

Master-Studiengang:

Globaler Wandel – regionale Nachhaltigkeit

Institut für Geographie

Leopold-Franzens-Universität Innsbruck

Projektbericht

Web mapping Projekt “Mapping project UBEVM”

Konzept: <http://mappingprojectuebvm.github.io/konzept.pdf>

Bericht: <http://mappingprojectuebvm.github.io/bericht.pdf>

Webseite: <http://mappingprojectuebvm.github.io/>

VU Geoinformatik: Web mapping

Sommersemester 2021

Dozenten:

Klaus Förster, BSc.

Mag. Bernhard Öggl

Bearbeiterinnen:

Julia Baumgartner, Matr.nr. 00317929

Elisa Kuntner, Matr.nr. 01622085

Innsbruck, am 16.06.2021

Inhalt

Abbildungsverzeichnis	2
1. Einführung (Kurzbeschreibung des Projektes).....	3
1.1. Thema	3
1.2. Beschreibung der Region und Schutzgebiete	3
1.3. Die Problemstellung des Projektes	4
1.4. Aufbau der Website	5
2. Implementierungsschritte & Techniken	6
2.1. Aufbau der HTML-Seiten (index.html, region.html, karte.html)	6
2.2. Aufbau des Stylesheets (main.css).....	10
2.3. Die Kartenseiten und Java-Skripte (region.js & main.js).....	11
2.3.1 Karten.....	11
2.3.2. Plugins	14
3. Resümee.....	17
4. Quellen der Daten & Plugins.....	18
4.1. Kartengrundlagen	18
4.2. Karten-Plugins	18
4.3. Tutorials & Konverter.....	19
4.4. Bilder und Icons	19
4.5. Literatur	20

Abbildungsverzeichnis

Abbildung 1: Header der region.html	7
Abbildung 2: HTML-Skriptteil des Headers	7
Abbildung 3: Quellenangabe mit geöffneten Dropdown-Menü.....	8
Abbildung 4: HTML-Skriptteil der Quellenangaben	8
Abbildung 5: Kontaktformular Teil 1.....	9
Abbildung 6: Kontaktformular Teil 2.....	9
Abbildung 7: Beispiel Stylesheet.....	10
Abbildung 8: Beispiel Responsive Webseite	11
Abbildung 9: Einbindung Kartengrundlagen.....	11
Abbildung 10: Variable ZONE erstellen.....	12
Abbildung 11: For-Schleife Daten einlesen.....	12
Abbildung 12: Zonierung zeichnen	12
Abbildung 13: Zonierung mit Pop-ups als Beschriftung.....	13
Abbildung 14: Kartengrundlagen Übersichtskarte	13
Abbildung 15: Layer-Control Übersichtskarte	13
Abbildung 16: Icons klassifizieren	14
Abbildung 17: Marker positionieren Ergebniskarte.....	14
Abbildung 18: Plugin Maßstab.....	14
Abbildung 19: Plugin Hash	15
Abbildung 20: Plugin Minimap.....	15
Abbildung 21: Plugin Nordpfeil	15
Abbildung 22: Plugin Legende	15
Abbildung 23: Beispiel Legende Stylesheet	16
Abbildung 24: Karte der Forschungsergebnisse	16

1. Einführung (Kurzbeschreibung des Projektes)

1.1. Thema

Wir haben uns im Rahmen der Vertiefungsrichtung „Raumentwicklung und Regionalforschung“ mit dem Thema Partizipation im „UNESCO Biosphärenreservat Engiadina Val Müstair“ (UBEVM) beschäftigt. Um Daten zu unserer Fragestellung zu erheben haben wir Leitfadeninterviews geführt und dabei zusätzlich ein partizipatives Kartentool integriert, in welchem die Teilnehmenden relevante Punkte verorten konnten. Die Ergebnisse und Analysen werden als Poster präsentiert und über einen QR-Code wird die Website abrufbar sein. Einige Ergebnisse des Kartentools und der Interviews werden interaktiv dargestellt. Wir wollen zu Beginn unseres Berichts einen kleinen Einblick in die räumliche Struktur der Region und der Schutzgebiete geben, um dann kurz unsere Problemstellung des Projekts darstellen. Im Anschluss wird die Erstellung der Website erläutert und eine kurze Reflexion rundet den Bericht ab.

1.2. Beschreibung der Region und Schutzgebiete

Das Unterengadin und Münstertal, sind zwei Täler im süd-östlichen Teil des Schweizer Kantons Graubünden, die lediglich durch die Ofenpassstraße miteinander verbunden sind. Auf einer Gesamtfläche von rund 1.200 km² ist die Zahl der Bewohner*innen mit 9.197 recht überschaubar (Bundesamt für Statistik Schweiz, 2021). Und auch wenn die Topographie auf den ersten Blick eine räumliche Trennung der zwei Täler vermuten lässt, so zeigt sich doch bei genauerer Betrachtung, wie eng vernetzt die zwei Täler, wie zum Beispiel im Natur- und Kulturreich, miteinander sind. So erklärt sich schließlich auch die administrative Verbindung der zwei Täler, die sich politisch in der Definition der Region als Verwaltungseinheit äußert und durch die Ausweisung mehrerer sich teils überlagernder Großschutzgebiete in der Region zustande gekommen ist. Der sich in der Region befindende *Schweizerische Nationalpark* (SNP) ist der erste und einzige Nationalpark der Schweiz, der bereits 1914 gegründet wurde (Schweizerischer Nationalpark, 2021b). Hier herrschen strenge Richtlinien zum Schutz der Natur. Der natürliche Lebensraum von Flora und Fauna ist seit Ausweisung des Gebiets komplett frei von menschlichen Eingriffen. Der Nationalpark darf beispielsweise nur über ausgewiesene Wanderwege betreten werden (Schweizerischer Nationalpark, 2021a).

Mit der 1979 zusätzlich erfolgten Ernennung zum UNESCO Biosphärenreservat stand der SNP allerdings zur Jahrtausendwende vor der Herausforderung, den Anforderungen seitens der UNESCO vorläufig nicht mehr gerecht zu werden. Mit der Verabschiedung der Sevilla-Strategie 1995 und des Erlasses des *Man and the Biosphere*-Programmes (MAB) können als Biosphärenreservate nur mehr Gebiete ausgewiesen werden, in denen sich das Paradigma des Naturschutzes „mit und durch den Menschen“ auch räumlich widerspiegelt (Deutsche UNESCO-Kommission, 2021): Ein Biosphärenreservat braucht neben einer Kernzone, in der die Natur unberührt bleibt, auch eine Pflege- und Entwicklungszone, in denen die Bewahrung „charakteristischer Lebensräume von Kulturlandschaften und deren Biodiversität“ (Braun et al., 2020, S. 34) gewährleistet wird und somit die menschliche Nutzung unter Einhaltung der Kriterien der Nachhaltigkeit ausdrücklich erwünscht ist (Braun et al., 2020, S. 36). Um dies auch in der Region Engiadina Bassa Val Müstair umzusetzen und damit das Gütesiegel der UNESCO zu behalten, haben sich der *Schweizerische Nationalpark* und die angrenzenden

