

MSc Geographie: Globaler Wandel – regionale Nachhaltigkeit  
**VU Geoinformatik Web mapping**  
716409 | 5 ECTS | SS23

*Klaus Förster*

*Bernd Öggl*

*Projektbericht: Astrophoto Tirol*

**Fabian Wild**

**Christopher Hartung**

Innsbruck, am 24.06.2023

1	Einleitung.....	3
2	Aufbau der Seiten.....	4
2.1	Hauptseite/ Landing Page.....	4
2.1.1	Plugins Hauptseite.....	4
2.2	Seiten der jeweiligen Photospots.....	5
2.2.1	Plugins Nebenseiten.....	5
3	Probleme & Herausforderungen.....	11
4	Diskussion & Fazit.....	<b>Fehler! Textmarke nicht definiert.</b>
5	Literaturverzeichnis.....	12

Abbildung 1:	Ordnerstruktur des Projekts.....	6
Abbildung 2:	Index.html der Hauptseite.....	7
Abbildung 3:	Index.html der Nebenseite - Lichtsee.....	7
Abbildung 4:	Main.js der Hauptseite.....	9
Abbildung 5:	Main.js der Nebenseite – Lichtsee.....	10

## 1 Einleitung

Willkommen bei Astrophoto Tirol, einem aufregenden Projekt, das es sich zum Ziel gesetzt hat, die besten Photospots für die Astrophotographie, oder einfach nur eine Nacht im Freien vorzustellen. Als leidenschaftliche Astrofotografen und Naturliebhaber sind wir von der beeindruckenden Pracht des Sternenhimmels und den unendlichen Möglichkeiten der Fotografie begeistert.

Die Tiroler Alpen bieten eine ideale Umgebung für unser Hobby, da viele Täler weit entfernt von den Lichtverschmutzungen der städtischen Gebiete liegen. Hier haben wir die Möglichkeit, den Nachthimmel in seiner vollen Pracht zu erleben und einzufangen. Sowohl das Stubaital, Zillertal und Ötztal eignen sich daher besonders.

Unser Projekt soll einen Überblick über unsere Lieblingssspots für Astrophotographie in Tirol geben.

Dazu geben wir den Usern verschiedene Tools an die Hand. Zum einen die Lichtverschmutzungskarte von David Lorenz um den Idealen Spot herauszufinden(David Lorenz, 2020). Zum anderen geben wir dem User einige Informationen zum Vorherrschenden Wetter.

Für die Einzelnen Spots haben wir jeweils einen Informellen Text, sowie eine Wanderroute die den Spot mit einbindet erstellt, die der User sich auch herunterladen kann. Dazu gibt es auch eine Wegbeschreibung.

## 2 Aufbau der Seiten

Unser Projekt besteht aus einer Hauptseite die auf insgesamt vier Seiten verlinkt. Im Folgenden wird erst auf die Hauptseite und dann auf die Seite der einzelnen Photospots eingegangen.

### 2.1 Hauptseite/ Landing Page

Unsere Hauptseite gibt eine Einführung in unser Projekt Astrophot Tirol. Nach einem Text folgt eine Kartenapplikation. In dieser Kartenapplikation sind unsere vier Lieblings-Astrophotospots in Tirol mittels Markern dargestellt. Beim Klicken auf die Marker erscheint jeweils der Name des Spots mit einem Vorschaubild. Sowie einer Verlinkung auf die Jeweilige Seite des Spots für weitere Informationen. Die Kartenapplikation beinhaltet auch das Leaflet Rainviewer Plugin. Unten links ist die Minimap dargestellt. Der Benutzer kann anhand der LayerControl aus vier Layern wählen. Eine topographische Karte, sowie eine Satellitenaufnahme; die weiteren Karten sind Lichtverschmutzungskarten für die Jahre 2016 und 2020, die bei Bedarf ausgewählt werden können. Der Nutzer kann außerdem durch das Hovern über das Legendensymbol in der rechten unteren Ecke eine Legende für die Klassen der künstlichen Himmelselligkeit einblenden.

