

Leopold-Franzens-Universität Innsbruck

Institut für Geographie

MSc Geographie



ibk-student-survival-guide – Projektbericht

VU-Geoinformatik: Webmapping

Betreuer:

Klaus Förster, Bsc

Mag. Bernhard Öggl

Abgabedatum:

25.06.2025

Verfasser:

Jeff Prumbaum (12002373)

Philip Price (12003224)

Yannick Vinti (11906972)

1	Beschreibung des Projektes	3
2	Darstellung der Implementierungsschritte und der eingesetzten Techniken.....	4
2.1	Grundgerüst	4
3	Daten & Plugins	6
4	Startseite	7
5	Bars und Restaurants:.....	8
6	MTB-Seite	11
6.1	Gestaltung der MTB-Seite	12
6.1.1	main.js – Karte 1 (Arzler Alm).....	13
6.1.2	map2.js - Karte 2 (Mutterer Drei-Almen-Runde).....	14
6.1.3	map3.js – Karte 3 (Patscherkofel Rundkurs)	14

1 Beschreibung des Projektes

Unser Projekt haben wir IBK-Student-Survival-Guide genannt. Es ist eine dreiteilige Webseite, die den Studierenden in Innsbruck einen schnellen und kompakten Überblick über Universität, Freizeit und Alltagsleben geben soll. Das Projekt bündelt standortrelevante Open-Data-Informationen auf interaktiven Leaflet Karten und kombiniert diese mit ausgewählten Tipps von Studierenden für Studierende.

Anfangen mit der Startseite unseres Guides, dient diese als Willkommens-Seite die den Studierenden die wichtigsten Eckdaten zu Innsbruck, sowohl zum akademischen Teil als auch zum Punkt Freizeit liefert.

Oben ein Panorama Header, das sich über alle drei Seiten zieht und diesen ein freundliches Gesicht gibt, mit den farbigen Häusern aus der Mariahilfstraße und der schneebedeckten Nordkette. Darunter befindet sich dann der Titel des Projektes sowie eine Navigation bestehend aus drei Links, mit denen man bequem zwischen den drei Seiten wechseln kann. Die Startseite mit einem Home-Icon, die Restaurants und Bars mit einem weiteren Icon und die MTB Innsbruck Seite mit einem Mountain-Icon. Im Hauptteil sind vier untergegliederte Absätze in den die Studenten viele Informationen erhalten. Sei es ein schneller Überblick über das Leben als Student:in, dass die Stadt eine einzigartige Verbindung vom urbanem Campus-Leben und der alpinen Natur bietet, oder die typischen Bedürfnisse wie Lernplätze oder Cafés angesprochen werden. Darauf folgt ein kleiner Abschnitt über einige Fakten zur Stadt, in dem zum Beispiel die Einwohnerzahl, Studienquote und die geographische Lage angesprochen werden. Darauf folgen zwei Abschnitte, einmal über Universitäten und Bildung, in der es hauptsächlich über die unterschiedlichen Fakultäten geht und einmal über Freizeit und Lebensqualität in der zum Beispiel über die Nordkette, Altstadt, Outdoorsportmöglichkeiten und das Nachtleben erzählt wird. Die Abschnitte werden mit selbst aufgenommenen Fotos ergänzt, die die Seite anspruchsvoll unterteilen. Durch diese erste Seite verstehen die Leser:innen in kurzer Zeit was Innsbruck als Studienort ausmacht, welche Themen die Seiten abdecken und wie sie zu den weiterführenden Karten gelangen.