Gemeinden Val Müstair und Scuol auf folgende Lösung geeinigt: Das gesamte Gemeindegebiet des Münstertals wurde zum *Regionalen Naturpark* auf Basis des Schweizer Natur- und Heimatschutzgesetzes ernannt, und die Gemeinde Scuol stellte ebenfalls Teile ihres Gemeindegebiets zur Verfügung, um die Zonierung zu ermöglichen. 2010 wurde dem Antrag des Münstertals seitens des Schweizer Bundes stattgegeben, sodass eidgenössische Förderinstrumente durch den neu etablierten *Regionalen Naturpark Biosfera Val Müstair* (RNP) in Anspruch genommen werden können (Braun et al., 2020, S. 40). Schließlich hat die UNESCO 2017 das Gesamtgebiet als Biosphärenreservat anerkannt: Der *Schweizerische Nationalpark* gilt nun nicht mehr nur als Großschutzgebiet eigener Art, sondern ist gleichzeitig Kernzone des *UNESCO Biosphärenreservats Engiadina Val Müstair* (UBEVM). Parallel ist das Münstertal *Regionaler Naturpark* und sein Gemeindegebiet in Teilen Pflege- wie auch Entwicklungszone. Und Teile des Gemeindegebiets von Scuol fließen ebenfalls in die Zonierung der Pflege- und Entwicklungszone ein (Filli & Abderhalden, 2020, S. 256; Schweizerischer Nationalpark, 2021c).

1.3. Die Problemstellung des Projektes

Die bereits erwähnte administrative Vereinigung der zwei Täler Unterengadin und Münstertal über die Ausweisung als Biosphärenreservat wirkt erst einmal sehr komplex, bringt aber aus institutioneller Sicht Vorteile mit sich: Nachdem die UNESCO keine finanziellen Mittel zur Verfügung stellt, kann die Region über die Institution des SNP sowie des RNP Gelder beim Bund lukrieren. Sie sollen helfen, den Anforderungen seitens der UNESCO gerecht zu werden, die örtliche Kulturlandschaft zu erhalten und die Region „in ökonomischer sowie soziokultureller Sicht“ gleichermaßen nachhaltig zu entwickeln (Braun et al., 2020, S. 34). Doch das im Vergleich zu anderen UNESCO Biosphärenreservaten wesentlich umfangreichere institutionelle Konstrukt bringt auch gewisse Hürden mit sich. So ist es für uns Studierende, die wir die Region erst im Zuge unseres Studienprojekts kennenlernen, eine Herausforderung, die regionalspezifische Entwicklung hin zum *UNESCO Biosphärenreservats Engiadina Val Müstair* auf Anhieb nachzuvollziehen. Und uns stellen sich insbesondere auch angesichts aktueller Erweiterungspläne des UBEVM folgende grundsätzliche Fragen: Wie wirken sich diese komplexen Strukturen auf die alltägliche Arbeit im Biosphärenreservat und damit auf dessen langfristige Wirkung in der Region aus? Wie gestaltet sich die Governance des Biosphärenreservats in der Region aus? Handelt es sich hierbei um partizipative und gleichzeitig auch handlungsfähige Strukturen? Unsere Forschungsfrage lautete demnach: *Welche Strukturen der Partizipation am UNESCO Biosphärenreservat Engiadina Val Müstair wurden in den letzten Jahren seit der Planung und Etablierung der Biosphäre etabliert, und inwieweit wirken sich diese auf das Nachhaltigkeitsverständnis der ortsansässigen Bevölkerung und damit auf die Haltung gegenüber den Erweiterungsplänen des Biosphärengiets aus?*

Um Antworten nicht ausschließlich durch Literaturrecherche einen Schritt näher zu kommen, wurden Leitfadeninterviews entwickelt und über ein Videokonferenzprogramm durchgeführt, als auch schriftliche Fragenkataloge verschickt. Aufgrund des ausschließlichen Online-Zugangs zur Region wurden mit den Interviewpartner*innen Informationen in einem digitalen Kartentool verortet, um eine gewisse Vorstellung der Gegebenheiten vor Ort gewinnen zu können. All diese Daten wurden ausgewertet und in diesem Web mapping Projekt dargestellt.

1.4. Aufbau der Website

Alle drei HTML-Seiten, sowie die Java- und CSS-Skripte sind miteinander verlinkt. Gearbeitet wird nur mit Open-Source-Daten und -Programmen, auch bei der Einbindung von Codes bzw. Teilen von Codes anderer User, Bildern und Icons haben wir auf die freie Nutzung geachtet und zitieren diese im Bericht und in den Skripten.

▪ *Über uns*

Für die erste Seite (index.html) haben wir eine Einführungsseite aufgebaut, in der wir unsere Arbeitsgruppe und unser Projekt präsentieren, dazu haben wir Bilder unserer Projektgruppe und ein Titelbild als Banner eingefügt. Für den unteren Bereich der Seite haben wir eine Wordcloud eingebaut, um uns bei allen Beteiligten zu bedanken.

▪ *Region und Biosphärenreservat (UBEVM)*

In diesem Abschnitt (region.html.) stellen wir die Region kurz vor, sowie das Biosphärenreservat und die Struktur der Schutzgebiete. Dabei haben wir eine Übersichtskarte der Region eingebunden, mit der Zonierung des *UNESCO Biosphärenreservats Engiadina Val Müstair*. Nach Rücksprache mit der Geschäftsführerin wurde auch die offizielle Seite des UBEVM verlinkt.

▪ *Forschungsergebnisse*

Auf der dritten Seite (karte.html) haben wir unsere interaktive Karte eingebunden, die die Ergebnisse des partizipativen Kartentools zeigt und deren Inhalte erläutert. Diese als auch die Übersichtskarte werden nach den kartographischen Richtlinien erstellt, daher haben wir eine Legende, einen Maßstab, einen Nordpfeil und die Quellen eingefügt. Zusätzlich haben wir eine kurze schriftliche Erläuterung der Ergebnisse formuliert. Die Kartengrundlage zeigt die Region des Biosphärenreservates und die Karte beinhaltet die Zonierung des UBEVM sowie Pop-ups, welche die wichtigsten Informationen aus den Interviews liefern. Zusätzlich haben wir ein Kontaktformular so umgebaut, dass Interessierte uns weitere Informationen zu den in der Karte dargestellten Themen schicken können. Diese würden dann per Mail versendet werden und können dann noch manuell von uns in die Karte integriert werden.

2. Implementierungsschritte & Techniken

Um in Zeiten von Covid-19 eine Gruppenarbeit erfolgreich erledigen zu können, braucht es eine gute Zusammenarbeit und die Nutzung von unterschiedlichen Tools. In diesem Fall wurde einerseits über „GitHub“ gearbeitet, andererseits haben wir ein Projekt im „Sharepoint“ angelegt, wo wir auch parallel arbeiten konnten, eine Excel-Datei zur Arbeitsplanung war die Grundlage unserer Arbeiten. Zudem haben wir uns auch persönlich getroffen, sobald dies möglich war, und zusammen an den Umsetzungsideen gearbeitet und Probleme besprochen, falls wir irgendwo keine Lösung finden konnten. Wir bauten eine Struktur auf und planten gemeinsam, was wir alles in die Seite implementieren wollten und wie diese aussehen sollte. Die Umsetzung aller Schritte ist in den einzelnen Skripten gut dokumentiert und mit Kommentaren versehen, sodass diese immer nachvollziehbar sind. Nachfolgend werden nun das Vorgehen und der Aufbau der einzelnen Skripte dargestellt.

