**Fallstudie**

**IPWA01-01**

**Aufgabenstellung 1**

**co2-Footprint**

von

**Kevin Faller**

Klosterstr. 30

77723 Gengenbach

Bearbeitungszeitraum: 1. November 2022 – 22. Juni 2023

GitHub <https://github.com/KevFaller>

Webseite [www.mein-spielplatz.com](http://www.mein-spielplatz.com)

Wissenschaftliche Kursleitung Prof. Dr. Marian Benner-Wickner

Tutor Paul Libbrecht

**Vorwort**

Dies ist eine Fallstudie zum Nachweis der Prüfungsleistung für das Modul IPWA01-01. Recht herzlich bedanken möchte ich mich bei Prof. Dr. Marian Banner-Wickner, sowie bei meinem Tutor Paul Libbrecht, die die Betreuung meiner Fallstudie übernommen haben.

Vielen Dank!

**Eidesstattliche Erklärung**

Hiermit versichere ich eidesstattlich, dass die vorliegende Fallstudie von mir selbstständig und ohne unerlaubte fremde Hilfe angefertigt worden ist, insbesondere, dass ich alle Stellen, die wörtlich oder annähernd wörtlich oder dem Gedanken nach aus Veröffentlichungen, unveröffentlichten Unterlagen und Gesprächen entnommen worden sind, als solche an den entsprechenden Stellen innerhalb der Arbeit durch Zitate kenntlich gemacht habe, wobei in den Zitaten jeweils der Umfang der entnommenen Originalzitate kenntlich gemacht wurde. Ich bin mir bewusst, dass eine falsche Versicherung rechtliche Folgen haben wird.



(Datum, Unterschrift)

24.06.2023

Diese Fallstudie ist urheberrechtlich geschützt, unbeschadet dessen wird folgenden Rechtsübertragungen zugestimmt:

• der Übertragung des Rechts zur Vervielfältigung der Bachelor-Thesis für Lehrzwecke an der IU-Internationale Hochschule (§ 16 UrhG),

• der Übertragung des Vortrags-, Aufführungs- und Vorführungs-rechts für Lehrzwecke durch Professoren der IU-Internationale Hochschule (§ 19 UrhG),

• der Übertragung des Rechts auf Wiedergabe durch Bild- oder Ton-

träger an die IU-Internationale Hochschule (§21 UrhG).

**Zusammenfassung**

Die vorliegende wissenschaftliche Arbeit basiert auf der Fallstudie mit dem Titel "Fallstudie IPWA01-01: co2-Footprint". Der Autor Kevin Faller, hat diese Fallstudie im Rahmen seines Studiums zum Softwareentwickler an der IU - Internationale Hochschule erstellt. Die Arbeit wurde von Prof. Dr. Marian Benner-Wickner betreut.

Das Ziel der Fallstudie war es, eine Webseite für eine fiktive Non-Profit-Organisation zu entwickeln, die sich mit dem Thema Klimawandel und co2-Emissionen beschäftigt. Die Webseite sollte einen Titel mit Logo, einen Header mit globaler Navigation, einen Content-Bereich und einen Footer mit rechtlichen Hinweisen enthalten. Die Webseite sollte responsive sein und sowohl auf Desktop-Computern als auch auf Smartphones gut lesbar sein. Der Content bestand aus einer Tabelle mit fiktiven co2-Werten, die nach Land und Unternehmen sortiert und gefiltert werden konnte. Zudem mussten die Eingabefelder gegen injizierten Code abgesichert werden.

Die verwendeten Technologien für die Implementierung der Webseite waren HTML, CSS und JavaScript. Zur Versionsverwaltung und Veröffentlichung des Quellcodes wurde ein GitHub-Repository genutzt. Visual Studio Code wurde als Entwicklungseditor verwendet und die Extension "Live Server" ermöglichte eine einfache visuelle Darstellung der Webseite während der Entwicklung.

Die Arbeit befasste sich auch mit den Grundlagen von HTML, CSS und JavaScript, sowie deren Bedeutung für die Frontend-Entwicklung. GitHub wurde als Versionsverwaltungssystem erklärt, und Visual Studio Code wurde als Quelltexteditor vorgestellt.