Die zweite Seite heißt Restaurants und Bars, auf dieser werden, wie der Name es verrät rund 40 Lokale als Marker auf einer Karte dargestellt. Restaurants und Bars werden durch unterschiedliche Icons kodiert, die Restaurants haben ein Besteck-Icon und die Bars ein Martiniglas-Icon. Klickt man solch einen Marker an, öffnet sich ein Popup mit dem Namen, der Adresse, des Website-linkes und der Telefonnummer des Betriebes. So können Studierende ihre nächste Essens- oder Ausgehoption direkt aus der Karte wählen. Auf der Karte wird dazu noch der Maßstab dargestellt als auch ein Kartenlayer in der man zwischen der OpenTopoMap und der OpenStreetMap

wählen kann. Über der Karte wird noch in einem kleinen Abschnitt erklärt um was es in der Karte geht und was man alles sehen kann.

Auf der dritten Seite mit dem Namen MTB Innsbruck präsentiert drei Mountainbike Routen aus dem offenen Datensatz der Stadt Innsbruck. Wir haben uns drei Strecken mit unterschiedlichen Schwierigkeitsgraden ausgewählt (leicht, mittel & schwer), die als GPX-Track geladen werden und von einem interaktiven Höhenprofil begleitet werden. Als Basiskarten können die Nutzer:innen wieder in einem Kartenlayer zwischen der eGrundkarte Tirol (Sommer & Winter) oder der Base-mapAt wählen, sodass diese zwischen Übersicht und Detailansicht wählen können, um ihre Tour optimal zu planen. Des Weiteren steht einem die Fullscreen Funktion, eine Minimap und ein Location Pfeil zur Verfügung. Alle Karten sind begleitet von einer Beschreibung des Weges mit den Informationen von Innsbruck Info.

2 Darstellung der Implementierungsschritte und der eingesetzten Techniken

2.1 Grundgerüst

1. Begonnen haben wir mit dem Schritt ein neues Repo zu erstellen. Dafür haben wir bei GitHub eine Organisation gegründet, zu der alle drei Teammitglieder Zugriff haben. Dann haben wir das Repo `ibk-student-survival-guide.github.io` erstellt.
2. Im `main.css` haben wir mit `@import` das Stylesheet `fonts.googleapis.com` geladen. Danach einen globalen CSS-Reset gemacht, so ist garantiert, dass das Layout auf jedem Gerät gleich startet.

```
/* Google Font */
@import url('https://fonts.googleapis.com/css2?family=Inter:wght@400;700&display=swap');

* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
```

3. Mit dem Card-Layout erzeugen wir in 6 Zeilen ein sauberes Grundgerüst für die Seite. Hierdurch wird zum Beispiel der Seiteninhalt mittig erscheinen, die Text, Bilder und Karten werden nicht an den Seiten kleben, Bilder werden abgerundet und es entsteht ein Schatten.

```
main {
  max-width: 1000px;
  margin: auto;
  background-color: white;
  padding: 2rem;
  border-radius: 12px;
  box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
}
```

4. Danach haben wir den Navigationsbanner erstellt, der auf großen Bildschirmen in einer Reihe steht und sich auf einem Mobilien Gerät zu einer Link-Liste umformt. Durch das nav-Element entsteht eine Gruppierung der einzelnen Links und im main.css haben wir dieses noch formatiert und Font-Awesome-Icons hinzugefügt. Der @media-Code sorgt dafür, dass die horizontale Menüleiste vom Desktop-Layout auf ein Spaltenlayout für die mobilen Geräte umschaltet. Dazu sorgt es dafür, dass das Layout auf Smartphones kompakt und navigierbar bleibt, ohne horizontal scrollen zu müssen.

```
<nav>
  IBK-Student-Survival-Guide:
  <div class="navigation">
    <a href="https://IBK-Student-Survival-Guide.github.io"><i class="fas fa-home"> </i> Startseite</a>
    <a href="https://IBK-Student-Survival-Guide.github.io/freizeitstudent"><i
      class="fas fa-utensils"></i> Restaurants & Bars</a>
    <a href="https://IBK-Student-Survival-Guide.github.io/freizeitstirol"><i class="fas fa-mountain"></i>
      MTB Innsbruck</a>
  </div>
</nav>
```

```
nav {
  background-color: #004d7a;
  color: white;
  padding: 1rem;
  border-radius: 8px;
  margin-bottom: 2rem;
  font-size: 1.2rem;
}

nav .navigation a {
  margin-right: 15px;
  color: #fff;
  text-decoration: none;
  font-weight: bold;
}

nav .navigation a:hover {
  text-decoration: underline;
}

@media (max-width: 600px) {
  body {
    padding: 10px;
  }

  nav {
    font-size: 1rem;
  }

  nav .navigation a {
    display: block;
    margin: 0.5rem 0;
  }
}
```