2.1. Aufbau der HTML-Seiten (index.html, region.html, karte.html)

Zuallererst wurde das HTML-Gerüst gebaut, dazu wurde der *Body* des HTML in einen *Header*, verschiedene *Sections* und *Footer* geteilt. Die *Sections* waren ursprünglich geplant, um diese Seite optisch zu unterteilen, jedoch erwies sich die Seite als so klein bzw. ist es aufgrund der wenigen Inhalten nicht notwendig eine zusätzliche Unterteilung vorzunehmen. Mit einheitlichen Farben und Formatierungen bleiben die Texte besser leserlich. Das Grundgerüst wurde für eine gleichbleibende Optik auf die anderen HTML-Seiten übertragen, ein Stylesheet (main.css) wurde für die grafische Darstellung angelegt. Oberhalb des *Bodys* werden alle nötigen Skripte, wie das Stylesheet und die Java-Skripte in die HTML geladen. Dabei wurden, abhängig von den verwendeten Plugins, verschiedene Skripte in die unterschiedlichen HTML-Files geladen. Eine detaillierte Auflistung aller eingebundenen Skripte ist in den Quellenangaben zu finden. In den beiden Seiten mit implementierten Karten wurden „Leaflet“-Skripte geladen, die für deren Darstellung notwendig sind. „Leaflet“ ist eine freie JavaScript-Bibliothek, welche es ermöglicht webbasierte Karten zu erstellen. Zudem bietet die Seite viele Möglichkeiten Basiskarten und Plugins zu implementieren. Über „Leaflet“ wurden auch die *Basemaps* implementiert. Das letzte Skript sollte das main.js sein, welches mit einem *defer* geladen wird. Damit wird das gesamte Skript ausgeführt, sobald das main.js funktioniert, dies ist sehr hilfreich für eine Fehlerkontrolle. Der *Header* der einzelnen HTML-Seiten verfügt über eine Menüleiste mit den Verlinkungen zu den jeweils anderen beiden HTML-Seiten. Diese Verlinkungen finden sich auch im *Footer* wieder. Zudem ist im **Header ein Bannerbild mit Quelle und die Überschrift** der Seite. Über die Klasse *container* wurde dieser Abschnitt im Stylesheet so formatiert, dass der Titel der jeweiligen Seite im Bild liegt.



Abbildung 1: Header der region.html

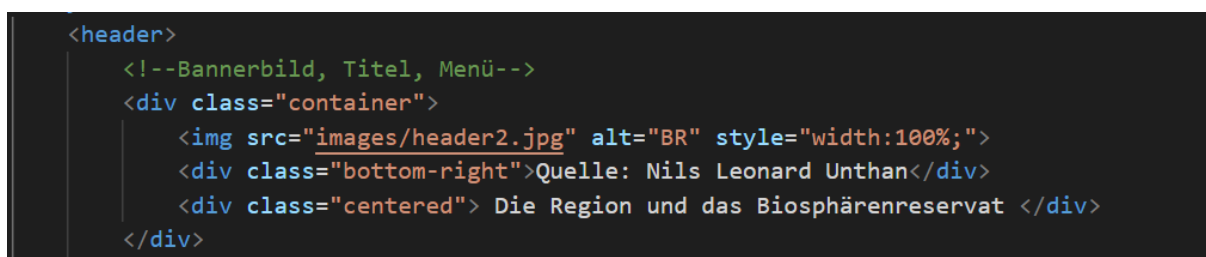


Abbildung 2: HTML-Skriptteil des Headers

In den *Sections* werden die Inhalte der Seite dargestellt: Fotos, Bilder, Texte und Karten. Die Bilder und Fotos wurden auf der frei zugänglichen Seite „RedKetchup“ bearbeitet und auf die richtige Größe gestellt. Die Seite ist ebenfalls eine Open-Source-Seite, aber die eingegebenen Informationen werden mit Google geteilt (siehe <https://redketchup.io/about>). Auf der *region.html* wurde zusätzlich eine **Verlinkung zu einer externen Seite** eingefügt. Dazu wurde ein Pfeil-Icon von „Font Awesome“ verwendet. Für die Verwendung muss man sich bei „Font Awesome“ anmelden, um einen Link zu einem Skript zu bekommen, der auf die HTML-Seite implementiert werden muss, damit anschließend die Icons eingebaut werden können. Am Ende der Seiten integrierten wir wo nötig ein **Quellenverzeichnis als Drop-down-Menü**. Für die einfachere Handhabung soll es per Klick geöffnet und geschlossen werden können. Dies ist vor allem dahingehend sinnvoll, da es sich am Seitenende befindet und im geöffneten Modus ansonsten den *Footer* und die Links zu den anderen Seiten verdecken würde. Die Implementierung erfolgt über das Java-Skript. Dort muss eine Funktion erstellt werden, um das Öffnen und Schließen per Klick zu ermöglichen, dabei wurde eine Online-Anleitung zur Hilfe genommen. Die Vorgehensweise war folgende: es werden mehrere Klassen für das Dropdown definiert, die *onclick*-Funktion wird eingesetzt, um die Information zu öffnen. Diese Texte werden über das Stylesheet gestaltet und die Text-Box wird direkt an der darüberstehenden Karte ausgerichtet, um ein einheitliches Bild zu zeigen. Der Text der Quellen ist verlinkt und führt direkt zu der jeweiligen Online-Ressource der Literatur, um die Suche zu vereinfachen.



Abbildung 3: Quellenangabe mit geöffneten Dropdown-Menü

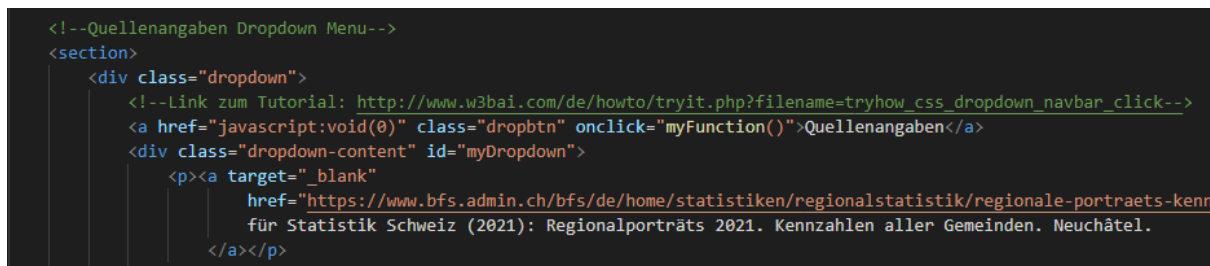


Abbildung 4: HTML-Skriptteil der Quellenangaben

Der *Footer* dient im Fall unserer **Website zur Weiterleitung zu den weiteren Seiten**, wird optisch durch eine Linie abgegrenzt, welche mit dem Tag *hr* erstellt wird. Für eine bessere Lesbarkeit wurde in allen Skripten auch häufig mit dem Tag *br* gearbeitet, um einen Abstand zwischen den einzelnen Elementen und innerhalb von Texten zu erzeugen.