Die Fallstudie wurde erfolgreich umgesetzt und die Webseite erfüllte alle gestellten Anforderungen. Die Arbeit schließt mit einer Diskussion die verwendeten Technologien thematisiert, Templates, GitHub und der Sicherung der Eingabefelder. Ein Fazit rundet die Arbeit ab.

Das Literaturverzeichnis und das Quellcodeverzeichnis sind in der Arbeit ebenfalls enthalten.

Inhaltsverzeichnis

[1 Einleitung 6](#_Toc138578413)

[1.1 Motivation 6](#_Toc138578414)

[1.2 Zielsetzung 6](#_Toc138578415)

[1.3 Vorgaben 6](#_Toc138578416)

[2 Fallvorstellung 6](#_Toc138578417)

[3 Verwendete Technologien 7](#_Toc138578418)

[3.1 HTML und HTTP 7](#_Toc138578419)

[3.2 CSS 7](#_Toc138578420)

[3.3 JavaScript 7](#_Toc138578421)

[3.4 GitHub 7](#_Toc138578422)

[3.5 Visual Studio Code 8](#_Toc138578423)

[3.5.1 Live Server 8](#_Toc138578424)

[4 Implementierung 8](#_Toc138578425)

[4.1 GitHub Repository 8](#_Toc138578426)

[4.2 Templates 8](#_Toc138578427)

[4.2.1 Logo 9](#_Toc138578428)

[4.2.2 Head 9](#_Toc138578429)

[4.2.3 Body 9](#_Toc138578430)

[4.2.4 Footer 10](#_Toc138578431)

[4.3 Schriftkultur RTL - LTR 11](#_Toc138578432)

[4.4 co2 – Tabelle 12](#_Toc138578433)

[4.4.1 Tabelle generieren 12](#_Toc138578434)

[4.4.2 Tabelle filtern 13](#_Toc138578435)

[4.5 Webseite absichern (injizierter Code) 13](#_Toc138578436)

[5 Diskussion 13](#_Toc138578437)

[5.1 Technologie 14](#_Toc138578438)

[5.2 Templates 14](#_Toc138578439)

[5.3 GitHub 14](#_Toc138578440)

[5.4 Injizierter Code 14](#_Toc138578441)

[5.5 Generierung der Tabelle 14](#_Toc138578442)

[5.6 Tabelle Filtern 14](#_Toc138578443)

[6 Fazit 14](#_Toc138578444)

[7 Literaturverzeichnis 16](#_Toc138578445)

[8 Quellcodeverzeichnis 16](#_Toc138578446)

[9 Abbildungsverzeichnis 16](#_Toc138578447)

# 1 Einleitung

## Motivation

Diese Fallstudie wurde im Rahmen meines Studiums zum Softwareentwicklers bei der IU – Internationale Hochschule als Prüfungsleistung für das Modul IPWA01-01 erstellt. Zur Überprüfung meiner Leistung habe ich mir die Aufgabe 1 „co2 - Footprint“ ausgesucht.

## Zielsetzung

Diese Fallstudie soll als Prüfungsleistung eine Dokumentation und ein Umsetzungskonzept darstellen. Als fiktiver Kunde wird eine Non-Profit-Organisation festgelegt. Die Webseite soll einen Titel mit einem Logo besitzen, ebenso einen Header mit einer globalen Navigation mit einem Content Bereich und einem Footer mit rechtlichen Hinweisen. Gemäß der Aufgabenstellung soll die Webseite responsive sein, so dass von einem Smartphone sowie von einem Desktop PC aus gut leserlich ist. Content ist eine Tabelle mit fiktiven co2-Werten, die sortiert und gefiltert werden kann. Aus Sicherheitsgründen müssen die Eingabefelder abgesichert werden, damit kein „injizierter Code“ ausgeführt werden kann.

## Vorgaben

Zu Beginn der Fallstudie werden momentan verfügbare Technogien untersucht, welche sich am besten zur Umsetzung eignen. Vorgegeben ist die Verwendung von HTML, CSS und JavaScript. Das Verwenden von weiteren JavaScript-Programmbibliotheken wie React.js oder Angula.js wurde freigestellt, ebenso die zu verwendende Entwicklungsumgebung.

