

Dokumentation FÜR Chinas Wirtschaftliche Geschichte Website VON *Leonardo, Marko und Leandro*

Dieses Projekt behandelt die wirtschaftliche Geschichte Chinas seit den 1970er Jahren. Wir haben eine Website erschaffen, die jene historischen Ereignisse zusammenfasst und auch ein Quiz enthält.



Abbildung 1: Grosser Sprung nach Vorn
(Quelle: <https://www.johndclare.net/China5.htm>)

Inhalt

Ziele und Planung	2
Spezifikation der Anforderungen.....	2
Zeitplanung.....	3
Analyse und Design	4
GUI.....	4
UML	5
Anwendungsfalldiagramm.....	5
Softwaredokumentation.....	6
Testfälle.....	6
Quiz (Leandro)	7
Parallaxe (Marko).....	8
Speichern des Hell- und Dunkelmodus (Leonardo).....	9
Reflexion	11
Rückblick // Marko	11
Rückblick // Leandro.....	11
Rückblick // Leonardo.....	12
Leistungsziele	13
Abweichung zum Plan	13
Teamarbeit.....	14

Ziele und Planung

Spezifikation der Anforderungen

Wir haben uns vorgestellt, eine Website zu kreieren, die über die wirtschaftliche Geschichte Chinas erzählt, spezifischer, über seinen Weg zur Wirtschaftsmacht.

Erstens mussten wir den Umfang unseres Themas, und somit den Umfang unserer Recherche bestimmen.

Wir legten den Zeitraum fest. Dieser sollte zirka dann anfangen, als China begann, seine Wirtschaft stark auszubauen (im Jahre 1978). Dazu sollte noch kurz über die wirtschaftliche Lage davor berichtet werden.

Dieser Zeitraum reicht bis und mit zur heutigen wirtschaftlichen Situation in China.

Somit legten wir die folgenden Unterthemen fest:

- China bevor den Reformen
- Einleitung der Reformen
- China nach den Reformen
- Heutige Situation

Zusätzlich hatten wir auch noch definiert, dass wir ein Quiz erstellen würden, welches die Informationen, die man sich auf unserer Webseite erlesen kann, testet.

Jetzt zum technischen Teil der Website. Hier haben wir uns vorgenommen, verschiedene Funktionen einzufügen. Funktionen wie die Gestaltung zwischen Dunkel- und Hellmodus und die Sprache zwischen Englisch und Deutsch zu wechseln. Aber auch viele Design und Animation verwandte Aspekte wollten wir realisieren, haben diese aber nicht klar definiert, sondern haben jene während dem Verlaufe des Projektes geschmiedet. Beispiele dazu wären der Navigationsbalken, und wie dieser schrumpft, oder das Hamburgermenu, und wie dieser beim Runterscrollen auftaucht.

Die Webseite sollte auf jeden Fall immer über eine Navigation verfügen, einen speziellen Knopf, um auf die Homepage zu gelangen haben, und in der Fusszeile Referenzen zu unserer Datenschutzerklärung, unseren Quellen und unserem Disclaimer haben.

Das war das Schema der Seite.

Zeitplanung

Zeitplan des Projekts				
Woche	1	2	3	4
1. Recherche				
Informationen über China				
Wirtschaft der China				
Geschichte der China				
Dokumentieren				
2. Planen				
Beispiele rausholen				
Interaktive Elemente Code Beispiele finden				
Mockup Webseite mit Visio				
Schritte dokumentieren				
3. Realisieren				
Mit dem HTML anfangen - Leonardo, Leandro				
Layout hinzufügen - Marko				
Script js interaktive Elemente coden				
Dokumentieren				

Abbildung 1: Unser Zeitplan

Wir haben uns einen einfachen Zeitplan erstellt.

In der ersten Woche liegt der Fokus bei der Sammlung des Inhalts, also der Recherche und dem Textschreiben. Halbwegs durch die erste Woche wird mehr und mehr Zeit für die Dokumentation unserer Fortschritte genommen, bis wir am Schluss nur noch dokumentieren.