Auf der Seite mit der Ergebniskarte wurde mithilfe von verschiedenen Tutorials bzw. Anleitungen ein **Kontaktformular** konstruiert. Die Besucher*innen der Seite sollen sich selbst einbringen können. Ursprünglich war es geplant, dass sie eigenständig Punkte in die Karte eintragen und kommentieren können, dies würde jedoch den Rahmen des Projektes, vor allem aber die zeitlichen persönlichen Ressourcen sprengen, und möglicherweise zu einer Überladung der Karte führen. Daher entschieden wir uns, dass die Website-Besucher*innen sich über ein Kontaktformular an uns wenden können und wir anschließend manuell die einzelnen Punkte in die Karte eintragen. Damit wird auch verhindert, dass unangemessene Diskussionen auf der Webseite entstehen. Das Kontaktformular beinhaltet unterschiedliche Klassen und div-Tags, wir fügen statt dem *php* die Aktion *mailto* ein, um die eingegebenen Kommentare als Mail zu erhalten. Im Tag *label* steht die Information, die über dem Input-Eingabefeld erscheinen soll. Im *placeholder*, wie der Name schon sagt, steht der Text, der als Platzhalter im Eingabefeld steht.

```
<div class="feedback">
  <div>
    <h2>Wenn Sie noch weitere Kommentare zu den Themen in der Karte haben, können Sie diese hier an uns schicken. Wir
      fügen sie dann in die Karte ein.</h2>
    <h2>Danke für Ihr Interesse!</h2>
  </div>
  <div class="row">
    <div class="column">
      <!--Quelle muss nicht angegeben werden-->
    </div>
    <div class="column">
      <!-- <form action="/action_page.php"> --><!--über ein php wäre eine Alternative Möglichkeit-->
      <form action="mailto:julia1983baumgartner@gmail.com" method="post" enctype="text/plain">
        <!--Eingabefelder -->
        <label for="kommentar">Dies ist ein Kommentar zu ...</label>
        <select id="kommentar" name="Kommentar">
          <option value="Gemeinde">... einer Gemeinde</option>
          <option value="Region">... der Region</option>
          <option value="Schutzgebiet">... einem Schutzgebiet</option>
          <option value="Bereich">... einem Bereich</option>
        </select>
        <label for="fname">Wo auf der Karte soll Ihr Kommentar platziert werden? (Angabe der Koordinaten, Gemeinde,
          Schutzgebiete, Zonierung oder ähnliches)</label>
        <input type="text" id="fname" name="Koordinaten" placeholder="Tragen Sie hier Ihre Information ein.">
        <label for="kategorie">Wählen Sie hier die Kategorie aus, der Sie Ihr Kommentar zuordnen würden.</label>
        <select id="kategorie" name="Kategorie">
          <option value="Vorteile">Vorteile</option>
          <option value="Nachteile">Nachteile</option>
          <option value="Entwickelt">Sehr gut entwickelt</option>
          <option value="Entwicklungsbedarf">Entwicklungspotenzial</option>
          <option value="Kommentar">Allgemeiner Kommentar</option>
        </select>
      </form>
    </div>
  </div>
```

Abbildung 5: Kontaktformular Teil 1

Select steht für die Funktion, eine Dropdownliste auszuwählen, in der *option* werden die Auswahlmöglichkeiten eingetragen. Als dritte Kategorie ist ein Feld für eine freie Texteingabe vorgesehen. *Input* stellt schließlich den Senden-Button dar.

```
    <label for="fname">Wo auf der Karte soll Ihr Kommentar platziert werden? (Angabe der Koordinaten, Gemeinde,
      Schutzgebiete, Zonierung oder ähnliches)</label>
    <input type="text" id="fname" name="Koordinaten" placeholder="Tragen Sie hier Ihre Information ein.">
    <label for="kategorie">Wählen Sie hier die Kategorie aus, der Sie Ihr Kommentar zuordnen würden.</label>
    <select id="kategorie" name="Kategorie">
      <option value="Vorteile">Vorteile</option>
      <option value="Nachteile">Nachteile</option>
      <option value="Entwickelt">Sehr gut entwickelt</option>
      <option value="Entwicklungsbedarf">Entwicklungspotenzial</option>
      <option value="Kommentar">Allgemeiner Kommentar</option>
    </select>
    <label for="subject">Text</label>
    <textarea id="subject" name="Feedback" placeholder="Schreiben Sie hier Ihren Text ...">
      style="height:170px; font-family: Arial"></textarea>
    <input type="submit" value="Senden">
  </form>
</div>
</div>
</div>
```

Abbildung 6: Kontaktformular Teil 2

Im Browser von Google-Chrome wird bei der Eingabe in unserem Formular angezeigt, dass dies keine sichere Verbindung sei. Dieses Problem taucht nur bei Google Chrome auf, da dies die Nutzer*innen vor unsicheren Seiten warnen und schützen will. Dies kann aber die Bereitschaft zum Ausfüllen des Formulars hemmen. Wenn man das Kontaktformular absenden will, wird zudem das Mailprogramm geöffnet, um es darüber zu versenden. Für manche Personen wird dies ein Problem darstellen, da sie somit nicht mehr anonym sind. Leider haben wir auch

für dieses Problem keine passende Lösung gefunden. Das Senden des Formulars über ein *php*-Formular wäre eine Lösung, aber leider scheiterten wir an der Implementierung.

2.2. Aufbau des Stylesheets (main.css)

Die HTML-Seiten wurden einheitlich mit einem CSS-File gestaltet, dabei wurde vorwiegend mit den Tags gearbeitet (Bsp. *header*, *div*, *footer*, *p* usw.), um ein einheitliches Gesamtbild zu gestalten. Für bestimmte Elemente wie *Header*, *Footer* und die Fotoboxen, wurden zusätzliche Klassen erstellt, um dort eine andere Schrift und eine bestimmte Anordnung der Elemente zu erreichen. IDs kommen seltener vor, wurden aber teilweise von anderen Skripten übernommen oder eben für einzelne Elemente von Klassen verwendet, denn die Eigenschaften der IDs überschreiben jene der Klassen und Tags, die Eigenschaften der Klassen jene der Tags. Um die farbliche Gestaltung der Website einheitlich zu machen, wurden die Farben der Boxen und Titel an jene der Bannerbilder angepasst. Dabei haben wir mit dem Tool „Adobe Color Wheel“ gearbeitet, welches online frei zugänglich ist.

```
/*HEADER*/  
/* Container für den Text und das Bild */  
.container {  
  position: relative;  
  text-align: center;  
  color: black;  
}  
/*Bottom right text - Figcapture / Bildquelle*/  
.bottom-right {  
  position: absolute;  
  bottom: 1px;  
  right: 2px;  
  font-size: 12px;  
  text-decoration: bold;  
  padding-left: 2em;  
  padding-top: 0.5em;  
  padding-bottom: 0.5em;  
  padding-right: 2em;  
  color: #b7c4b8;  
}  
/*Schrift Titel*/  
.centered { /* Schrift mit Schatten, versetzt */  
  position: absolute;  
  top: 70%;  
  left: 50%;
```

Abbildung 7: Beispiel Stylesheet

Der Nachteil an der Arbeit mit den Tags ist, dass manche Angaben mehrmals angegeben werden müssen und man aufpassen muss, dass es nicht für Elemente übernommen wird, die anders gestylt werden sollten. Hätte man vermehrt mit Klassen und IDs gearbeitet, wäre dies nicht nötig gewesen. Leider war uns nicht immer von Anfang an klar, was wir alles noch einbauen wollen oder müssen. Damit war das Planen nicht so einfach und es entstanden am Ende sehr viele verschiedene Klassen und IDs.