# Fallvorstellung

In dieser Fallstudie soll eine öffentlich zugängliche Webseite für eine Non-Profit-Organisation entwickelt werden, die sich mit dem Thema Klimawandel beschäftigt. Das Ziel ist es, Transparenz über die jährlichen co2-Emissionen von Unternehmen und Ländern zu schaffen. Im zuge der vorliegenden Arbeit soll eine Webseite konzipiert und umgesetzt werden. Entwickelt werden soll eine Webanwendung, die folgende Anforderungen erfüllt:

a. Die Webseite enthält einen Titel und ein Logo.

b. Es gibt einen Header mit globaler Navigation, einen Content-Bereich und einen Footer mit rechtlichen Hinweisen.

c. Das Menü mit lokalen Links soll je nach Schriftkultur der Besucher rechts oder links dargestellt werden.

d. Die Seite soll responsiv sein und sowohl auf Desktop-Monitoren als auch auf Tablets oder Smartphones gut lesbar sein.

e. Die Seite soll eine Tabelle mit (fiktiven) co2-Emissionsdaten enthalten, die nach Land und Unternehmen sortiert und gefiltert werden kann.

f. Alle Eingabefelder müssen gegen die Ausführung von injiziertem Code abgesichert sein.

# Verwendete Technologien

## HTML und HTTP

Die Auszeichnungssprache HTML ist neben der Stilsprache CSS und der Programmiersprache JavaScript eine der drei groben Sprachen, die für die Frontend-Entwicklung wichtig sind. (Ackermann, 2021, p. 47) *Das HyperText Transfer Protocoll* (HTTP), das ein Anwendungsprotokoll ist, definiert, wie Webserver und Webbrowser unterereinander kommunizieren. Das Protokoll selbst ist relativ einfach gehalten, um den Webserver möglichst wenig zu belasten. Die Kommunikation selbst setzt ein Transportprotokoll wie TCP/IP voraus. Eine Kommunikation über HTTP ist immer gleich aufgebaut (Herold, 2017, p. 478)

## CSS

HTML beschreibt, was dargestellt werden soll, […]. Aber es beschreibt nicht, wie die Information dem Benutzer präsentiert werden soll. Über so genannte Cascading Style Sheets (CSS) kann dem Browser mitgeteilt werden, in welchen Schriftstil, -größe, -farbe, Hintergrund, Ausrichtung und Abstand der jeweilige Text darstellen soll. Teilweise kann auch noch das Verhalten bei Benutzerinteraktion modifiziert werden. (Herold, 2017, p. 479)

## JavaScript

Reine HTML-Dokumente erlauben nur sehr begrenzte Interaktion mit dem Benutzer, über Hyperlinks zu andere Dokumenten sowie über Formulare. Dabei steckt die ganze Logik hinter der Interaktion in der Implementierung des Servers. Schon Ende 1995 wurde erkannt, dass für eine bessere Bedienung eine echte Programmiersprache nötig ist, und mit JavaScript eine solche entwickelt. Trotz der Namensähnlichkeit hat JavaScript nichts mit der Sprache Java zu tun. (Herold, 2017, p. 480)

## GitHub

GitHub setzt sich aus zwei Worten zusammen, Git und Hub. Git ist eine Versionsverwaltungssoftware, entwickelt von dem Linuxerfinder Linus Torvalds. Hub wird ein Ort oder eine Logik bezeichnet Beispiele dafür sind Verteilzentren, Verkehrsknotenpunkte oder Güterverteilzentren. Das Besondere daran ist, dass man seinen Quellcode nicht nur verwalten, sondern auch anderen Menschen zur Verfügung stellen kann. GitHub steht so für Verwalten und Teilen.

## Visual Studio Code

Visual Studio Code von der Firma Microsoft, ist ein kostenloser Quelltext Editor. Er ist plattformübergreifend für die drei gängigen Betriebssysteme Windows, macOS und Linux verfügbar. Das besondere daran ist, dass man für viele Programmiersprachen Extensions installieren und diese verwenden kann. Somit ist es möglich seine Integrated Development Environment zu Deutsch: Integrierte Entwicklungsumgebung Projektspezifisch frei anzupassen.

### Live Server

Live Server ist eine Extension für Visual Studio Code mit der man in der Entwicklungsphase von Webseiten zur einfachen visuellen Darstellung die Seite Hosten kann und diese direkt lokal im Browser sieht.