Die zweite Woche sieht die Festlegung der Fundamente und Vorbereitung zur Realisierungsphase der Webseite. Hier würden wir unser Mockup erschaffen, Ideen aus dem Internet sammeln und uns Beispiele ansehen, anhand denen wir uns orientieren können. Hier sind die einzelnen Schritte ein bisschen verteilter, also fangen wir nicht sofort mit allem an, sondern gehen Schritt für Schritt fort. Wie in der ersten Woche, liegt am Ende der Fokus zum Grossteil beim Dokumentieren unserer Schritte.

Die dritten und vierten Wochen erleben das Realisieren unserer Webseite. Jetzt sind wir bereit und haben eine klare Vorstellung unseres Endprodukts. Wir Arbeiten gezielt zuerst an der HTML-Struktur, dann an der Gestaltung und schlussendlich an der Interaktivität und der Responsivity. Wie gehabt, ist am Ende der Woche die Dokumentation angesagt.

Analyse und Design

GUI

Dies ist das Mockup der Website (Abbildung 3 links), welches Marko entworfen hat. Man kann hier sehen, wie unsere ersten Vorstellungen aussahen. Das Endprodukt entsprach vieles vom



angezeigtem. Die Formatierung der Inhaltstexte, der Hintergrund und die Kredite sind Beispiele dieser Konsistenz.

Wie aber schon erwähnt, gab es manche Überlegungen, die nicht klar definiert wurden.

Im originalen Mockup gab es keinen Navigationsbalken, doch die Idee war, die einzelnen Referenzen auf den Drachen im Header zu platzieren.

Mit der Zeit, entschieden wir uns aber dagegen, und platzierten den Balken unterhalb des Headers.



Abbildung 4: Endgültiger Header

Andere Änderungen im Header, die wir im Endprodukt haben, im Vergleich zum Mockup, kann man hier auch sehen. Der Start Knopf ist anders, den Drachen haben wir weiss gemacht und neulich gibt es auch ein Knopf für die Einstellungen.

Der Header hat aber auch noch weitere Änderungen, wie ein neulich eingeführter Knopf für die Einstellungen, und einen weissen statt schwarzen Drachen.



Abbildung 3: Mockup

Die endgültige Fusszeile unserer Webseite ähnelt sich dem des Mockups stark im Vergleich zum Header.

Ideen, wie zum Beispiel das Video, welche eingeblendet werden soll (bei den violetten Elementen im Mockup), haben wir verworfen.



Abbildung 5: Endgültiger Fusszeile

UML

Anwendungsfalldiagramm

Hier ist zu sehen, welche Interaktionen der Benutzer mit unserer Website haben kann. Zum einen, kann er sich durch die Themen navigieren und die Informationen aus dem Text erlesen, zum anderen kann er dann sein Wissen mit unserem Quiz testen. Die Funktionen, um zwischen der Sprache und zwischen Hell- und Dunkelmodus zu wechseln, sind auch enthalten.