3 Daten & Plugins

Die Daten, welche für die Karten genutzt wurden, sind aus unterschiedlichen Quellen. Die Daten für die Karte Bars & Restaurants sind von den Gewerben selbst die Koordinaten der Gewerbe haben wir mit Hilfe von Google Maps herausgefunden die Daten wurden in einem LibreOffice Calc Dokument festgehalten und in einem weiteren schritt im JSON-Daten um gewandelt dies mit Hilfe der Internetseite Convert CSV to JSON (<https://www.convertcsv.com/csv-to-json.html>).

Die GPX-Daten für die Mountainbike Trails sind von Innsbruck Info, die Daten waren frei verfügbar auf der Seite (<https://www.innsbruck.info/radsport/d/mountainbiketouren.html>).

Alle Plugins wurden auf der Seite jsdelivr.com gesucht und die Codes wurden von der Seite kopiert auf dem jeweiligen Link zu GitHub wurde dann geschaut wie die Codes im main.js angewendet werden müssen.

- Leaflet.coordinates (<https://www.jsdelivr.com/package/npm/leaflet.coordinates>)
- Leaflet.fullscreen (<https://www.jsdelivr.com/package/npm/leaflet-fullscreen>)
- Leaflet.mimimap (<https://www.jsdelivr.com/package/npm/leaflet-minimap>)
- Leaflet.locatecontrol (<https://www.jsdelivr.com/package/npm/leaflet.locatecontrol>)
- Leaflet.elevatio (<https://www.jsdelivr.com/package/npm/@raruto/leaflet-elevation>)

Die Funktionsweise der Plugins wird in Kapitel 5 genauer erläutert.

4 Startseite

1. Im index.html der Startseite ist als erstes ein Willkommensabschnitt eingetragen. Dieser ist mittels margin-bottom:2rem;) klar getrennt vom Header. Im main.css wird der Überschrift eine Farbe #004d7a zugeordnet.

```
<div class="infos">  
  <h2 class="inns"> Neu in Innsbruck oder schon mittendrin? Dieser Guide ist für dich. </h2>  
  <p> Du bist neu in Innsbruck oder willst einfach wissen, was die Stadt für Studierende wirklich zu...</p>  
</div>
```

```
h2.inns {  
  color: #004d7a;  
  margin-bottom: 0.8rem;  
}
```

2. Figure und Caption diesen als visuelles Element. Das <figure>-Tag gruppiert das Bild und Beschreibung, es wird erkannt, dass beides zusammengehört. Im main.css wird und figure img, wird die Größe des Bildes skaliert und überschüssiges wird abgeschnitten. Dann wird dem Bild mit Box-shadow und dem figure img:hover ein Schatten um die Bilder erstellt und es entsteht ein leichter Zoom-effekt wenn man mit der Maus über das Bild gehen würde. Des weiteren wird ebenfalls eine Figcaption mit Icon erstellt, bei dem Font-Awesome-Symbol «fa-image» ein Bild-Icon erstellt, dahinter dann eine Bildunterschrift kommt und noch die Quelle angezeigt wird.

```
header img, figure img {  
  border-radius: 8px;  
  box-shadow:  
    0 2px 6px rgba(0, 0, 0, 0.15),  
    0 6px 18px rgba(0, 0, 0, 0.10);  
  transition: transform 0.25s ease;  
}  
  
figure img:hover {  
  transform: scale(1.01);  
}  
  
figure {  
  margin: 2rem 0;  
}  
  figcaption {  
    margin-top: .5rem;  
    text-align: center;  
    font-style: italic;  
    color: #666;  
  }
```

```
<figure>
  
  <figcaption><i class="fa-regular fa-image"></i> Maria-Theresien-Straße (Quelle: eigene Aufnahme)
</figcaption>
</figure>
```

