

Projektbericht zu Train2POW

VU Geoinformatik: Webmapping

Lehrveranstaltungsleitung:

Förster K. & Öggel B.

Baumann, Silja Lillian (silja.baumann@student.uibk.ac.at)
Siebenbrunner, Anna (anna.siebenbrunner@student.uibk.ac.at)
Stang, Anna Klara (anna.stang@student.uibk.ac.at)
Geographie, Master of Science

Inhaltsverzeichnis

Abbildungsverzeichnis.....	2
Einleitung.....	3
Protokoll der Implementierungsschritte und eingesetzten Techniken.....	4
1. Erste Vorlagen der Webseiten	4
2. Seiten mit Textinhalten befüllen	4
3. Aufbau und Erstellung der Karten	5
4. Style der Seiten an die von POW anpassen.....	9
5. Responsives Navigationselement.....	9
6. Kontaktformular einfügen	11
Literaturverzeichnis:.....	13

Abbildungsverzeichnis

Abbildung 1: Vorbereitung der HTML Seite	4
Abbildung 2: Übersichtskarte der POW-Standorte hinzufügen	5
Abbildung 3: Popups der POW-Standorte hinzufügen.....	5
Abbildung 4: JavaScript für die POW-Standorte befüllen	6
Abbildung 5: JavaScript-Datei mit Informationen über Skigebiete befüllen.....	6
Abbildung 6: Marker für die Skigebiete erstellen und Abhängigkeit von Layern definieren	7
Abbildung 7: Minimap der Karte hinzufügen	8
Abbildung 8: Overlay je Bundesland hinzufügen	8
Abbildung 9: Favicon hinzufügen	9
Abbildung 10: nav-Element für die Erstellung des Navigationsmenüs	10
Abbildung 11: Erstellung eines Flex-Containers für nav ul.....	10
Abbildung 12: Setzen der Schriftgröße von nav li auf 1.5 em	10
Abbildung 13: Erstellung media query für größere Displays.....	10
Abbildung 14: Anpassen des nav Elements auf flexible Breite und Erstellen des Nachglüheffekts	10
Abbildung 15: Farbwechsel für Anzeige der aktuellen Seite.....	11
Abbildung 16: Erstellen des Rollovereffekts	11
Abbildung 17: Kontaktformular auf der Webseite.....	11
Abbildung 18: Kontaktformular in HTML einbauen	12

Einleitung

In dieser Projektarbeit wollen wir uns mit der Erreichbarkeit von Skigebieten mit öffentlichen Verkehrsmitteln beschäftigen. Hierfür wurden drei HTML Seiten angelegt, die miteinander verknüpft wurden und somit einfach zu bedienen und gegenseitig zu erreichen sind. Die drei Webseiten werden im Folgenden kurz vorgestellt:

1. WAS ist POW?

Um die Idee, aus der heraus wir das Projekt ausgewählt haben, besser zu verstehen, gibt unsere erste Seite einen kurzen Einblick in [POW](#) AT (Protect Our Winters Austria) und dessen Ideen und Ziele, sowie dessen Entstehung. Auf dieser Seite gibt es zudem eine Übersichtskarte, wo in Österreich POW vertreten ist und an wen man sich wenden kann, wenn man an einem POW-Stammtisch teilnehmen möchte.

2. Warum TRAIN2POW?

Auf unserer zweiten Webseite wollen wir die Besucher*innen unserer Website auf die Problematik des Wintersports in Kombination mit CO₂-Ausstoß aufmerksam machen. Wir erklären hier, wo Emissionen entstehen und welche Alternativen zum Individualverkehr es gibt. Die Seite zeigt somit, wie wichtig eine Diskussion um eine klimafreundliche An- und Abreise in die Skigebiete ist.

3. CO₂ neutral ins Skigebiet

Die letzte Seite stellt unser Herzstück dar. Durch eine interaktive Karte ist es Besuchenden der Seite möglich, Skigebiete in ihrer Umgebung zu finden und durch den eingebauten Scotty-Link der ÖBB eine direkte Bahnverbindung herauszusuchen. Jedes Skigebiet wird mit einem Schneeflocken Icon dargestellt und ist mit einem Popup ausgestattet, welches Infos über das Skigebiet enthält. Für eine übersichtlichere Darstellung ist es möglich, sich jeweils die Skigebiete je Bundesland anzeigen zu lassen. Zudem haben Besucher*innen die Möglichkeit weitere Skigebiete vorzuschlagen, indem sie ein Formular mit den wichtigsten Eckdaten ausfüllen. Dieses wird dann an uns weitergeleitet und wir können diese dann integrieren.