# Implementierung

## GitHub Repository

Ein Repository ist die Sammlung (das Depot, das Archiv) aller Dateien eines Projekzs inklusive ihrer Geschichte. Sie können das Repository wie ein Datenbanksystem sehen, das sämtliche Zustände des Projekts von der ersten Datei bis zum aktuellen Zustand enthält, samt Informationen wer wann welche Änderungen durchgeführt hat. (Öggl & Kofler, 2022, S. 78) Zur Versionsverwaltung und zum Veröffentlichen des Quellcodes zur Prüfungskontrolle wurde ein GitHub-Repository verwendet. Um dies zu verwenden, muss ein Account auf github.com erstellt werden. Zur einfacheren und benutzerfreundlichen Verwendung kann GitHub-Desktop installiert werden. Um ein lokales Repository nun mit der Versionskontrolle von GitHub zu verwalten, muss diese verknüpft werden. Nach dem verknüpfen besteht die Möglichkeit über den Button „push“ das lokale Repository zu laden. Sollten nun Änderungen an dem Repository lokal vorgenommen werden, so sind diese auch in GitHub-Desktop zu sehen. Durch einen Commit können Änderungen beschrieben werden, und diese ebenfalls über den „push“ Button hochgeladen werden. Nun besteht die Möglichkeit, die Änderungen am Quellcode zu dokumentieren und gegebenenfalls zu einem gewünschten Änderungspunkt zurückzuspringen.

## Templates

Um den gleichbleibenden Stil über die Webseite hinweg zu gewährleisten, wurde zu Beginn ein Template erstellt, von dem ausgehend die Index.html und style.css sowie alle weiteren Seiten erstellt wurden.

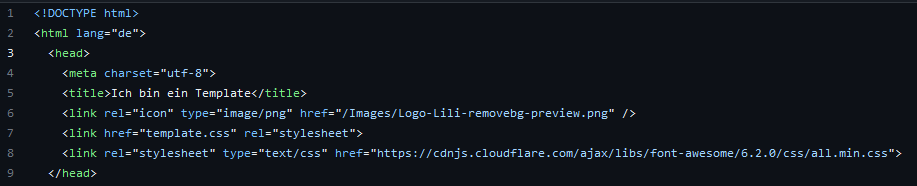
### Logo

Das Logo wurde von Lili Marie Faller Entworfen, um keine Copyright verletzungen zu begehen. Die Implementierung wurde von Kevin Faller übernommen und im head bereichs der HTML Seite eingebunden. (Quellcod-1 head)

Abbildung Logo

### Head

Der vorliegende (Quellcode-1) Code repräsentiert die grundlegende Struktur einer HTML-Seite mit Spracheinstellung für die deutsche Sprache. Im <head>-Bereich werden relevante Metadaten definiert, wie die Zeichenkodierung (UTF-8), den Titel der Webseite sowie das Favicon, welches ein Bild oder Symbol ist, das im Browser-Tab angezeigt wird. Des Weiteren werden zwei externe CSS-Dateien verknüpft: "template.css" für das individuelle Design der Webseite und "all.min.css" von Font Awesome, welches eine umfangreiche Sammlung an vorgefertigten Icons bereitstellt. Diese Strukturierung des Codes ist essenziell für die Gestaltung und Darstellung einer professionellen Webseite.



Quellcode head

### Body

<body> Definiert den Hauptinhalt eines HTML-Dokuments. Kann innerhalb eines HTML-Dokuments nur einmal vorkommen. (Ackermann, 2021, S. 615) Im Rahmen einer Webseite wird der <body>-Bereich als bedeutender Bestandteil betrachtet, da er den Hauptinhalt sowie häufig die Menüleiste enthält. Um eine strukturierte Darstellung zu gewährleisten, wird der <body> in zwei Abschnitte unterteilt: den "header" für das Kopfelement des <body> und den <main>-Bereich für den Hauptinhalt. Diese Unterteilung ermöglicht eine gezielte Gestaltung und Organisation der Webseite, um eine ansprechende Benutzererfahrung zu gewährleisten.

#### Header

In diesem Bereich wird das Logo, die Überschrift sowie das Globale Dropdownmenü festgelegt. Im Menüabschnitt wird im HTML-Code auch die Beschriftung des Menüpunktes und dessen Untergliederung und die referenzierenden links festgelegt.