Im weiteren Verlauf der Seite befindet sich ein Fenster, in das ein Bild der Milchstraße als Superzoom implementiert ist. Außerdem wird ein Video eingebettet, das der Nutzer zum Thema Airglow ansehen kann.

#### 2.1.1 Plugins Hauptseite

##### **Leaflet FullScreen**

Das Leaflet Fullscreen Plugin ist leicht zu integrieren und zu verwenden. Es erfordert das Einbinden der Plugin-Dateien in das Webprojekt sowie das Hinzufügen einiger Codezeilen zur Initialisierung der Karte und zur Aktivierung des Vollbildmodus.

Insgesamt ist das Leaflet Fullscreen Plugin eine praktische Erweiterung für Leaflet-basierte Kartenanwendungen, die den Benutzern eine einfache Möglichkeit bietet, die Kartenansicht in den Vollbildmodus zu versetzen und dadurch ein beeindruckendes visuelles Erlebnis zu schaffen (B., 2012/2023).

##### **Leaflet Rainviewer**

Mit dem Leaflet Rainviewer Plugin können Benutzer Niederschlagsradardaten über eine Zeitleiste anzeigen lassen, um den zeitlichen Verlauf des Niederschlags zu verfolgen. Die Daten werden in Form von Overlay-Kacheln auf der Karte dargestellt, wobei verschiedene Farben den Intensitätsgrad des Niederschlags anzeigen (Marcin, 2019/2023).

##### **Leaflet World Minimap**

Das Leaflet World Minimap Plugin stellt eine verkleinerte Version der Weltkarte dar, die den sichtbaren Ausschnitt der Karte in Echtzeit widerspiegelt. Dem Benutzer wird mit Verschieben und Zoomen der Karte der aktuelle Ausschnitt auf der Minimap dargestellt (Friedl, 2022/2022).

##### **Leaflet Legend**

Mit dem Leaflet.Legend Plugin können benutzerdefinierte Legenden erstellt werden, die den individuellen Vorstellungen des Nutzers entsprechen (ptma, 2020).

## 2.2 Seiten der jeweiligen Photospots

Die Webseiten der einzelnen Photospots enthalten Informationen über den jeweiligen Ort. Eine interaktive Karte, welche einen GPX Track darstellt mittels dem Leaflet Elevation Plugin. Ebenfalls werden die Höhenprofile dargestellt und der GPX Track kann heruntergeladen werden. Durch Klicken auf die Karte wird für den angeklickten Ort auf der Karte die Wettervorhersage von MET Norway generiert.

