

Leopold-Franzens-Universität Innsbruck

Institut für Geographie

LV.-Nr.: 716409

VU Geoinformatik: Web mapping

22S

LV.-Leitung: Klaus Förster, BSc & Mag. Bernhard Öggl

Abschlussprojekt AlmzentrenTirol –

Bericht

Franz Gatt (11807388)

Franz.Gatt@student.uibk.ac.at

Mario Lehmeyer (12110435)

Mario.Lehmeyer@student.uibk.ac.at

28.06.2022

Projektbeschreibung

Für das Abschlussprojekt der VU Webmapping im Sommersemester 2022 ist von uns das Thema **Almzentren in Tirol** gewählt worden. Das Projekt beinhaltet eine Website als Startseite (index.html) mit Einführung in das Thema (Quelle: <https://www.tirol.gv.at/landwirtschaft-forstwirtschaft/agrар/almwirtschaft-in-tirol/>), Beschreibung des Themas, inklusive Bilder, Quellen und Verlinkung zu unserer zweiten Website, die aus einer interaktiven Leaflet-Karte (karte.html) besteht. Diese zweite Website besteht aus einer Mischung des Wien-Beispiels und des Wettervorhersagen-Beispiels aus dem Kurs. Hier sollen verschiedene Basemap-Layer implementiert werden. Mittels eines Popup-Markers mit Lat-Lng-Koordinaten mit dem Verweis auf die Almzentren sollen diese entsprechend referenziert werden. Der entsprechende Datensatz hierfür kommt von: https://www.data.gv.at/katalog/dataset/land-tirol_almzentrenintiroл. Die Pop-Ups sollen auch das Objekt, Name, Objektbezeichnungen, Gemeindenummer und den Erfassungsmaßstab beinhalten. Des Weiteren sollen Leaflet-Plugins zum Einsatz kommen. Diese sind für die Karte der Rainviewer und Velocity als On-Off-Toggle, Minimap, Fullscreen, Leaflet providers plugin, Markercluster, Maßstab und Suchfunktion der Namen der Almzentren.

Geplante Umsetzung

Als Grundlage für die Umsetzung der Marker bzw. Popups werden die Daten (https://www.data.gv.at/katalog/dataset/land-tirol_almzentrenintiroл) im GeoJSON-Format genutzt. Die Einführung in das Thema oder auch die Beschreibung des Themas wird mittels der Rubrik „Almwirtschaft in Tirol“ der offiziellen Seite des Bundeslandes Tirols umgesetzt (<https://www.tirol.gv.at/landwirtschaft-forstwirtschaft/agrар/almwirtschaft-in-tirol/>). Die interaktive Karte wird mit Hilfe von Leaflet implementiert und die zugehörigen Plugins werden über die Website „cdnjs.com“ in den Code geladen. Als Kartengrundlage dienen verschiedene anwählbare Basemaps, die von folgender Website bezogen werden: <https://leaflet-extras.github.io/leaflet-providers/preview/>. Innerhalb der Karte werden die Marker visuell abgeändert. Hierzu werden die frei verwendbaren Map-Icons von „<https://mapicons.mapsmarker.com/>“ genutzt und farblich mit den Codes von „clrs.cc“ eingefärbt. Für die erste Website mit der Beschreibung und der Einführung des Themas werden lizenzfreie Bilder von „RedKetchup“ und „Pixabay“ verwendet. Bezüglich der

Schriftarten auf beiden Websites wird „fontawesome“ und „Google-fonts“ genutzt. Zuletzt werden die Wettericons über „api.met.no“ implementiert.

Der Aufbau des Projekts erfolgt in zwei HTML-Dateien, zwei CSS-Dateien zur Visualisierung und einer Javascript-Datei zur notwendigen Karteninteraktion.