<label for="products\_checkbox">Menu-1</label>

                    <ul class="dropdown">

                      <li><a href="tabelle.html">Tabelle</a></li>

                      <li><a href="#">BBBB</a></li>

                      <li><a href="#">CCCC</a></li>

                    </ul>

Quellcode Menupunkt 1

Ein Bild, das Text, Screenshot, Schrift, Logo enthält.

Automatisch generierte Beschreibung

Abbildung Logo-Menü

#### Main

Im „main“ Bereich, wird der Hauptinhalt dargestellt. Dieser wird wiederum in einzelne „article“ Bereichen unterteilt, um die einzelnen Artikel besser voneinander visüll abgrenzen zu können. Die „article“ Bereiche können Tabellen, Bilder oder auch Artikel sein.

<article>

            <h1>Ich bin eine Überschrifft</h1>

            <p>Beispieltext...</p>

            <p>Beispieltext...</p>

            <p>Beispieltext...</p>

            <p>Beispieltext...</p>

        </article>

Quellcode article

### Footer

In der Fußzeile dem sogenannten „footer“ sind die rechtlichen Hinweise einer Webseite wie beispielsweise das Copyright. Ebenso wird dieser Bereich auch zur Verlinkung und für Verweise auf die Indexseite sowie zu verschiedenen Social-Media-Kanälen genutzt.

<h4>Social Media</h4>

            <div class="social-links">

              <li><a href="https://www.facebook.com/kevin.faller.33"><i class="fab fa-facebook-f "></i></a></li>

              <li><a href="https://github.com/KevFaller"><i class="fab fa-github"></i></a></li>

              <li><a href="https://www.instagram.com/kevinfallerkevdev/"><i class="fab fa-instagram"></i></a></li>

              <li><a href="https://www.linkedin.com/in/kevin-faller-a4b707205"><i class="fab fa-linkedin-in"></i></a></li>

            </div>

Quellcode Social Media

Ein Bild, das Screenshot, Text, Schrift, Kreis enthält.

Automatisch generierte Beschreibung

Abbildung Social Media

## Schriftkultur RTL - LTR

Die Umstellung von LTR (Left-to-Right) zu RTL (Right-to-Left) wurde durch das Hinzufügen einer JavaScript-Funktion realisiert. Durch Klicken auf den "RTL"-Link unter dem Menüpungt-1 wird die dir-Eigenschaft des documentElement auf "rtl" gesetzt, wodurch die Ausrichtung der Webseite auf RTL geändert wird. Ähnlich wird durch Klicken auf den "LTR"-Link die dir-Eigenschaft auf "ltr" gesetzt, um zur LTR-Ausrichtung zurückzukehren.

<label for="products\_checkbox">Menu-1</label>

                    <ul class="dropdown">

                      <li><a href="tabelle.html">Tabelle</a></li>

                      <li><a href="#"onclick="toggleRTL()">RTL</a></li>

                      <li><a href="#"onclick="toggleLTR()">LTR</a></li>

Quellcode 5 Menu-RTL\LTR

<script>

          function toggleRTL() {

            document.documentElement.dir = 'rtl'; // Setze die Ausrichtung auf RTL

          }

          function toggleLTR() {

            document.documentElement.dir = 'ltr'; // Setze die Ausrichtung auf LTR

          }

        </script>

Quellcode 6 script-RTL\LTR

Ein Bild, das Text, Screenshot, Schrift, Design enthält.

Automatisch generierte Beschreibung

Abbildung Toggelmenü

## co2 – Tabelle

### Tabelle generieren

Um eine automatische Generierung einer co2-Tabelle mit fiktiven Daten zu ermöglichen, wurde eine JavaScript-Funktion implementiert. Diese Funktion greift auf ein im Skript definiertes Array namens "array" zu, welches manuell mit den entsprechenden Daten gefüllt wurde. Die Funktion ermöglicht das Abrufen der Daten aus diesem Array und verwendet sie zur dynamischen Erstellung der Tabelle. Dadurch wird eine effiziente und automatisierte Darstellung der co2-Daten gewährleistet.

        id:"8",

        name:"Lind-Schokolade",

        land:"Schweiz",

        co2:"8763",

    }

]

