GeoE3 – Use Case #2:
Using Public Open Data for Electric vehicles and trucks

Requirements and Architecture

GEOE3 Aventi

Picture: Norsk elbilforening

Revision	Change	Approved by project manager		Approved by project owner	
1	New document	Bjørn Elnes	30.06.2022	N/A	N/A
2	Adding functionality	Bjørn Elnes	02.01.2024	N/A	N/A
3	Finalized	Bjørn Elnes	27.02.2024	N/A	N/A

Co-funded by the European Union

Table of Contents

1 INTRODUCTION

This cross-domain use case provides benefits in the battle against climate change: electric vehicles and automated driving. It has long been known that automation that exploits geographic information reduces energy consumption. Benefits can be realized as electric vehicles calculate how much power they will use to reach the top of a mountain pass, and how much they can recover when they descend the mountain on the other side. It will also be useful as part of the ADAS (Automated Driver Assist System) in heavy goods vehicles (HGV), where the cruise control can take into account upcoming slopes and sharp curves on the road before shifting gear.

In this project we will provide the vehicles with data make the above-mentioned calculations.

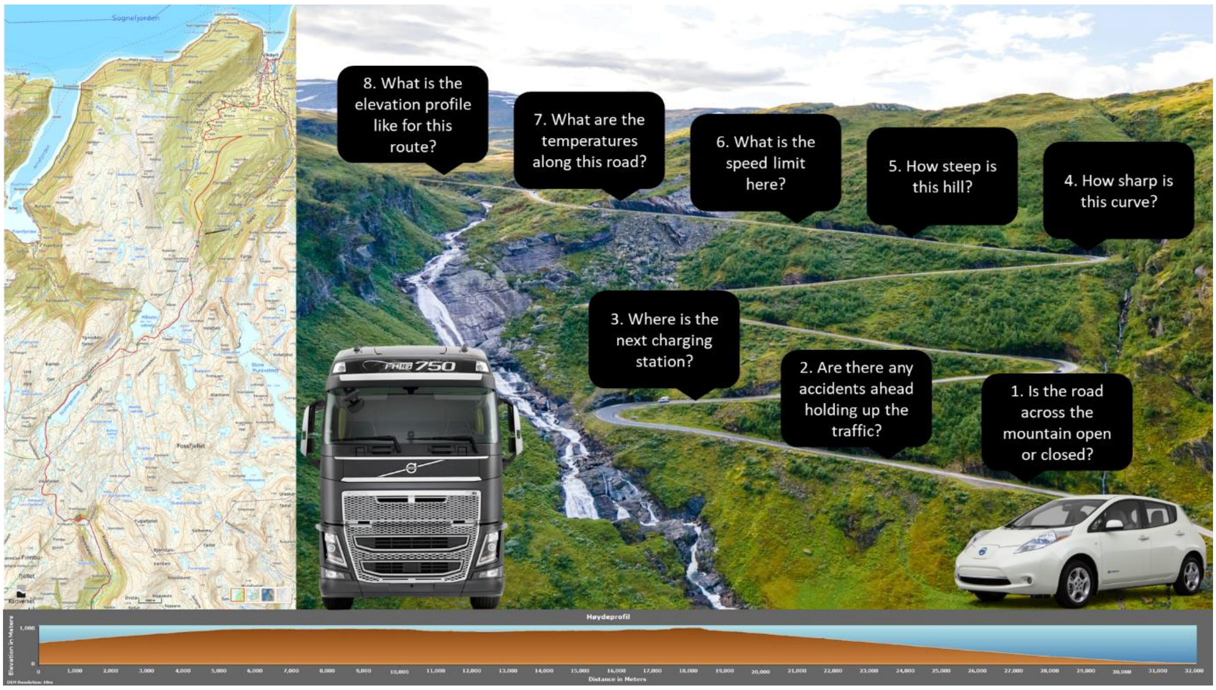


Figure 1: Types of information sent to the vehicles.

This will be achieved by concentrating relevant **public open data** and passing this on to the vehicles.

The development of accurate range calculations for electric vehicles, predictive cruise control for diesel trucks, and sharp-curve-warnings are complex endeavors. In this project we will only implement some simple proof-of-concepts showing how public open data can be used in these cases. For more information about the complex nature of these challenges, see these articles:

- Calculate correct driving range for electric vehicles under various Nordic conditions:
  - https://www.motor.no/aktuelt/dette-stjeler-rekkevidde-fra-elbiler-se-triksene/222152
  - https://www.recurrentauto.com/research/winter-ev-range-loss
  - https://www.naf.no/elbil/bruke-elbil/elbilen-kan-bremse-uventet-pa-glatt-fore
- Using topography data to control the powertrain for trucks:
  - https://youtu.be/lX0iOZVN_L0?si=aU3wMagEoL448223
  - https://www.oemoffhighway.com/electronics/smart-systems/press-release/21138173/mercedesbenz-10-facts-about-predictive-cruise-control-in-trucks

- Determine correct vehicle speed for negotiating sharp curves. You can read more about sharp and dangerous curves here, including two reports that contradict each other:
    - https://vegvesen.brage.unit.no/vegvesen-xmlui/handle/11250/2614570
    - https://tsr.international/TSR/article/view/25044
    - https://www.vegvesen.no/globalassets/fag/veg-og-gate/trafikkskilt-og-vegoppmerking/skilting-av-farlige-kurver.pdf

2 FUNCTIONAL REQUIREMENTS

Public Open Data will be gathered from sources described in section 2.1, 2.2, 2.3 and 2.4. This data will be used as described in section 2.5 and 2.6.