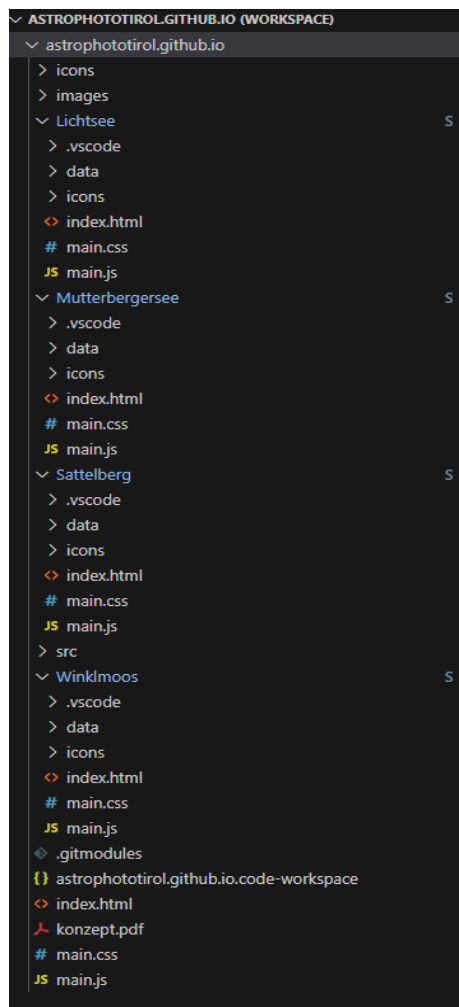
### 2.2.1 Plugins Nebenseiten

#### **Leaflet Elevation**

Eine der Hauptfunktionen des Plugins ist die Erstellung von Höhenprofilen entlang einer bestimmten Strecke. Benutzer können einfach eine Linie auf der Karte zeichnen, und das Plugin generiert automatisch ein Höhenprofil, das die Höhenänderungen entlang dieser Strecke anzeigt. Dies ist besonders nützlich für Wander- oder Fahrradroutenplanung oder für die Analyse von Geländeprofilen. Das Leaflet Elevation Plugin bietet verschiedene Anpassungsoptionen, um die Darstellung der Höhendaten anzupassen. Benutzer können die Auflösung der Höhenlinien, die Farbgebung und das Erscheinungsbild des Höhenprofils anpassen, um die visuelle Darstellung den eigenen Bedürfnissen anzupassen. Darüber hinaus unterstützt das Plugin auch Interaktionen mit den Höhendaten. Benutzer können beispielsweise den Mauszeiger über das Höhenprofil bewegen, um die Höhenwerte an bestimmten Punkten anzuzeigen, oder auf bestimmte Punkte im Höhenprofil klicken, um die entsprechende Position auf der Karte zu markieren (Bache, 2013/2023).

## 3 Vorgehensweise und Beispielcode

### 3.1 Datenstruktur



Aufgrund der hohen Datenmenge wurde beschlossen, eine Ordnerstruktur für das gesamte Projekt anzulegen und zunächst einzelne Repositories als Submodul der Hauptseite in Github zu integrieren, sodass die Handhabung und Übersichtlichkeit besser gewährleistet werden konnte.

Das übergeordnete Repository `astrophotographie.github.io` besitzt demnach drei Ordner ("Icons", "images" und einen Ordner "src", wo die .js und .css Dateien von Plugins lokal gespeichert werden).

Zusätzlich wird im `.gitmodules` die Zuordnung der Submodule festgelegt, was für das Deployment der Webapp auf Github wichtig ist. Standardmäßig besitzt dieses Repository eine `Index.html`, `main.css` und `main.js` Datei.

Die Submodule haben jeweils einen Ordner namens "data", in welchem `.gpx` Tracks und Bilder gespeichert werden. Zusätzlich dazu existiert für jedes Submodul ein Ordner mit den Icons für die Wettervorhersage

Abbildung 1: Ordnerstruktur des Projekts

## 3.2 Beispielcode

Im Folgenden wird für die Hauptseite, sowie anhand eines Beispiels für eine Nebenseite, der Aufbau der Index.html, sowie der main.js Dateien gezeigt.

### 3.2.1 Html Hauptseite

```
<!DOCTYPE html>
<html lang="de">

<head>
</head>

<body>
  <main>
    <header>
    </header>
    <article>...
    </article>
    <div id="map"></div>
    <article>
    </article>
    <article>
    </article>
    <div id="superzoom"></div>
    <article>...
    </article>
    <iframe width="1202" height="720" src="https://www.youtube.com/embed/3E1TgsGoLIM" ...
    </iframe>
    <footer>
    </footer>
  </main>
</body>
</html>
```

Abbildung 2: Index.html der Hauptseite

Abbildung 2 zeigt den Codeaufbau der Index.html der Hauptseite. Es werden zunächst Titel der Seite, sowie Plugins eingebunden. Innerhalb des <body> Tags werden <header> Elemente, Texte (<article>), sowie division-Tags (<div>) für neue eigene Abschnitte verwendet. Auf dieser Seite sind zwei Leaflet Elemente, „map“ und „superzoom“ in jeweils ein <div> Element eingebunden. Am Ende der Seite werden eine Youtube Video über einen <iframe> eingebettet und im Footer die Datenquellen angegeben.