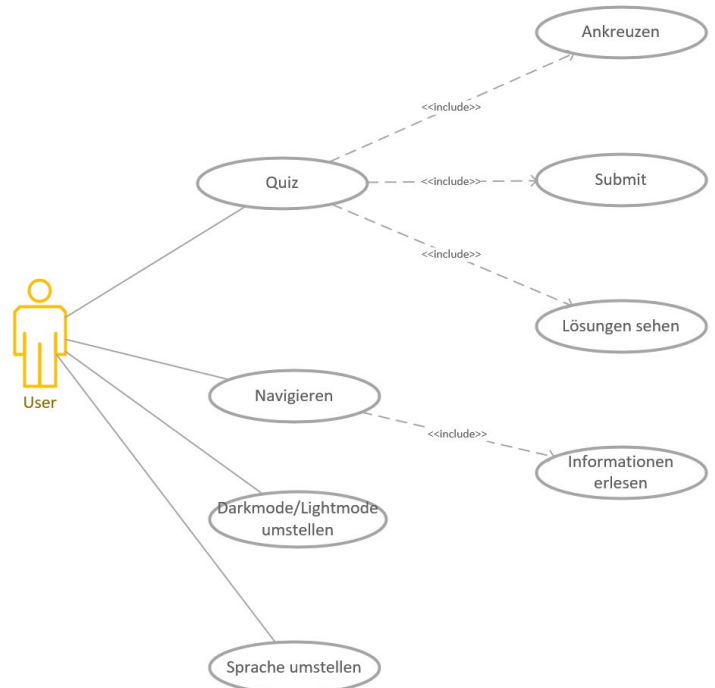


Abbildung 6: Anwendungsfalldiagramm

Softwaredokumentation

Testfälle

ID	BESCHREIBUNG	SCHRITTE	UMFELD	ERWARTETES ERGEBNIS	TATSÄCHLICHES ERGEBNIS	STATUS
01	Funktionalität der Navigation	Alle Navigationslinks anklicken	Desktop + Mobile	-Jeder Link führt zur korrekten Seite -Seite ladet korrekt	-Jeder Link führt zur korrekten Seite -Seite ladet korrekt	Funktioniert
02	Funktionalität des Quiz	-Alles beim Quiz ankreuzen -Antworten abgeben	Desktop + Mobile	-Zeigt erreichte Punktzahl und maximale Punktzahl	-Zeigt erreichte Punktzahl und maximale Punktzahl	Funktioniert
03	Hell- und Dunkelmodus	In den Einstellungen den Modus umstellen	Desktop + Mobile	-Textfarbe passt sich an -Modus wird gespeichert und auf anderen Seiten automatisch gewählt	-Textfarbe passt sich an -Modus wird gespeichert und auf anderen Seiten automatisch gewählt	Funktioniert
04	Wechseln der Sprache	In den Einstellungen die Sprache wechseln	Desktop + Mobile	-Leitet auf die Seite mit der erwünschten Sprache weiter	-Leitet auf die Seite mit der erwünschten Sprache weiter	Funktioniert
05	Responsivität für verschiedene Bildschirmgrößen	-Auf Mobile prüfen -Browser-Fenster-Grösse verändern	Desktop + Mobile	-Text, Bilder und Navigationsbalken passen sich an	-Text, Bilder und Navigationsbalken passen sich an	Funktioniert

Quiz (Leandro)

Anhand dieses kleinen Code-Ausschnitts aus dem JavaScript des Quiz werde ich kurz erklären, wie ich das Quiz codiert habe.

```
questions.forEach( callbackfn: question : Element => {  
    const correctAnswers : string[] = question.dataset.correct.split( separator: ',' );  
    totalCorrectAnswers += correctAnswers.length;  
    const correctAnswer : string = question.dataset.correct  
    const checkboxes : NodeListOf<Element> = question.querySelectorAll( selectors: ".q" );  
  
    checkboxes.forEach( callbackfn: checkbox : Element => {  
        if (checkbox.checked && correctAnswer.includes(checkbox.id)) {  
            score += 1;  
        } else if (checkbox.checked && !correctAnswer.includes(checkbox.id)) {  
            score -= 1;  
        }  
    });  
});
```

Abbildung 7: Code aus dem JavaScript des Quiz

Das Quiz ist so aufgebaut, dass sobald man durch alle Fragen gegangen ist, und sich diese angekreuzt hat, und man auf den Submit-Button klickt, dieser dann eine Funktion im JavaScript durchführt, in dessen auch dieser Code miteinbegriffen ist. Der Grund, warum ich diesen Ausschnitt zeige, ist, weil er berechnet, wie viele Punkte der Benutzer erreicht hat, und somit die dem am Benutzer auffälligste Änderung auf der Seite vornimmt.

Mit der `forEach` Funktion geht es hier alle Fragen durch. Es nimmt zuerst die korrekten Antworten aus dem Dataset (was im HTML definiert ist) von dieser Frage. Somit fügt es dann die Anzahl der korrekten Antworten, die es pro Frage gibt, dem Total hinzu, damit ich am Ende anzeigen kann, wie viele richtige Antworten es gab, und es mit der Anzahl Antworten, die der User korrekt hatte, vergleichen kann. Dann nimmt es die korrekten Antworten nochmals aber nicht um die Anzahl zu bestimmen, sondern um dann später zu bestimmen, ob der User diese Antworten ausgewählt hat. Dann selektiert es die Checkboxes. Somit geht es dann durch jede Checkbox und vergibt einen Punkt, falls eine korrekte Antwort angekreuzt wurde. Falls eine falsche Antwort angekreuzt wurde, zieht es einen Punkt ab. Am Schluss wird das Total der abgezogenen Punkte vom Total der vergebenen subtrahiert. Die Summe wird schlussendlich neben der maximal-erreichbaren Punktzahl angezeigt.

