Technologies:

- Language: C#
- Framework: WPF (Windows Presentation Foundation)
- Libraries:
    - -Windows.

Design Pattern: MVVM

Functionalities:

- Display electricity data on graphs or plots
    - Adjust the parameters of the visualisation (timeline, power source, etc.)
    - Combine electricity data in one view.
    - Calculate and visualise of percentages of different power forms.
- Display weather data on graphs or plots
    - Adjust the parameters of the visualisation (timeline, location, selected weather information)
    - Calculate and visualise average temperatures at certain location in certain month.
    - Calculate and visualise average maximum and average minimum temperatures at certain location in certain month.
- Display combined data of weather and electricity into one window.
- Initiate data collection.
- Save certain data sets and produce visualisation.
- Save preferences for producing visualisations (parameters)

Modules:

- UI
    - Electricity consumption (load)
        - 24hr forecast
    - Electricity production (generation)
        - 24hr forecast (hourly energy)
        - Wind power forecast
        - Nuclear power
        - Hydro power
    - Visualisations of these stats and percentages between them
        - https://www.c-sharpcorner.com/UploadFile/mahesh/charting-in-wpf/
- Core
    - All business logic for handling program functionality and data
        - Combining data
        -
- Data
    - getting data from HTTP servers
    - Interfaces for reading this data in a specific format

Source APIs:

- https://data.fingrid.fi/open-data-api/
- https://data.fingrid.fi/en/dataset
- https://data.fingrid.fi/en/dataset?groups=load-and-generation
- https://data.fingrid.fi/en/dataset?groups=state-of-power-system
- https://en.ilmatieteenlaitos.fi/open-data-manual

Tasks:

- Create a module to pull information from source APIs
    - Most likely using HTTPclient
- Create a module to display information in the UI
    - Need to determine the format of the information

Interfaces:

- Weather Data
    - Temperature
    - Observations
        - Wind
        - Cloudiness
    - Predictions
        - Wind
        - Temperature
    - Supplied by Data module
- Power Data
    - Consumption
    - Production
    - Source (type of power, nuclear, hydro, etc)
    - Supplied by Data module
- Grapher
    - Implemented by UI module. Allows for drawing graphs and other visualizations.

Environment variables:

- FINGRID_API: API key for fetching power data from Fingrid

The Core module can request information from the HTTP source API:s through the Data module This will happen most likely through a query event activated by the user through the UI, or a running timer in the

case of automatically updating real-time data. The Data module will process HTTP raw data (XMLs, JSON) into C# readable datatypes. Data combination and processing into various forms for display in the UI will be done in the Core module.