Laufend verlief die Anpassung der Elemente über *@media* auf andere Bildschirme wie Handys, Tablets, usw. Das Bannerbild soll auch bei großen Bildschirmen bis zum Rand gehen, die Fotoboxen sollten bei kleineren Bildschirmen untereinander sein. Zudem erstellten wir eine Einstellung, dass der Text bei zu großen Bildschirmen nicht zu weit auseinandergezogen wird. Unsere Webseite funktioniert somit auch für kleinere Bildschirme, aber sie ist optimiert für Computerbildschirme, da dort die Handhabung der Karte einfacher ist. Dies ist leider nicht optimal, da in der heutigen Zeit Webseiten meistens über Smartphone/Tablet-Bildschirme betrachtet werden.

```
/*Bildschirmeinstellung Responsive */
@media screen and (max-width:900px) {

  /*HEADER Texte*/
  .centered {
    font-size: 40px;
    position: center;
    list-style: 0;
    width: 100%;
    flex-wrap: wrap;
    line-height: 1;
    top: 50%;
  }

  .bottom-right {
    font-size: 7px;
  }

  /*Bilder*/
  .danke {
    width: 100%;
  };
};
```

Abbildung 8: Beispiel Responsive Webseite

2.3. Die Kartenseiten und Java-Skripte (region.js & main.js)

2.3.1 Karten

Wir haben zwei Karten erstellt: eine Übersichtskarte und eine Ergebniskarte. Dazu wurden aufgrund der Übersichtlichkeit zwei getrennte Java-Skripte erstellt. Die Übersichtskarte der Region bekam in der region.html die ID *overviewmap* und die Karte der Ergebnisse in der karte.html die ID *map*. Damit konnten beide im selben CSS-File gehandelt werden, und über die richtige Verlinkung in den verschiedenen Java-Skripten bearbeitet werden. Bei beiden Karten wurde die *Openstreetmap* als Kartengrundlage verwendet und über das Erstellen einer *map*-Variable eingebunden, sowie bestimmt welcher Ausschnitt der Karte gezeigt werden soll. Die Quellen der Kartengrundlage und aller weiteren Elemente in den Karten wurden im unteren Ende der Karte über die nachfolgend zu sehende *attribution* verlinkt:

```
const map = L.map('map', { //muss Map heißen, weil das DIV-Element dazu im Index.html so definiert ist.
  center: [46.7159,10.2526],
  zoom: 10,
  fullscreenControl: true, //Fullscreen Plugin
});
L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {
  attribution: '&copy; <a target="_blank" href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors, <a target="_blank" href="https://www.geocat.ch/geonetwork/srv/ger/md.viewer#/full_view/c212a19c-6e06-4ca4-bdee-3ec62ccd7b1e/tab/complete">Zonierung</a>, <a target="_blank" href="https://pixabay.com/vectors/arrow-map-north-orienteeing-152596/">Nordpfeil</a>, Icons von <a target="_blank" href="https://icons8.de">Icons8</a>: <a target="_blank" href="https://icons8.de/icon/61876/gef%C3%BCllte-sprechblase">Sprechblase</a>, <a target="_blank" href="https://icons8.de/icon/DigGIRktG1KK/minus">Minus</a>, <a target="_blank" href="https://icons8.de/icon/zddfYtvTqQmM/volle-batterie">Volle Batterie</a>, <a target="_blank" href="https://icons8.de/icon/2d4uwoSaZeK0/batterie-voll-geladen">Batterie geladen</a>'
}).addTo(map);
```

Abbildung 9: Einbindung Kartengrundlagen

Zudem wurde in beide Karten die Zonierung des Biosphärenreservates eingebaut. Diesen Layer erhielten wir als Shape-File auf Anfrage von der Geschäftsführerin des Biosphärenreservates Engiadina Val Müstair. Der Layer wurde über den gratis zugänglichen „Aspose Converter“ in eine GeoJSON-Datei umgewandelt und zugleich wurde das Koordinatensystem auf dasselbe wie bei der *Openstreetmap* (WGS 84) gestellt. Dies funktionierte jedoch leider nicht

richtig, da die Zonierungslinien der *Openstreetmap* nicht ganz genau über jenen der Biosphärenzonierung liegen. Wir versuchten es mit verschiedenen Convertern, das Problem wurde damit zwar besser, die Verschiebung blieb jedoch minimal bestehen. Möglicherweise liegt es an einem Fehler in den Daten.

Um die Zonierung in beide Karten einzubinden, wurde zunächst eine Konstante erstellt.

```
const ZONE = [{  
  title: "Zonierung des UBEVM",  
  data: "data/UBEVM_Zonierung_Perimeter.json",  
}]
```

Abbildung 10: Variable ZONE erstellen

Daraufhin wurde über diese Konstante eine *For*-Schleife erstellt, die über die ganzen Daten läuft und diese einliest.

```
//Forschleife machen, die über die ganzen Geojson-Daten läuft  
for (let config of ZONE) {  
  fetch(config.data)  
    .then(response => response.json()) //innere runde Klammer: Funktionsaufruf, damit es gestartet/ausgeführt wird  
    .then(geojsonData => {  
      //console.log("Data: ", geojsonData); //Immer mal wieder, zum schauen ob es funktioniert oder Fehler gibt.  
      if (config.title == "Zonierung des UBEVM") {  
        drawGeometry(geojsonData);  
      }  
    });  
};
```

Abbildung 11: For-Schleife Daten einlesen

Im nächsten Schritt wurde die Zonierung gezeichnet. Dazu wurde vorher eine weitere Konstante, nämlich *col* erstellt, in der die verschiedenen Farben der Zonierung definiert werden. Diese werden dann hier eingebaut.

```
//Zonierung "zeichnen" + hinzufügen  
let drawGeometry = (geojsonData) => {  
  //console.log("Geometry", geojsonData); //schauen ob es bisher geklappt hat  
  L.geoJson(geojsonData, {  
    style: (feature) => {  
      let col = COLORS[feature.properties.Zone];  
      return {  
        color: col,  
        fillOpacity: 0.2,  
      }  
    }  
  })
```

Abbildung 12: Zonierung zeichnen

Innerhalb dieser Funktion wurde auch festgelegt, dass bei einem Klick auf die Zonierung ein Textfeld aufpoppt, welches die Zone und dessen Gesamtfläche anzeigt:

```
}, //Popups einbinden und beschriften, damit man auf die Zone klicken kann
onEachFeature: (features, layer) => {
  layer.bindPopup(`<strong>Zonierung des UBEVM</strong>
  <hr>
  ${features.properties.Zone || ""}<br>
  Gesamtgröße: ${features.properties.Area_ha || ""} ha
  `);
},
}).addTo(map)
}
```

Abbildung 13: Zonierung mit Pop-ups als Beschriftung

In der Übersichtskarte wurden zusätzlich zu der *Basemap* noch eine Topographische und eine Bildkarte eingefügt. Die Kartengrundlage kann über eine *Layer-Control* gewechselt werden und zudem kann damit in der Übersichtskarte auch die Zonierung aus- und einblendet werden.

```
let baselayers = {
  standard: L.tileLayer.provider("OpenStreetMap.Mapnik"),
  topographie: L.tileLayer.provider("OpenTopoMap"),
  imagery: L.tileLayer.provider("Esri.WorldImagery"),
};
```

