS-Framework, das darauf abzielt, das Erstellen von benutzerdefinierten Benutzeroberflächen schnell und effizient zu gestalten. Es zeichnet sich durch eine große Anzahl vordefinierter Utility-Klassen aus, die es Entwicklern ermöglichen, schnell und einfach Stile und Layouts auf ihre HTML-Elemente anzuwenden. Mit Tailwind können Entwickler die Gestaltung ihrer Webseiten detailliert anpassen, indem sie Klassen wie "bg-blue-500" für die Hintergrundfarbe oder "p-4" für den Padding-Wert verwenden.

Das Framework fördert eine pragmatische Herangehensweise an das Styling von Websites und bietet Flexibilität und Skalierbarkeit, ohne die Notwendigkeit von eigenen CSS-Regeln.

Abschluss Realisieren

Durch meine systematische Vorgehensweise und die Anwendung des Gelernten konnte ich mein Projekt erfolgreich umsetzen und ein professionelles Portfolio mit React, GSAP und Three.js entwickeln. Mein Portfolio bietet nun eine benutzerfreundliche Benutzeroberfläche, beeindruckende Animationen und 3D-Modelle, die meine Projekte, Fähigkeiten und Erfahrungen auf eine ansprechende Weise präsentieren.

Kontrollieren

Nach dem Realisierungsprozesses habe ich meine Webseite auf verschiedenen Geräten und Browsern getestet, um sicherzustellen, dass sie responsive ist und ein konsistentes Nutzererlebnis bietet. Ich habe auch die Ladezeiten optimiert, um eine schnelle und reaktionsschnelle Webseite zu gewährleisten.

Ich habe folgendes Testprotokoll genutzt, um meine mein Projekt zu kontrollieren:

Testprotokoll:

Testdatum: 30.06.2023

Name des Testers: Mustafa Sagaaro

Test	AR	Aktion	Was wird geprüft	<u>Passed/Failed</u>
1	1	Webseite beachten	Benutzerfreundlich	<u>passed</u>
2	2	Webseite beachten	Schöne Animationen	<u>passed</u>
3	3	Kontakt Formular ausfüllen	Mail wird an mich geschickt mit den Angaben	<u>passed</u>
4	3	Webseite beachten	3d Modelle eingefügt	<u>passed</u>
5	4	Webseite beachten	Kontakt Möglichkeit vorhanden und funktionstüchtig	<u>passed</u>

Schlussfolgerung: Alles Gut und ist fehlerfrei

Was mir nicht so gut gefällt ist, dass die Webseite noch nicht so responsive ist wie gewünscht.

4. Auswertung

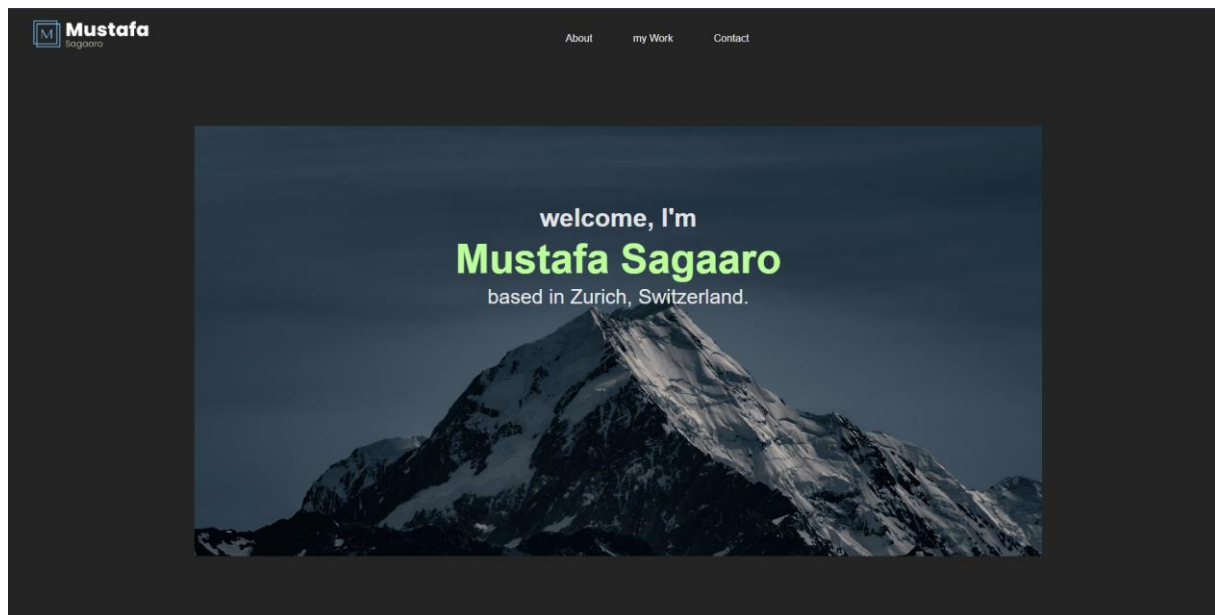
4.1 Fazit

Im Großen und Ganzen bin ich mit meinem Projekt zufrieden. Das Endergebnis gefällt mir und ich bin mit meiner Vorgehensweise weitgehend zufrieden. Eine Herausforderung war, dass ich viel Zeit damit verbracht habe, verschiedene Ansätze auszuprobieren und einige davon wieder zu verwerfen. Meine Unsicherheit führte dazu, dass ich mir zu hohe Ziele gesetzt habe und letztendlich nicht alles umsetzen konnte, wie ich es geplant hatte. Trotzdem ist das Endergebnis sehr erfolgreich.

Für zukünftige Projekte werde ich mehr auf die Zeit achten und ein detaillierteres Wireframe erstellen, dem ich zu 100% folgen kann. Ich habe viel Zeit damit verschwendet, Fehler eigenständig zu lösen, anstatt Hilfe von Freunden oder dem Coach in Anspruch zu nehmen. Neben dem Zeitmanagement plane ich auch ausreichend Zeit für die Dokumentation ein, um sicherzustellen, dass alle Aspekte des Projekts angemessen dokumentiert sind.

4.2 Selbsteinschätzung

Auf einer Scala von 1-6 würde ich mir selber die Note 5.25 geben.



Quellen

3d Modelle:

Hologram:

<https://sketchfab.com/3d-models/hologram-earth-dark-land-2b353748066649849d5def841324b229>

Camping:

<https://sketchfab.com/3d-models/camping-buscraft-ambience-7b65e4df95c3492fbf4e0641e3b472c1>

Bilder:

Unsplash:

<https://unsplash.com/>

Socials Template:

<https://codepen.io/Stockin/pen/mzydLz>

Kleine Fehlerbehebungen :

Chatgpt