Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

IKG
Institute of Cartography
and Geoinformation

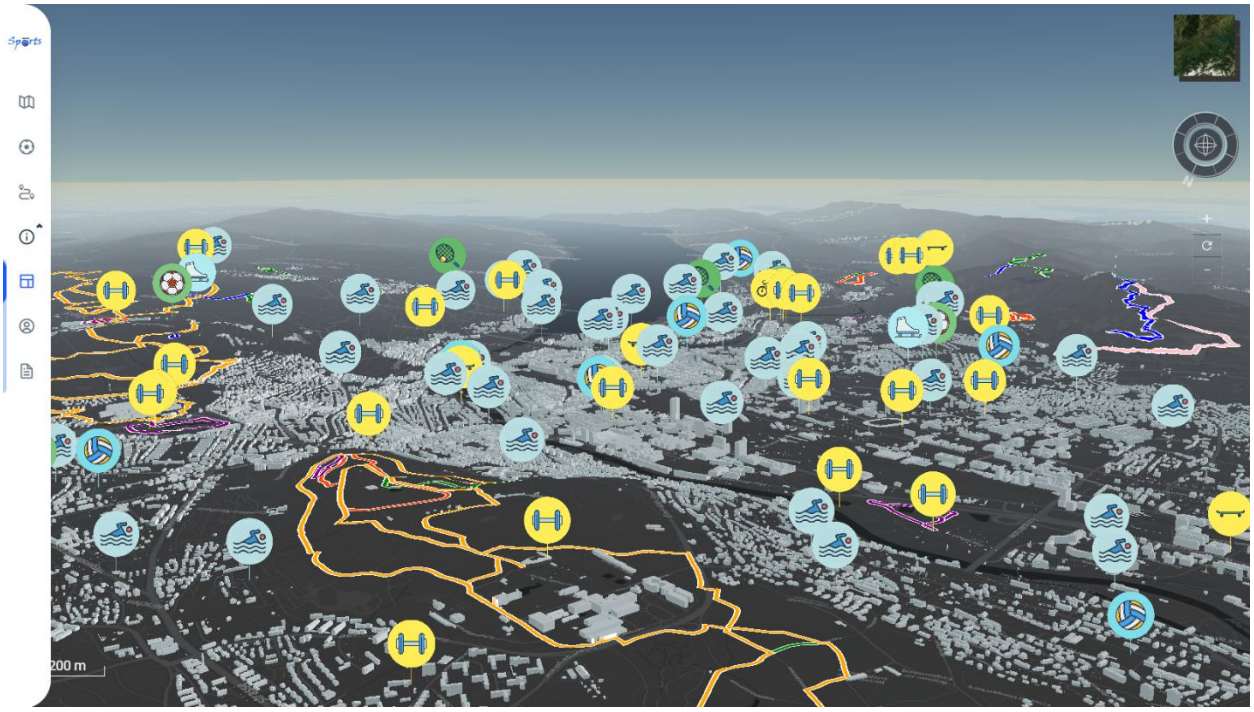
Cartography III (103-0227-00L)

Autumn semester 2020

Course lead: Prof. Dr. Lorenz Hurni

A 3D Sports Map for Teenagers in Zurich

Project report



Code repository: https://gitlab.ethz.ch/carto3-hs20/yue

Web application: http://carto3.ethz.ch/project/qm2NL4tM/

Author: Yuanwen Yue

Study program: M.Sc. Geomatics

yuayue@student.ethz.ch

Jining, 08.01.2021

Introduction

Sports help teenagers live a more active life and reduce the risks of diseases like obesity. However, various electronic products seem to have changed teenagers' movement patterns and encourage them to sit more and be less active. A study led by the World Health Organization (WHO) shows that teenagers worldwide, including the Swiss, lack enough exercise [1]. The main purpose of this project is to develop an interesting and attractive 3D web map to encourage teenagers to go outdoors and help them find sports more easily.

Some existing applications such as Google Maps can be used to find sports locations. However, there are two main limitations. First, Google maps only have the location of sports facilities, no sports routes. Second, the information displayed on Google Maps is 2D and static, which is not attractive enough for teenagers.

To solve these problems, the project integrates many features, including but not limited to the following. First, two thematic layers are integrated into one web app, one for sports facilities and the other for sports routes. Second, 2D map is extended to 3D map, and various map animations are used, such as virtual flight, model animation, etc. Moreover, the project designs eye-catching user interfaces and friendly interaction for teenagers. All these features make the project both practical and fun, so it will be more interesting to teenagers.