Implementierungsschritte/Techniken

Wie schon erwähnt, war vor Projektbeginn eine Strukturierung der Schritte notwendig. So war zuallererst die Idee im Raum, gefolgt von der Umsetzung mittels den erlernten Techniken der Ausgangspunkt. Wir haben uns bewusst für ein Projekt entschieden, welches Geojson-Daten über die bereits von uns vertrauten Quellen, anbietet. Wie in Abbildung 1 ersichtlich, fiel die Wahl passenderweise zu unserem Projekt der Almen in Tirol mit data.gv.at zusammen.

The screenshot shows the 'Katalog Almzentren Tirol' page on data.gv.at. The page includes a description of alpine pastures in Tyrol, a list of data formats (HTML, WFS, REST API, SHP, JSON, KML), and a table of contact information. The 'JSON' format is circled in red. The right sidebar contains metadata such as the publishing organization (Land Tirol), category (Land and Forestry), keywords, API link, RSS feeds, and last update date.

Daten und Ressourcen	
Almzentren Tirol	HTML
Almzentren Tirol	WFS
Almzentren Tirol	rest api
Almzentren Tirol	SHP
Almzentren Tirol	JSON
Almzentren Tirol	KML

Metadaten	
Datenverantwortliche Stelle	Abteilung Agrarwirtschaft
Kontaktseite der datenverantwortlichen Stelle	https://www.tirol.gv.at/telefonbuch/bww/organisationseinheit/oe/300095/
Datenverantwortliche Stelle - E-Mailkontakt	agrarwirtschaft@tirol.gv.at
Veröffentlichende Stelle	Land Tirol
Kontaktseite der	https://www.tirol.gv.at/landwirtschaft-forstwirtschaft/agrar/

Abbildung 1: Datenquellen aus data.gv.at (<https://www.data.gv.at/katalog/dataset/41760DF5-1EBF-424C-8251-EA0466BC2EE5>)

Aufbau Struktur des Abschlussprojekts

Nach erfolgreicher Datenbeschaffung und Konzeption des Projekts werden in der Organisation „AlmzentrenTirol“ die erforderlichen Dateien erstellt. So wird für die geojson-Daten ein eigener Ordner data angelegt und die Daten dorthin abgelegt. Das Projekt besteht so (auch u.A. laut Vorgaben) aus folgenden Dateien:

- index.html
- karte.html
- kartemain.css
- main.css
- kartenmain.js
- main.js
- folder (data, icons, wettericons und lib)

index.html

In der index.html-Datei geht es vor allem darum, die Grundstruktur der Willkommenseite zu erstellen. Im <head> werden Metasettings gesetzt, wie etwa der Zeichenformatierung für Westeuropa – UTF8. Desweiteren werden hier diverse Plugins eingebaut. Für unsere Website ist dies ein TabvorschauBild, welches mittels Link verlinkt wird, sowie die Verknüpfung mit dem dazugehörigen Stylsheet, der main.css-Datei. Es wird aber auch die FontAwesome-Funktion über cdnjs eingebaut.

Im <body> Bereich des html-Files wird der Infotext zur Almwirtschaft Tirol angezeigt, sowie auf die Originalquelle verwiesen. Mittels <h1>, <h2> bzw. <h3> Elementen werden Überschriften in verschiedenen Größen eingebaut. Weiters werden die dazugehörigen Links als `unsorted` list angegeben. Im Footer des html-Files wird auf die karte.html-Seite verlinkt, welche die Leaflet-Karte unseres Projekts darstellt. In Abbildung 2 ist der Inhalt des index.html dargestellt.