Ich muss aber zugeben, dass dieser Code nicht optimal ist, da ich zweimal die korrekten Antworten aus dem Dataset ziehe und die Namensgebung etwas ungünstig ist, da man die einzelnen Referenzen leicht verwechseln kann. Worauf ich aber geachtet habe beim Codieren ist, dass nichts «Hardcoded» war im JavaScript. Also, dass, wenn man neue Fragen hinzufügen wollte, man sie nur im HTML hinzufügen musste, und das JavaScript sich automatisch den neuen Umständen anpasste und ohne weiteres wie erwartet funktionierte.

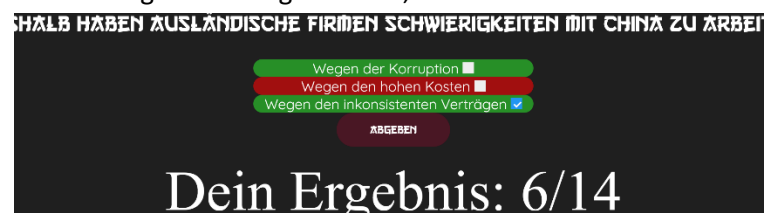


Abbildung 8: Screenshot aus dem Quiz

Parallaxe (Marko)

Auf allen Plattformen (ausser iPad und iPhone) befindet sich im Hintergrund der Webseite ein gestalterisches Element. Es ist ein wellenförmiges Muster, dass die spezielle Eigenschaft hat, eine 3-dimensionale Illusion auf der Webseite zu schaffen. In den 4 Bildern unten wird diese Funktionalität gezeigt.