// show table data

function showtable(curarray){

    document.getElementById("mytable").innerHTML =`

        <tr class="bg-primary test-white fw-bold">

        <td>ID</td>

        <td>Name</td>

        <td>Land</td>

        <td>co2</td>

    `;

Quellcode Tabelle generieren

Ein Bild, das Text, Screenshot, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Abbildung Tabelle

### Tabelle filtern

Die Filterfunktion wurde mittels eines JavaScript Funktion implementiert. Zunächst wurde ein Event-Listener auf das Eingabefeld registriert, um auf Benutzereingaben zu reagieren. Bei jeder Eingabe wird der eingegebene Suchbegriff mit den vorhandenen Daten abgeglichen. Durch Iteration über die Daten und Anwendung von Filtermechanismen wird eine Liste mit übereinstimmenden Ergebnissen erstellt. Diese Ergebnisse werden dann dynamisch in die Webseite eingefügt und angezeigt. Nicht übereinstimmende Inhalte ausgeblendet werden.

Ein Bild, das Text, Multimedia, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Abbildung Tabelle gefiltert

## Webseite absichern (injizierter Code)

Es wurde eine Funktion namens sanitizeInput implementiert. Diese Funktion hat den Zweck, Benutzereingaben von potenziell schädlichen Codes zu bereinigen. Dies wird erreicht, indem ein temporäres DOM-Element erstellt wird und der Eingabetext als reiner Text in den Inhalt dieses Elements gesetzt wird. Durch die Verwendung des innerHTML-Attributs wird sichergestellt, dass der Text als Text behandelt wird und nicht als ausführbarer HTML-Code interpretiert wird. Diese Maßnahme trägt dazu bei, die Sicherheit der Webseite zu erhöhen und potenzielle Angriffe durch injizierten Codes zu verhindern.

// Funktion zur Säuberung der Benutzereingaben von potenziell schädlichem Code

function sanitizeInput(input) {

    var tempElement = document.createElement("div");

    tempElement.textContent = input;

    return tempElement.innerHTML;

}

Quellcode injizierter Code

# Diskussion

Das Ziel dieser Fallstudie ist es, im Rahmen einer Prüfungsleistung für das Modul IPWA01-01 als Softwareentwickler eine Dokumentation und ein Umsetzungskonzept zu erstellen. Dabei soll eine öffentlich zugängliche Webseite für eine Non-Profit-Organisation entwickelt werden, die sich mit dem Thema Klimawandel beschäftigt. Die Webseite soll Transparenz über die jährlichen CO²-Emissionen von Unternehmen und Ländern schaffen.

## Technologie

Die gewählten Technologien (HTML, CSS, JavaScript) waren gut geeignet, um die Anforderungen der Webseite zu erfüllen. Sie ermöglichten eine effiziente Entwicklung und Umsetzung des Projekts.

## Templates

Die Verwendung von Templates erwies sich als vorteilhaft, da dadurch ein einheitlicher Stil über die gesamte Webseite hinweg gewährleistet werden konnte. Dadurch konnte Zeit gespart und eine konsistente Benutzererfahrung geschaffen werden.

## GitHub

Die Implementierung des GitHub-Repositories war eine sinnvolle Entscheidung, da dadurch eine Versionskontrolle des Quellcodes ermöglicht wurde.

## Injizierter Code

Die Absicherung der Eingabefelder gegen injizierten Code war ein wichtiger Sicherheitsaspekt. Durch geeignete Maßnahmen wie das implementieren der sanitizelnput Funktion konnte verhindert werden, dass schädliche Codes ausgeführt und die Webseite anfällig für Angriffe wurde.

## Generierung der Tabelle

Die Generierung der co2-Tabelle mit Hilfe von JavaScript war effizient und ermöglichte eine automatisierte Darstellung der Daten. Dadurch konnte Zeit gespart werden und eine dynamische Aktualisierung der Tabelle bei neuen Daten war möglich.

## Tabelle Filtern

Die Implementierung der Filterfunktion in der co2-Tabelle verbesserte die Benutzererfahrung, indem Benutzer die Daten nach Land und Unternehmen sortieren und filtern konnten. Dadurch konnten Benutzer schnell die gewünschten Informationen finden.

