# Prototyping Projektdokumentation Simon Lindenmann

Name: Simon Lindenmann

E-Mail: lindesi2@students.zhaw.ch

URL der deployten Anwendung: <https://vacation-planer-simon.netlify.app/>

## Einleitung

*Meine Idee war es ein Urlaub Planer zu machen bei dem man ein Trip in ein bestimmtes Land planen kann und in diesem Land verschiedene Reiseziele besuchen kann. Ich wollte dabei vor allem einen realistischen Datensatz verwenden, der auch ziemlich gross ist. Ich hatte am Anfang mühe etwas passendes zu finden und habe es zuerst mit daten von Kaggle versucht, weil ich das von Data Management schon kannte und wusste, dass es Datensätze hat die man auch gut in eine Datenbank importieren kann. Ich habe dann auf Kaggle nichts passendes gefunden und hab mir überlegt das mit mockaroo zu machen aber dort müsste man alles selber machen und ich wollte nicht 1000 daten von Hand schreiben. Ich habe dann ein bisschen mit Hilfe von ChatGPT nach geforscht und habe heraus gefunden, dass man von Wikipedia mit Queries Daten exportieren kann. Ich habe die Daten dann zu einem csv exportiert. Ich hatte schon ein bisschen Erfahrung mit Python und csv Import in Python und konnte dann die Daten von den Ländern und den Destinationen trennen und als 2 separate json files ausgeben. Diese konnte ich dann in die mongodb importieren und so konnte ich einen sehr grossen Datensatz beschaffen. Es war für mich spannend aber auch sehr zeitaufwändig, weil es immer wieder Probleme gab aber am Schluss hat es sich gelohnt weil es die Funktionalität um einiges besser macht, deshalb erwähne ich es auch bei Erweiterungen.*

## Datenmodell

## Ein Bild, das Screenshot, Diagramm, Kreis, Text enthält. Automatisch generierte Beschreibung

Jede Destination gehört nur zu genau einem Land und jeder Trip findet nur in genau einem Land statt in diesem können aber im gleichen Trip mehrere Destinationen des Landes besucht werden. Länder und Destinationen können logischerweise von mehreren verschiedenen Trips besucht werden und in einem Land hat es meistens mehrere Destinationen.

## Beschreibung der Anwendung

*Startseite: Gif von Fahrenden Auto mit einem Knopf Manage Trips, wo man zum zu der wichtigsten Seite kommt und zu Oberst eine Navigations-Bar.*

*Ein Bild, das Fahrzeug, Landfahrzeug, Rad, Auto enthält.

Automatisch generierte Beschreibung*

### Trips

Ist die wichtigste Seite zeigt alle Trips an aber nur das Land und den Namen des Trips.

Ein Bild, das Text, Screenshot, Grafikdesign, Design enthält.

Automatisch generierte Beschreibung

Dateien:

* routes/trips/+page.svelte
* routes/trips/+page.server.js

### Trips Details

Zeigt die genaueren Informationen vom Trip an dabei hat es bei Country und den Destinations die Buttons Details, mit denen man genauere Informationen vom Jeweiligen Objekt ansehen kann.

Ein Bild, das Text, Screenshot, Software, Zahl enthält.

Automatisch generierte Beschreibung

Dateien:

* routes/trips/[trip\_id]/+page.svelte
* routes/trips/[trip\_id]/+page.server.js

### Countries Details

Ein Country Objekt hat nur 2 Attribute und zwar die country\_id und den country\_name, darum wäre eine Detail Seite über das Objekt sehr unnötig, weil man alles was einem interessiert schon weiss. Stattdessen werden einfach alle Destinations, die es in diesem Country gibt aufgelistet und mit dem Knopf Details kann man sich diese noch genauer anschauen.

Ein Bild, das Text, Screenshot, Software, Computersymbol enthält.

Automatisch generierte Beschreibung

Dateien:

* routes/countries/[country\_name]/+page.svelte
* routes/ countries/[country\_name]/+page.server.js

### Destinations details

Hier sieht man eine Destination genauer. Von der liste bei der Country Detail Seite sieht man hier nicht mehr viel neues ausser man braucht die Country oder Destination ID wenn man aber von wo anders kommt zum Beispiel von Trips Details kann man neue Dinge nützliche Details auf dieser Seite sehen.

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

Dateien:

* routes/destinations/[destination\_id]/+page.svelte
* routes/destinations/[destination\_id]/+page.server.js

### Countries

Auf dieser Seite werden Alle Countries der Datenbank gezeigt. Wenn man auf View Destinations geht, dann kommt man wieder auf die Seite «routes/countries/[country\_name]/+page.svelte» wo man die Liste aller Destinations von dem Land das man auswählt sieht.

Ein Bild, das Text, Screenshot, Zahl, Software enthält.

Automatisch generierte Beschreibung

Dateien:

* routes/ countries /+page.svelte
* routes/countries/+page.server.js

Destinations

Auf dieser Seite sieht man alle Destinations auf gelistet. Das sollten ca 10'000 sein also macht es sinn nur so wenig wie möglich an zu zeigen. Mit View Details kommt man auf die Detail Seite von der Destination die man auswählt.

Ein Bild, das Text, Screenshot, Software, Webseite enthält.

Automatisch generierte Beschreibung

Dateien:

* routes/ destinations /+page.svelte
* routes/ destinations /+page.server.js

## Erweiterungen

Ich habe zusätzliche Dateien in documentation -> extra\_files abgelegt

### Datenbeschaffung

Ich habe wie bereits schon in der Einleitung erwähnt, Daten von Wikipedia mit einer Query herunter geladen:

Ein Bild, das Text, Screenshot, Software, Computersymbol enthält.

Automatisch generierte Beschreibung

Hier habe ich 10'000 Daten exportiert in ein CSV. Ich habe zuerst gedacht ich könnte einfach 2mal ein Export machen aber das ging auch nicht weil die Daten dann nicht überein stimmten. Also musste ich ein Python Skript schreiben. Um die Countries und Destinations zu trennen und die Daten zu bereinigen. Die Getrennten Daten habe ich in 2 json files geschrieben, die ich dann in die MongoDB Importieren konnte.

Ein Bild, das Text, Screenshot, Software, Display enthält.

Automatisch generierte Beschreibung

Die Daten konnte ich Importieren ich hatte viele Fehler die ich korrigieren musste als ich dann alle Destinations aus einem Land anzeigen konnte, wusste ich das der Datensatz funktioniert.

Ein Bild, das Text, Screenshot, Software, Zahl enthält.

Automatisch generierte Beschreibung

Dann habe ich die Trips als json importiert, das war sehr schwierig richtig zu machen, denn der Array machte viele probleme.

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte Beschreibung

Ich habe wegen vielen Komplikationen mit den Arrays und so ziemlich lange gebraucht, bis ich eine Query machen konnte, die so geht aber am Schluss ist die Query gelaufen und ich wusste so, dass die Daten bereit für den Gebrauch sind.

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte Beschreibung

Später beim programmieren war der grösste aufwand das db.js file ich habe dort sehr viel neue Funktionen geschrieben (auch mit Hilfe von ChatGPT) und ich musst viele Fehler suchen vor allem mit Arrays und weil ich die Daten verknüpft habe also nicht immer mit Object\_id gearbeitet habe.

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte Beschreibung