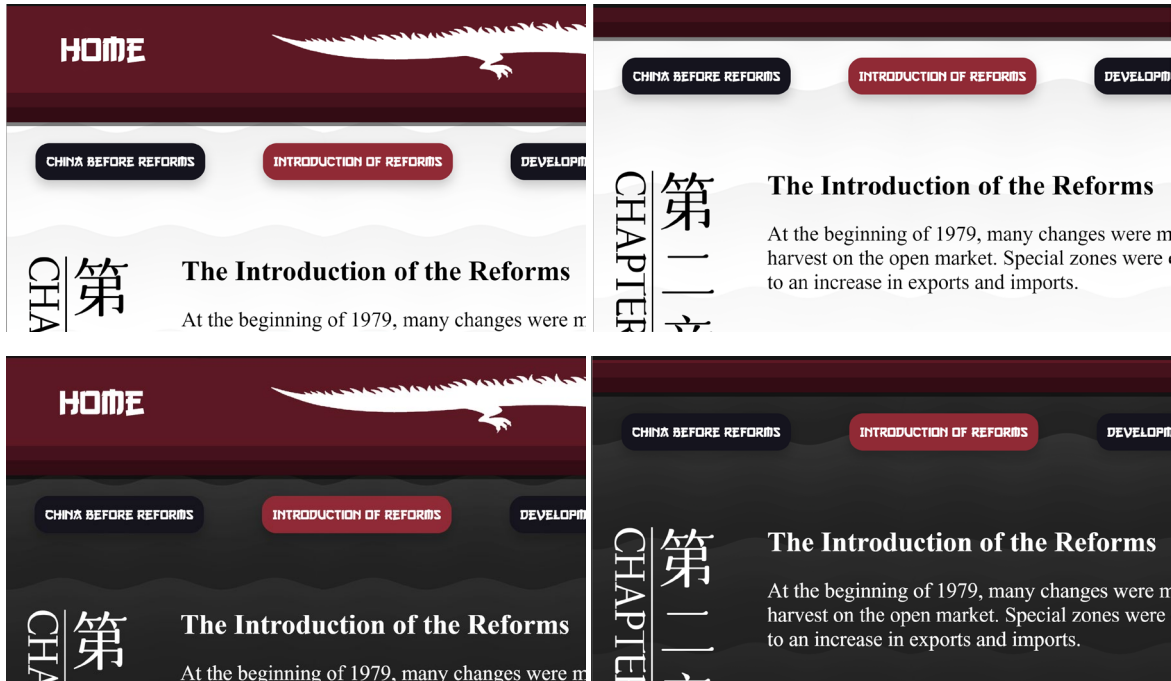


Abbildung 9-12: Demonstration Parallaxen-Effekt mit Hell- und Dunkelmodus

Es **scheint**, als ob sich das Muster während dem Runterscrollen nur leicht aufwärts-, beim Hochscrollen leicht abwärtsbewegt, im Gegensatz zu den anderen Elementen wie der Text.

Code:

```
window.addEventListener( type: "scroll", listener: function(event :Event ) :void {  
  
    let top :number = this.pageYOffset;  
    let scrollTop :number = window.scrollToY;  
    /*console.log(scrollTop)*/  
    /*console.log(window.innerWidth)*/  
    let layers :HTMLCollectionOf<Element> = document.getElementsByClassName( className: "parallax");  
    let layer, speed, yPos;  
    for (let i :number = 0; i < layers.length; i++) {  
        layer = layers[i];  
        speed = layer.getAttribute( qualifiedName: 'data-speed');  
        let yPos :number = -(top * speed / 100);  
        layer.setAttribute( qualifiedName: 'style', value: 'transform: translate3d(0px, ' + yPos + 'px, 0px)');  
    }  
}
```

Abbildung 13: Code aus dem JavaScript des Parallaxen-Effekts

Kurze Erläuterung des Code-Abschnitts:

Es wird in einem «scroll» Eventlistener die gescrollte Distanz auf der Webseite gespeichert. Dann werden alle Elemente mit der Klasse «parallax» gespeichert.

Jedes dieser Elemente hat ein Attribut «data-speed», das einen Float-Wert enthält. Mit diesem Wert und der gescrollten Distanz wird eine Abweichung berechnet, um der die relevante Parallaxe dann auf der Y-Achse der Webseite bewegt wird. Es gilt: desto mehr man runter scrollt, desto weiter nach oben werden die Parallaxen versetzt.

Wieso nach oben?:

Eigentlich sind diese Parallaxen auf position:fixed gesetzt. Das heisst, dass sie immer auf der genau gleichen Position des Bildschirms bleiben, unabhängig davon, was auf der Seite passiert. Dieser Code bewegt die Elemente auf dem Bildschirm langsam aufwärts, wenn man nach unten scrollt, um die Illusion eines 3-dimensionalen Bildes zu erschaffen, in dem sie von der Perspektive des Benutzers weiter hinten zu sein scheinen als die Hauptelemente wie Text und Bilder.

Speichern des Hell- und Dunkelmodus (Leonardo)

Haben Sie bemerkt, dass, wenn Sie zum Dunkelmodus wechseln, dies auf der gesamten Webseite übernommen wird? Dies funktioniert durch einige Funktionen, die ich in JavaScript geschrieben habe. Ein wichtiger Teil davon war das Wechseln der Klassen eines HTML-Elements.

```
document.querySelectorAll( selectors: ".dropdown-darkTheme").forEach( callbackfn: (i : Element) : void => {
    i.classList.remove( tokens: "dropdown-darkTheme");
    i.classList.add("dropdown-lightTheme");
});
document.querySelectorAll( selectors: ".arrowColorDark").forEach( callbackfn: (i : Element) : void => {
    i.classList.remove( tokens: "arrowColorDark");
    i.classList.add("arrowColorLight");
});
document.querySelectorAll( selectors: ".fontColorDark").forEach( callbackfn: (i : Element) : void => {
    i.classList.remove( tokens: "fontColorDark");
    i.classList.add("fontColorLight");
});
```