Auf allen drei Seiten haben wir versucht unseren Stil an den der offiziellen Webseite anzupassen.

Protokoll der Implementierungsschritte und eingesetzten Techniken

1. Erste Vorlagen der Webseiten

Zuerst wurden jeweils eine Vorlage von HTML, CSS und JS aufgesetzt. Zunächst haben wir drei html5 Seiten erstellt, sowie ein CSS-Stylesheet. Außerdem wurden zwei JavaScript-Dokumente angelegt, eines für die erste html-Seite und eins für die dritte html-Seite. Schließlich wurden die html-, CSS-, und Javascript Dokumente miteinander verknüpft. Als Beispiel ein Screenshot von der Seite „was-ist-pow.html“ in Abbildung 1.

Die drei Seiten wurden außerdem mit Hilfe einer Media Query in CSS an verschiedene Fenstergrößen angepasst. Für Bildschirme, die maximal 900px weit sind, ist sich die maximale Breite der Seite auf 90% gestellt.

```
<> was-ist-pow.html > html > body > header
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <!-- FontAwesome -->
9      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/
10 5.15.3/css/all.min.css" />
11      <!-- Stylesheet main.css -->
12      <link rel="stylesheet" href="main.css">
13      <!-- Leaflet -->
14      <link rel="stylesheet" href="https://unpkg.com/leaflet@1.7.1/dist/leaflet.css"
15          integrity="sha512-xodZBNTC5n17Xt2atTPuE1HxjVMSvLVW9ocqUKLsCC5CXdbqCmblAsho
16MAS6/keqq/sMzMZ19scr4PsZChSR7A=="
17          crossorigin="" />
18      <script src="https://unpkg.com/leaflet@1.7.1/dist/leaflet.js"
19          integrity="sha512-XQoYMqMTK8LvdxXYG3nZ448hOEQiglfqkJs1NOQV44cWnUrBc8PkA0cX
20y20w0vlaXaVUearIOBhiXZ5V3ynxwA=="
21          crossorigin=""></script>
22      <!-- Leaflet Providers -->
23      <script src="https://cdnjs.cloudflare.com/ajax/libs/leaflet-providers/1.12.0/
24leaflet-providers.min.js"
25          integrity="sha512-LixflAm9c0/q0NbZ9F1Ept
26+QJ6QBpb7wUlpuyv1EHloTVgwSK8j3yMV3e1nElGQcv7Y5QTF1F/FqyeE/N4LnKQ=="
27          crossorigin="anonymous"></script>
28      <!-- Kartenscript -->
29      <script defer src="pow.js"></script>
30      <link rel="icon"
31          href="https://protectourwinters.at/wp-content/uploads/2020/09/
32cropped-android-chrome-512x512-1-32x32.png"
33          sizes="32x32" />
34      <title>Protect Our Winters Austria</title>
35  </head>
```

Abbildung 1: Vorbereitung der HTML Seite

2. Seiten mit Textinhalten befüllen

Um die drei Seiten mit Hintergrundinformationen zu füllen, wurden vor allem für die erste und dritte Seite einige Informationen von protectourwinters.at übernommen. Für die zweite Seite mit dem Thema „Warum CO2-neutral ins Skigebiet?“ wurden auch weitere Quellen herangezogen.

3. Aufbau und Erstellung der Karten

- *Karte 1: Übersichtskarte der POW-Niederlassungen*

Zuerst wurde eine Karte mithilfe von Leaflet Maps hinzugefügt. Diese wurde auf Österreich eingestellt, sodass alle Standorte von POW Austria sichtbar sind.

```
// Karte initialisieren und auf Österreichs Mittelpunkt blicken
let powMap = L.map("powMap", {
  fullscreenControl: true,
  center: [47.71216, 13.34290],
  zoom: 6.25,
  layers: [
    L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png')
  ]
});
```