5 Bars und Restaurants:

1. Wir fangen damit an eine Leaflet-Karte zu initialisieren und binden dazu eine Basemap. Mit `setView` wählen wir im `main.js` Innsbrucks Stadtzentrum als Startpunkt, das wir vorher festgelegt haben. Damit alle Bars und Restaurants in einem Blick zu erkennen sind, haben wir den Zoom-Level auf 13 gestellt.

```
// Innsbruck als Ausgangspunkt
let ibk = {
  lat: 47.267222,
  lng: 11.392778
};
```

```
let map = L.map("map").setView([ibk.lat, ibk.lng], 13)
```

2. Dann haben wir im `main.js` eine Konstante `DATA` angelegt, mit den Daten der rund 40 Bars und Restaurants. Die Daten bestehen jeweils aus dem Namen, den Koordinaten, die persönliche Informationen für das Popup (TEL, WEB & ADR) und der Kategorie (KATE) ob es sich um eine Bar oder ein Restaurant handelt.

```
const DATA = [
  {
    "NAME": "Dinzler",
    "lat": 47.26948,
    "lng": 11.39915,
    "ADR": "Kaiserjägerstraße 1, 6020 Innsbruck",
    "WEB": "http://www.dinz.at/",
    "EMAIL": "office@dinz.at",
    "TEL": "/",
    "KATE": "Bar"
  },
  ...
]
```

3. Nun erstellen wir im `main.js` die Marker und fangen an mit `Data.forEach(item => {`, somit verläuft durch das `forEach` jedes Objekt im Array «Data». Jedes Objekt repräsentiert nun ein Lokal mit den Koordinaten und Attributen. Die Icon Auswahl entscheidet sich dann durch das `KATE`-Feld. Für Bars wird ein Martiniglaß-Icon und für Restaurants ein Besteck-Icon zugeordnet. Falls einmal ein nicht klassifizierter Eintrag auftauchen würde, haben wir die «else» Zeile

hinzugefügt, dass in dem Fall ein neutrales Marker Symbol eingesetzt wird.

```
// Marker erstellen
DATA.forEach(item => {
  let iconHtml;
  if (item.KATE === "Bar") {
    iconHtml = '<i class="fa-solid fa-martini-glass-empty icon-bar"></i>';
  } else if (item.KATE === "Restaurant") {
    iconHtml = '<i class="fa-solid fa-utensils icon-resto"></i>';
  } else {
    iconHtml = '<i class="fa-solid fa-map-marker-alt icon-standa"></i>'; // Standard Icon
  }
}
```

4. Dann erzeugen wir das LeafletIcon, L.divIcon verpackt das HTML in ein als Marker nutzbares div. Mit iconSize wird die Breite und Höhe des Icons definiert und der IconAnchor sorgt dafür, dass die untere Spitze exakt mittig zur auf der Geo-Position sitzt.

```
const icon = L.divIcon({
  html: iconHtml,
  className: '',
  iconSize: [24, 24],
  iconAnchor: [12, 24]
});
```

