|  |
| --- |
| Alexander Christopher Bock |
| IPWA01-01 |
| Fallstudie |

|  |
| --- |
| Moritz Döring  Informatik  32210824  10.10.2023 |
|  |
|  |
|  |

Inhaltsverzeichnis

1. [Konzept 3](#_Toc153792020)
2. [Umsetzung 3](#_Toc153792021)
   1. [Allgemein 3](#_Toc153792022)
   2. [Header 4](#_Toc153792023)
   3. [Contentbereich 4](#_Toc153792024)
   4. [Navigationsleiste 4](#_Toc153792025)
   5. [Footer 4](#_Toc153792026)

# Konzept

Für die Fallstudie habe ich mir die Aufgabenstellung 1.1 „CO2-Footprint“ ausgesucht. Dafür soll die Internetpräsenz einer Non-Profit-Organisation erstellt werden. Die Organisation beschäftigt sich mit dem Klimawandel und veröffentlicht CO2-Emissionsdaten von Unternehmen und Ländern. Das Design der Internetseite soll die Bemühungen der Organisation widerzuspiegeln.

Die Seite soll dafür modern, aber schlicht sein. Die Seite soll größtenteils Weiß gehalten sein, da zum Beispiel bunte Hintergründe die Seriosität untergraben könnten. Die Seite soll anderseits Farbakzente besitzen, um nicht langweilig zu wirken. Es wurden auf Grund dem Bezug zum Klimaschutz der Organisation ein grünes Farbkonzept gewählt. Das Logo der Internetseite ist eine Fabrik, aus der ein Blatt aus dem Schornstein emporsteigt. Es nimmt damit auch wieder Bezug zu dem Klimaschutz und weist auf die Zukunftsvision der Organisation hin.

# Umsetzung

Für die Erstellung der Internetseite wurde Visual Studio Code und Bootstrap verwendet.

# HTML-Kopf

Die folgenden Elemente sind für jede Seite der Website gleich und werden deshalb nur am Beispiel der Startseite erklärt.

In HTML-Dateien wird als erstes angegeben, dass es sich um eine HTML-Datei handelt (<!DOCTYPE html>). Danach wird deklariert, in welcher Sprache das Dokument verfasst ist (<html lang=“de“). Nach diesen beiden Angaben wird der HTML-Kopf angegeben. Dort werden üblicherweise nur Informationen angegeben, die nicht im Anzeigebereich des Browsers dargestellt werden sollen. Als erstes wird in diesem Fall die Zeichenkodierung, hier UTF-8 angegeben. Danach wird mit dem Viewport der Anzeigebereich der Website angepasst. Mit „width=device-width“ passt sich der Anzeigebereich auf die Bildschirmbreite des Gerätes an. Das Titel Element setzt den Namen in der Tableiste. In der nächsten Zeile wird Bootstrap per CDN eingebunden. Danach wird das Stylesheet „navbar text.css“ eingebunden und zum Schluss noch das Logo.

Jetzt folgt der sichtbare teil der Internetseite. Dieser besteht aus einem Header, einem Contentbereich und einem Footer. Der Contentbereich besteht selber noch aus einer Navigationsleite und dem eigentlichen Inhalt der Seite.

# Header

Auf der linken Seite des Headers kann man das Logo und den Namen der Organisation sehen. Die Standardfarbe für Unternehmensnamen in Navbars ist schwarz und das passte leider nicht zu dem restlichen Design des Headers. Ich konnte den Text nicht in einem Paragraphen-Element (<p> </p>) einfügen und die Farbe mit der „text-success“ Klasse auf grün setzen, da der Text sonst nicht auf einer Höhe mit den anderen Elementen des Headers ist. In der CSS Datei „navbar text.css“ wird daher die Standardfarbe von Bootstrap für diese spezielle Navbar Art auf grün überschrieben.

Rechts in dem Header stehen zwei Links. Die Links verweisen auf die Seite mit den Kontaktangaben und zu der Seite mit den Angaben für Unterstützerspenden. Der Header ist nicht „sticky“. Das bedeutet, dass wenn man herunterscrollt der Header nach oben verschwindet. Die Bootstrap Klasse „navbar“ hat bereits viele Funktionen implementiert. Ich konnte dadurch leicht die Farbe des Headers ändern und ihn auf die komplette Bildschirmbreite setzen. Durch die „collapse“ Klasse werden die Links in einem aufklappbaren Menü versteckt, falls der Bildschirm zu klein wird.

## Contentbereich

Um den Contentbereich responsive zu gestalten, wurde er mit Flexboxen umgesetzt. Der Header ist zwar ebenfalls eine Flexbox, aber im Gegensatz zu dem Contentbereich ist die Position immer die gleiche und es gibt auch keine zwei Flexboxen nebeneinander.

Dadurch, dass die Navigationsleiste sich direkt neben dem eigentlichen Inhalt der Seite befindet, muss man ein paar Dinge beachten. Bootstrap nutzt für die Flexboxen das Gridsystem. Hierbei werden die einzelnen Flexboxen nach dem zur Verfügung stehenden Platz angeordnet. Die Boxen werden wie in einer Tabelle nach Zeilen und Spalten angeordnet. Falls sich die Fenstergröße ändert, ordnet Bootstrap die Boxen neu an und positioniert nebeneinanderliegende Boxen in einer Zeile aus Platzgründen automatisch untereinander.

## Footer