Abbildung 2: Übersichtskarte der POW-Standorte hinzufügen

Die POW Standorte wurden in die pow.js Datei eingetragen und mit Informationen befüllt, um diese dann auf der Karte anzeigen zu lassen. Abbildung 3 zeigt, wie die Popups eingebaut wurden, die die Informationen zu den Kontaktpersonen geben. Die Daten zu den Standorten wurden selbst zusammengetragen und aufbereitet. Als Icon für die Popups wurden, passend zum Thema, Schneeflocken gewählt. Dieses wurde über von der Original POW-Webseite bezogen.



```
// Marker an POW-Standorten setzen + Popup
var marker = (function () {
  for (let index = 0; index < standorte.length; index++) {
    L.marker([standorte[index].lat, standorte[index].lon], {
      icon: snowflake
    })
    .bindPopup(
      '<h2>' + standorte[index].name + '</h2><br>' +
      '<p><b>Stammtisch: </b>' + standorte[index].stammtisch + '</p>' +
      '<p><b>Ansprechperson: </b>' + standorte[index].ansprechperson + '</p>' +
      '<p><a href=' + standorte[index].mail +
      '><i class="fas fa-envelope"></i>Schreib ein Mail!</a></p>'
    )
    .addTo(powMap);
  }
})();
```

Abbildung 3: Popups der POW-Standorte hinzufügen

```
var standorte = [{  
  "name": "Innsbruck",  
  "lat": 47.26913970430914,  
  "lon": 11.395510708574795,  
  "bundesland": "Tirol",  
  "stammtisch": "jeden zweiten Mittwoch im Monat",  
  "ansprechperson": "Verena",  
  "mail": "mailto:innsbruck@protectourwinters.at"  
},
```

Abbildung 4: JavaScript für die POW-Standorte befüllen

- *Karte 2: CO2 neutral ins Skigebiet*

Zuerst wurde eine Karte von Österreich auf der Seite eingefügt. Dann wurde eine weitere JavaScript Datei angelegt, in der die Skigebiete aufgelistet werden. Mit der `const`-Funktion wird die Variable `SKIGEBIETE` definiert. Jedem Skigebiet wurden eine ID, Name, die Koordinaten, das Bundesland, ein Link zur Webseite und der Scotty Link zugewiesen.

```
const SKIGEBIETE = [{  
  "id": "kuehtai",  
  "name": "Kühtai",  
  "lat": 47.213244507445054,  
  "lon": 11.017325791147448,  
  "bundesland": "Tirol",  
  "bundeslandId": "tir",  
  "link": "https://www.kuehtai.info/",  
  "scotty": "https://fahrplan.oebb.at/bin/query.exe/dn?REQ0JourneyStopsZID=A=1@O=K%Fchtai%20Gaiskogelbahn@X=1"  
},
```

Abbildung 5: JavaScript-Datei mit Informationen über Skigebiete befüllen

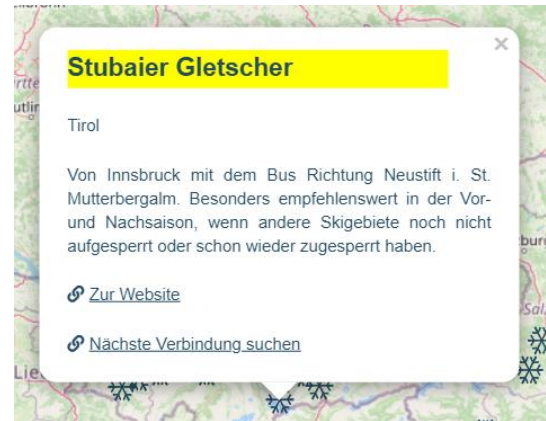
Die Daten wurden händisch zusammengetragen. Für die jeweiligen Skigebiete wurde auch immer geprüft, ob eine Anreise mit öffentlichen Verkehrsmitteln sinnvoll ist.

Dann wurde für jedes Skigebiet ein Marker angelegt, der die Informationen zu dem Skigebiet wiedergibt. Die Marker wurden in `main.js` angelegt wie in Abbildung 6 sichtbar ist.

Zudem wurde für jedes Bundesland ein eigener Layer erstellt, um so einfacher durch die Karte navigieren zu können (Abbildung 8). Die Anzeige wird jeweils durch eine `if`-Abfrage gesteuert.

Um die verschiedenen Bundesländer betrachten zu können, wurde mit `Overlays` gearbeitet. Hierzu wurde jeweils ein Layer pro Bundesland hinzugefügt. Über die `LayerControl` Funktion wird eine Kartenmenü eingefügt, in dem der jeweilige Layer ausgewählt werden kann.