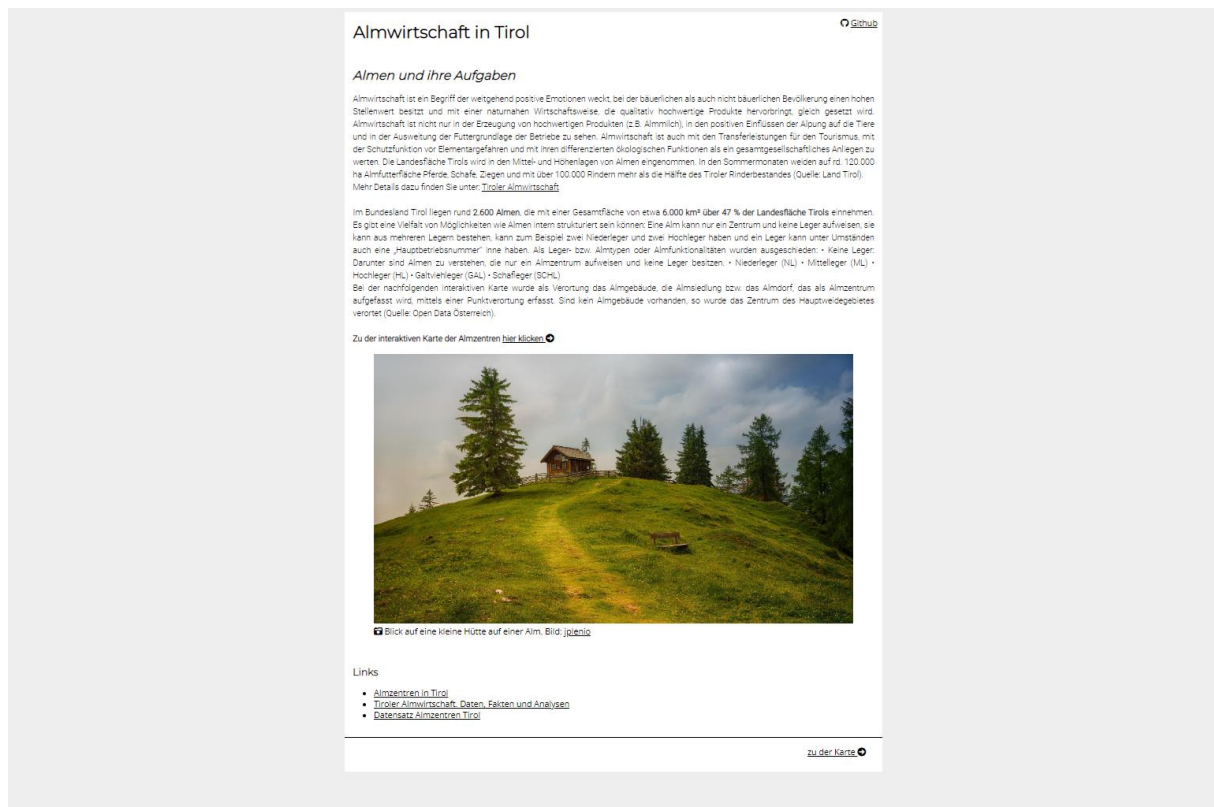


Abbildung 3: index.html im Browser

karte.html

In der karte.html-Datei werden nun die für unser Projekt erforderlichen Leaflet-plugins implementiert. Mittels `<link rel="shortcut icon" href="icons/mountains.png" type="image/png">` wird wieder auf das icon verlinkt und aufgerufen. Auch Font-awesome wird wieder implementiert.

Als nächstes wird das Herzstück der Karte implementiert, die hosted Version von Leaflet, welche über die offizielle Leaflet-Seite kopiert werden kann (Abbildung 4).



an open-source JavaScript library
for mobile-friendly interactive maps

[Overview](#) [Tutorials](#) [Docs](#) [Download](#) [Plugins](#) [Blog](#)

Download Leaflet

Version	Description
Leaflet 1.8.0	Stable version, released on April 18, 2022.
Leaflet 1.7.1	Previous stable version, released on September 3, 2020.
Leaflet 1.8-dev	In-progress version, developed on the main branch.

[View Changelog](#)

Note that the main version can contain incompatible changes, so please read the changelog carefully when upgrading to it.