The project takes the City of Zurich as the spatial extent, and uses the latest available sports data provided by the government.

Data

Data processing mainly includes two parts: Convert SHP to Geojson and generate CZML file.

- Convert SHP to Geojson: The original sports facilities and routes data are multiple SHP files. The multiple SHP files are merged using ***Merge*** function in ArcGIS toolbox. Then, ***Project*** function is used to project the data from CH1903+_LV95 to WGS_1984. For the sports routes data, ***Calculate Geometry*** function is used to add a field of route length. Finally, mapshaper.org is used to convert the SHP files to Geojson.
- Generate CZML file: ***Feature Vertices To Points*** in ArcGIS toolbox is used to convert sports routes data to points data. Then the coordinates of these points are exported to a csv file. Manual work is carried out to add two more fields, time dimension and altitude. The reconstructed data is used to fill the filed of cartographicDegrees in CZML file.

Design

Maps

- Design guidelines: a. Keep neat and avoid extraneous information and reduce occlusion; b. Object symbolization should adapt to different viewing; c. Coherent style of color and abstraction designed for teenagers;
- Base layer: Two base layer options are provided. The dark style base map can produce a sense of adventure, which will be more attractive to teenagers. Satellite base map is also provided for teenagers to better locate sports facilities and the surrounding environment.
- Sports facilities layer: As shown in Fig.3, different kinds of sports facilities are symbolized by different billboards. Anchoring is used for better visibility. Scaling is used to reduce overlap.
- Sports routes layer: As shown in Fig.4, different kinds of sports routes are symbolized by different colored lines. The line is highlighted when clicked. As shown in Fig.5, 3D model animation is used to make the map more interesting.

User interface

- Design guidelines: a. Interface elements should not affect the quality of the map; b. The UI should be self-explaining; c. Eye-catching interfaces and friendly interaction for teenagers.
- Cartographic UI elements: Legend is provided for sports routes because the lines are not very indicative. Feature query is used to provide more information for sports features.
- General UI elements: The dynamic sidebar is used to switch layers and trigger the dialogs. A spatial navigation bar is provided to make the map easier to control.
- Advanced functionality: Routing can help teenagers plan the itinerary to sports facilities of interest. Three travel modes are provided and travel time and distance are calculated.

Implementation

- Code structure: As shown in Fig.10, the components are logically related. CesiumScene.vue and its sub-components, and Cesium.js are mainly responsible for the interaction of Cesium elements. Sidebar.vue is mainly responsible for the general UI elements management. eventVue.js is designed for the communication between sibling components.
- Cesium functions: IonResource.fromAssetId, GeoJsonDataSource.load, Cartesian3.fromDegrees, UrlTemplateImageryProvider, CesiumTerrainProvider, Cartographic.fromCartesian, HeightReference, PolylineOutlineMaterialProperty, ScreenSpaceEventHandler, Math.toDegrees, Cartesian3.fromDegreesArrayHeights, Color, FeatureDetection, etc.
- HTML elements or Vue components: div, table, img, vs-button, vs-sidebar, vs-card, vs-dialog, etc.
- Routing implementation: First, the coordinates of starting point and end point are converted from Cartesian to degrees. Second, router.calculateRoute function from HERE Maps JS is used to calculate routes. Then, the coordinates of the routes are extracted from the returned result. Finally, a polyline is created using the coordinates and added to the entities of viewer. Travel time and distance are also parsed from the returned result.
- Clean code: Eslint is used to format the code automatically when saved. Common code is written into functions to eliminate redundancies. The return data of sports facility and route is encapsulated. Human-readable variable names are used. Add comments when necessary.
- Alternatives: Many map APIs can provide routing functions, such as Google Map, Mapbox. The reason for using HERE Maps JS is that the coordinates used to create the Cesium polyline can be extracted from the results returned by the HERE routing API with almost no additional processing, which is very convenient.

Conclusion

- Satisfied things: In this project, I have witnessed a small demo gradually grow into a product which is useful for real users, and this process makes me very satisfied. The user interface is clear and intuitive for teenagers.
- Challenges: First, it is difficult to come up with a novel topic and find available data. Second, the design and organization of Vue components is challenging. Besides, the interaction and coordination between multiple development packages is critical.
- Limitations: First, the sports facilities and routes data are not real-time and contain missing values. Second, the advantages of Vue are not fully utilized. Third, more functions can be added, such as reservation for sports activities.
- Lessons: First, checking official documents can usually solve problems in programming quickly. Second, when building applications, we must not only consider functionality, but also pay attention to compatibility and error prevention, etc.

