

Public Toilet Finder – Kiel Project Report

Student: Furkan Yilmaz

Student Number: 953661

Course: Geographical Web Applications

University: HAW Kiel

Semester: Winter Semester 2025/26

Lecturer: Hendrik Oenings

Submission Date: 08 December 2025

1. Introduction

Public facilities are an essential part of urban infrastructure. Among these facilities, public toilets play a particularly important role for elderly people, tourists, families with children, and individuals with disabilities. Despite their importance, reliable and easy-to-access information about public toilets is often missing or fragmented. Many people do not know whether a toilet is free or paid, whether it is wheelchair accessible, or during which hours it is open.

The motivation behind this project is to solve this problem by developing a modern web-based geographical information system that provides detailed and accurate information about public toilets in the city of Kiel. The application is based on freely available OpenStreetMap (OSM) data and aims to improve accessibility and urban orientation for both residents and visitors.

The main objective of this project is to design and implement a standards-compliant, accessible, and interactive web mapping application using Vue.js, OpenLayers, GeoServer, WMS, WFS, and TypeScript. The application fulfills the technical and design requirements of the Geographical Web Applications course.

2. Data Source and Data Processing

The primary data source for this project is OpenStreetMap (OSM). OpenStreetMap is a collaborative and community-driven project that provides a free and openly accessible geographic database of the world. Millions of contributors continuously update and maintain spatial information such as roads, buildings, public facilities, and points of interest. Due to its openness and high level of detail, OSM is an ideal data source for web-based geographic applications.

For this project, all spatial features tagged with the attribute amenity = toilets were extracted from the OpenStreetMap dataset. These features represent public toilet facilities distributed across the city of Kiel. The extraction ensures that only relevant and meaningful data is used for the application.

The following attributes were selected and used inside the application:

- wheelchair – indicates whether the toilet is accessible for wheelchair users.
- fee – specifies whether the toilet is free or requires payment
- charge – provides detailed price information, if available
- opening_hours – describes the operating times
- male, female, unisex – represent gender availability
- changing_table – indicates whether a baby changing table is available
- access – defines access restrictions such as public or customers only

After extraction, the data was converted into GeoJSON format. GeoJSON is a lightweight and web-friendly data format that is fully supported by modern web GIS frameworks such as OpenLayers. This conversion enables efficient client-server communication via standardized web services.

The resulting GeoJSON file was then uploaded to GeoServer, where it was published as a vector layer. GeoServer provides this dataset through a Web Feature Service (WFS), which allows the application to dynamically load, filter, and display individual toilet features based on the current map view and user interaction.