### 3.2.2 Html Nebenseite (Lichtsee)

```
<head>
</head>

<body>
  <main>
    <header>
      <h2><i class="fa-solid fa-camera-retro"></i> Photospot Lichtsee</h2>
      <nav>...
    </nav>
    </header>
    <article>
      <h1><a href=""></a>Spot 2: Lichtsee</h1>
      <img>
      <p>Der Lichtsee liegt im hinteren Obernbergtal. Ausgangspunkt ist der Ort Obernberg am Brenner. Dieser See ...
      </p>
      <div id="map"></div>
      <div id="profile"></div>
      <h3>Streckenbeschreibung</h3>
      <p>Mit dem Bus oder Auto von Innsbruck fährt man etwa 40min bis zum Parkplatz Waldesruh, von hier aus geht
        und später den See erreicht.</p>
    </article>
    <h2 class="w3-center">Dieser Spot ist auch Tags sehr schön!</h2>
    <div class="w3-content w3-display-container">...
    </div>
    <script>
    </script>
    <footer>
    </footer>
  </main>
</body>
</html>
```

Abbildung 3: Index.html der Nebenseite - Lichtsee

Abbildung 3 zeigt die Index.html der Nebenseite am Beispiel Lichtsee. Es werden zunächst Titel und Plugins im <head> Tag implementiert. Nach der Überschrift wird ein Photo für den jeweiligen Spot vom Github Repository abgerufen. Anschließend wird durch zwei <div> Tags ein Abschnitt für eine Karte, sowie ein Fenster für die Höhenprofildarstellung festgelegt. Nach der Streckenbeschreibung wird noch ein carousel Item für weitere Bilder eingebunden. Innerhalb dieses Items kann durch den <script> Tag Javascript Code ausgeführt werden, sodass die komplette Funktion in der Index.html bestehen bleiben kann.



### 3.2.3 main.js Hauptseite

```
/* Übersichtskarte Astropots Tirol */

// Tirol
> let tirol = { ...
};

// Karte initialisieren
> let map = L.map("map", { ...
}).setView([tirol.lat, tirol.lng], 8);

> let themalayer = { ...
}

// Hintergrundlayer
let layerControl = L.control.layers({
  "Esri WorldTopoMap": L.tileLayer.provider("Esri.WorldTopoMap").addTo(map),
  "Esri WorldImagery": L.tileLayer.provider("Esri.WorldImagery"),
> "Lichtverschmutzung 2016": L.tileLayer(...
  }),
> "Lichtverschmutzung 2020": L.tileLayer(...
  }).},
> { ...
},
).addTo(map);
layerControl.expand();

//Marker Mutterbergersee
> var mutterbergersee_icon = L.icon({ ...
});
//Marker Steinplatte/Winklmoosalm

> L.marker([47.657568, 12.579869], { icon: mutterbergersee_icon }).addTo(map).bindPopup(`<h4>Winklmoos/Steinplatte</h4>...
  <a href = "https://astrophototirol.github.io/Winklmoos/" > Hier gehts zum Spot -> </a>`).openPopup();

//Marker Lichtsee
> L.marker([47.831409, 11.405618], { icon: mutterbergersee_icon }).addTo(map).bindPopup(`<h4>Lichtsee</h4>...
  `).openPopup();

//Marker Sattelberg
> L.marker([47.010163, 11.477516], { icon: mutterbergersee_icon }).addTo(map).bindPopup(`<h4>Sattelberg</h4>...
  <a href = "https://astrophototirol.github.io/Sattelberg/" > Hier gehts zum Spot -> </a>`).openPopup();

//Marker Mutterbergersee
> L.marker([47.0166667, 11.129], { icon: mutterbergersee_icon }).addTo(map).bindPopup(` ...
  `).openPopup();

map.eachLayer(function (layer) {
  layer.closePopup();
});

// Maßstab
> L.control.scale({ ...
}).addTo(map);

// Create the image overlay
let imageBounds = [[0, 0], [window.innerHeight, window.innerWidth]];
let imageUrl = 'images/Unbenanntes_Panorama-1.jpg'; // Replace with the path to your actual large image file
let imageOverlay = L.imageOverlay(imageUrl, imageBounds);

// Create the image frame map
> let imageFrameMap = L.map('superzoom', { ...
});

// Set the max bounds of the image frame map to match the image bounds
imageFrameMap.setMaxBounds(imageBounds);

// Adjust image bounds on window resize
> window.addEventListener('resize', () => { ...
});

// Fit the image overlay to the map bounds initially
imageFrameMap.fitBounds(imageBounds);

// Add the image overlay to the image frame map
imageOverlay.addTo(imageFrameMap);

> L.control.Legend({ ...
}).addTo(map);

> L.control.rainviewer({ ...
}).addTo(map);
```