Appendix

References:

[1] SwissInfo, WHO: Swiss teens are too sedentary, https://www.swissinfo.ch/eng/global-health-report-_who--swiss-teens-are-too-sedentary--/45385780

Data

Table 1: Data sources used in this project

Description	URL
Shapefile of Swiss boundaries	Null (Provided by teaching assistants)
CityGML with 3D buildings of Zurich	Null (Provided by teaching assistants)
Shapefile of sports facilities of Zurich	https://www.stadt-zuerich.ch/geodaten/download/Sport
Shapefile of sports routes of Zurich	https://www.stadt-zuerich.ch/geodaten/download/Erholungs__und_Sporteinrichtungen

Maps

Figure 1: base layer - dark

Figure 2: base layer - satellite

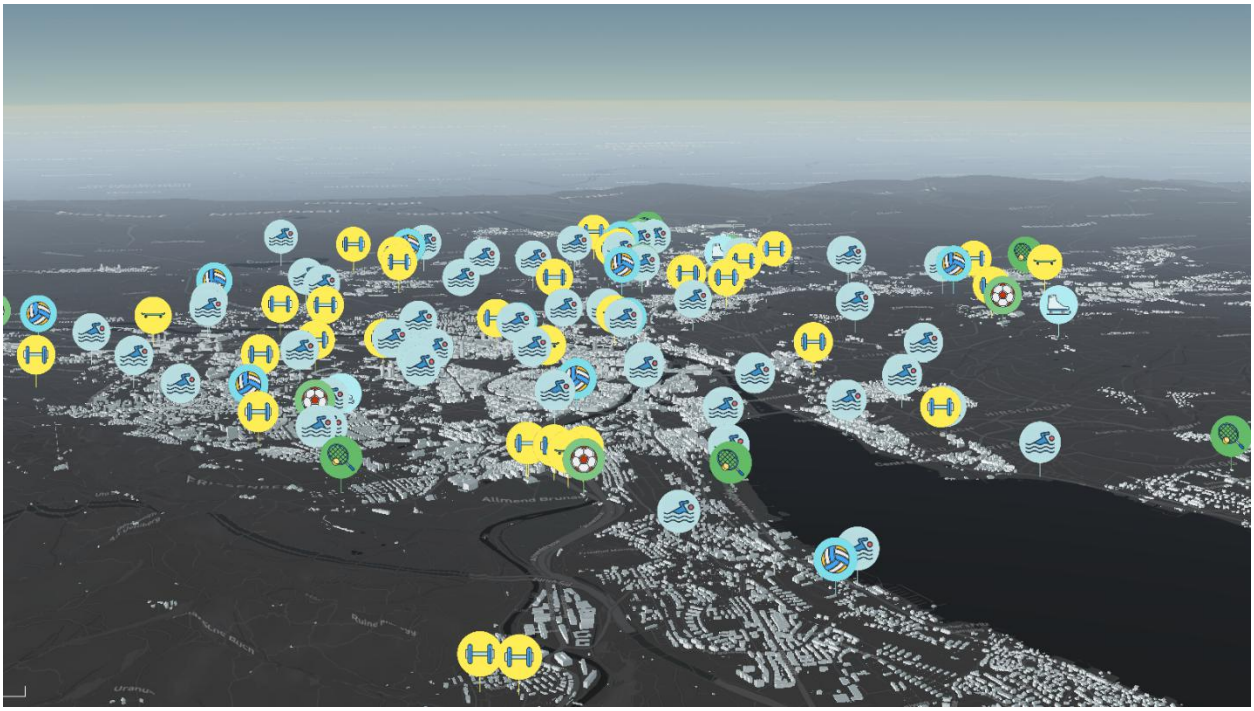


Figure 3: sports facilities layer

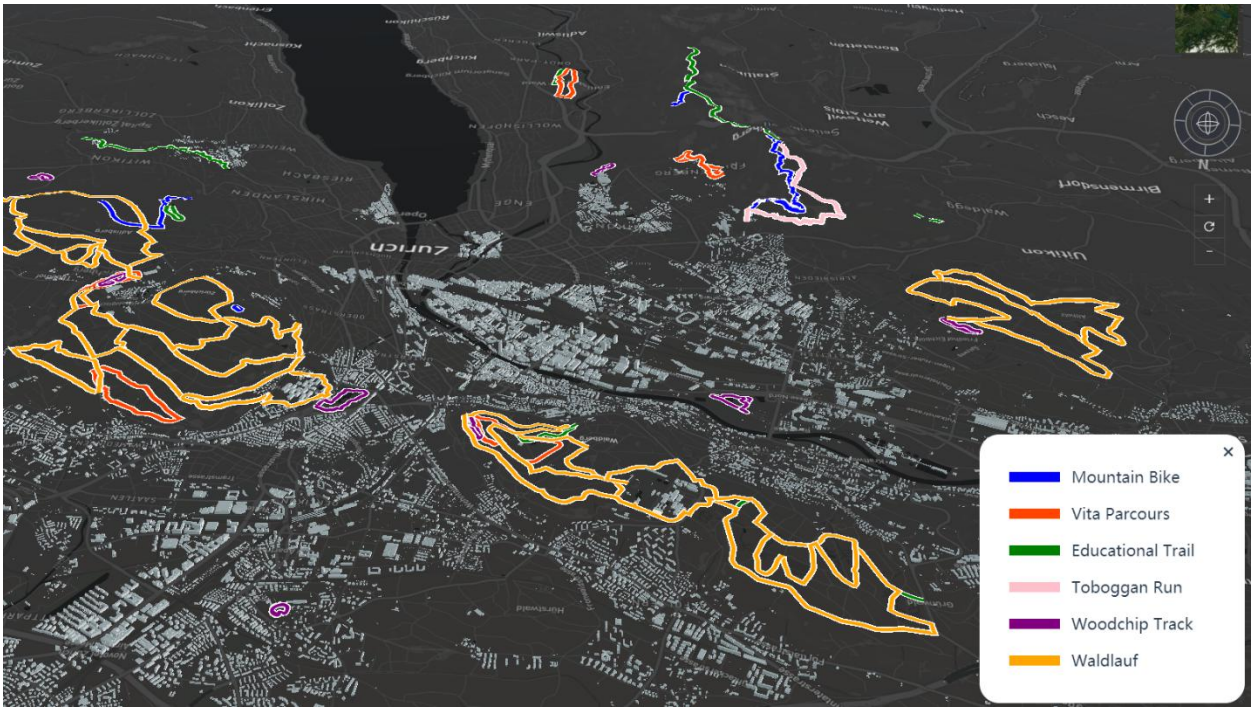


Figure 4: sports routes layer

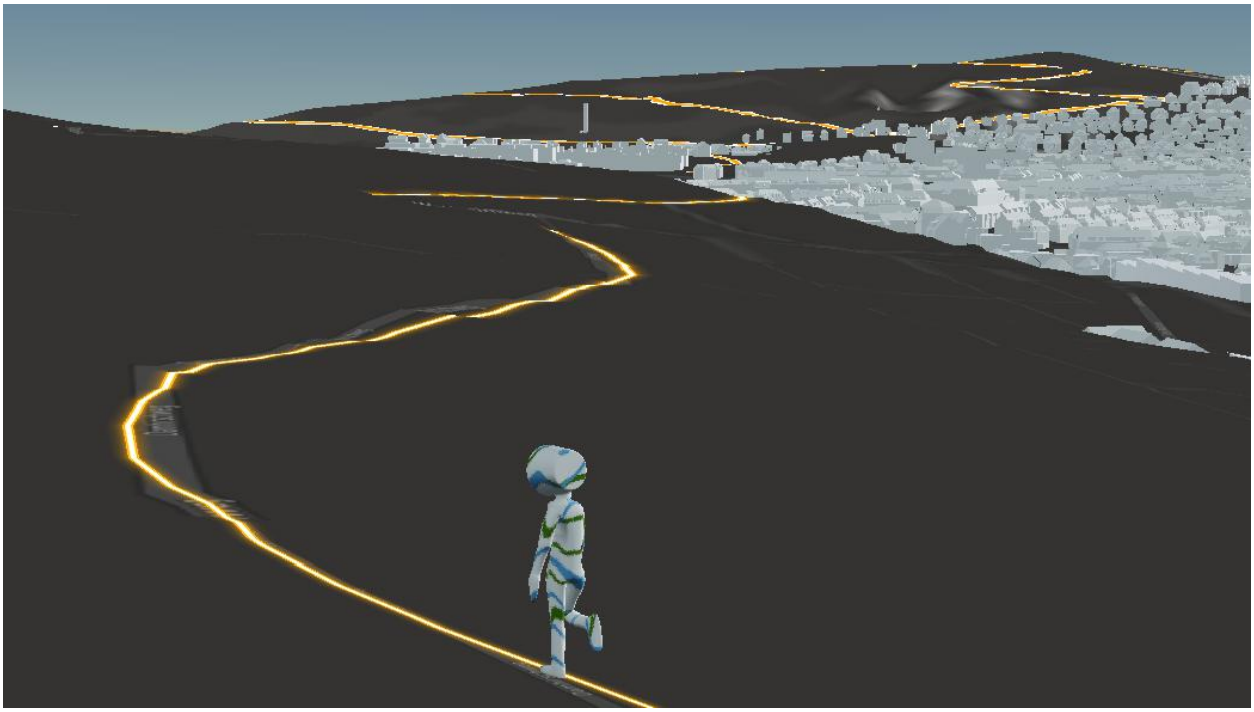


Figure 5: model animation

User interface

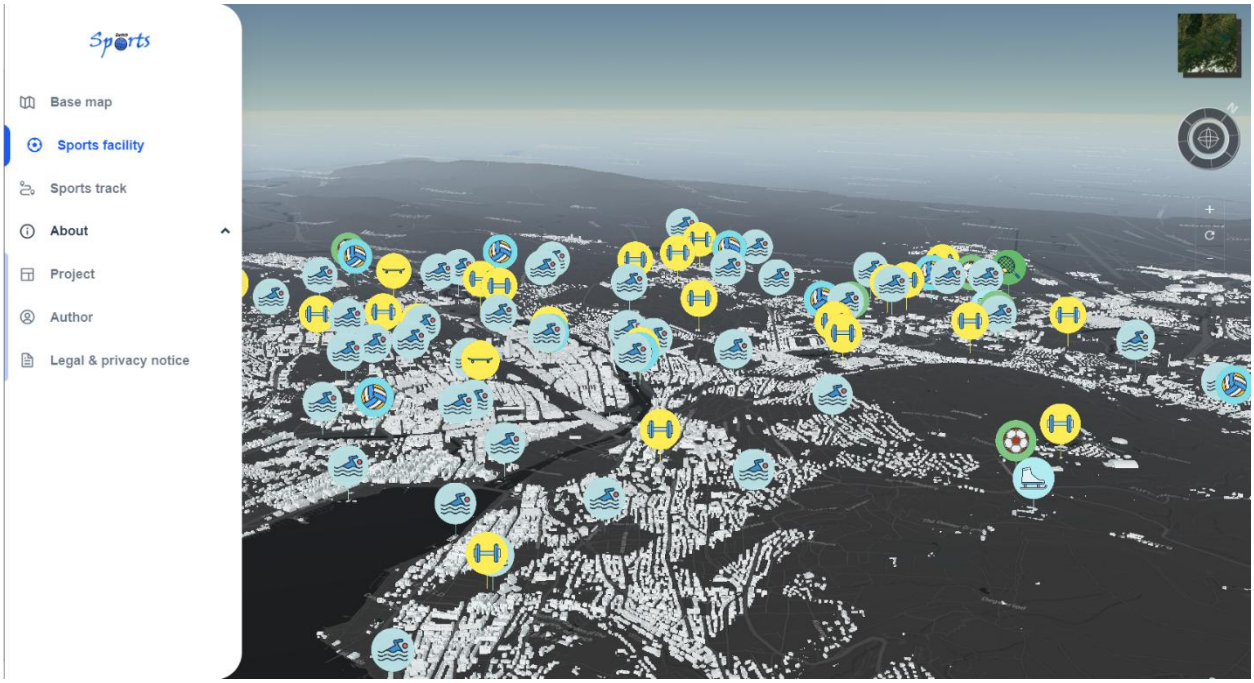


Figure 6: overall layout