Abbildung 14: Kartengrundlagen Übersichtskarte

```
//Kartenhintergründe und Overlays zur Layer-Control hinzufügen
let layerControl = L.control.layers({
  "Standard": baselayers.standard,
  "Topographie": baselayers.topographie,
  "Bildkarte": baselayers.imagery,
},
{ //Klammer erneut innerhalb der runden Klammer öffnen, damit es eine visuelle Abtrennung gibt zwischen den verschiedenen Overlays
  "Zonierung des Biosphärenreservates": overlays.geometry,
}).addTo(map);

overlays.geometry.addTo(map) //Overlays in der Karte anzeigen lassen
```

Abbildung 15: Layer-Control Übersichtskarte

In der Ergebniskarte wurden zusätzlich Pop-ups mit unseren Ergebnissen in Form von kurzen Texten implementiert. Dazu wurde ein weiteres Java-Skript erstellt (*comments.js*), dieses beinhaltet unter anderem die Koordinaten, IDs für die Zuordnung und einen Text, der in den Pop-ups aufscheinen soll. Dafür wurden die Inhalte der Interviews generalisiert, dann nochmal zusammengefasst, mit einer ID versehen und in die *comments.js* Datei übertragen.

Bei der Ergebnisdarstellung war uns wichtig, dass verschiedene Icons eingebaut werden können, um die vier Typen (Vorteil, Nachteil, Entwicklungspotenzial und Gut Entwickelt) voneinander zu unterscheiden. Dazu wurde zuallererst eine Variable namens *Icons* erstellt, in der den IDs der Texte Icon-Bildern zugeordnet werden. Gleichzeitig wurden geeignete Icons auf der Seite „icons8.de“ gesucht, die auch ein passendes Dateiformat hatten und farblich aneinander angepasst. Anschließend wurde eine *For*-Schleife verwendet, in welcher eine *If*-Abfrage eingebunden ist, die abfragt bei welcher ID welches Icon verwendet werden soll.

```
console.log(COMMENTS) //Schauen ob die Verknüpfung funktioniert und etwas in der Console gelogged wird
for (let entry of COMMENTS) { // "entry" wird hier als Begriff für die einzelnen Variablen bzw Keys in COMMENTS definiert
//wählen wann welches Icon zum Einsatz kommt:
  if (icons[entry.id == "Entwickelt"]) {}
  if (icons[entry.id == "Entwicklungsbedarf"]) {}
  if (icons[entry.id == "Vorteil"]) {}
  if (icons[entry.id == "Nachteil"]) {}
  if (icons[entry.id == "keineZuordnung"]) {}
}
```

Abbildung 16: Icons klassifizieren

Im Anschluss daran wird eine Marker-Variable *mrk* erstellt, in welcher definiert wird, wie die Marker aussehen sollen, deren Position und Größe, sowie welche Inhalte im zugehörigen Pop-up aufscheinen sollen. *Entry* ist hierbei wieder eine stellvertretende Variable, über die auf die Variablen in der Datei *comments.js* zugegriffen wird.

```
//Icons einsetzen:
let mrk = L.marker([entry.lat, entry.lng], {
  icon: L.icon({
    iconUrl: `icons/${icons[entry.id]}`,
    iconSize: [37, 37], //Array mit Höhe u Breite, kann ich im img anschauen. Mit der Size ist das Icon genau mittig auf den
    Koordinatenpunkt, passend für unsere Karte.
    popupAnchor: [0, -19], //Damit das Popup am ende des Icons aufploppt und nicht in der Mitte und die Hälfte überdeckt, verschieben
    wir es um die Hälfte ca. nach oben.
  })
}).addTo(map);
mrk.bindPopup(`
  <strong> ${entry.ort}</strong>
  <hr>${entry.txt}
`);
}
```

Abbildung 17: Marker positionieren Ergebniskarte

2.3.2. Plugins

Die meisten Plugins wurden über „Leaflet“ geladen und implementiert. Diese frei zugängliche JavaScript-Bibliothek ist speziell für Websites erstellt worden, die interaktive Karten einbinden möchten. Die Arbeitsschritte sind gut dokumentiert und die Codes können frei übernommen werden. Laufend werden die Codes von den Usern aktualisiert und weiterentwickelt. Wir verwenden für beide Karten folgende Plugins:

Plugin Fullscreen: Dieses Plugin erstellt einen Button im linken oberen Eck der Karte, womit die Karte auf Vollbild gestellt werden kann. Dazu muss ein Skript ins HTML-File geladen werden, sowie bei der Definition der Variable *map* die *fullscreenControl* auf *true* gestellt werden (siehe Abbildung 9).

Maßstab: Bei beiden Karten wurde ein Maßstab über eine *Layer-Control* implementiert.

```
//Maßstab siehe: https://leafletjs.com/reference-1.7.1.html#control-scale
L.control.scale({
  imperial: false
}).addTo(map)
```

Abbildung 18: Plugin Maßstab

Plugin Hash: Dieses Plugin zeigt das Zoom und die Koordinaten der Karte in der URL-Leiste an, in der sich der User gerade befindet.


```
// hash: zeigt in der URL-Leiste die Koordinaten und das #Zoom des Kartenausschnittes an  
var hash = new L.Hash(map); //var hash steht hier nur weil man eine neue Variable erstellt, falls  
man sie später noch mal braucht.
```

Abbildung 19: Plugin Hash

Plugin Minimap: Dieses Plugin ermöglicht es eine kleine Übersichtskarte in die rechte untere Ecke der Karte zu platzieren und eine ausgewählte Kartengrundlage in einer vergrößerten Form darzustellen. Mit der Einstellung *toggleDisplay* wird die Minimap für den User der Karte ein- und ausklappbar. Die Startposition der Ansicht wird über das *minimized* definiert.

```
//Minimap  
var miniMap = new L.Control.Minimap(L.tileLayer("https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png"), {  
  toggleDisplay: true, //Minimap wird damit ein- und ausklappbar  
  minimized: true //mit true zeigt es die Minimap im eingeklappten Zustand an.  
}).addTo(map);
```

Abbildung 20: Plugin Minimap

Plugin Nordpfeil: Das Bild für den Nordpfeil wurde von „Pixabay“ heruntergeladen, im Ordner gesichert und dann über den untenstehenden Code in das main.js und das uebersicht.js eingefügt. Der Code wurde von einem Forum auf der Seite „StackOverflow“ übernommen, da sich dieser als funktional erwies. Die Quelle des Bildes wurde in dem Code an unseren Ordner angepasst. Die Onlineresource des Bildes wird am Fußende der Karte verlinkt.

```
//Nordpfeil einfügen  
var north = L.control({  
  position: "bottomleft"  
});  
north.onAdd = function (map) {  
  var div = L.DomUtil.create("div", "info legend");  
  div.innerHTML = '';  
  return div;  
}  
north.addTo(map);
```