5. Als nächsten Schritt wird dem Marker das zuvor angelegte divIcon zugewiesen und die Popups definiert. Durch die Backticks (` `) entsteht eine kleine HTML-Visitenkarte für alle Felder im Objekt. Hier werden dann Icons in das Popup implementiert (address-book, paperplane...) die das Popup attraktiver gestalten. Zuletzt werden die Marker den richtigen Overlay hinzugefügt. Die zwei layer-Gruppen (overlays.Bars & overlays.Restaurants) wurden vorher erstellt und sind in der Layer-Control sichtbar. So kann, die Nutzer:in die Bars und Restaurants einzeln an- und ausschalten (siehe Punkt 4). Der Website-Link wird mit target="_blank" kodiert, damit die externe Seite sich in einer neuen Registerkarte öffnet.

```
const marker = L.marker([item.lat, item.lng], { icon: icon });
marker.bindPopup(`
  <h2>${item.NAME}</h2>
  <ul>
    <li><i class="fa-solid fa-address-book"></i> ${item.ADR}</li>
    <li><i class="fa-solid fa-paper-plane"></i> ${item.EMAIL}</li>
    <li><i class="fa-solid fa-phone"></i> ${item.TEL}</li>
    <li><a href="${item.WEB}" target="_blank">Website</a></li>
  </ul>
`);
//Zuordnung
if (item.KATE === "Bar") {
  marker.addTo(overlays.Bars);
} else if (item.KATE === "Restaurant") {
  marker.addTo(overlays.Restaurants);
}
});
```

6. Um die Marker separat an und ausschalten zu können, erzeugt man zwei `L.LayerGroup()`-Objekte und übergeben diese als Overlays an die Leaflet-Layer-Control. In der Layer-Control, rendert Leaflet nun für jede LayerGroup eine Checkbox, mit der man die jeweiligen Icons dann ein- oder ausdrücken kann.

```
//LayerGroup erstellen
let overlays = {
  Bars: L.layerGroup().addTo(map),
  Restaurants: L.layerGroup().addTo(map),
};

// Kartenlayer
L.control.layers({
  "OpenStreetMap": L.tileLayer.provider('OpenStreetMap.Mapnik').addTo(map),
  "OpenTopoMap": L.tileLayer.provider('OpenTopoMap'),
}, overlays).addTo(map);
```

7. Als letztes kommen wir zu den eingesetzten Leaflet-Plugins. Hierzu haben wir die Plugins Minimap, Fullscreen, Locate Control und Coordinates implementiert. Der Minimap-Plugin blendet unten links eine Verkleinerungskarte ein, die immer den aktuell angezeigten Ausschnitt zeigt. Das Fullscreen-Plugin fügt oben rechts einen Button hinzu der die Karte auf dem Desktop bildschirmfüllend darstellt. Der Leaflet-Locate Control springt man auf der Karte per Klick auf die eigene Position und markiert diese. Durch Leaflet-Coordinates werden in der Kartenecke live die Mauskoordinaten angezeigt.

```
// MiniMap
var gkTirol = new L.TileLayer("https://wmts.kartetirol.at/gdi_summer/{z}/{x}/{y}.png");
var miniMap = new L.Control.MiniMap(gkTirol, {
  toggleDisplay: true
}).addTo(map);

// Fullscreen
map.addControl(new L.Control.Fullscreen());

// Locate control
var lc = L.control.locate({
  position: "topright"
}).addTo(map);

// Koordinatenanzeige
L.control.coordinates({
  position: "bottomleft",
  decimals: 5,
  decimalSeperator: ".",
  labelTemplateLat: "Latitude: {y}",
  labelTemplateLng: "Longitude: {x}",
  enableUserInput: true,
  useDMS: false,
  useLatLngOrder: true,
  markerType: L.marker,
  markerProps: {}
}).addTo(map);
```

Zum main.css der Bars und Restaurants kann man noch sagen dass der #map-Selektor eine feste Höhe von 600px festlegt und einen 1px Grauton als Border um die Karte setzt. Mit .icon-bar und .icon-resto werden die Font-Awesome Icons auf 25px skaliert.

```
#map {
  width: 100%;
  height: 600px;
  border: 1px solid gray;
}

.icon-bar {
  font-size: 25px;
}

.icon-resto {
  font-size: 25px;
}
```

6 MTB-Seite

Die Datei index.html bildet die Grundlage der Webseite und gliedert sich in folgende relevante Bereiche für die Mountainbike-Routen:

1. **Titel und Header:** Die Seite hat den Titel „freizeittirol“ und zeigt im <h1> die Überschrift „Mountainbikestrecken rund um Innsbruck“. Ein Bild mit Innsbruck-Panorama dient als optischer Einstieg ().