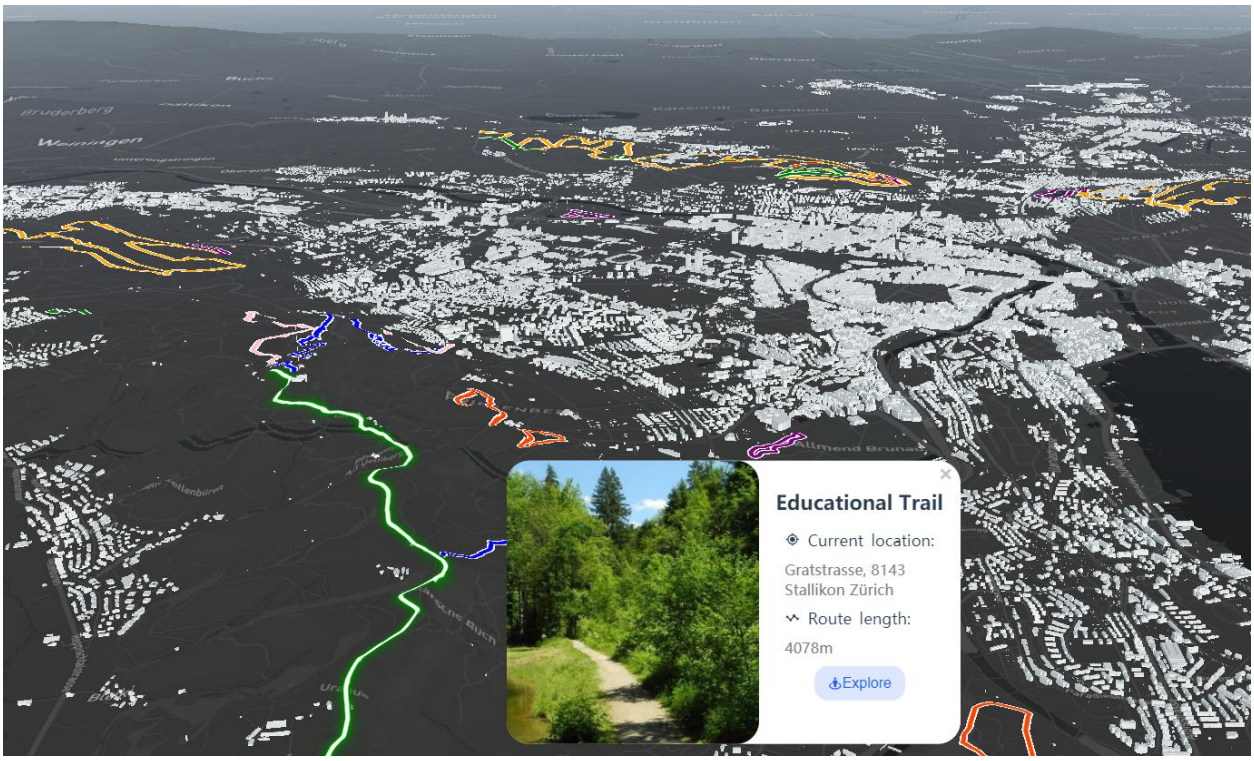


Figure 7: sports routes popup

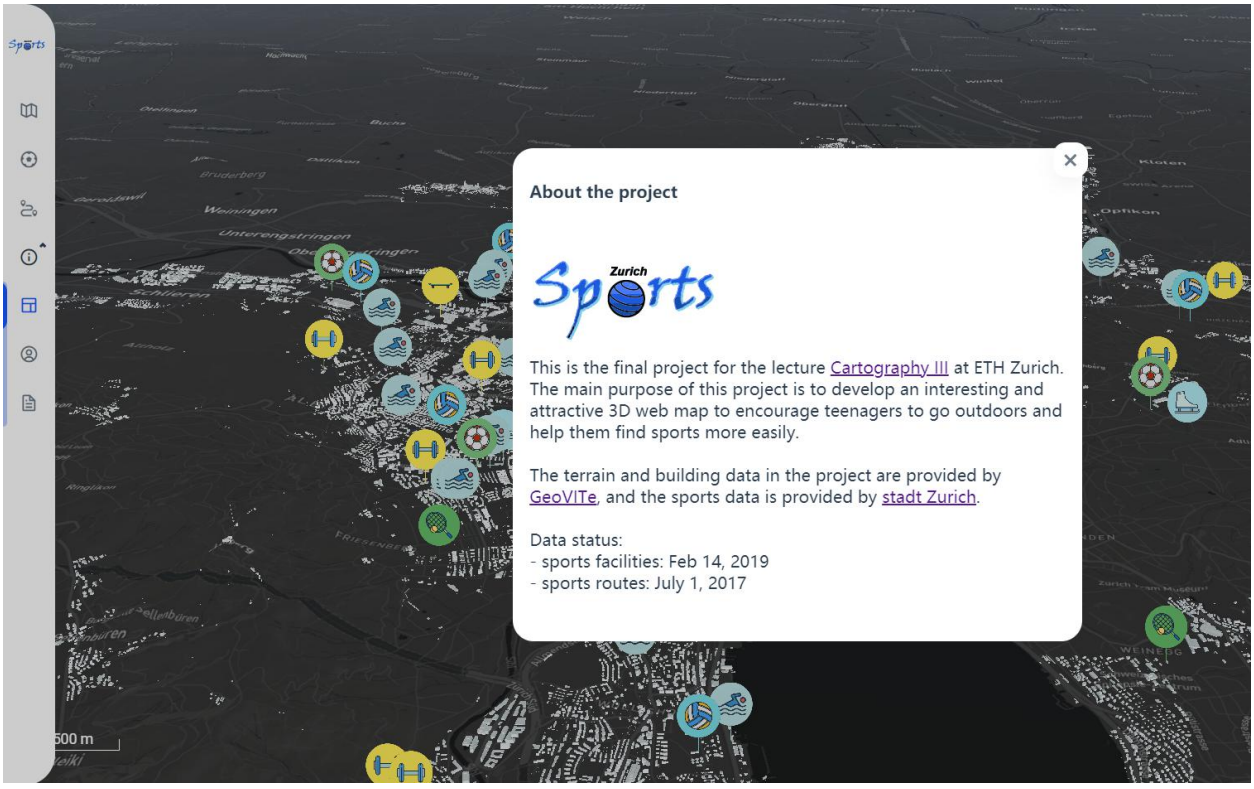


Figure 8: project description

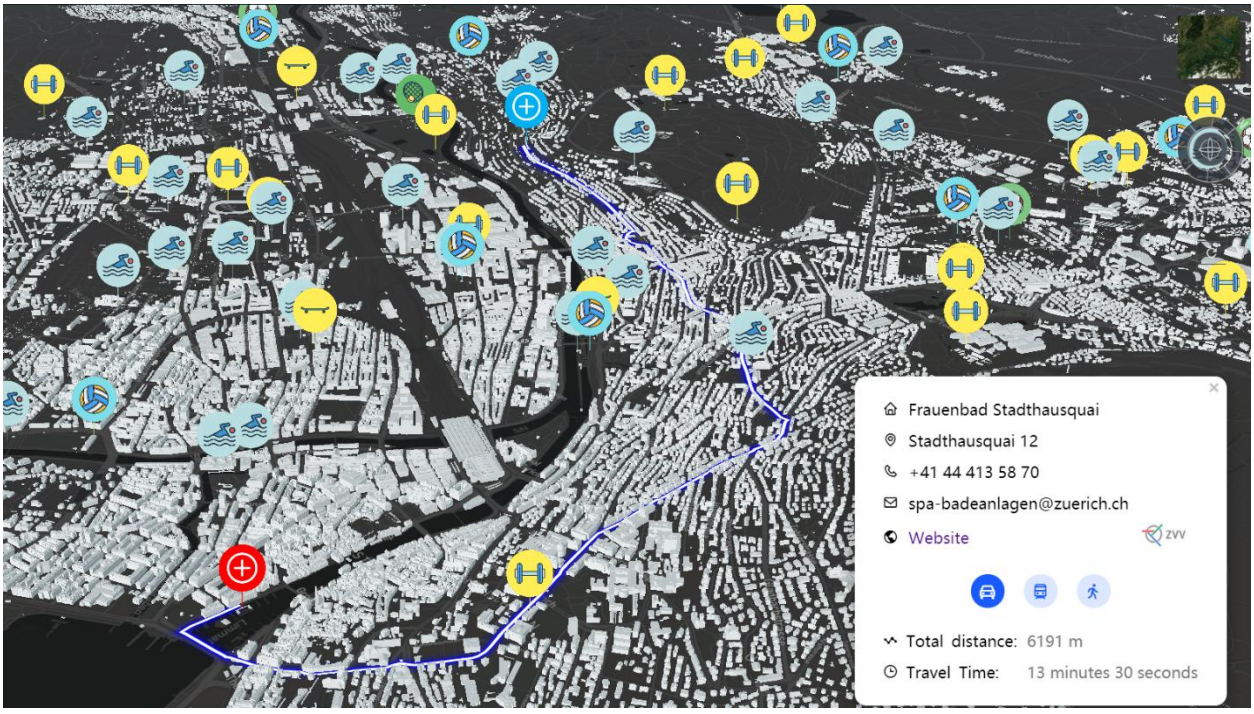


Figure 9: routing function

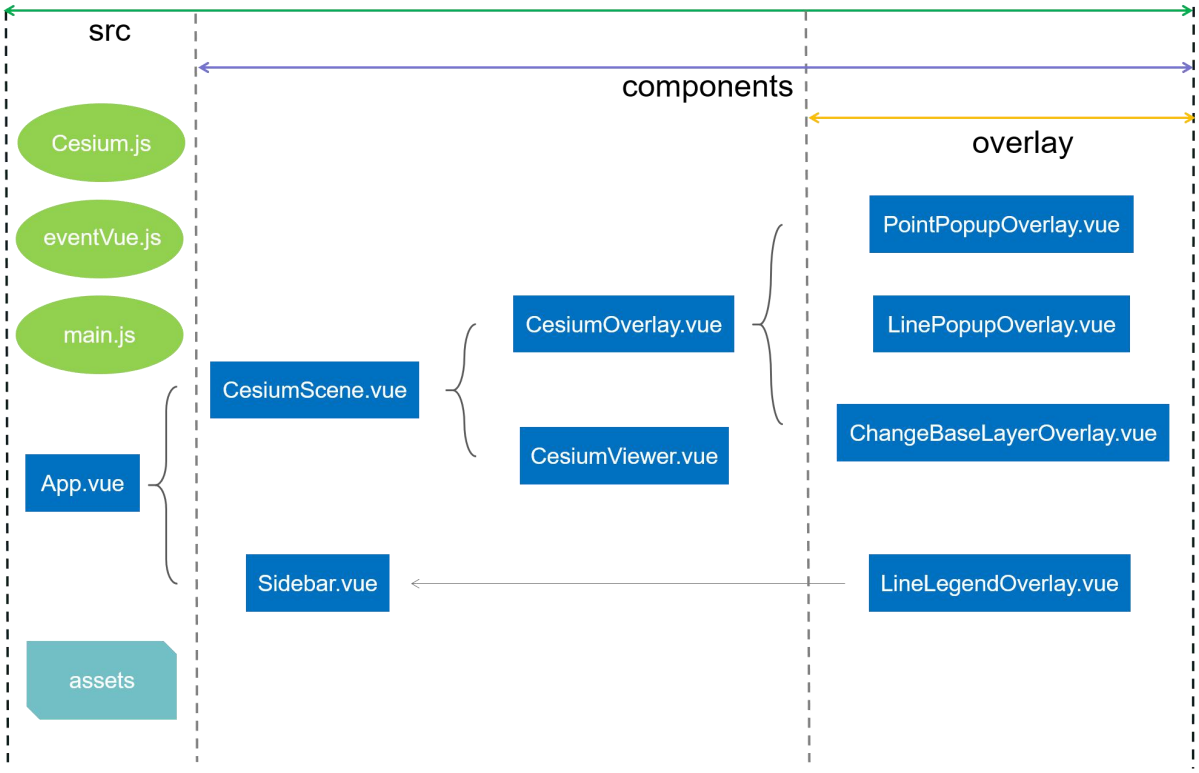


Figure 10: structure of src

Software

Table 2: Web frameworks and libraries used in this project

Name	**Description**	**URL**	**License**
CesiumJS	Virtual globe, 3D maps	https://github.com/CesiumGS/cesium	Apache 2.0
Vue.js	User interface	https://github.com/vuejs/vue	MIT
Vuesax	User interface	https://vuesax.com/	GPLv3
BoxIcons	Web friendly icons	https://github.com/atisawd/boxicons	CC 4.0
HERE Maps JS	Routing, Reverse Geocode	https://developer.here.com/develop/javascript-api	Proprietary
Mapbox GL JS	Raster Tiles	https://docs.mapbox.com/mapbox-gl-js/api/	Proprietary
cesium-navigation-es6	Cesium plugin, compass, navigator, distance scale	https://github.com/richard1015/cesium-navigation-es6	MIT