Using a Hosted Version of Leaflet

The latest stable Leaflet release is available on several CDN's — to start using it straight away, place this in the head of your HTML code:

```
<link rel="stylesheet" href="https://unpkg.com/leaflet@1.8.0/dist/leaflet.css" integrity="sha512-hoalwL9h8Z+o1wuBQ+h7WfzTJ9l8w7ZKgsi2RcLhJJ++svsF04HeZp6BOIDLanUE0wpUQZlh3DDwgAoDgd" />  
<script src="https://unpkg.com/leaflet@1.8.0/dist/leaflet.js" integrity="sha512-BB3hKbD3w+niB58nu+mbIgs9ll+ERoQTzJ1FODzKY9L2yy9Ux838LL+gahPYpT3BbBXZrv4k89+OPKh9vwG0UY5Rz0x" />
```

Note that the [integrity hashes](#) are included for security when using Leaflet from CDN.

Abbildung 4: Leaflet

Über die Website cdjs.com (Abbildung 5) werden weitere, für die Karte benötigte Plugins installiert.

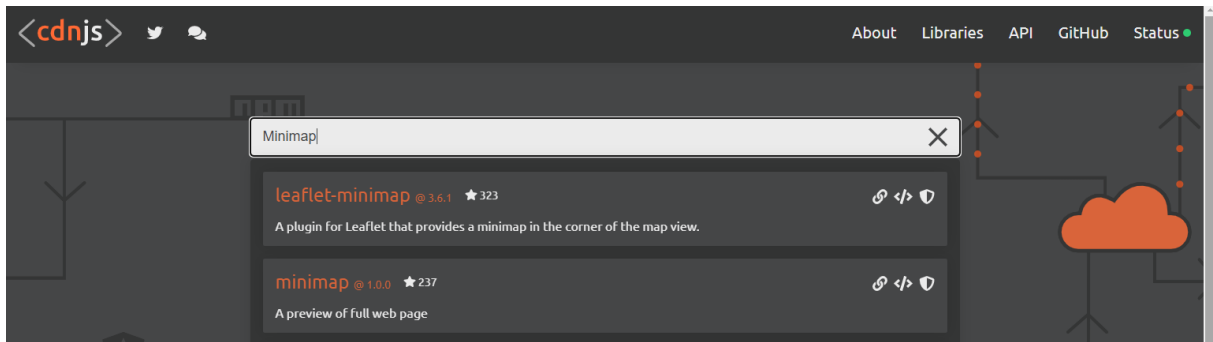


Abbildung 5: cdnjs.com

Diese lauten:

- Leaflet providers plugin (für die verschiedenen Basemap-Layer)
- Leaflet MiniMap (stellt im rechten unteren Eck eine Übersichtskarte dar)
- Leaflet markercluster (clustert die Marker zu einem Kreis mit der Anzahl an Elementen zusammen)

Über andere "Anbieter" werden noch

- Fullscreen plugin (über api.mapbox.com - Button ermöglicht einen Vollbildmodus der Karte)
- Leaflet Velocity (über die js & css -Files verlinkt - über Wetterdaten werden angezeigt)

Implementiert.

Desweiteren besteht das karte.html-File im <body> noch aus der Überschrift „Karte Almtzentren Tirol“ und der Karte von der id map.

[kartemain.js](#)

Hier werden die Funktionen der Karte programmiert. Eine let-Variable wird hier mit den Koordinaten auf Innsbruck definiert. Anschließend wird der Start-Layer (BasemapAT.grau) auch als let Variable definiert und über die Leaflet (L.) Funktion aufgerufen. In der let-Variable map wird nun die Karte initialisiert und mit den entsprechenden Koordinaten für Innsbruck gestartet. Der Zoom-Level auf 12 gibt den Start-„Maßstab“ vor. Dies sieht folgendermaßen aus:

```
let innsbruck = {
  lat: 47.267222,
  lng: 11.392778,
  title: "Innsbruck"
};

let startLayer = L.tileLayer.provider("BasemapAT.grau")

let map = L.map("map", {
  center: [innsbruck.lat, innsbruck.lng],
  zoom: 12,
  layers: [
    startLayer
  ]
});
```