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>freizeittirol</title>
```

```
<body>
  <main>
    <h1>Mountainbikestrecken rund um Innsbruck</h1>
    <article>
      <header>
        <!--https://pixabay.com/photos/innsbruck-austria-winter-snow-5278879/-->
        
      </header>
```

2. **Scripts im <head>:** Alle für die MTB-Funktionalität notwendigen Leaflet-Plugins (z. B. leaflet-elevation, MiniMap, Fullscreen, Locate) werden eingebunden. Ebenso werden main.js, map2.js und map3.js geladen, welche jeweils eine Strecke anzeigen.

```
<!--CSS und JS-->
<link rel="stylesheet" href="main.css">
<script defer src="main.js"></script>
<script defer src="map2.js"></script>
<script defer src="map3.js"></script>
```

3. **Navigation:** Eine einfache Navigationsleiste verlinkt auf die Startseite, Restaurants & Bars sowie die aktuelle MTB-Seite. Icons von Font Awesome ergänzen die Benutzerführung visuell.

```
<nav>
  IBK-Student-Survival-Guide:
  <div class="navigation">
    <a href="https://IBK-Student-Survival-Guide.github.io"><i
class="fas fa-home"> </i> Startseite</a>
    <a href="https://IBK-Student-Survival-
Guide.github.io/freizeitstudent"><i
class="fas fa-utensils"></i> Restaurants & Bars</a>
    <a href="https://IBK-Student-Survival-
Guide.github.io/freizeittirol"><i class="fas fa-mountain"></i>
      MTB Innsbruck</a>
    </div>
  </nav>
```

4. **Einleitungstext:** Ein einleitender Absatz gibt eine Übersicht über die Intention der Seite, nämlich Studierenden Mountainbike-Strecken unterschiedlicher Schwierigkeitsstufen in Innsbruck näherzubringen.

```
<div>
  <p>Raus aus dem Hörsaal, rauf aufs Bike: Innsbruck ist wie gemacht
  für Mountainbiker*innen, egal ob
    Anfänger*in, Fortgeschrittene*r oder Trail-Profi. Auf dieser
  Seite stellen wir dir drei ausgewählte
    Mountainbike-Strecken vor, jeweils mit unterschiedlichem
  Schwierigkeitsgrad, damit für jede
    Erfahrungsstufe etwas dabei ist. Jede Tour zeigt dir nicht nur,
  wie vielseitig Innsbrucks Umgebung
    ist, sondern bietet dir auch die perfekte Gelegenheit, nach
  einem Unitag den Kopf freizubekommen.
    Was du zu jeder Strecke siehst ist eine interaktive Karte mit
  Streckenverlauf, ein detailliertes
    Höhenprofil und Infos zu Streckenlänge und Höhenmetern. Egal ob
  gemütlich bergauf zur Alm oder
    technisch anspruchsvoll bergab, such dir die Tour aus, die zu
  dir passt, und entdecke Innsbruck von
    seiner sportlichen Seite.</p>
</div>
```

Drei Sektionen mit Streckeninformationen: Jede MTB-Route hat ihren eigenen Abschnitt (<section>):

- **Arzler Alm (leicht)** mit <div id="map1"> und <div id="profile1">
- **Mutterer Drei-Almen-Runde (mittel)** mit <div id="map2"> und <div id="profile2">
- **Patscherkofel Rundkurs (schwer)** mit <div id="map3"> und <div id="profile3">

Diese Container sind Platzhalter für interaktive Leaflet-Karten und Höhenprofile, die durch JavaScript befüllt werden.