Data Processing Screenshots

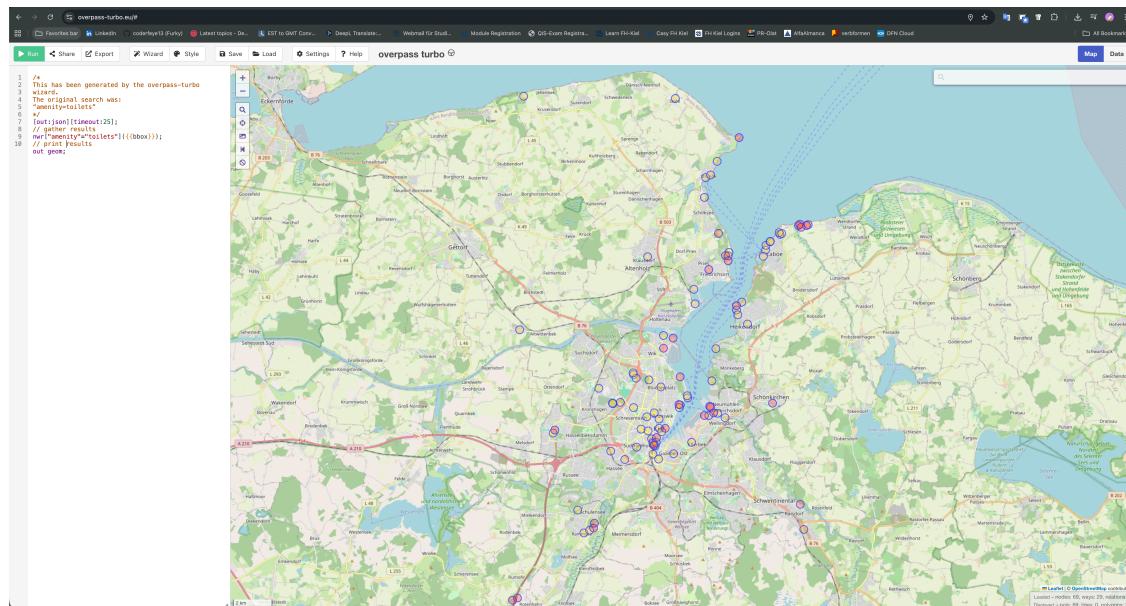


Figure 1: Screenshot of public toilet features extracted from OpenStreetMap using Overpass Turbo.

```
< → C localhost:8080/geoserver/ptf/ows?service=WFS&version=1.1.0&request=GetFeature&typeName=ptf:toilets_kiel...
Pretty-print
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "id": "toilets_kiel.1",
      "geometry": {
        "type": "Point",
        "coordinates": [10.229965, 54.4111242]
      },
      "geometry_name": "geom",
      "properties": {
        "@id": "node/29897960",
        "@access": null,
        "name": "toilets",
        "building": null,
        "seamark:type": "small_Craft_facility:category",
        "seamark:type": null,
        "name": null,
        "name": null,
        "operator": null,
        "source": null,
        "wheelchair": null,
        "wheelchair": null,
        "showers": null,
        "fee": null,
        "opening_hours": null,
        "material": null,
        "toilets:hourly_charge": null,
        "toilets:wheelchair": null,
        "drinking_water": null,
        "addr:houseNumber": null,
        "addr:street": null,
        "check_date": "2025-10-11Z",
        "toilets:disposal": null,
        "building:levels": null,
        "check_date:opening_hours": null,
        "centralkey": null,
        "roof:shape": null,
        "unisex": null,
        "female": null,
        "material": null,
        "toilets:position": null,
        "indoor": null,
        "level": null,
        "website": null,
        "building:type": null,
        "centralkey": null,
        "charge": null,
        "supervised": null,
        "disabled": null,
        "wheelchair:description": null,
        "changing_table:count": null,
        "changing_table:location": null,
        "check_date:wheelchair": null,
        "changing_table:signed": null,
        "entrance": null,
        "note:de": null,
        "addr:floor": null,
        "addr:bedroom": null,
        "baby_feeding:description": null,
        "changing_table:fee": null,
        "changing_table:description": null,
        "hot_water": null,
        "all_access": null,
        "toilets:handwashing": null,
        "toilets:num_chambers": null,
        "toilets:paper_supplied": null,
        "seasonal": null
      }
    }
  ]
}
```

Figure 2: Screenshot of the GeoServer WFS response in GeoJSON format showing public toilet features and attributes.