Anschließend werden die Layer für unsere Karte definiert:

```
// Layercontrol
let layerControl = L.control.layers({
  "Basemap Österreich Grau": startLayer,
  "Basemap Österreich Standard": L.tileLayer.provider("BasemapAT.basemap"),
  "Basemap Österreich High-DPI": L.tileLayer.provider("BasemapAT.highdpi"),
  "Basemap Österreich Gelände": L.tileLayer.provider("BasemapAT.terrain"),
```



```

    "Basemap Österreich Oberfläche":
L.tileLayer.provider("BasemapAT.surface"),
    "Basemap Österreich Orthofoto":
L.tileLayer.provider("BasemapAT.orthofoto"),
    "Basemap Österreich Beschriftung":
L.tileLayer.provider("BasemapAT.overlay"),
    "Basemap Österreich mit Orthofoto und Beschriftung": L.layerGroup([
        L.tileLayer.provider("BasemapAT.orthofoto"),
        L.tileLayer.provider("BasemapAT.overlay"),
    ]),
    "Esri World Imagery": L.tileLayer.provider("Esri.WorldImagery"),
    "Esri World Topo Map": L.tileLayer.provider("Esri.WorldTopoMap"),
    "Esri World Street Map": L.tileLayer.provider("Esri.WorldStreetMap")
}).addTo(map);

```

Mit weiteren L.control werden Maßstab, Fullscreen und die Minimap in die Karte mit .addTo(map) eingefügt:

```

// Maßstab hinzufügen
L.control.scale({
    imperial: false,
}).addTo(map);

// Fullscreen hinzufügen
L.control.fullscreen().addTo(map);

// Minimap hinzufügen
let miniMap = new L.Control.MiniMap(
    L.tileLayer.provider("BasemapAT"), {
        toggleDisplay: true
    }
).addTo(map);

```

Nun werden die Marker für die Almzentren erstellt und auf die Karte gebracht. Mittels L.geoJSON werden nun die Geojson-Dateien auf die entsprechende Stelle auf der Karte geladen und in den Popups die Werte Namen, Gemeindenummer, Objektbezeichnung und Erfassungsmaßstab angezeigt. Die async-Funktion wird im Laufe des Projekts durch if-Abfragen ersetzt (mehr dazu später in Herausforderungen im Projekt) Dies sieht im Code wie folgt aus:

```

// Almzentren Marker und Pop-Up mit Beschriftung
async function loadSites(url) {
    let response = await fetch(url);
    let geojson = await response.json();

    let overlay = L.markerClusterGroup();
    layerControl.addOverlay(overlay, "Almzentren");
    overlay.addTo(map);
}

```

```

L.geoJSON(geojson, {
  pointToLayer: function (geoJsonPoint, latlng) {
    let popup = `
      <strong>${geoJsonPoint.properties.NAME}</strong>
      <hr>
      Gemeindenummer: ${geoJsonPoint.properties.GEMNR}<br>
      Objektbezeichnung:
      ${geoJsonPoint.properties.OBJEKTBEZEICHNUNG}<br>
      Erfassungsmaßstab:
      ${geoJsonPoint.properties.ERFASSUNGSMASSTAB}
    `;
    return L.marker(latlng, {
      icon: L.icon({
        iconUrl: "icons/mountains.png",
        iconAnchor: [16, 37],
        popupAnchor: [0, -37]
      })
    }).bindPopup(popup);
  }
}).addTo(overlay);
}
loadSites("https://data-tiris.opendata.arcgis.com/datasets/tiris::almzentren-1.geojson");

```

Ergebnis Karte

Durch die Implementierung der einzelnen Schritte ist bei Kartenaufruf Innsbruck im Zentrum. Einzelne Almen sind bereits in Cluster zu erkennen. Andere, einzelne als Icons. Desweiteren ist bei Start die Wettervorhersage aufgerufen, welche per beliebigen Klick auf einem Punkt in der Karte aktualisiert wird. In Abbildung 6 ist die Karte zu sehen.