```
<section>
  <h2>Arzler Alm (leicht)</h2>
  <p>Leichte Mountainbikestrecke für Einsteiger bis zur Arzler Alm
  und gleichzeitig beliebte Enduro-Tour
    mit Singletrail-Option.
    Start beim Parkplatz Nordkettenbahn, der MTB-Beschilderung zur
  Arzler Alm zunächst auf Teerstrasse,
    später auf Schotterstraße folgen. Der Weg ist meist mäßig
  steil, mit kurzen Steilstücken bis zum
    Einstieg Arzler Alm Singletrail, ab hier die letzten Meter
  leicht bergab zur Alm rollen. Auf 1.067
    Metern Seehöhe am Fuße der Nordkette gelegen, bietet die Arzler
  Alm eine feine Sonnenterrasse,
    Kinderspielplatz, schöne Aussicht und sehr gute Küche.
    Tipp: Kurz vor der Arzler Alm zweigt Bikers left der
  Singletrail Arzler Alm ab. Fortgeschrittene Biker
    und viele Einheimische nutzen diesen als ansprechende Enduro-
  Tour. Schwierigkeit: S2
    (Singletrailskala). Variante: Nach 1,5 km zum Arzler
  Schießstand abzweigen, vor der Arzler Alm
    Anschluss zur Rumer Alm. Nach weiteren 1,5 km zweigt eine
  Verbindung zum Höttinger Bild und
    Achselwaldweg ab. Vor der Höttinger Alm Anschluss zur
  Bodensteinalm.
  <a
  href="https://www.innsbruck.info/radsport/mountainbike/mountainbike-
  touren/touren/arzler-alm-505.html">Quelle:
    Innsbruck Info</a>
  </p>
  <div id="map1"></div>
  <div id="profile1"></div>
</section>
```

6.1 Gestaltung der MTB-Seite

Die Datei main.css sorgt für das Layout und die visuelle Gestaltung. Relevante Punkte für die MTB-Seite:

1. **Kartenelemente (#map1, #map2, #map3):** Die Kartencontainer sind jeweils 100 % breit und 600 px hoch, mit grauem Rahmen versehen, was für eine gleichmäßige Darstellung sorgt.

```
#map1 {  
  width: 100%;  
  height: 600px;  
  border: 1px solid gray;  
}
```

2. **Höhenprofile (#profile1, #profile2, #profile3):** Ebenfalls einheitlich gestaltet mit grauem Rand und 0.3em Abstand zur Karte

```
#profile1{  
  margin-top: 0.3em;  
  border: 1px solid grey;  
}
```

3. **Schwierigkeit-Theme:** Für jede Route gibt es eigene CSS-Klassen (.leicht, .mittel und .schwer), die jeweils über CSS Custom Properties (--ele-area, --ele-bg) die Farben für das Höhenprofil anpassen:

- leicht: hellblau
- mittel: rotbraun
- schwer: dunkelgrau

```
.leicht {  
  --ele-area: #33CFFF;  
  --ele-bg: white;  
}
```

6.1.1 main.js – Karte 1 (Arzler Alm)

1. **Initialisierung der Leaflet-Karte (map1):** Karte wird auf Innsbruck zentriert (lat: 47.267222, lng: 11.392778) mit Zoomstufe 9.

```
// Innsbruck  
let ibk = {  
  lat: 47.267222,  
  lng: 11.392778  
};  
  
// Karte 1 initialisieren  
let map1 = L.map("map1").setView([ibk.lat, ibk.lng], 9);
```

2. **Hintergrundlayer:** Verwendung der „eGrundkarte Tirol“ in den Varianten Sommer, Orthofoto und Nomenklatur. Diese werden gruppiert und über ein Layer-Control-Element auswählbar gemacht.