Abbildung 21: Plugin Nordpfeil

Plugin Legende: Die Legende wurde, wie der Maßstab, über eine *Layer-Control* implementiert, der dazugehörige Code ist von einem User aus einem Forum übernommen und dementsprechend angepasst worden. Die Position wurde auf *bottomleft* gestellt und für die Icons haben wir statt dem ursprünglichen *i style=" " die Bilder der Icons (Vorteil, Nachteil usw.) und Farben für die Zonierung verlinkt. Über main.css wurde die Beschriftung verändert.*

```
//Legende:  
var legend = L.control({  
  position: "bottomleft",  
});  
  
legend.onAdd = function (map) {  
  var div = L.DomUtil.create("div", "legend");  
  div.innerHTML += "<h4>Ergebnisse</h4>";  
  div.innerHTML += '<img src=icons/vorteil.png class="imglegend mr-L"><span id=textlegend>Vorteile</span><br>';  
  div.innerHTML += '<img src=icons/nachteil.png class="imglegend mr-L"><span id=textlegend>Nachteile</span><br>';  
  div.innerHTML += '<img src=icons/entwickelt.png class="imglegend mr-L"><span id=textlegend>Sehr Gut Entwickelt</span><br>';  
  div.innerHTML += '<img src=icons/entwicklungsbedarf.png class="imglegend mr-L"><span id=textlegend>Entwicklungspotenzial</span><br>';  
  div.innerHTML += '<img src=icons/comment-map-icon.png class="imglegend mr-L"><span id=textlegend>Kommentare</span><br>';  
  div.innerHTML += '<i style="background: darkgreen"></i><span>RNP</span><br>';  
  div.innerHTML += '<i style="background: darkred"></i><span>SNP</span><br>';  
  div.innerHTML += '<i style="background: orange"></i><span>UEBVM</span><br>';  
  
  return div;  
};  
legend.addTo(map);
```

Abbildung 22: Plugin Legende

```
.legend {  
  padding: 6px 8px;  
  font: 14px Arial, Helvetica, sans-serif;  
  background: white;  
  background: rgba(255, 255, 255, 0.8);  
  line-height: 24px;  
  color: #555;  
}  
  
.legend h4 {  
  text-align: left;  
  font-size: 16px;  
  margin: 2px 12px 8px;  
  color: #777;  
  margin-left: 0.1em;  
  padding-left: 0;  
}  
  
.legend span {  
  position: relative;  
  bottom: 3px;  
  text-align: center;  
}
```

Abbildung 23: Beispiel Legende Stylesheet

In der Abbildung 24 ist die fertige Karte der Forschungsergebnisse zu sehen mit allen implementierten Plugins.

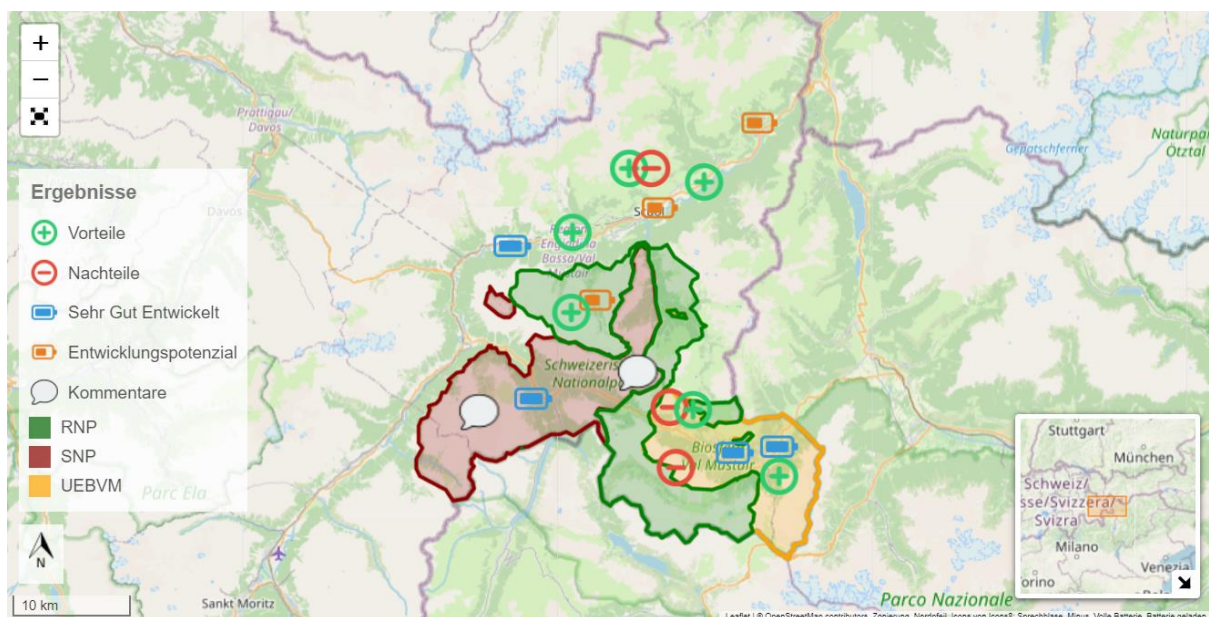


Abbildung 24: Karte der Forschungsergebnisse

Die Quellen aller Kartengrundlagen, Tutorials, Plugins, sowie der notwendigen Skripte, die im HTML verknüpft werden müssen, sind im Quellenverzeichnis zu finden.

Die Website wird auch nach Abschluss der Lehrveranstaltung noch weiterbearbeitet werden, da die Seite für externe Personen erst ab der Posterpräsentation zugänglich sein wird. Davor werden noch ein paar Korrekturschleifen stattfinden und diverse Inhalte, vor allem Texte, noch angepasst werden.

3. Resümee

Bei der Umsetzung der Ideen starteten wir bei den wichtigsten Schritten, also alles, was technisch funktionieren muss, während einige Plugins zusätzlich bzw. optional geplant waren. Außerdem war es eine Herausforderung das Design der Website von Vorneherein festzulegen. Dies liegt vor allem an den vielen unterschiedlichen Möglichkeiten und wir haben uns von Zeit zu Zeit in Designthemen "verfranzt". Dass das Stylesheet nicht so gut strukturiert ist, wie anfangs gehofft, lässt sich auch auf diese Punkte zurückführen. Es verfügt über viele Klassen, die man zusammenfassen hätte können, wenn man von Anfang an gewusst hätte ob und wie die einzelnen Plugins zu implementieren und zu designen sind. Zudem können zu viele Klassen zu Verwechslungen führen. Aber ein gut durchkommentiertes Skript ist hilfreich, um dennoch den Überblick zu bewahren.

Und nachdem wir erfolgreich unsere erste eigene Webseite gebaut haben, wissen wir, dass ein solches Projekt inklusive Design gleich von Anfang an besser strukturiert werden sollte. Auch optische Gestaltungen sollten eher am Ende des Projekts eingeplant werden. Viel Zeit geht in das Design und optische Elemente, man verliert sich in Kleinigkeiten wie Farben, Icons, Bilder und so weiter. Zu bemerken ist aber, dass die Seite öffentlich zu sehen sein wird und zusätzlich an die Dozenten von Raum und Region und die Interviewpartner*innen verschickt werden wird. Da unsere Webseite bei einer Posterpräsentation auch öffentlich präsentiert werden soll, haben wir mehr Zeit in das Design und die Formulierungen der wissenschaftlichen Texte und Ergebnisse investiert.