Abbildung 4: Main.js der Hauptseite

Abbildung 4 zeigt den Aufbau und die Implementierung der Websitekomponenten innerhalb der Datei main.js der Hauptseite. Zunächst werden Beispielkoordinaten für das Zentrum von Tirol, in diesem Fall Innsbruck, deklariert und im Anschluss die Karte initialisiert, sodass die Karte im Richtigen Ausschnitt angezeigt wird. Zusammen mit zwei Esri-Hintergrundkarten werden im Anschluss zwei Lichtverschmutzungskarten als TileLayer vom Github Repository des Erstellers (David Lorenz, 2020) in die LayerControl geladen und der Karte hinzugefügt. Im Anschluss werden die Marker für jeden einzelnen Astropot implementiert. Mit dem befehl map.eachLayer und Layer.closePopup werden alle Popups beim Aufruf der Seite geschlossen, sodass diese nicht direkt angezeigt werden. Im Folgenden Schritt wird ein Maßstabsbalken eingefügt. Hiernach wird das „superzoom“ Foto implementiert und als ImageOverlay in einen Leaflet Kartenframe eingebunden. Im nächsten Schritt wurde eine Legende mit Hilfe des Plugins Leaflet.Legend erstellt um die Farben der Lichtverschmutzungskarten nach wissenschaftlichen Vorgaben zur Klasseneinteilung zu beschriften. Diese besitzt insgesamt 15 Icons. Zu guter Letzt wird das Rainviewer Plugin geladen.

### 3.2.4 Main.js Lichtsee Nebenseite (Lichtsee)

```
/* Lichtsee */

// Innsbruck
> let ibk = { ...
};

// Karte initialisieren
> let map = L.map("map", { ...
}).setView([ibk.lat, ibk.lng], 9);

// thematische Layer
> let themaLayer = { ...
}

// WMTS Hintergrundlayer der eGrundkarte Tirol definieren
> let eGrundkarteTirol = { ...
}

// Hintergrundlayer eGrundkarte Tirol mit GPX overlay
> let layerControl = L.control.layers({ ...
  "Vorhersage durch Klicken auf Karte": themaLayer.forecast.addTo(map)
}).addTo(map);

// Maßstab
> L.control.scale({ ...
}).addTo(map);

// GPX Track visualisieren
> let controlElevation = L.control.elevation({ ...
}).addTo(map);
controlElevation.load("./data/lichtsee.gpx")

// Wettervorhersage MET Norway
> async function showForecast(url, latlng) { ...
}

// auf Kartenklick reagieren
> map.on("click", function(evt) { ...
});

// Klick auf Innsbruck simulieren
> map.fireEvent("click", { ...
})
```

Abbildung 5: Main.js der Nebenseite – Lichtsee

Abbildung 5 zeigt den Aufbau der Datei main.js einer Nebenseite anhand des Beispiels Lichtsee. Zunächst wird die Karte zusammen mit Koordinaten von Innsbruck Initialisiert. Diese sind in diesem Fall unwichtig, da die Karte später automatisch auf den Ausschnitt des befindlichen .gpx Tracks ausgerichtet wird. Es werden Themalayer für den .gpx Track, sowie den später implementierten forecast implementiert. Die Grundkarte wird im selben Schritt eingebunden. Im weiteren Verlauf werden .gpx

Tracks visualisiert und die Wettervorhersagen des MET Norway mit Hilfe einer asynchronen Funktion als Popup eingebunden. Dabei wird eine weitere Funktion zur Hilfe gezogen, die auf den Map Klick reagieren soll und so die Koordinaten an die Funktion showForecast übergeben kann.