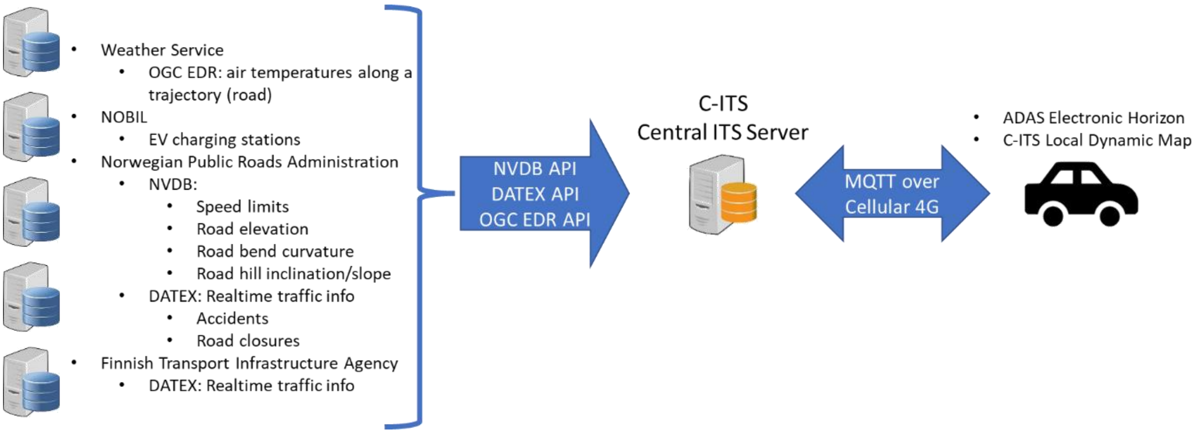


Figure 2: Public Open Data APIs used to provide relevant information for the vehicles.

2.1 OGC EDR – OUTDOOR AIR TEMPERATURE

[*This was implemented in 2022-23*]

Specification for OGC EDR Trajectory query can be found here:
https://docs.ogc.org/is/19-086r6/19-086r6.html#_5181b3c7-ada4-40d9-bcf0-4c890a6087f1

For the GeoE3 project, the Finnish Meteorological Institute has implemented functionality for OGC EDR air temperatures along a trajectory (road/route), using a query like this:
https://opendata.fmi.fi/edr/collections/pal_skandinavia/trajectory?coords=LINESTRING(24.4839+60.9723,24.6864+61.0426,24.8562+61.1020,25.0196+61.0436,25.0196+60.9279,24.9738+60.8516)&datetime=2023-12-28T19:00:00Z&parameter-name=Temperature

2.2 NVDB - ROAD MEASUREMENT PROPERTIES

The following road measurements must be collected from NVDB by the Central ITS Server.

*(For detailed description of the NVDB objects and their properties, go to the data dictionary, click on "˅**Se detaljer**", and then click on a property.)*

2.2.1 Speed limit

[*This was implemented in 2022-23*]

NVDB object name:
105 - Fartsgrense (Speed limit)

Object properties relevant for this project:

A. 19885 – 5 (Speed limit 5 km/h)
B. 11576 - 20 (Speed limit 20 km/h)
C. 2726 - 30 (Speed limit 30 km/h)
D. 2728 - 40 (Speed limit 40 km/h)
E. 2730 - 50 (Speed limit 50 km/h)
F. 2732 - 60 (Speed limit 60 km/h)

G. 2735 - 70 (Speed limit 70 km/h)
H. 2738 - 80 (Speed limit 80 km/h)
I. 2741 - 90 (Speed limit 90 km/h)
J. 5087 - 100 (Speed limit 100 km/h)
K. 9721 - 110 (Speed limit 110 km/h)
L. 19642 - 120 (Speed limit 120 km/h)

Data dictionary:
https://datakatalogen.atlas.vegvesen.no/#/105%20-%20Fartsgrense

Example:
https://vegkart.atlas.vegvesen.no/#kartlag:geodata/@265615,6654298,14/hva:!(id~105)~/valgt:91486307:105

2.2.2 Road sign – speed limit
[*This was implemented in 2022-23*]

NVDB object name:
96 – Skiltplate (sign plate)

Object properties relevant for this project:
A. 11578 - 362.20 - Fartsgrense 20 km/t
B. 8872 - 362.30 - Fartsgrense 30 km/t
C. 8873 - 362.40 - Fartsgrense 40 km/t
D. 8874 - 362.50 - Fartsgrense 50 km/t
E. 8875 - 362.60 - Fartsgrense 60 km/t
F. 8876 - 362.70 - Fartsgrense 70 km/t
G. 8877 - 362.80 - Fartsgrense 80 km/t
H. 8878 - 362.90 - Fartsgrense 90 km/t
I. 8879 - 362.100 - Fartsgrense 100 km/t
J. 11816 - 362.110 - Fartsgrense 110 km/t
K. 12146 - 364.20 - Slutt på særskilt fartsgrense 20 km/t
L. 9155 - 364.30 - Slutt på særskilt fartsgrense 30 km/t
M. 9156 - 364.40 - Slutt på særskilt fartsgrense 40 km/t
N. 9157 - 364.50 - Slutt på særskilt fartsgrense 50 km/t
O. 9158 - 364.60 - Slutt på særskilt fartsgrense 60 km/t
P. 21392 - 364.70 - Slutt på særskilt fartsgrense 70 km/t

Data dictionary:
https://datakatalogen.atlas.vegvesen.no/#/96%20-%20Skiltplate

Example:
https://vegkart.atlas.vegvesen.no/#kartlag:geodata/@265745,6654204,14/hva:!(filter~!(operator~**E~type*_id~5530~verdi~!8873)~id~96)~/valgt:1014756523:96

Image files (Forbudsskiltene):
https://www.vegvesen.no/fag/veg-og-gate/trafikkskilt-og-vegoppmerking/filer-og-fargekoder-for-trafikkskilt/

2.2.3 Road elevation

[*This was implemented in 2024*]

NVDB object name:
640 - Kurvatur, vertikalelement (vertical element curvature)

Object properties relevant for this project:

A. 5831 - Høyde start (elevation start)
B. 5832 - Høyde slutt (elevation end)

Data dictionary:
https://datakatalogen.atlas.vegvesen.no/#/640%20-%20Kurvatur,%20vertikalelement