Abbildung 14: Code der Klassenänderung

Es funktioniert so, dass, wenn man auf den Modus Knopf drückt, Klassen gelöscht und hinzugefügt werden, wie in Abbildung 14 zu sehen ist. Die Klassen sind bereits im CSS definiert und im HTML als helle Version vorhanden. Wenn man jedoch auf den Knopf drückt, ändern sich die Klassen dieser HTML-Elemente und werden somit dunkel oder wieder hell. Die Einstellung wird mittels des lokalen Speichers (local Storage) gespeichert.

```
function changeThemeLight() : void {
    if (localStorage.getItem( key: "themeLight") !== null) {
        localStorage.removeItem( key: "themeLight")
    }
    localStorage.setItem("themeLight", "true")
}
```

Abbildung 15: Local Storage Nutzung

Wie im vorausgehenden Bild (Abbildung 15) zu sehen ist, verwende ich den lokalen Speicher. Dies dient dazu, Daten im aktuellen Browser des Nutzers zu speichern. Was ich hier programmiert habe,

ist, dass zunächst geprüft wird, ob ein Schlüssel namens "themeLight" vorhanden ist. Wenn ja, wird dieser gelöscht und durch einen neuen Wert ersetzt.

```
function content_loader() :void {  
    if (localStorage.getItem( key: "themeLight") !== null) {  
        let value :string = localStorage.getItem( key: "themeLight")  
        if (value === "true"){/*appapp*/  
            changeThemeLight()  
            if (document.getElementById( elementId: "keyart-0")) {  
                document.getElementById( elementId: "keyart-0").style.backgroundImage = "url('../img/background waves 3.png')";  
            }  
        } else {  
            changeThemeDark()  
            if (document.getElementById( elementId: "keyart-0")) {  
                document.getElementById( elementId: "keyart-0").style.backgroundImage = "url('../img/wavy background idaf prwr bl.png')";  
            }  
        }  
    }  
}
```

Abbildung 16: Laden des Dokuments

Zuletzt habe ich programmiert, dass bei Ladung des HTML-Inhalts mit JavaScript kontrolliert wird, ob der lokale Speicher einen Wert, der auf «true» gesetzt ist, enthält. Falls schon, ist die Webseite auf Hellmodus zu stellen. Falls nicht, wird sie auf Dunkelmodus gestellt.

Reflexion

Rückblick // Marko

Das Projekt fing mit einem guten Start an und hörte mit einem guten Ende auf. Der Weg dazwischen hatte ihre Stolpersteine und Hürden, aber die allgemeine Stimmung und Leistungserbringung blieb hoch oben.

Hinsichtlich gab es manche Sachen, die ich lieber anders hätte machen wollen.

Zum Beispiel die Dokumentation vom ersten Tag aus zu führen, oder die HTML-Struktur Flex- und nicht Manuell-basiert aufzubauen.

Aber allgemein bin ich stolz auf unseren Arbeitsprozess.

Insbesondere auf die Kooperation im Team und die Rücksicht, die wir gegenseitig aufeinander nahmen.

Rückblick // Leandro

Ich persönlich muss sagen, dieses Projekt ist viel besser gelaufen als jegliche vorherigen Projekte, die ich einst gemacht habe. Normalerweise, wenn man in einer Gruppe arbeitet, habe ich die Erfahrungen gemacht, dass jemand 80% macht und der Rest dann nur sehr wenige 20%. Doch ich kann mit viel Stolz sagen, dass uns das nicht geschehen ist. Wir hatten alle unsere Aufgaben, und haben alle unseren Teil zum Projekt beigetragen. Hätte einer von uns, seine Aufgaben nicht erfüllt, wäre unsere Website sehr mangelhaft. Und deshalb finde ich, dass unser Projekt ein Erfolg war. Auch wenn die Website nicht perfekt ist, und es generell nicht ganz per Plan verlaufen ist, würde ich sagen, dass die Teamarbeit, wie ich finde das wichtigste Element, eine sehr gute Dynamik hatte. Das hat viel Vertrauen ineinander gebraucht und dafür bin ich auch meinen Kollegen sehr dankbar.

