

Learning experience

Introduction

I have written a program that queries the Finnish Metereological Institute (FMI) and displays the temperature, wind speed, and pressure at all the stations closest to Oulu. These values are shown in a QML window, overlapping a map of Oulu. I also distinguish the stations that have the highest temperature, strongest wind speed, and lowest air pressure on the map with 3 markers and an additional UI element.

The data used is the most recent available for each station. The data is refreshed every 15 seconds.

Architecture

There is a clean separation between front and back-end. The back-end is exposed to QML as a singleton object.

The back-end follows the process flow. There are 4 functions:

- 1) one for sending a GET request to FMI,
- 2) one for extracting the raw data from the XML response,
- 3) one for converting the raw data into an intermediate structure, whilst keeping only the most recent measurements, and
- 4) one for finding the stations with highest temperature, wind speed, and lowest pressure.

This final function is the only one responsible for communicating with the front-end via signals. All the functions are slots and use signals to pass data along the pipeline. Since all the containers used are implicitly shared, they are passed by value.

Decisions

I have chosen to optimize the program for readability. This means that I could realistically have done everything in a single function, with no custom data structures. Maybe this would have improved performance slightly, but it would have reduced modularity and readability. Still, local performance decisions were taken when appropriate (e.g., not parsing the station ↔ coordinate mapping every time the data is refreshed, since that doesn't change).

I have also assumed that the upstream data schema is invariant (i.e., XML tags don't change names, measurements are always grouped by stations and ordered by ascending unix time), so I have implemented limited error checking as regards the data quality.

Experience

I encountered no major problems during this assignment, but the things which took me the most time were: 1) reading and understanding FMI's data format and API parameters, and 2) setting up an Android emulator and OpenSSL. Other than that, I just had to read the documentation.

Future direction

A possible improvement to the program could be to display a countdown until the next data refresh. I would also like to make the station markers on the map clickable and redirect them to their respective section in Oulu's FMI page. An additional one could be to have a field where the user would input the name of a city and the program would work there instead, so we're not limited to only Oulu.