Außerdem wurde für die Karten eine Minimap eingebaut (Abbildung 7). Auch diese wurde mithilfe von Leaflet implementiert und bezieht sich auf eine OpenStreetMap. Mithilfe der `zoomLevelOffset` wurde der Bezug zur Hauptkarte hergestellt und fixiert, wie viel kleiner der Ausschnitt dargestellt werden sollte.

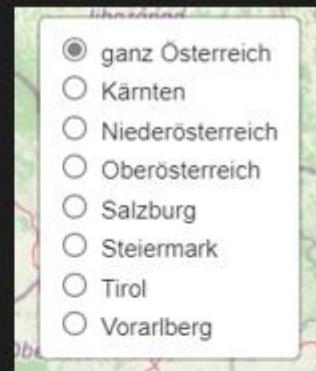


```
var marker = (function () {
  for (let index = 0; index < SKIGEBIETE.length; index++) {
    let marker = L.marker([SKIGEBIETE[index].lat, SKIGEBIETE[index].lon], {
      icon: snowflake
    })
    marker.bindPopup(`
      <h2>${SKIGEBIETE[index].name}</h2>
      <p>${SKIGEBIETE[index].bundesland}</p>
      <p>${SKIGEBIETE[index].info || ''}</p>
      <p><a href=${SKIGEBIETE[index].link, target="_blank"}><i class="fas fa-link"></i>Zur Website</a></p>
      <p><a href=${SKIGEBIETE[index].scotty}><i class="fas fa-link"></i>Nächste Verbindung suchen</a></p>
    `)
    .addTo(overlays.at)
    // Mit den nachfolgenden if-Abfragen wird für jedes Bundesland noch ein eigener Layer angelegt
    if (SKIGEBIETE[index].bundeslandId == "ktn") {
      let marker = L.marker([SKIGEBIETE[index].lat, SKIGEBIETE[index].lon], {
        icon: snowflake
      })
      marker.bindPopup(`
        <h2>${SKIGEBIETE[index].name}</h2>
        <p>${SKIGEBIETE[index].bundesland}</p>
        <p>${SKIGEBIETE[index].info || ''}</p>
        <p><a href=${SKIGEBIETE[index].link, target="_blank"}><i class="fas fa-link"></i>Zur Website</a></p>
        <p><a href=${SKIGEBIETE[index].scotty}><i class="fas fa-link"></i>Nächste Verbindung suchen</a></p>
      `);
      marker.addTo(overlays.ktn);
    }
  }
})
```

Abbildung 6: Marker für die Skigebiete erstellen und Abhängigkeit von Layers definieren

```
// Overlays für die Themen zum Ein- und Ausschalten definieren
```

```
let overlays = {
  at: L.featureGroup(),
  ktn: L.featureGroup(),
  noe: L.featureGroup(),
  ooe: L.featureGroup(),
  sbg: L.featureGroup(),
  stmk: L.featureGroup(),
  tir: L.featureGroup(),
  vbg: L.featureGroup()
};
```



```
// Overlays zur Layer-Control hinzufügen
```

```
let layerControl = L.control.layers({
  "ganz Österreich": overlays.at,
  "Kärnten": overlays.ktn,
  "Niederösterreich": overlays.noe,
  "Oberösterreich": overlays.ooe,
  "Salzburg": overlays.sbg,
  "Steiermark": overlays.stmk,
  "Tirol": overlays.tir,
  "Vorarlberg": overlays.vbg
})
.addTo(map);
overlays.at.addTo(map);
```

Abbildung 8: Overlay je Bundesland hinzufügen

```
let miniMap = new L.Control.MiniMap(
  L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png'), {
    toggleDisplay: true,
    minimized: false,
    zoomLevelOffset: -4,
    width: 100,
    height: 100
  }
).addTo(map);
// source: https://github.com/Norkart/Leaflet-MiniMap
```