The screenshot shows a code editor interface with two main panes. The left pane, titled 'EXPLORER', displays a file tree for a project named 'public-toilet-finder'. The tree includes a 'PUBLIC-TOILET...' folder containing 'vscode', 'data', and 'toilets_kiel.geojson'. Under 'data', there are 'dist', 'node_modules', 'public', 'reports', and 'screenshots'. The 'src' folder contains 'assets', 'components' (which further contains 'icons' with files like 'AppFooter.vue', 'AppHeader.vue', and 'MapView.vue'), 'config' (with 'App.vue', 'main.ts', '.gitattributes', and '.gitignore'), and 'index.html'. The right pane, titled 'toilets_kiel.geojson', shows the content of the selected JSON file. The JSON object represents a feature collection for a toilet location in Kiel, Germany, with coordinates approximately at 10.1780802, 54.329994.

```
data > toilets_kiel.geojson > ...
You, 2 days ago | Last author (You)
1   <!-- Created by OSM2JSON - Initial commit - Public Toilet Finder ... -->
2   "type": "FeatureCollection",
3   "generator": "overpass-turbo",
4   "copyright": "The data included in this document is from www.openstreetmap.org. The data is made available under ODbL.",
5   "timestamp": "2025-12-01T14:04:28Z",
6   "features": [
7     {
8       "type": "Feature",
9       "properties": {
10         "@id": "way/26741445",
11         "access": "private",
12         "amenity": "toilets",
13         "building": "yes",
14         "seamark:small_craft_facility:category": "toilets;showers",
15         "seamark:type": "small_craft_facility"
16       },
17       "geometry": {
18         "type": "Polygon",
19         "coordinates": [
20           [
21             [
22               [
23                 [
24                   [
25                     [
26                       [
27                         [
28                           [
29                             [
30                               [
31                                 [
32                                   [
33                                     [
34                                       [
35                                         [
36                                           [
37                                             [
38                                               [
39                                                 [
40                                                   [
41                                                     [
42                                                       [
43                                                       "id": "way/26741445"
44         ],
45       }
46     ]
47   ]
48 }
```

Figure 3: Screenshot showing the structure of the GeoJSON file containing extracted toilet features and attributes.

 **GeoServer**

Logged in as admin. [Logout](#) en

Edit Layer

Edit layer data and publishing **ptf:toilets_kiel**

Configure the resource and publishing information for the current layer

Data Publishing Dimensions Tile Caching Security

Edit Layer

Basic Resource Info

Store Name: toilets_kiel_gpkg3
Native Name: toilets_kiel

Name: toilets_kiel
 Enabled
 Advertised
Title i18n
toilets_kiel
Abstract i18n

Keywords

Current Keywords: features
 Remove selected
New Keyword:
Vocabulary:
 Add Keyword

Metadata links

No metadata links so far
 Add link Note only FGDC and TC211 metadata links show up in WMS 1.1.1 capabilities

Data links

No data links so far
 Add link

Coordinate Reference Systems

Native SRS: EPSG:4326 EPSG:WGS 84...

Declared SRS: EPSG:4326 Find... EPSG:WGS 84...

SRS handling: Force declared

Bounding Boxes

Native Bounding Box

Min X	Min Y	Max X	Max Y
9.9870877	54.27528639999999	10.3020229	54.4443251

Compute from data
Compute from SRS bounds

Lat/Lon Bounding Box

Min X	Min Y	Max X	Max Y
9.9870877	54.27528639999999	10.3020229	54.4443251

Compute from native bounds

Curved geometries control

Linear geometries can contain circular arcs
Linearization tolerance (useful only if your data contains curved geometries)

Feature Type Details

Customize attributes

Property	Type	Nullable	Min/Max Occurrences
geom	Point	true	0/1
id	String	true	0/1
@id	String	true	0/1
access	String	true	0/1
amenity	String	true	0/1
building	String	true	0/1
seamark:small_craft_facility:category	String	true	0/1
seamark:type	String	true	0/1
name	String	true	0/1
note	String	true	0/1
operator	String	true	0/1
source	String	true	0/1
wheelchair	String	true	0/1
wikidata	String	true	0/1
shower	String	true	0/1
fee	String	true	0/1
opening_hours	String	true	0/1
material	String	true	0/1
opening_hours:lastcheck	Date	true	0/1
toilets:wheelchair	String	true	0/1

Figure 4: Screenshot showing the GeoServer layer configuration for the public toilet dataset (ptf:toilets_kiel).

3. System Architecture

The application follows a client–server architecture, where spatial data processing and map services are handled on the server side, while visualization and user interaction take place on the client side. This separation ensures better performance, modularity, and maintainability.