Example:
https://vegkart.atlas.vegvesen.no/#kartlag:topo4/@265693,6654553,15/hva:!(id~640)~/valgt:326516146:640

2.2.4 Point elevation

[*This was implemented in 2024*]

NVDB object name:
642 - Kurvatur, vertikalpunkt (vertical point curvature)

Object properties relevant for this project:

A. 8320 - Toppunkt, høybrekk (top point of vertical curve)
B. 8321 - Bunnpunkt, lavbrekk (bottom point of vertical curve)
C. 9108 - Markert knekkpunkt, høy (significant high break point of vertical curve)
D. 9109 - Markert knekkpunkt, lav (significant low break point of vertical curve)
E. 9110 - Bruddpunkt (break point between two vertical curves)
F. 5845 - Høyde (elevation - given in meters above sea level)

Data dictionary:
https://datakatalogen.atlas.vegvesen.no/#/642%20-%20Kurvatur,%20vertikalpunkt

Example:
https://vegkart.atlas.vegvesen.no/#kartlag:topo4/@265693,6654553,15/hva:!(id~642)~/valgt:326531459:642

2.2.5 Road inclination

[*This was implemented in 2022-23*]

NVDB object name:
825 - Kurvatur, stigning (inclination curvature)

Object properties relevant for this project:

A. 9396 - Stigning (inclination)

Data dictionary:
https://datakatalogen.atlas.vegvesen.no/#/825%20-%20Kurvatur,%20stigning

Example:
https://vegkart.atlas.vegvesen.no/#kartlag:geodata/@265676,6654537,14/hva:!(id~825)~/valgt:1013848392:825/vegsystemreferanse:242391.06:6585544.419

2.2.6 Road curvature

[*This was implemented in 2022-23*]

NVDB object name:
639 - Kurvatur, horisontalelement (horizontal element curvature)

Object properties relevant for this project:

A. 5842 - Type
B. 5827 - Radius

Data dictionary:
https://datakatalogen.atlas.vegvesen.no/#/639%20-%20Kurvatur,%20horisontalelement

Example:
https://vegkart.atlas.vegvesen.no/#kartlag:geodata/@265603,6654426,14/hva:!(id~639)~/valgt:1013848522:639

2.3 DATEX

The DATEX messages are translated to C-ITS DENM in JSON format by the Central ITS Server before being sent via MQTT to the Android devices.

2.3.1 DATEX – Norway

[*This was implemented in 2022-23*]

NPRA DATEX specification (NPRA_DATEXII_3_1_Specification.pdf):
https://git.vegvesen.no/projects/DATEX2/repos/datex2-spesifications/browse/3.1

DATEX publications relevant for this project:

A. 2.2 - Weather
B. 2.4 - Road Traffic Information

Example:
https://www.vegvesen.no/trafikk/kart#/tra/NPRA_HBT_25-05-2023.75552?lat=59.95376&lng=10.78792&zoom=11&layer=fer,tra,ctv,tfl

2.3.2 DATEX – Finland

[*This was implemented in 2022-23*]

Following API endpoints are used to fetch DATEX in Finland

DATEX messages - https://tie.digitraffic.fi/api/traffic-message/v1/messages.datex2

Situation locations - https://tie.digitraffic.fi/api/traffic-message/v1/locations

2.4 EV CHARGING STATIONS

[*This was implemented in 2024*]

NOBIL’s API provides information about electric vehicle charging stations in Norway (and Sweden): https://info.nobil.no/api

JSON-data relevant for this project (to be reviewed when we get API access):

A. “geolocation” or “Position”
B. “id”
C. “active”
D. “name” or “owner” or “Owned_by”
E. “address” or “Street” and “House_number”
F. “zip”
G. “city”
H. “countrycode”
I. “placetype”
J. “accessibility”
K. “usercomment” or “Description_of_location”
L. “Available_charging_points”
M. “Connector”
N. “chargerspeed” or “Chargingcapacity”

Explanation of available information can be found here: https://info.nobil.no/innhold

Here is an overview of other smartphone apps and services using this API from NOBIL: https://info.nobil.no/index.php/tjenester

Currently this API does not provide live data about operational status and vacancy for charging stations, but that may be offered by mid-2024. You can read more about it here: https://info.nobil.no/nyheter/important-update-on-nobil-data

2.5 ANDROID APP FEATURES

The android app will help drivers of electric vehicles and trucks to cross the mountains in a more predictable manner, answering the following questions:

- Is the road across the mountain open or closed?
- Are there any accidents ahead holding up the traffic?
- Where is the next charging station?
- How sharp is this curve?
- How steep is this hill?
- What is the speed limit here?
- What are the temperatures along this road?
- What is the elevation profile like for this route?

This video shows what the Android app looked like on the 30th of April 2023:
https://youtu.be/b7RxJ3cX0rQ?t=196