Abbildung 7: Minimap der Karte hinzufügen

4. Style der Seiten an die von POW anpassen

- Logo und Bilder einfügen
 - Das Pow AT Logo konnte über ein div Element in den html Seiten eingebaut werden und ist über a href="" zur Website von POW AT verlinkt.
 - Die header-Bilder stammen von der Opensource-Plattform Pixabay, wurden mit Picasa auf 1280x365 zugeschnitten und in CSS über eine media-query an verschiedene Anzeigegrößen der Website angepasst. Die Quellen und Urheber der Bilder sind ebenfalls verlinkt. Deren Erscheinungsbild wurde über die Klasse class="white" in CSS gestylt.
- Schrift und Farben wählen
 - Zunächst wurde mit Hilfe des Inspector-Tools die POW AT Seite untersucht, dadurch konnten wir die passenden Farben für unsere Seiten festlegen.
 - In CSS wurden damit der Style für Schriftgrößen, Schriftfarben, Abstände und so weiter festgelegt.
- Buttons, Icons und Hover anpassen
 - Mithilfe von :hover wurde es möglich, den Hintergrund bei Links und Buttons beim Darüberfahren gelb zu färben.
 - Icons wurden auf FontAwesome gesucht und dann mit Hilfe von <i class=""></i> eingefügt. Damit dies funktioniert musste vorher im HTML das „stylesheet“ <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css" /> eingefügt werden
- Favicon einbauen
 - Um dasselbe Symbol in der Taskleiste wie auf der POW Seite angezeigt zu bekommen, wurde das Favicon eingebaut.

```
<link rel="icon"
href="https://protectourwinters.at/wp-content/uploads/2020/09/cropped-android-chrome-512x512-1-32x32.png"
sizes="32x32" />
```

Abbildung 9: Favicon hinzufügen

5. Responsives Navigationselement

Nach einem Beispiel von wiki.selfhtml.org wurde ein responsives Navigationselement eingebaut, das auf größeren Geräten horizontal angeordnet wird und auf kleineren Geräten untereinander. Zunächst wurde eine Liste in html angelegt und dann die passenden Style Element in CSS.

```
<nav>
  <ul>
    <li aria-current="page"><a href="https://train2pow.github.io/was-ist-pow.html">Was ist POW?</a>
    <li><a href="https://train2pow.github.io/warum-train2pow.html">Warum TRAIN2POW?</a>
    </li>
    <li><a href="https://train2pow.github.io/co2-neutral-ins-skigebiet.html">CO2-neutral ins Skigebiet</a>
    </li>
  </ul>
</nav>
```

Abbildung 10: nav-Element für die Erstellung des Navigationsmenüs

In HTML wurde eine Auflistung erstellt, da diese Struktur gibt und die Elemente `ul`, `li` und `a` leicht in CSS gestylt werden können. Die Auflistung wird in ein `nav`-Element eingebettet, damit es unabhängig von anderen Aufzählungen und Verweisen formatiert werden kann.

Schritt für Schritt Erklärung Styling des nav-Elements in CS

```
nav ul {  
  margin: 0;  
  padding: 0;  
  display: flex;  
  flex-direction: column;  
}
```

Durch `display: flex;` wird `nav ul` zu einem „Flex-Container, in dem die Kindelemente durch `flex-direction: column;` untereinander angeordnet werden“ (wiki.selfhtml.org).

Abbildung 11: Erstellung eines Flex-Containers für `nav ul`

```
nav li {  
  list-style: none;  
  margin: 0.5em;  
  padding: 0;  
  font-size: 1.5em;  
  color: #fff
```

Die Schriftgröße wird für `nav li` auf `1.5 em` gesetzt, damit die Navigation auch auf kleineren Displays gut lesbar ist.

Abbildung 12: Setzen der Schriftgröße von `nav li` auf `1.5 em`

```
@media (min-width: 45em) {  
  nav ul {  
    flex-direction: row;  
  }  
  nav li {  
    flex: 1;  
    font-size: 1em;  
  }  
}
```

Mit Hilfe einer media query wurde für größere Displays ab `45em` Breite eine andere Formatierung festgelegt. Auf die horizontale Darstellung wird mit `flex-direction: row;` gewechselt und `nav li` erhält in diesem Fall wieder eine normale Größe.

Abbildung 13: Erstellung media query für größere Displays

```
nav a {  
  display: block;  
  padding: 0.4em;  
  text-decoration: none;  
  font-weight: bold;  
  text-align: center;  
  border: 1px solid #234b64;  
  color: yellow;  
  transition: all .25s ease-in;  
}
```

Mit `display: block;` kann das Linkelement die gesamte zur Verfügung stehende Breite einnehmen. Wird ein Element ausgewählt/ drüber gehovert ändert sich die Farbe durch einen Rollovereffekt. Dies geschieht mit `0.25` Sekunden Verzögerung, um einen Nachglüh-Effekt zu erzeugen.