## 4 Probleme & Herausforderungen

Herausforderungen beim Erstellen der Web Applikation waren vor allem einen Überblick über die gewünschten Funktionen zu behalten, und sich vorher Gedanken zu machen, welche Funktionen denn überhaupt ohne zu großen Aufwand technisch durchführbar sind. Es ist daher ratsam, zunächst einen genauen Plan zu erstellen, wie die Website im Detail aussehen soll.

Bis zur Entscheidung, die einzelnen Nebenseiten als Submodule zu implementieren waren oftmals mehrere Instanzen von VS Code geöffnet, was schnell sehr unübersichtlich wurde. Erst der oben genannte Schritt machte das Arbeiten angenehmer und übersichtlicher, auch wenn dazu weitere Schritte z.B. das Erstellen der .gitmodules Datei notwendig war.

Des Weiteren sind viele der Leaflet Erweiterungen veraltet und funktionieren daher nicht oder nicht richtig, da die Arbeit nichtmehr vom Entwickler fortgeführt wird. Dies machte es schwierig, geeignete Plugins zu finden.

## 5 Fazit und Ausblick

Insgesamt lässt sich sagen, dass die Implementierung der Webapp bis auf einige Probleme reibungslos verlief. Die Datenstruktur des Projekts wurde nach anfänglichen Schwierigkeiten gut organisiert, mit einem übergeordneten Repository und Submodulen für die einzelnen Seiten.

Während des Projekts traten zwar einige Probleme und Herausforderungen auf, wie die Auswahl und Integration der richtigen Plugins und die Organisation des Codes; eine sorgfältige Planung und ein genauer Überblick über die gewünschten Funktionen führten später dazu, diese Herausforderungen zu bewältigen.

Für die Weiterentwicklung des Projekts würde sich ein Open Source Ansatz anbieten, wobei Benutzer ihre eigenen Spots eintragen können und somit eine Datenbank für gute Astropots entsteht.

## 6 Literaturverzeichnis

B., B. (2023). *Leaflet.Control.FullScreen* [HTML]. <https://github.com/brunob/leaflet.fullscreen>

(Original work published 2012)

Bache, F. (2023). *Leaflet.Elevation* [JavaScript]. <https://github.com/MrMufflon/Leaflet.Elevation>

(Original work published 2013)

David Lorenz. (2020). *Light Pollution Atlas 2006, 2016 and 2020*.

<https://djlorenz.github.io/astrometry/lp2016/overlay/dark.html>

<https://djlorenz.github.io/astrometry/lp2020/overlay/dark.html>

Friedl, M. (2022). *Leaflet.WorldMiniMap* [JavaScript].

<https://github.com/maneoverland/leaflet.WorldMiniMap> (Original work published 2022)

Marcin, W. (2023). *Leaflet.Rainviewer* [JavaScript]. <https://github.com/mwasil/Leaflet.Rainviewer>

(Original work published 2019)

Ptma (2020). *Leaflet.Legend* [JavaScript].

<https://github.com/ptma/Leaflet.Legend> (Original work published 2020)

Wild, F. (2023). *Astrophotos Mutterbergersee, Lichtsee, Winklmoos und Sattelberg*

<https://github.com/AstrophotoTirol/astrophototirol.github.io/tree/main/images> (Original work)

Wild, F. (2023). *GPX Tracks von Outdooractive*

Winklmoos:

<https://www.outdooractive.com/de/route/wanderung/chiemgau/winklmoos/271117914/>

Lichtsee:

<https://www.outdooractive.com/de/route/wanderung/wipptal/lichtsee-fototour/267838325/?share=%7Ezvxqmokn%244ossnnsu>

Mutterbergersee:

<https://www.outdooractive.com/de/route/bergtour/stubaital/mutterbergersee/265389300/?share=%7Ezvtzbnx%244ossdnmv>

Sattelberg:

<https://www.outdooractive.com/de/route/bergtour/wipptal/sattelberg/265389599/?share=%7Ezvkuzvhi%244ossdnmx>