Frontend

The frontend is implemented as a Single Page Application (SPA) using:

- Vue 3
- TypeScript
- HTML5 & CSS3
- Vite
- OpenLayers

HTML is used for the semantic page structure, CSS for responsive layout and visual design, and TypeScript for application logic and map interaction. OpenLayers handles map rendering and communication with map services.

Backend / Data Services

The backend is provided by GeoServer, which publishes the dataset through standard OGC services:

- WMS (Web Map Service) for raster visualization
- WFS (Web Feature Service) for vector-based features and interaction

This ensures full compliance with international OGC standards.

Data Flow

The overall data flow of the system is as follows:

OpenStreetMap → GeoJSON → GeoServer → WMS / WFS → OpenLayers → Vue Application

1. Public toilet data are extracted from OpenStreetMap using amenity = toilets.
2. The data are converted into GeoJSON format.
3. The GeoJSON dataset is published as a vector layer in GeoServer.
4. The layer is made available through WMS and WFS.
5. OpenLayers loads and visualizes the services.
6. Vue manages all user interaction and UI logic.

This architecture provides a scalable and standards-compliant solution for web-based GIS applications.

4. Application Features

The application provides several core functionalities for interactive exploration and accessibility-focused usage.

Interactive Map Display

The web map is implemented using OpenLayers. Users can navigate the map through panning and zooming. The map dynamically updates its content based on the visible area.

Visualization with WMS and WFS

The dataset is visualized using both OGC services:

- WMS is used for raster-based visualization.
- WFS provides vector features in GeoJSON format for interaction.

This hybrid approach enables efficient visualization and advanced feature interaction.

Click-Based Information Popups

Each toilet feature is clickable and displays a popup with detailed information such as:

- Price (free or paid)
- Wheelchair accessibility
- Opening hours
- Gender availability
- Baby changing table availability
- Public access status

Wheelchair Accessibility Filter

A filter allows users to display only wheelchair-accessible toilets (wheelchair = yes). All other features are hidden dynamically without reloading the page.

Custom Map Control Buttons

Custom floating map buttons are implemented using HTML and CSS:

- Zoom In
- Zoom Out
- Recenter Map

All buttons are fully accessible via keyboard and include ARIA labels.

5. Accessibility Implementation

Accessibility is a central requirement of this project. The application was evaluated using the Google Lighthouse audit tool in order to measure performance, accessibility, best practices, and SEO compliance. The test results demonstrate that the application fully complies with modern web standards and accessibility principles.