# Fazit

Im Rahmen der Fallstudie "CO²-Footprint" wurde eine öffentlich zugängliche Webseite für eine Non-Profit-Organisation entwickelt, die sich mit dem Thema Klimawandel und CO²-Emissionen beschäftigt. Das Ziel bestand darin, mehr Transparenz über die jährlichen CO²-Emissionen von Unternehmen und Ländern zu schaffen. Die Webseite wurde entsprechend der vorgegebenen Aufgabenstellung entworfen und umgesetzt.

Zur Umsetzung der Webseite wurden verschiedene Technologien eingesetzt. HTML wurde verwendet, um die grundlegende Struktur der Webseite zu definieren. CSS kam zum Einsatz, um das Design und das Erscheinungsbild der Webseite zu gestalten. JavaScript wurde verwendet, um interaktive Elemente und Funktionen auf der Webseite zu implementieren.

Ein öffentliches Code-Repository wurde in GitHub eingerichtet, um den Quellcode der Webseite zu verwalten und den Überprüfungsprozess zu ermöglichen. Die Entwicklungsumgebung Visual Studio Code wurde genutzt, um den Code zu schreiben und zu bearbeiten. Die Live Server Extension in Visual Studio Code ermöglichte eine einfache lokale Hostung der Webseite während der Entwicklungsphase.

Die Webseite erfüllt die gestellten Anforderungen der Fallstudie. Sie enthält einen Titel und ein Logo, einen Header mit globaler Navigation, einen Content-Bereich und einen Footer mit rechtlichen Hinweisen. Das Menü mit lokalen Links wurde entsprechend der Schriftkultur der Besucher rechts oder links dargestellt. Die Webseite ist responsiv und kann auf verschiedenen Geräten wie Desktop-Monitoren, Tablets und Smartphones gut gelesen werden. Eine Tabelle mit fiktiven CO²-Emissionsdaten wurde implementiert und ermöglicht das Sortieren und Filtern nach Land und Unternehmen. Alle Eingabefelder wurden abgesichert, um die Ausführung von injizierten Codes zu verhindern.

Die Implementierung der Webseite erfolgte unter Berücksichtigung bewährter Praktiken und Standards in der Webentwicklung. Der Quellcode wurde im GitHub-Repository abgelegt und dokumentiert.

Insgesamt wurde die Aufgabenstellung der Fallstudie erfolgreich umgesetzt. Die erstellte Webseite ermöglicht es der Non-Profit-Organisation, den CO²-Footprint von Unternehmen und Ländern transparent darzustellen und trägt somit zur Sensibilisierung für den Klimawandel bei. Durch die Verwendung geeigneter Technologien und die Beachtung von Sicherheitsaspekten wurde eine professionelle und benutzerfreundliche Webanwendung entwickelt.

# Literaturverzeichnis

Öggl, B., & Kofler, M. (2022). *Git - Projektverwaltung für Entwickler und DevOps-Teams.* Rheinwerk.

Ackermann, P. (2021). *Webentwicklung.* Rheinwerk Compting.

Herold, L. W. (2017). *Grundlagen der Informatik.* Pearson.

# Quellcodeverzeichnis

[Quellcode 1 head 9](#_Toc138422232)

[Quellcode 2 Menupunkt 1 10](#_Toc138422233)

[Quellcode 3 article 10](#_Toc138422234)

[Quellcode 4 Social Media 10](#_Toc138422235)

[Quellcode 5 Menu-RTL\LTR 11](#_Toc138422236)

[Quellcode 6 script-RTL\LTR 11](#_Toc138422237)

[Quellcode 7 Tabelle generieren 12](#_Toc138422238)

[Quellcode 8 injizierter Code 13](#_Toc138422239)

# Abbildungsverzeichnis

[Abbildung 1 Logo 9](https://iubhfs-my.sharepoint.com/personal/kevin_faller_iubh-fernstudium_de/Documents/IUBH-Studium/Module/Programmierung%20von%20WEB-Anwendungspberflächen/Fallstudie/Fallstudie-Kevin%20-%201.0.1.docx#_Toc138422210)

[Abbildung 2 Logo-Menü 10](#_Toc138422211)

[Abbildung 3 Social Media 11](#_Toc138422212)

[Abbildung 4 Toggelmenü 11](#_Toc138422213)

[Abbildung 5 Tabelle 12](#_Toc138422214)

[Abbildung 6 Tabelle gefiltert 13](#_Toc138422215)