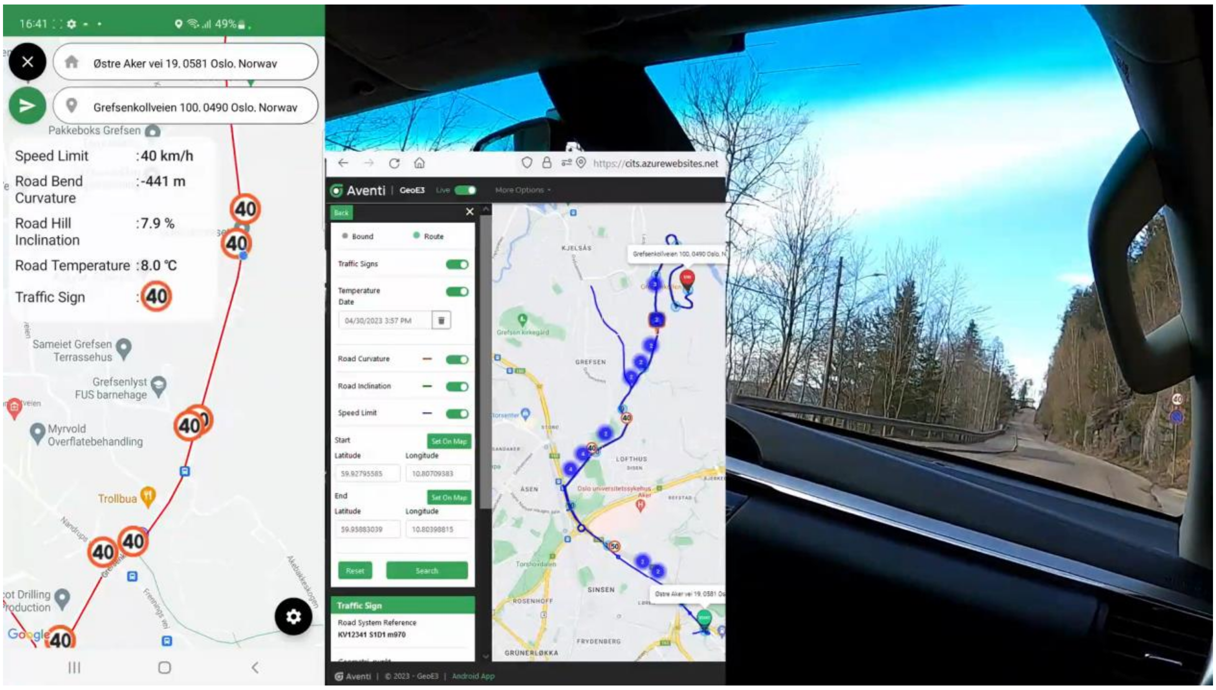


Figure 3: The Android app and Central ITS Server as of 30th of April 2023

[The subsections below describe features and functionality that was implemented in January and February of 2024]

2.5.1 UX Design

For this project, we will only run the GeoE3 app on standalone Android tablets and Android phones, and not Android Auto or Android Automotive OS. However, the UX design should consider aspects from Google Design for Driving with respect to “look & feel”, simplifying future compliance with Android Auto/Automotive-OS:

- https://youtu.be/ii7PGDG0G5g?si=g9iqapvpzVBmySbf
- https://developers.google.com/cars/design

2.5.2 Map display self-vehicle

A. The self-vehicle shall normally be displayed automatically in the middle near the bottom of the map while driving (and the map viewing angle is in perspective-mode). Here are examples from Google Maps and Waze:
https://youtu.be/LpFJRjNKyPE?t=259

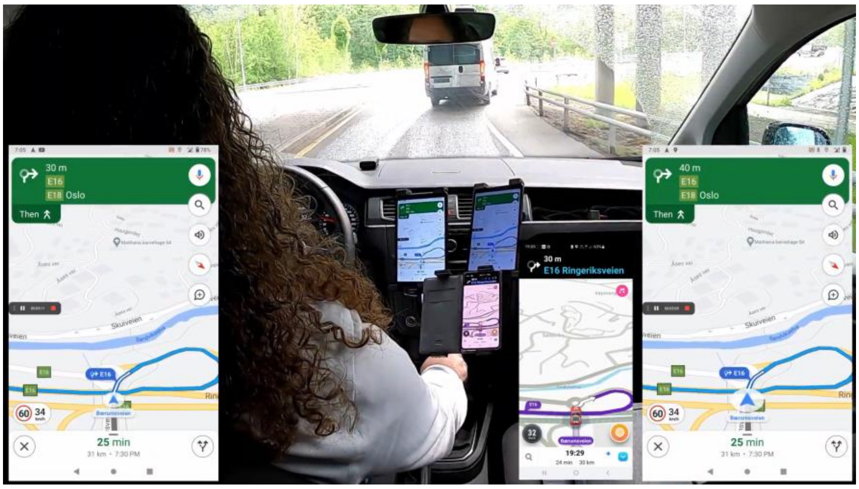


Figure 4: Example of self-vehicle in Google Maps and Waze

B. It should be possible to manually zoom in and out of the map, and move it to other locations, but it must be easy to return to the mode described in A above.

2.5.3 Map display viewing angles

The following Google Map display features for viewing angles should be supported:

A. North-up
B. Overview
C. Perspective

Here are examples from Google Map in Android Auto:
https://youtu.be/crew4Khugfk?t=116

For more information, see this:
https://developers.google.com/maps/documentation/android-sdk/views

2.5.4 Enable/disable GeoE3 features

2.5.4.1 Enable Electric Vehicle or Truck

In the settings of the Android app, there will be two radio buttons where the user can enable the following options with sub-settings:

A. Option for enable/disable GeoE3 range calculation for Electric Vehicle (default on)
   a. Setting for maximum charging limit for the battery, 70% to 100% (default 80%)
   b. Setting for minimum discharge limit for the battery, 10% to 30% (default 20%)
   c. Setting for battery size in kWh, 20 to 200 (default 70). The user may also pick vehicle from this list: https://ev-database.org/cheatsheet/useable-battery-capacity-electric-car
   d. Setting for Wh/km, 50 to 500 (default 200). The user may also pick vehicle from this list: https://ev-database.org/cheatsheet/energy-consumption-electric-car
   e. Setting for lost kWh per 1000 meter increasing altitude, 0 to 20 kWh (default 6 kWh)
   f. Setting for gained kWh per 1000 meter decreasing altitude (regenerative braking), 0 to 20 kWh (default 3 kWh)

g. Setting for lost driving range in percent per 10 °C outdoor air temperature below 20 °C, 0% to 50% (default 20%)
   for more info: https://www.geotab.com/blog/ev-range-impact-of-speed-and-temperature/

B. Option for enable/disable GeoE3 predictive cruise control warnings for truck (default off)
   a. Settings for steep incline warning,
      i. Steep incline greater than, 2% to 25% (default 7%)
      ii. Distance before incline, 0 m to 500 m (default 100 m)
   b. Settings for sharp curve warning
      i. Sharp curve radius less than, 0 m to ±500 m (default ±200)
      ii. Distance before curve, 0 m to 500 m (default 100 m)