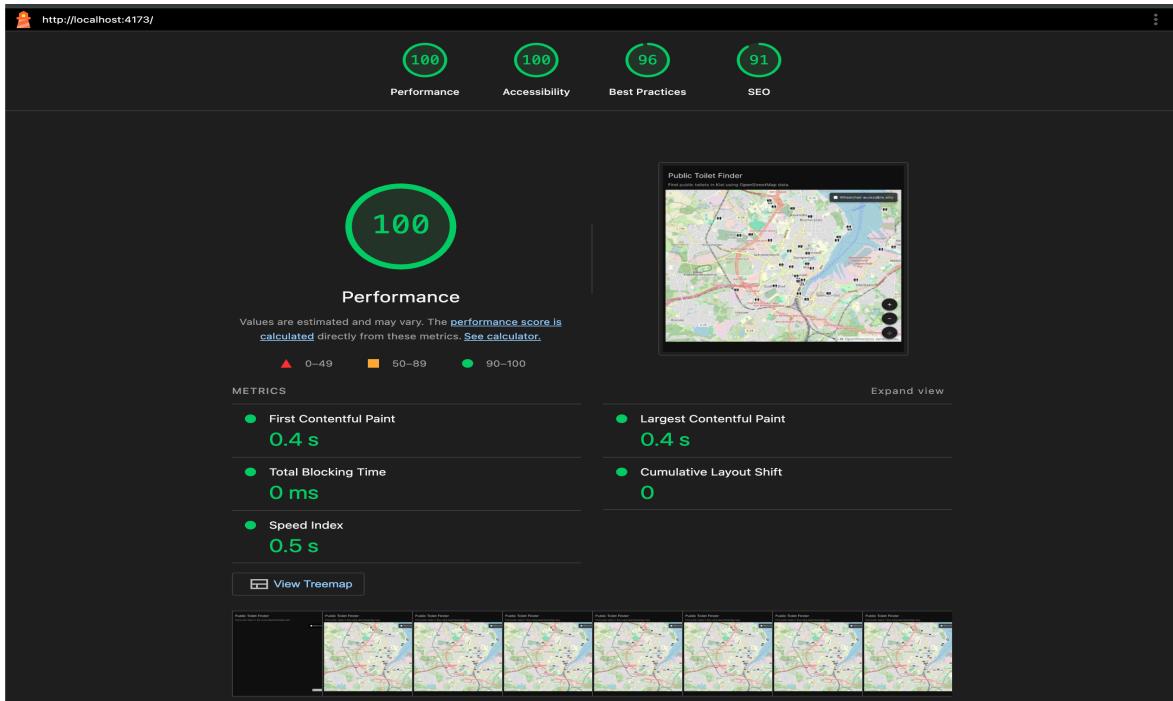


Figure 6: Google Lighthouse audit results showing performance, accessibility, best practices, and SEO scores for the desktop version of the application.

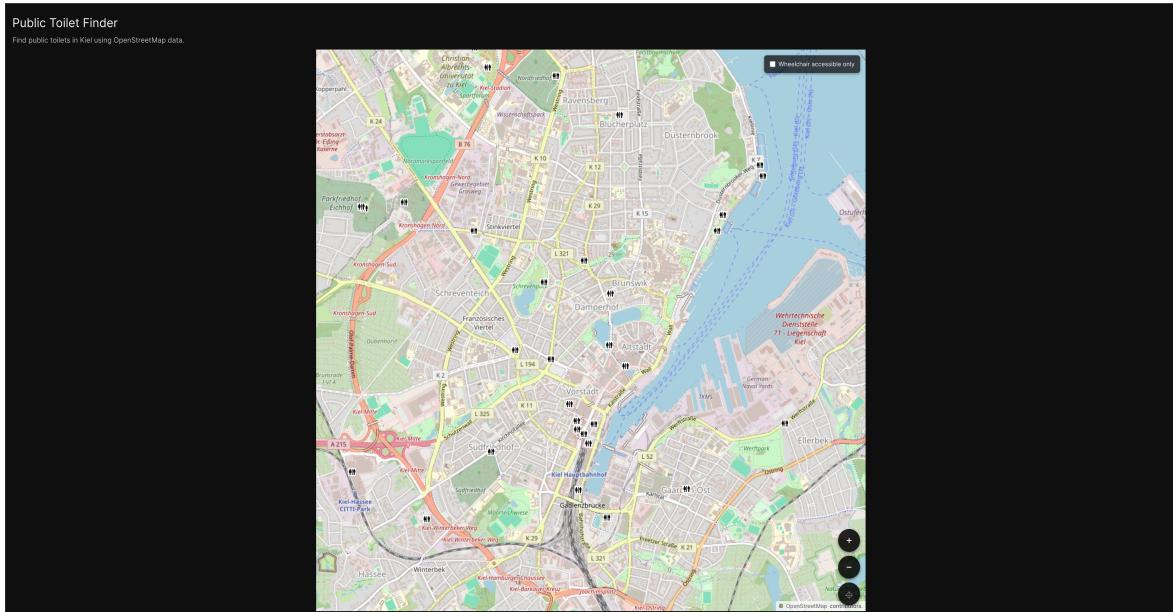


Figure 7: Desktop version of the application tested with Lighthouse showing optimized performance and full feature availability.