Abbildung 14: Anpassen des `nav` Elements auf flexible Breite und Erstellen des Nachglüheffekts

```
nav li[aria-current] a {  
  color: yellow;  
  background-color: #234b64;  
}
```

Abbildung 15: Farbwechsel für Anzeige der aktuellen Seite

Außerdem wurde noch über `[aria-current]` festgelegt, dass die Schaltfläche der aktuellen Seite in einer anderen Farbe dargestellt werden soll.

```
nav a:focus,  
nav a:hover,  
nav li[aria-current] a:focus,  
nav li[aria-current] a:hover {  
  color: #234b64;  
  background-color: yellow;  
}
```

Abbildung 16: Erstellen des Rollovereffekts

Die oben erwähnte Farbänderung der Schaltflächen durch den Rollovereffekt kann mit `a:focus`, und `a:hover`, erzeugt werden.

6. Kontaktformular einfügen

Um die Webseite interaktiver zu gestalten, wurde die Möglichkeit, eigene Skigebiete vorzuschlagen, eingebaut. Auf diese Weise kann die Karte mit den Skigebieten auch zukünftig noch wachsen. Das ausgefüllte Formular wird an uns per Mail weitergeleitet, sodass wir die Skigebiete ergänzen können.

Du kennst noch weitere Skigebiete, die gut mit Öffis erreichbar sind?

Dann lass es uns einfach wissen, indem du das nachfolgende Formular ausfüllst. Ein Community-Projekt wie dieses lebt von den Inputs aus der Community, daher freuen wir uns über Hinweise!
Dein Name

Deine E-Mail-Adresse

Das mit Öffis gut erreichbare Skigebiet, das dir auf unserer Karte noch fehlt

Das Bundesland, in dem das Skigebiet liegt

Link zur Website des Skigebiets

☐ Ich bestätige, dass ich mit der Verarbeitung meiner Eingaben zum Zwecke der Anfragebearbeitung, einverstanden bin.

Absenden

Abbildung 17: Kontaktformular auf der Webseite

Die Vorlage für das Formular stammt von form.taxi. Der HTML-Code kann einfach von der Seite übernommen werden. Außerdem wird ein CSS-Dokument zum Download bereitgestellt, beides kann individuell angepasst werden. Unten findet sich ein Screenshot von der Anwendung in HTML. Der Style des Formulars wurde mithilfe von form.css an den POW-Style angepasst.

```
<form action="mailto:anna@protectourwinters.at" id="ft-form" method="POST" accept-charset="UTF-8">
  <fieldset>
    <label>
      Dein Name
      <input type="text" name="Name" required>
    </label>
    <label>
      Deine E-Mail-Adresse
      <input type="email" name="E-Mail" required>
    </label>
    <label>
      Das mit Öffis gut erreichbare Skigebiet, das dir auf unserer Karte noch fehlt
      <textarea rows="1" name="Nachricht" required></textarea>
    </label>
    <label>
      Das Bundesland, in dem das Skigebiet liegt
      <textarea rows="1" name="Nachricht" required></textarea>
    </label>
    <label>
      Link zur Website des Skigebiets
      <textarea rows="1" name="Nachricht" required></textarea>
    </label>
    <div>
      <label>
        <input type="checkbox" name="Datenverarbeitung bestätigt" value="Ja" required>
        Ich bestätige, dass ich mit der Verarbeitung meiner Eingaben zum Zwecke der
        Anfragebearbeitung, einverstanden bin.
      </label>
    </div>
  </fieldset>
  <div class="btns">
    <input type="text" name="_gotcha" value="" style="display:none;">
    <input type="submit" value="Absenden">
  </div>
</form>
```

Abbildung 18: Kontaktformular in HTML einbauen

Literaturverzeichnis:

FontAwesome

Stylesheet, <https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css>, zuletzt aufgerufen am 15.06.21

Leaflet Stylesheet, <https://unpkg.com/leaflet@1.7.1/dist/leaflet.css>, zuletzt aufgerufen am 15.06.21 und <https://unpkg.com/leaflet@1.7.1/dist/leaflet.js>, zuletzt aufgerufen am 15.06.21