2.5.4.2 Options for enabling route information

In the settings of the Android app, there will be eight radio buttons where the user can enable the following options:

A. Outdoor air temperature (default on) according to 2.1
B. Speed limit (default on) according to 2.2
C. Speed limit sign (default on) according to 2.2
D. Road elevation (default on) according to 2.2
E. Point elevation (default on) according to 2.2
F. Road inclination (default on) according to 2.2
G. Road curvature (default on) according to 2.2
H. Traffic information (default on) according to 2.3
I. EV charging stations (default on for EV, default off for truck) according to 2.4

2.5.5 GeoE3 functionality common for both Electric Vehicle and Truck

2.5.5.1 Lay in a trip

Before the trips starts, the driver must enter the destination and optionally the starting point:

A. It must be possible to type in a destination.
B. The starting point must default to the current location, but it must also be possible to type in a starting location in the form of a place or street address.
C. Enabled route information from 2.5.4.2 will be downloaded from start to destination along the route.
D. If 2.5.4.1 is enabled then
   a. EV charging stations will be included if they have been enabled in 2.5.4.2 and are located less than 5 km from the main route.
   b. The driver will be asked to enter the current SOC (State Of Charge) of the battery in percent.

2.5.5.2 Overview map of trip

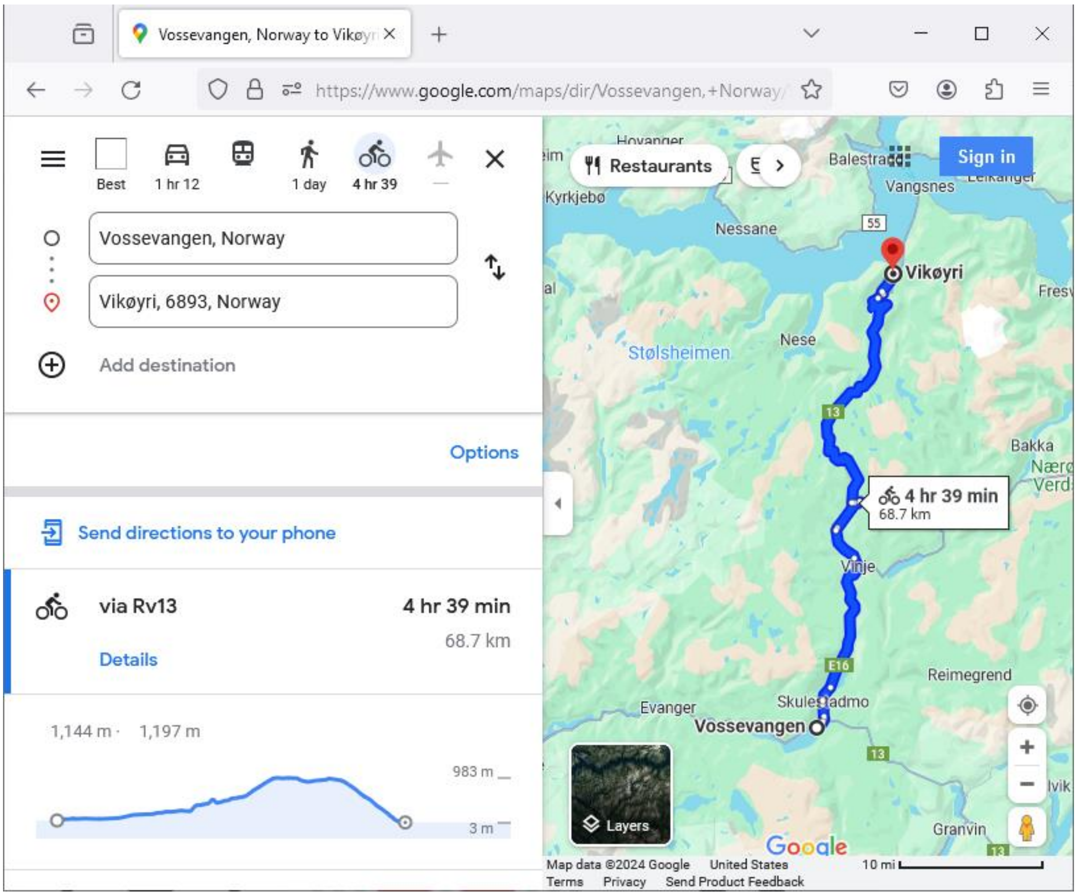
For a trip laid in according to 2.5.5.1, an overview map of the route will be available showing the following information:

A. Starting point and destination
B. EV charging stations according to 2.5.5.1.D (if option for electric vehicles is enabled) where suggested stops at some of the EV charging stations are highlighted according to the range calculation in 2.5.6.
C. Steep inclines according to 2.5.4.1.B.a.i (if option for trucks is enabled)
D. Locations with air temperatures at or below 0 °C

2.5.5.3 Elevation profile of trip

Based on 2.5.4.2.D and 2.5.4.2.E (alternatively similar elevation data from Google Map API), for a trip laid in according to 2.5.5.1, a height profile of the route will be available showing the following:

A. Line chart with elevations along the y-axis and kilometers along the x axis
B. Starting point at the beginning of the chart and destination at the end of the chart
   Here is an example of elevation profile from Google Maps (for bicycle):



2.5.5.4 Pop-up information while driving

While driving, the information enabled in 2.5.4.2 will pop-up in the map:

A. This will be done in the same manner as in the Android app of 30th of April 2023
B. However, the look-and-feel will be more in line with 2.5.1
C. It will be possible to hide or unhide the pop-up information box.