Jetzt zu dem, was wir hätten besser machen können. Also zum einen hatten wir Zeitmanagement nicht ernst genommen, haben zwar ein Gantt-diagramm gemacht, aber überhaupt nicht beachtet. Zeitlich gesehen haben wir aber trotzdem viel Zeit in das Projekt eingesteckt und haben sowohl einzeln als auch zusammen viele Stunden daran gearbeitet.

Die Rollenverteilung hat sich auch nicht ganz bewährt, was aber am Ende trotzdem eigentlich noch gut funktionierte. Zusätzlich haben wir diese Dokumentation auch sehr vernachlässigt für die ersten 3 Wochen, begonnen haben wir in der Woche der Abgabe der Webseite. Also zum Glück noch früh genug, dass wir keinen Zeit-Stress hatten, aber wir hätten das auf jeden Fall besser machen können.

Im Grossen und Ganzen würde ich schlussendlich sagen, dass die grössten Mängel an unserem Projekt in der Planung und dem Zeitmanagement lagen. Aber wie ich schon erwähnt habe, bin ich sehr glücklich gewesen mit der Teamdynamik und auch mit der Website selbst. Somit kann ich auch mit Sicherheit sagen, dass ich nicht nur eine gute Zeit hatte, aber auch, dass es eine Erfahrung war, die mir viel Lehrreiches mit auf den Weg gegeben hat.



Rückblick // Leonardo

Schon zu Beginn hatten wir grosse Hoffnungen auf eine Webseite, die fehlerfrei funktioniert. Nun kann ich über die endgültigen Resultate sagen, dass dies der Fall sei. Unsere Motivation, China als Thema zu wählen, liegt darin, dass es in unserer gegenwärtigen Gesellschaft sehr präsent ist, beispielsweise durch Apps wie TikTok und Unternehmen wie Tencent. Uns hat es verwundert, warum China heutzutage so präsent ist, weshalb wir seine Geschichte studiert haben.

Kurz möchte ich meinen Teamkollegen danken, da ich finde, dass wir sehr gut zusammengearbeitet haben; besonders möchte ich Leandro danken, da er die Texte für Marko und mich verfasst hat. Ebenso möchte ich Marko Anerkennung zollen, weil er die grundlegende Idee präsentiert hat.

Jedoch stiessen wir mitten im Projekt auf ein gravierendes Problem: Die HTML-Struktur war fehlerhaft, was zu erheblichem Stress führte. Zudem war die Zeit ein Gegner, den wir letztendlich besiegt haben.

Obwohl wir viele Hürden überwinden mussten, bin ich stolz darauf sagen zu können, dass das Projekt erfolgreich abgeschlossen wurde.

Leistungsziele

Knopf, um auf die Index-Seite zurückzugelangen: präsent

Über die Geschichte berichtender Text auf der Webseite: präsent

Es sind zum Text passende Bilder zu sehen: ja

Englisch und Deutsch als verfügbare Sprachen: ja

Drache: präsent

Rötliche Farben: präsent

Dunkel- und Hellmodus: verfügbar

Webseite ist responsive: ja

Auftauchende Videos, wenn man mit der Maus über einem violetten Knopf schwebt: nicht präsent

-> Wieso:

- Keine Zeit mehr
- Meine Katze hat die Pläne dazu zerfetzt

Abweichung zum Plan

Time Management:

Weiter oben im Dokument wurde ein Gantt-Diagramm gezeigt, dass wir geplant haben. Unser realer Arbeitsablauf entsprach diesem nicht genau. Das Diagramm war ursprünglich nur dazu da, dass wir den groben Ablauf der Prozesse visuell dargestellt hatten. Diesem folgten wir aber schlussendlich auch nicht.

Recherche und Gestaltung der Webseite begannen gleichzeitig.

Mockup der Website wurde am ersten Tag erstellt.

Dokumentiert wurde anfangs der ersten Woche, aber schnell vernachlässigt.

Letztendlich wurde gleichzeitig recherchiert und Inhalt geschrieben sowie die Webseite codiert und gestalten.