Karte Almzentren Tirol

 [GitHub](#)

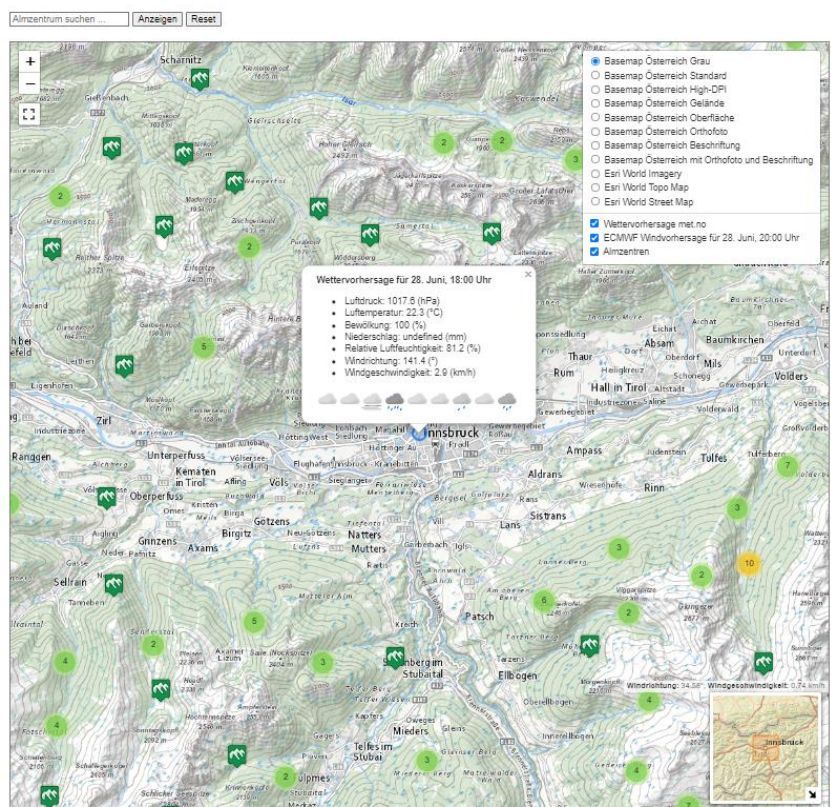


Abbildung 6: Karte beim Start

Bei Klick auf ein Alm-Icon wird dieses mit den entsprechenden Elementen mit dem Javascript-File angezeigt. Hier ein Beispiel der Pröfogl-Alm:

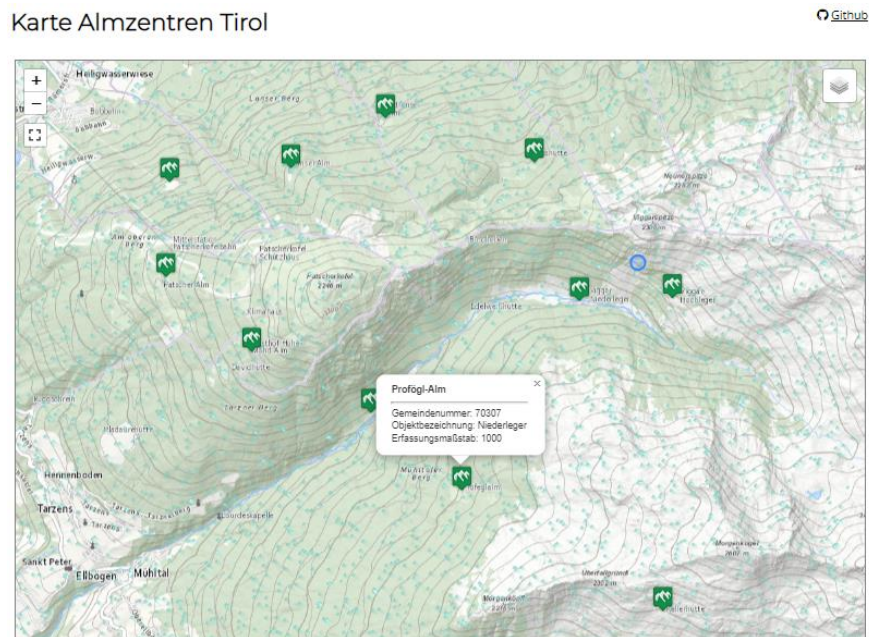


Abbildung 7: Almzentrum

Herausforderungen im Projekt

Grundsätzlich war die größte Schwierigkeit ein geeignetes Projektthema mit entsprechenden Daten zu bekommen. Desweiteren sollte das Projekt aufgrund des zeitlichen Horizonts nicht allzu stark von den Übungsbeispielen aus der VU abweichen, weshalb wir uns für einen Mix aus all den erlernten Praktiken entschieden haben. Wir haben unser Konzept so wie erarbeitet gut umsetzen können. Herausforderung war allerdings noch das Sortieren der Suchfunktion und die lange Ladedauer der knapp 3800 Almen. Für die verkürzte Ladedauer haben wir mit Hilfe folgenden Code implementiert:

```
let option = document.createElement('option');  
  
    option.value = geoJsonPoint.properties.NAME;  
  
    searchList.appendChild(option);
```

Durch die Ersetzung durch den ursprünglich innerHTML-Code konnte so die Ladezeit von knapp 17 Sekunden auf 2 Sekunden reduziert werden.

Beim Sortieren wurde die async-Funktion also mit if-Verzweigungen abgeändert, da für die anonyme Suchfunktion kein boolean-Wert (true/false) akzeptiert werden kann.

```
Beim Sortieren wurde // Almzentren nach Name sortieren
    geojson.features.sort(function (a, b) {
        if (a.properties.NAME.toLowerCase() > b.properties.NAME.toLowerCase())
        {
            return 1;
        }

        if (a.properties.NAME.toLowerCase() < b.properties.NAME.toLowerCase())
        {
            return -1;
        }

        return 0; // wenn die Namen identisch sind
```

Abschließend möchten wir festhalten, dass uns die VU Webmapping zu Beginn nicht viel sagte, aber jetzt am Ende der VU sind wir begeistert, was wir alles in einem Semester lernen konnten. Die Unterstützung während dem Semester war auch top. Hierfür möchten wir uns herzlich bedanken.

Der vollständige und aktuelle Code, sowie das Projekt sind unter dem Link <https://almzentrentirol.github.io/> erreichbar bzw. in unserer Repository AlmzentrenTirol ersichtlich. Dort können auch detaillierte Commits, welche während der Projekterstellung gemacht wurden, nachgesehen werden.

Quellenverzeichnis/Links

Folgende Quellen sind für das Abschlussprojekt verwendet worden:

Abschlussprojekt Almzentren Tirol

- <https://almzentrentirol.github.io/>

Daten

- https://www.data.gv.at/katalog/dataset/land-tirol_almzentrentirol

Funktionelle Seiten

- <https://api.met.no/weatherapi/locationforecast/2.0/documentation>
- <https://cdnjs.com/>
- <https://www.ecmwf.int/en/forecasts/datasets/open-data>
- <https://fontawesome.com/>
- <https://leafletjs.com/>
- <https://leaflet-extras.github.io/leaflet-providers/preview/>
- <https://mapicons.mapsmarker.com/>

Land Tirol

- https://www.tirol.gv.at/fileadmin/themen/land-forstwirtschaft/agrar/zahlen-daten-agrarberichte/TirolerAlmwirtschaft_DatenFaktenAnalysen_DerAlmundBergbauer_Maerz2021.pdf
- <https://www.tirol.gv.at/landwirtschaft-forstwirtschaft/agrar/almwirtschaft-in-tirol/>