LocAround: How to make it work?

Software downloading:

pgAdmin
Geoserver
osm2pgsql (PostgreSQL)

Data downloading:

Geofabrik in format .osm.pbf (Karlsruhe's data)
Karlsruhe public transport

Setup of the Routing Server (local)

The first-time setup only needs to be completed once. Afterward, you can use the script to start the server easily.

Replace directory placeholders like ~/graphhopper with the actual paths on your system.

1. Installation (First-Time Setup Only)

- Clone the GraphHopper repository:

```
git clone https://github.com/graphhopper/graphhopper.git ~/graphhopper

cd ~/graphhopper
```

- Build the project:

```
mvn clean package -DskipTests
```

- Download OpenStreetMap (OSM) data:

```
wget https://download.geofabrik.de/europe/germany/karlsruhe-regbez-latest.osm.pbf - karlsruhe-regbez-latest.osm.pbf
```

- (Optional) Download GTFS data for public transport:

```
wget https://example.com/path/to/gtfs-data.zip -O gtfs-data.zip
```

- Edit config.yml and update:

```
datareader.file: karlsruhe-regbez-latest.osm.pbf

gtfs.file: gtfs-data.zip
```

- Import graph data:

```
java -jar web/target/graphhopper-web-*.jar import config.yml
```

2. Starting the Server

After completing the first-time setup, you can simply use the provided script to start the GraphHopper server.

- Navigate to the GraphHopper directory:

```
cd ~/graphhopper
```

- Run the script to start the server:

```
./start-graphhopper.sh
```

Access the server at: http://localhost:8989

How to adjust the data from Geofabrik in pgAdmin ?

osm2pgsql needs to be downloaded. Once it's done, open it to create a database:

```
psql -U postgres
```

Link the PostGIS extension in pgAdmin

```
CREATE DATABASE osm_data;
\c osm_data
CREATE EXTENSION postgis;
```

In PostgreSQL, import data

```
osm2pgsql -c -d osm_data -U postgres -W -S default.style /path/to/yourfile.osm.pbf
```

change "/path/to/yourfile.osm.pbf" to the reel path where you downloaded the file

To verify the import of the data, write this code in pgAdmin

```
SELECT * FROM planet_osm_point LIMIT 10;
SELECT * FROM planet_osm_polygon LIMIT 10;
```

For the table planet_osm_point, you to change the polygon into centroid, those point will be created at the center of the polygon, and it's those point that would appear in the map:

```
ALTER TABLE planet_osm_polygon ADD COLUMN centroid geometry(Point, 4326);

UPDATE planet_osm_polygon
SET centroid = ST_Centroid(way);
```

To change the data to be compatible with our app, you need to apply this changes in pgAdmin:

```
-- For the table planet_osm_point
ALTER TABLE planet_osm_point
ADD COLUMN indoor BOOLEAN,
ADD COLUMN outdoor BOOLEAN,
ADD COLUMN spectator BOOLEAN,
ADD COLUMN actor BOOLEAN;

-- For the table planet_osm_polygon
ALTER TABLE planet_osm_polygon
ADD COLUMN indoor BOOLEAN,
ADD COLUMN outdoor BOOLEAN,
ADD COLUMN spectator BOOLEAN,
ADD COLUMN actor BOOLEAN;

-- Update data in planet_osm_point
UPDATE planet_osm_point
SET indoor = true WHERE amenity IN ('museum', 'restaurant', 'cinema', 'theatre', 'bar', 'cafe',
'library');
```

```
SET outdoor = true WHERE leisure = 'park' OR leisure = 'stadium' OR tourism =
'theme_park';
SET spectator = true WHERE amenity IN ('restaurant', 'cinema', 'theatre', 'bar', 'cafe',
'library') OR tourism = 'stadium';
SET actor = true WHERE amenity = 'museum' OR leisure = 'park' OR tourism = 'theme_park'
OR leisure = 'sports_centre';

-- Update data in planet_osm_polygon
UPDATE planet_osm_polygon
SET indoor = true WHERE amenity IN ('museum', 'restaurant', 'cinema', 'theatre', 'bar', 'cafe',
'library');
SET outdoor = true WHERE leisure = 'park' OR leisure = 'stadium' OR tourism =
'theme_park';
SET spectator = true WHERE amenity IN ('restaurant', 'cinema', 'theatre', 'bar', 'cafe',
'library') OR tourism = 'stadium';
SET actor = true WHERE amenity = 'museum' OR leisure = 'park' OR tourism = 'theme_park'
OR leisure = 'sports_centre';
```

How to put your table in Geoserver ?

The app will run locally. After changing the data in the table and downloading Geoserver correctly, it can be opened.

In order to change any server settings or configure data, a user must first be authenticated. Navigate to the upper right of the web interface to log into GeoServer. The default administration credentials are:

User name: admin
Password: geoserver

The next step is to create a workspace for the data. A workspace is a container used to group similar layers together. After logging in, navigate to **Data > Workspaces**.
Click the Add new workspace button.

You will be prompted to enter a workspace Name and Namespace URL.
Enter the Name as data and the Namespace URL as http://geoserver.org/data.

change /data by the name given to the data

Click the Submit button. The workspace will be added to the Workspaces list.

Navigate to **Data > Stores**
Click on Add new Store

Navigate to **Data > Layers**
Click Add a new resource.
From the New Layer chooser menu, select data : datatable

Finalize the layer configuration by scrolling to the bottom of the page and clicking Save.
In order to verify that the data layer is published correctly, we can preview the layer.
Navigate to the Layer Preview screen and find the data : datatable layer.
Click the OpenLayers link in the Common Formats column.
An OpenLayers map will load in a new tab and display the shapefile data with the default line style. You can use this preview map to zoom and pan around the dataset, as well as display the attributes of features.

For more help, everything is explain here

How does the frontend work ?

Things to know:

If the logo is not displaying, ensure that the image is located in the root directory of your code file. If the path is not showing, check the routing server section for any issues. The location button might have problems depending on the internet connection and how it's linked to the servers.

Main steps:

First, the user needs to click on the map or use the "use my location" button to show where they are or where they want to start. Next, they must choose the activities they want to do, such as visiting museums or parks. Then, they select any filters, like whether they want an indoor or outdoor activity. Finally, they click the "search" button to confirm their choices.

Display:

Once the user selects the filters and activities they want, points will appear on the map. These points represent the buildings and locations the user chose. The location information will be shown above each point. If the user wants to go to one of the locations, they need to click on the corresponding point. Four possible routes will then be shown: walking, biking, driving, and public transport. The user can choose one of these routes to reach their destination.