2.5.6 GeoE3 range calculation for Electric Vehicle

Range calculations will be based on the information in 2.5.4.1.A and 2.5.5.1:

A. The app will plan a trip with stops at relevant charging stations along the route of 2.5.5.1, making sure the battery's SOC always will be below 2.5.4.1.A.a and above 2.5.4.1.A.b.
B. The calculation will take into consideration the size of the battery in 2.5.4.1.A.c and how many kilometers this will last (within the limits of 2.5.4.1.A.a and 2.5.4.1.A.b) with the power usage in 2.5.4.1.A.d.
C. The range will be reduced according to increasing altitude in 2.5.4.1.A.e based on elevation profile in 2.5.5.3.
D. The range will be extended according to decreasing altitude in 2.5.4.1.A.f based on elevation profile in 2.5.5.3.
E. The range will be reduced according to outdoor air temperatures in 2.5.4.1.A.g based on 2.5.4.2.A (ideally this would be updated as the trip progresses and temperatures change, but that will be a future feature for the app).

2.5.7 GeoE3 predictive cruise control warnings for truck

While driving, the warnings configured in 2.5.4.1.B will be activated when relevant:

A. Based on 2.5.4.2.F, when the truck is at a distance of 2.5.4.1.B.a.ii before an incline of 2.5.4.1.B.a.i, then pop up a warning message combined with text-to-speech: "Steep incline ahead".
B. Based on 2.5.4.2.G, when the truck is at a distance of 2.5.4.1.B.b.ii before a curve of 2.5.4.1.B.b.i, then pop up a warning message combined with text-to-speech: "Sharp curve ahead" (make sure negative and positive curve values are treated the same way.

2.6 Central ITS Server features

The Central ITS Server for this project will build on the Central ITS Server that Aventi has developed for the Norwegian Public Roads Administration for the Nordic Way projects E8 Boralis, E6 Patterød, and Central C-ITS Server (also called ETSI-translator). It also builds on Aventi's project Pilot-T ASAM, funded by the Research Council of Norway. The Central ITS server for E8 Borealis and E6 Patterød had been merged, and is now called Norway Central ITS Server, and together they have been running more or less continuously for five years.

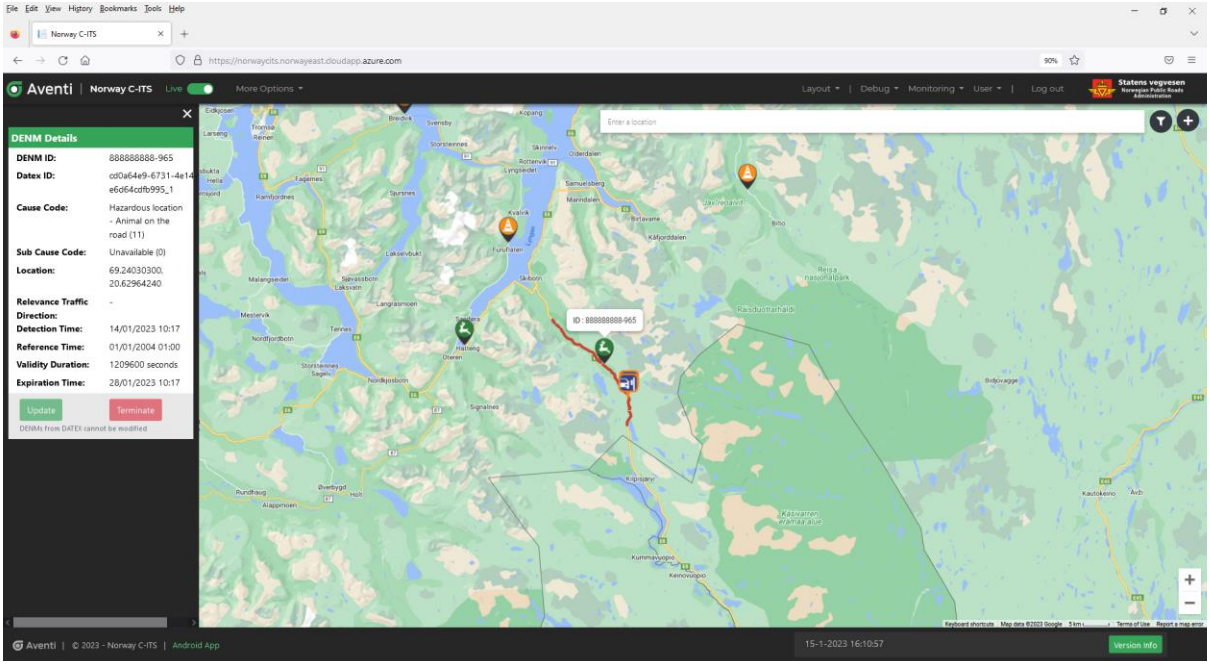


Figure 5: Norway Central ITS Server

2.6.1 Azure tenant

Aventi's Microsoft Azure tenant will be used for this project.

2.6.2 Azure Active Directory

Set up Azure Active Directory for the new Azure tenant and add necessary users.

2.6.3 Set up a copy of Central ITS Server

The Central ITS Server from the previous Pilot-T project will be used in this project, which runs all services in Kubernets. It shows CAM for all vehicles with the same blue icons and side bar that we currently use for all projects.

The MQTT quadtree/quadkey-map-tile structure for topics on the MQTT broker will remain as currently developed for Aventi's Central ITS Server.

2.6.4 Implement GeoE3 functionality

2.6.4.1 Display available and relevant Public Open Data

From the Central ITS Server, it will be possible to display the following information along a route:

A. OGC EDR air temperature along a route-trajectory as described in section 2.1.
B. NVDB road measurements as described in section 2.2.
C. DATEX messages as described in section 2.3.
D. EV charging stations as described in section 2.4.