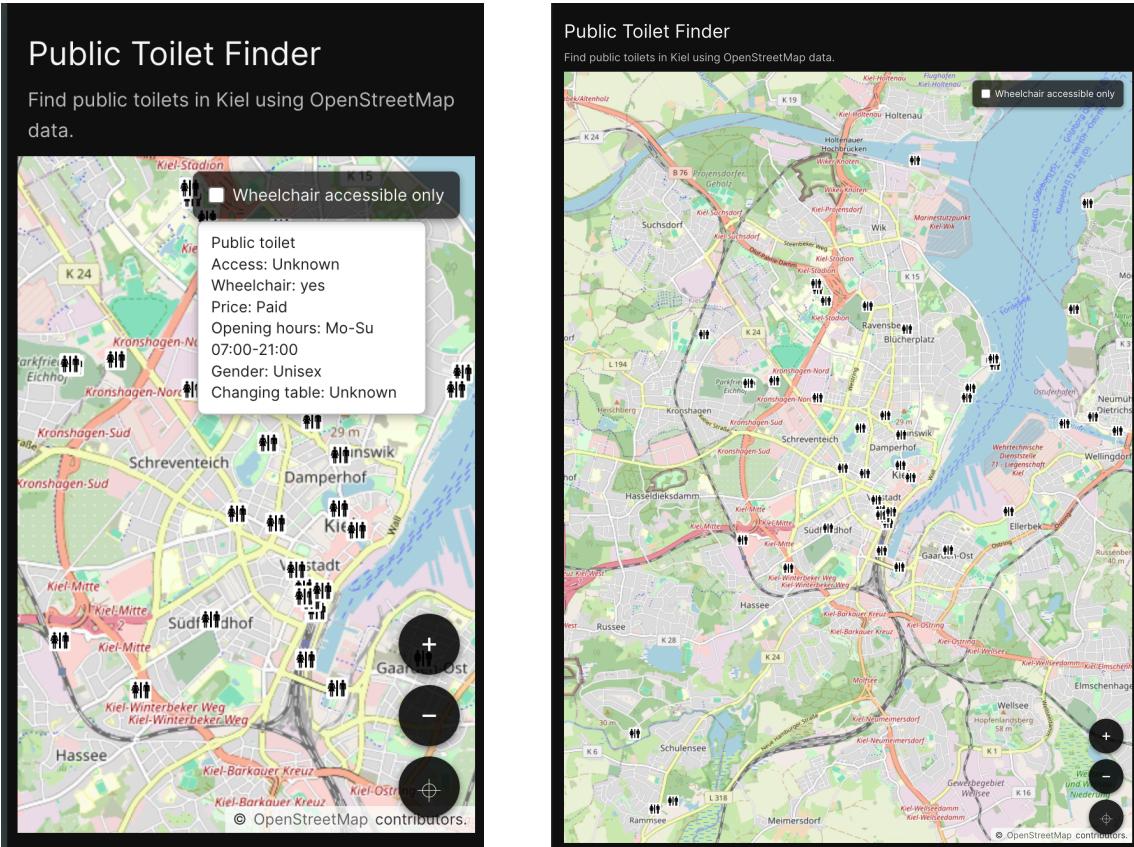


Figure 8: Mobile version of the Public Toilet Finder application tested with Lighthouse showing responsive layout and accessible UI elements.

7. Version Control and GitHub

The project is fully managed using Git as the version control system...

Repository: [GitHub](#)

8. Conclusion and Future Work

This project successfully demonstrates the development of a web-based GIS application for visualizing public toilet locations in the city of Kiel. By integrating OpenStreetMap data, GeoServer services, and an OpenLayers-based frontend within a Vue 3 application, a fully functional and standards-compliant system has been implemented.

The application provides interactive map visualization, detailed feature information, accessibility filtering, and a responsive user interface for both desktop and mobile devices. The high Lighthouse performance and accessibility scores confirm the technical quality of the system.

Future improvements may include additional attribute filters, routing to the nearest toilet, clustering techniques for dense areas, and user-driven data updates through WFS-T.