Leaflet

Leaflet Providers, <https://cdnjs.cloudflare.com/ajax/libs/leaflet-providers/1.12.0/leaflet-providers.min.js>, zuletzt aufgerufen am 15.06.21

Leaflet hash, <https://cdnjs.cloudflare.com/ajax/libs/leaflet-hash/0.2.1/leaflet-hash.js>, zuletzt aufgerufen am 15.06.21

Leaflet Fullscreen, <https://cdnjs.cloudflare.com/ajax/libs/leaflet.fullscreen/2.0.0/Control.FullScreen.css>, zuletzt aufgerufen am 15.06.21

<https://cdnjs.cloudflare.com/ajax/libs/leaflet.fullscreen/2.0.0/Control.FullScreen.js>, zuletzt aufgerufen am 15.06.21

Leaflet Minimap, <https://cdnjs.cloudflare.com/ajax/libs/leaflet-minimap/3.6.1/Control.MiniMap.css>, zuletzt aufgerufen am 15.06.21

<https://cdnjs.cloudflare.com/ajax/libs/leaflet-minimap/3.6.1/Control.MiniMap.js>, zuletzt aufgerufen am 15.06.21

Formular

form.taxi, HTML Vorlage Kontaktformular, <https://form.taxi/de/sample-forms/contact-form>, zuletzt aufgerufen am 15.06.21

Bilder und Icons

Protect our winters Austria, pow main Logo, <https://protectourwinters.at/wp-content/uploads/2020/09/pow-main-logo.svg>, zuletzt aufgerufen am 15.06.21

Cropped Android Chrome, <https://protectourwinters.at/wp-content/uploads/2020/09/cropped-android-chrome-512x512-1-32x32.png>, zuletzt aufgerufen am 15.06.21

Pixabay, <https://pixabay.com/de/photos/rau-horn-alpine-tannheimer-berge-2146181/>, zuletzt aufgerufen am 15.06.21

<https://pixabay.com/de/photos/panorama-fr%C3%BCh-%C3%BCber-den-wolken-3241426/>, zuletzt aufgerufen am 15.06.21

<https://pixabay.com/de/photos/skifahren-schnee-winter-k%C3%A4lte-4835024/>, zuletzt aufgerufen am 15.06.21

Weblinks

Scotty Webservices - die ÖBB-Fahrplanauskunft für Ihren Webauftritt, <https://fahrplan.oebb.at/bin/query.exe/dn?application=INPUTGEN&genType=LINK>, zuletzt aufgerufen am 15.06.21

Protect our winters Austria, <https://protectourwinters.at/>, zuletzt aufgerufen am 15.06.21

Zusatzlinks

<https://kombitickets.railtours.at/de/wintersport-kombitickets>

ÖBB Rail Tours, Kombitickets Austria, <https://kombitickets.railtours.at/de/kombitickets-austria>, zuletzt aufgerufen am 15.06.21

ÖBB, Haus-Haus-Gepäck, <https://www.oebb.at/de/reiseplanung-services/vor-ihrer-reise/haus-haus-gepaeck>, zuletzt aufgerufen am 15.06.21

railaxed, Mit der Bahn auf die Piste, <https://www.railaxed.at/de/reisetipps/lifestyle/mit-der-bahn-auf-die-piste>, zuletzt aufgerufen am 15.06.21

Literatur

Protect our winters Austria, Nachhaltige Mobilität, <https://protectourwinters.at/mission/themen-und-ziele/electrifying-transit/>, zuletzt aufgerufen am 15.06.21

5 Tipps wie ein Skitag ohne Auto zum Hit wird, <https://protectourwinters.at/5-tipps-wie-ein-skitag-ohne-auto-zum-hit-wird/>, zuletzt aufgerufen am 15.06.21

Der Spiegel, Adrian Meyer: Darf man in Zeiten des Klimawandels noch Skifahren?, <https://www.spiegel.de/wissenschaft/natur/skifahren-gibt-es-nachhaltigen-wintersport-a-1302413.html>, zuletzt aufgerufen am 15.06.21

Alpine Pearls, Nachhaltig Skifahren – aber wie?, <https://www.alpine-pearls.com/aktuelles/newsblog/detail/nachhaltig-skifahren-aber-wie/>, zuletzt aufgerufen am 15.06.21