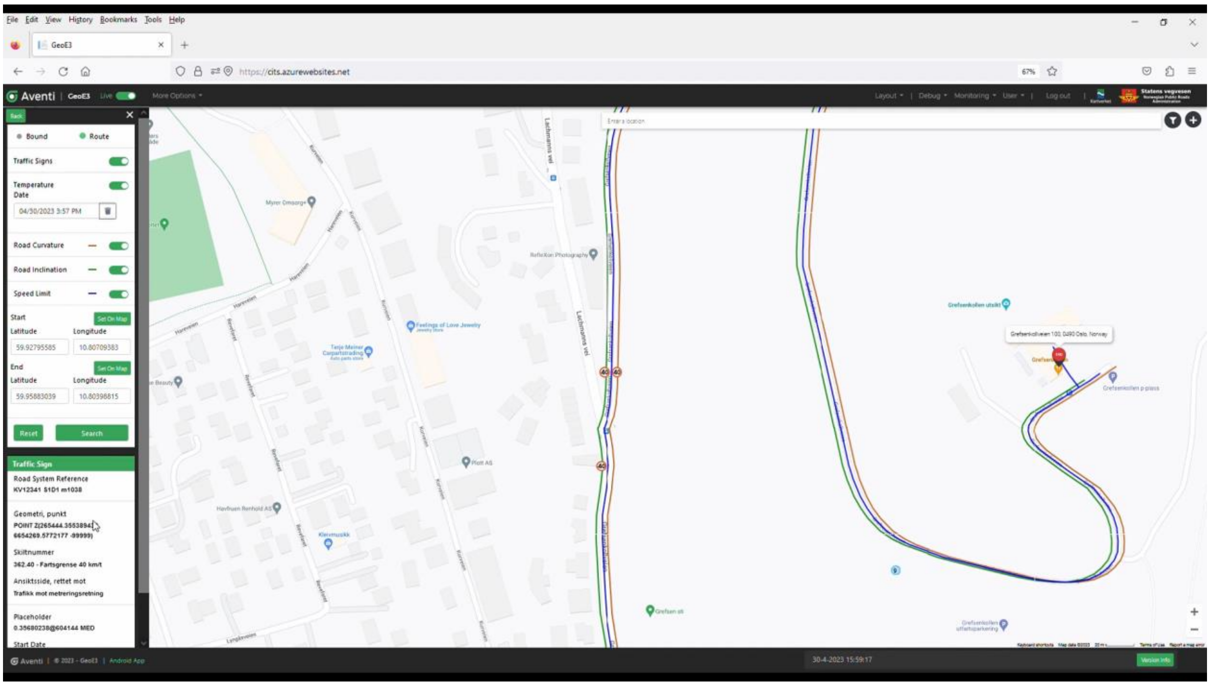


Figure 6: Displayed GeoE3 information as of April 30th, 2023
[Elevation data from NVDB and EV charging stations from NOBIL was added in 2024]

3 INTERFACE REQUIREMENTS

External interface requirements are types of functional requirements that ensure the system will communicate properly with external components, such as:

3.1 USER INTERFACES

1. The smartphone app will be running on an Android Device
2. The user interface for the Central ITS Server will be a web page (see figures in section 2.6)

3.2 HARDWARE INTERFACES

1. The SmartPhone app will be running on an Android Device
2. The Central ITS Server will be running in Kubernetes Cluster which will run on a Virtual Machine Scale Set in Microsoft's Azure datacenter in Oslo. There will be no hardware located along the roads or at the Aventi office.

3.3 SOFTWARE INTERFACES

1. The smartphone app will connect to:
   a. Central ITS Server via MQTT
   b. Central ITS Server web page for app download
   c. Google Map API and the.
   d. NVDB API at the Norwegian Public Roads Administration (Through Central ITS Server)
   e. OGC EDR API at the Finnish Meteorological Institute (Through Central ITS Server)
2. The Central ITS Server will connect to
   a. Google Map API and the Norwegian Mapping Authority API.
   b. NVDB API at the Norwegian Public Roads Administration
   c. DATEX API at the Norwegian Public Roads Administration
   d. DATEX API for Finland
   e. OGC EDR API at the Finnish Meteorological Institute

3.4 COMMUNICATION INTERFACES

1. The smartphone app will connect to its software interfaces via
   a. Internet via cellular connection
   b. Internet via Wifi
2. The Central ITS Server will connect to its software interfaces via
   a. Internet via cloud solutions

4 NONFUNCTIONAL REQUIREMENTS

4.1 PRIVACY

A. Drivers participating in this project must give consent to have their driving location tracked and monitored.
B. When installing the app on Android devices, it will ask for several permissions, but we consider this safe.

4.2 SECURITY

A. User login to the Azure tenant requires 2-factor-authentication.
B. All exposed API interfaces from the Central ITS Server requires authentication.
C. The Central ITS Server’s MQTT broker requires authentication, must not use default username and password, and must in addition use PKI certificates.
D. It must not be possible to access any public IP addresses listed in Azure-portal without proper authentication.
E. Any password provided to Aventi for accessing public APIs must not be exposed to the public.
F. MongoDB Atlas access must not be accessible from “anywhere” but must instead use IP-address whitelisting or VPN. All access must require authentication.

4.3 CAPACITY

A. Android app: Capable of handling data and calculations as described in sections 2.1 and 2.5.
B. Central ITS Server: Capable of handling data and calculations as described in sections 2.1 and 2.6.

4.4 COMPATIBILITY

Our solution will strive to be as compatible as possible with relevant C-Roads (https://www.c-roads.eu) profiles for C-ITS messages like CAM and DENM. But our solution uses cellular communication while C-Roads so far primarily has focused on ITS-G5 communication.

4.5 RELIABILITY AND AVAILABILITY

No SLA, meaning there is no guaranteed uptime for the Central ITS Server in this project.

4.6 SCALABILITY

The Central ITS Server is based on an architecture with microservices running in Kubernetes. This is horizontally scalable to support a dozen test vehicles (Smartphones) to millions of ordinary vehicles (Smartphones).