Eine weitere Herausforderung war die Fehler (Engl.: *bugs*) zu finden, so verbringt man stundenlang an der Fehlersuche und am Ende war es doch nur ein fehlender Beistrich. Auch ist es manchmal schwierig, die eigenen Vorstellungen umzusetzen, jedoch gibt es zahlreiche Online-Foren, die viel Unterstützung und auch Ideen bieten. Unsere Tutorin war zusätzlich eine gute Hilfe.

Unsere ursprünglichen Pläne, auch Bilder und Fotos in die Karte zu implementieren, erwies sich als nicht notwendig. Weitere Inhalte hätten die Karte eher überladen. Genauso verhielt es sich mit dem Plan, dass die Website-Besucher*innen die Möglichkeit haben sollten, eigenständig Punkte mit Informationen in die Karte hinzufügen zu können. Anhand der Quellen wäre es ein sehr herausforderndes Plugin, zusätzlich könnte es mehrere Probleme verursachen, wenn die Seite von mehreren Personen gestaltet werden kann. Daher reduzierten wir unsere Pläne auf die Verwendung einer Kommentarfunktion, die die Informationen über eine Mailadresse schickt, welche wir händisch in die Karte übertragen.

Die Zusammenarbeit funktionierte recht gut, es wird aber eine regelmäßige und gute Absprache nötig. Wenn die Internetverbindung gerade nicht funktioniert und der Pull nicht angenommen wird, kann dies zu einigen Problemen führen. Zum Glück ist das nur einmal passiert. Persönliche Treffen sind allerdings notwendig, ausschließlich online zu arbeiten ist nicht bei allen Schritten effektiv. Alles in allem war es ein spannendes und lehrreiches Projekt.

Fazit: Web mapping macht Spaß.

4. Quellen der Daten & Plugins

4.1. Kartengrundlagen

Leaflet-Skripte in HTML-File für die Kartendarstellung

Stylesheet:

- <https://unpkg.com/leaflet@1.7.1/dist/leaflet.css>

Java-Skript:

- <https://unpkg.com/leaflet@1.7.1/dist/leaflet.js>

Leaflet Providers für Basemaps

Java-Skript:

- <https://cdnjs.cloudflare.com/ajax/libs/leaflet-providers/1.12.0/leaflet-providers.min.js>

Kartengrundlagen

Quelle:

- <http://leaflet-extras.github.io/leaflet-providers/preview/>

Davon gewählt wurden:

- OpenStreetMap.Mapnik
- OpenTopoMap
- Esri.WorldImagery

Shapefile der Biosphärenreservatszonierung

Quelle:

- Geschäftsführerin des Biosphärenreservats
- Online unter: https://www.geocat.ch/geonetwork/srv/ger/md.vierwer#/full_view/c212a19c-6e06-4ca4-bdee-3ec62ccd7b1e/tab/complete

4.2. Karten-Plugins

Plugin LayerControl

Tutorial:

- <https://leafletjs.com/reference-1.7.1.html#control-layers>

Plugin Fullscreen

Tutorial:

- <https://leafletjs.com/plugins.html#fullscreen-controls>

Stylesheet:

- <https://cdnjs.cloudflare.com/ajax/libs/leaflet.fullscreen/2.0.0/Control.FullScreen.css>

Java-Skript:

- <https://cdnjs.cloudflare.com/ajax/libs/leaflet.fullscreen/2.0.0/Control.FullScreen.js>

Plugin Maßstab

Tutorial:

- <https://leafletjs.com/reference-1.7.1.html#control-scale>

Plugin Hash

Tutorial:

- <https://github.com/mlevans/leaflet-hash>

Java-Skript:

- <https://cdnjs.cloudflare.com/ajax/libs/leaflet-hash/0.2.1/leaflet-hash.js>

Plugin Minimap

Tutorial:

- <https://github.com/Norkart/Leaflet-MiniMap>

Stylesheet:

- <https://cdnjs.cloudflare.com/ajax/libs/leaflet-minimap/3.6.1/Control.MiniMap.css>

Java-Skript:

- <https://cdnjs.cloudflare.com/ajax/libs/leaflet-minimap/3.6.1/Control.MiniMap.js>

Plugin Legende

Tutorial:

- <https://codepen.io/haakseth/pen/KQbjdO>

Plugin Nordpfeil

Tutorial:

- <https://stackoverflow.com/questions/22325460/how-can-i-add-a-north-arrow-to-a-leaflet-js-map>

4.3. Tutorials & Konverter

Kontaktformular

Tutorials:

- https://www.w3schools.com/howto/tryit.asp?filename=tryhow_css_contact_section
- <https://stackoverflow.com/questions/13906990/how-do-i-link-up-the-contact-form>

Quellenverzeichnis

Tutorial:

- http://www.w3bai.com/de/css/tryit.php?filename=trycss_dropdown_navbar

Shapefile zu GeoJSON

Konverter:

- <https://products.aspose.app/gis/conversion/shapefile-to-json>

Bilderkonverter

Konverter:

- <https://redketchup.io>

Adobe Color Wheel

- <https://color.adobe.com/de/create/color-wheel>

4.4. Bilder und Icons

Icons

- <https://icons8.de>
- <https://fontawesome.com>

Nordpfeil

- <https://pixabay.com/vectors/arrow-map-north-orienteeing-152596/>

Fotos

- Über uns: Julia Baumgartner, Elisa Kuntner, Carlotta Schlosser
- Bilder der Region für den Header: Nils Leonard Unthan
- Wordcloud: Eigene Darstellung

Kontaktformular

- <https://pixabay.com/de/illustrations/r%C3%BCckmeldung-gruppe-kommunikation-2044700/>

4.5. Literatur

Bundesamt für Statistik Schweiz (2021). Regionalporträts 2021: Kennzahlen aller Gemeinden. <https://www.bfs.admin.ch/bfs/de/home/statistiken/regionalstatistik/regionale-portraets-kennzahlen/gemeinden.assetdetail.15864450.html> (letzter Aufruf am 24. Mai 2021).

Braun, V., Humer-Gruber, A., Heinrich, K. & H. Job (2020). Synopsis der Biosphere Reserves in Deutschland, Österreich und der Schweiz. In A. Borsdorf, M. Jungmeier, & V. Braun (Hrsg.), Biosphäre 4.0. UNESCO Biosphere Reserves als Modellregionen einer nachhaltigen Entwicklung (S. 33–60).

Filli, F., & A. Abderhalden (2020). Der Weg vom Réserve de Biosphère Parc Suisse zum UNESCO Biosphärenreservat Engiadina Val Müstair: eine basisdemokratische Herausforderung. In: A. Borsdorf, M. Jungmeier, & V. Braun (Hrsg.), Biosphäre 4.0. UNESCO Biosphere Reserves als Modellregionen einer nachhaltigen Entwicklung (S. 255–264).

Schweizerischer Nationalpark (2021a). Besonderheiten. <https://www.nationalpark.ch/de/about/ueber-uns/naturoase/besonderheiten/> (letzter Aufruf am 14. Juni 2021).

Schweizerischer Nationalpark (2021b). Über uns. <https://www.nationalpark.ch/de/about/ueber-uns> (letzter Aufruf am 24. Mai 2021).

Schweizerischer Nationalpark (2021c). UNESCO-Biosphärenreservat Engiadina Val Müstair. <https://www.nationalpark.ch/de/about/ueber-uns/unesco-biosphaerenreservat/> (letzter Aufruf am 14. Juni 2021).