```
// WMTS Hintergrundlayer der eGrundkarte Tirol  
let eGrundkarteTirol = {  
  sommer: L.tileLayer("https://wmts.kartetirol.at/gdi_summer/{z}/{x}/{y}.png", {  
    attribution: `Datenquelle: <a href="https://www.data.gv.at/katalog/dataset/land-tirol_elektronischekartetirol">eGrundkarte Tirol</a>`  
  }),  
  ortho: L.tileLayer("https://wmts.kartetirol.at/gdi_ortho/{z}/{x}/{y}.png", {  
    attribution: `Datenquelle: <a href="https://www.data.gv.at/katalog/dataset/land-tirol_elektronischekartetirol">eGrundkarte Tirol</a>`  
  }),  
  nomenklatur:  
  L.tileLayer("https://wmts.kartetirol.at/gdi_nomenklatur/{z}/{x}/{y}.png", {  
    attribution: `Datenquelle: <a href="https://www.data.gv.at/katalog/dataset/land-tirol_elektronischekartetirol">eGrundkarte Tirol</a>`,  
    pane: "overlayPane",  
  })  
};
```

```
// Layer control mit eGrundkarte Tirol und Standardlayern  
L.control.layers({  
  "eGrundkarte Tirol Sommer": L.layerGroup([  
    eGrundkarteTirol.sommer,  
    eGrundkarteTirol.nomenklatur  
  ]).addTo(map1),  
  "eGrundkarte Tirol Orthofoto": L.layerGroup([  
    eGrundkarteTirol.ortho,  
    eGrundkarteTirol.nomenklatur  
  ]),  
  "OpenStreetMap": L.tileLayer.provider("OpenStreetMap.Mapnik"),  
  "Esri WorldImagery": L.tileLayer.provider("Esri.WorldImagery"),  
}).addTo(map1);
```

3. **Zusatzfunktionen:** Maßstabanzeige, Minimap mit Toggle-Funktion, Fullscreen-Modus, Geolokalisierung und Koordinatenanzeige
4. **Höhenprofil:** Import der Strecke über leicht.gpx und Einbindung in ein Höhenprofil mit dem Container #profile1. Die visuelle Gestaltung wird über das Theme "leicht" bestimmt.

```
// Elevation-Profil (GPX laden)
const controlElevation = L.control.elevation({
  theme: "leicht",
  time: false,
  elevationDiv: "#profile1",
  height: 300,
  slope: true,
}).addTo(map1);

controlElevation.load("gpx/leicht.gpx");
```

6.1.2 map2.js - Karte 2 (Mutterer Drei-Almen-Runde)

Diese Datei basiert stark auf main.js, hat aber folgende Änderungen:

- Die Karte wird in map2 initialisiert.
- Das Höhenprofil verwendet mittel.gpx und das Theme "mittel".
- Die Karten- und Profildaten werden in die Elemente #map2 und #profile2 geladen.

Unterschiede im Vergleich zu main.js:

- Abweichender GPX-Dateiname
- Eigener DOM-Container
- Eigene Farbgebung (CSS-Klasse .mittel)

6.1.3 map3.js – Karte 3 (Patscherkofel Rundkurs)

Analog zu map2.js, aber mit folgenden Unterschieden:

- Karte map3 mit GPX-Datei schwer.gpx
- Höhenprofil-Theme "schwer"
- Ziel-Container: #map3 und #profile3

Bei map2.js und map3.js trat ein Problem auf da wir in main.js die gleichen Funktionen verwendet haben liefen die Karte 2 und 3 anfangs nicht das Problem lag darin, dass die Funktionen immer wieder abliefen, was nicht funktioniert. Um dieses Problem zu beheben, wurde in map2.js und map3.js die Funktion immediately Invoked Function Expression hinzugefügt. Diese Funktion kapselt den Code so ab, dass dieser nur lokal abgerufen wird und das Problem wurde so behoben.