Aufbau der Seite:

Die Webseite ähnelt sich stark dem originalen Mockup, und enthält mehrere darauf illustrierte Konzepte.

Beispiele sind Fusszeile, Kredite, Hintergrund und Farbenwahl.

Es gibt aber auch grössere Abweichungen, die dem Mockup nur teilweise, oder gar nicht entsprechen.

Beispiele dazu wären der Start-Knopf, Drache, auftauchende Videos und Bilder sowie roter und violetter Text.

Teamarbeit

Anfangs hatten wir die Rollen (*grob*) folgend aufgeteilt:

Leandro:

- *Recherchieren (rotiert jede Woche auf die nächste Person)*
- Inhalt der Webseite schreiben (rotiert auch wöchentlich, wäre aber hauptsächlich von Leandro übernommen)

Leonardo:

- Funktionalität der Webseite

Marko:

- Gestaltung der Webseite

Wie es aber kam, wühlten sich die Rollen schon am ersten Tag durcheinander. Zur Zeit der Abgabe sah die Rollenaufteilung etwa so aus:

Leandro:

- Recherchieren
- Inhalt der Webseite schreiben
- Quiz erstellen, codieren und Grundgestaltung festlegen

Leonardo:

- HTML-Struktur festlegen, aufbauen und existierende Strukturen überarbeiten
- Funktionalität der Webseite
- Allgemeine Grundgestaltung festlegen

Marko:

- Die Gestaltung der ganzen Webseite überarbeiten
- Zusätzliche Funktionalität, die zur Gestaltung dient, codieren (z.B. interaktive Elemente)

Diese Rollenänderungen wurden in unseren Aufgaben reflektiert.

Folgend sind die offensichtlichsten Abweichungen zu unserem ursprünglichen Plan:

Schlussendlich rotierten die Rollen «Recherchieren» und «Inhalt der Webseite schreiben» nie, und Leandro hat fast all den textlichen Inhalt der Seite selbst geschrieben. Dazu hatte er am meisten von uns recherchiert.

Leonardo musste die erste HTML-Struktur überarbeiten, da sie am Anfang unpraktisch aufgebaut war und nicht richtig funktionierte. Er musste auch die Grundgestaltung der Elemente festlegen, da unsere Seite sonst nicht dem Mockup entsprechen würde, und er nicht wusste, wie er die HTML-Struktur aufbauen sollte.

Ich (Marko) musste gewissermassen für bestimmte Gestaltungsaspekte Funktionalitäten codieren, da sie sonst nicht funktionieren, oder nicht schön aussehen würden. Ein Beispiel davon ist die Parallaxe im Hintergrund, oder das Auftauchen und Verschwinden des Hamburgermenus auf kleineren Bildschirmen beim Scrollen.

Trotz diesen Rollenvermischungen hatten wir alle unseren eigenen ungefähren Arbeitsbereich, und konnten so die Webseite in allen Aspekten auf unserer Weise aufbauen und erweitern.

Wir stellten sicher, dass bei grossen Entscheidungen und Änderungen alle einverstanden waren. Wir stellten aktiv sicher, dass alle mit ihren Aufgaben zufrieden waren, und versuchten diese fair und effizient aufzuteilen.

Um die Übersicht unserer Fortschritte zu behalten, arbeiteten wir, wenn möglich, zusammen. Am Wochenende wachten wir jeweils früh auf und blieben lange zusammen in einem Voice-Call, stellten sicher, dass niemand die gleiche Aufgabe hatte und, dass alle Aspekte der Webseite zueinander passten und zusammen funktionierten.

Wir versuchten so oft wie möglich einander zu helfen, und unsere Moral weit oben zu behalten. Wir motivierten uns gegenseitig, wenn die Lagen schief aussahen, und belohnten uns nach harten und langen Arbeitstagen (oft sassen wir bis zu 10 Stunden in einem Call, siehe Abbildung unten).

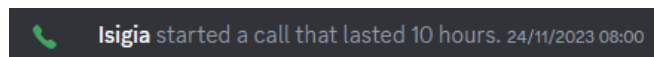


Abbildung 17: eines der mehreren 10 Stunden langen Coding-Marathons, die wir im Team hatten