

## Assignment 2

### Map Awareness

**Course:** Mobile App Development, WiSe 2025/26

**Group name:** Map Awareness

Pratik Kumar: general UI

Marvin Dorsch: Routing

Gabriel Glaser: Location & Warnings and UI & UX improvements

## Routes

The Routes screen presents two options to choose from, the first is to plan a route, and the second option is to switch to the already saved routes. Calculating a new route is kept simple it only requires a starting location and a destination. The route is being calculated by an API called "Graphhopper", which first converts the input of the starting and ending points to longitude and latitude coordinates using the "Geocoding" services. Then using the "Routing" services it calculates the Route. The API returns a collection of Polyline points and street names. With the street names, we can determine which "Autobahn" is on the route. With the names of the "Autobahn" we can use the "Autobahn API" to get a list of all the roadworks on the Autobahn. Using the Polyline coordinates and the coordinates of the roadworks from the API we can filter out the roadworks which are not on the route. Once the route is calculated it can be expanded to see a list of widgets which represent the collection of roadworks on the route.

The Routes Screen also shows a quick summary of the Weather conditions and Weather warnings.

Adding a Gemini API key grants the option to generate an AI summary on the Weather and Routes conditions.

The last button on the screen is a redirection to the Map screen.

## Warnings and Locations

The Warning screen shows all Weather warnings from the "Deutscher Wetter Dienst" API in a specified Location or using the phones GPS data on your current location in a radius from 1 to 100km.

We also provides the option to get an Summary from Gemini, once the user provides an API key.

The weather warnings are color coded from red as storm warnings and similar dangerous warnings to orange for less dangerous warnings.

## Map

If the user selected a route, the map will show the route and the roadworks along it. Otherwise it shows the radius from the Warning screen.

## Challenges

The first problem was calculating the route and getting the coordinates from the starting and end points. But thanks to Graphhopper calculating the route was resolved quickly, and using the Geocoding service it also cleared the problem of getting the coordinates.

We also ran into an issue with GitHub where we could not merge the feature branch of the Warnings and Locations feature back together with the main branch. We resolved this by simply using the branch as the new main branch. That means our latest version off the App is in the "working-version,-new-merge" branch