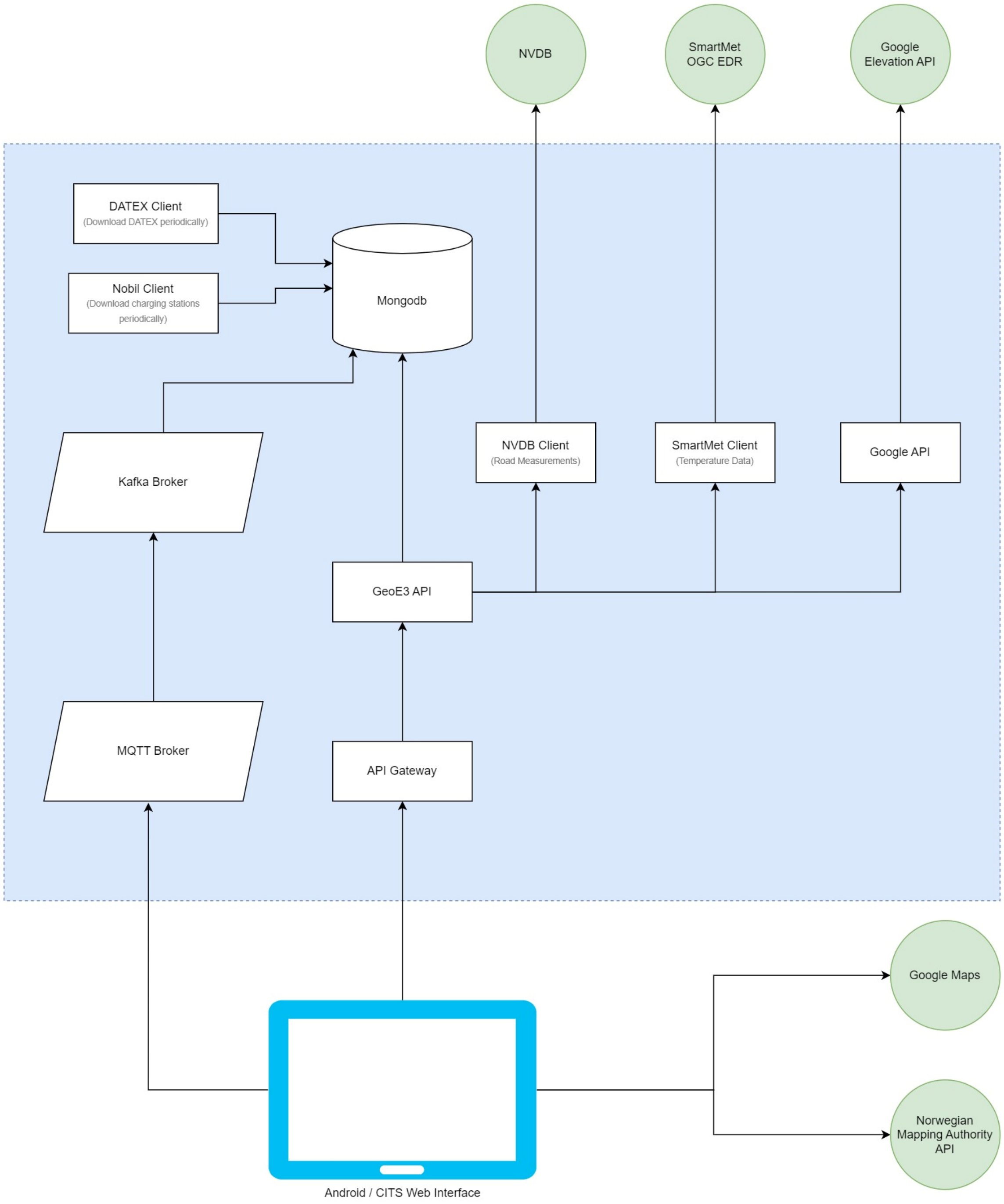
4.7 MAINTAINABILITY

We use CI/CD deployment of new Docker images from GitHub. When we have made changes to the C# code in Visual Studio, this will be updated in GitHub. GitHub actions pushes new Docker image to Azure Container Registry. Kubernetes launches the new image.

4.8 USABILITY

Both the Android app and the Central ITS Server have very intuitive user interfaces. In addition, we will make instructional step by step videos.

5 ARCHITECTURE



6 APPENDIX

Here you'll find some additional information about this project.

6.1 EXPLANATION OF EV RANGE CALCULATIONS

First, we obtain the route geometry from the route planner API. For each point along this geometry, we gather elevation data from Google APIs and temperature data from the OGC EDR APIs for each point in the route. This information is processed on the backend and then sent to the mobile application.

In the mobile application, we start by calculating the **current battery level** before the trip begins. This calculation uses the State of Charge (SOC) and the Minimum Discharge Limit. if the vehicle needs charging at the start of the trip, we locate the nearest charging station from the current location and add it as a suggested option. The **current battery level** is then updated using the Maximum Charging Limit. Following this, we continue with the energy consumption calculations.

The energy consumption calculation iterates through all the points obtained from the route planner's geometry and consists of three main adjustments:

Energy Consumption according to Travelled Distance: This step calculates the initial energy consumption based on the travelled distance, multiplying it by the rate of energy consumption specified in the settings.

Temperature Adjustment: If the temperature is below 20°C, we assume that energy consumption will increase due to the colder temperature. The increase is calculated by determining the difference between 20°C and the actual temperature, dividing that difference by 10, and then multiplying it by the temperature loss percentage specified in the settings to find the temperature loss factor. This factor is then used to increase the initial energy consumption.

Elevation Adjustment: If there is an elevation gain (positive elevation change) between two points, the initial energy consumption is increased based on the amount of energy lost per 1000 meters of elevation gain, as defined in the settings. If there is an elevation loss (negative elevation change) between two points, the initial energy consumption is decreased based on the amount of energy gained per 1000 meters of elevation descent, as defined in the settings.

After calculating the energy consumed, it is subtracted from the **current battery level**. We then identify the location where the **current battery level** would reach zero and search for the nearest charging stations from that location. If a charging station is found, we add it as a suggested option. If no charging station is found, we backtrack along the route to find a charging station. Once a charging station is located, we redo our calculations starting from the nearest point to the charging station with the maximum **current battery level**. This process is repeated until the route is completed.

6.2 VIDEOS

Here you'll find a short demo video from the final test:
https://youtu.be/q5gcU7PsN1Q?si=iV49uiCb1yzOZvhl

Here you'll find the complete video from the final test:
https://www.youtube.com/watch?v=7ry301SlrJ0