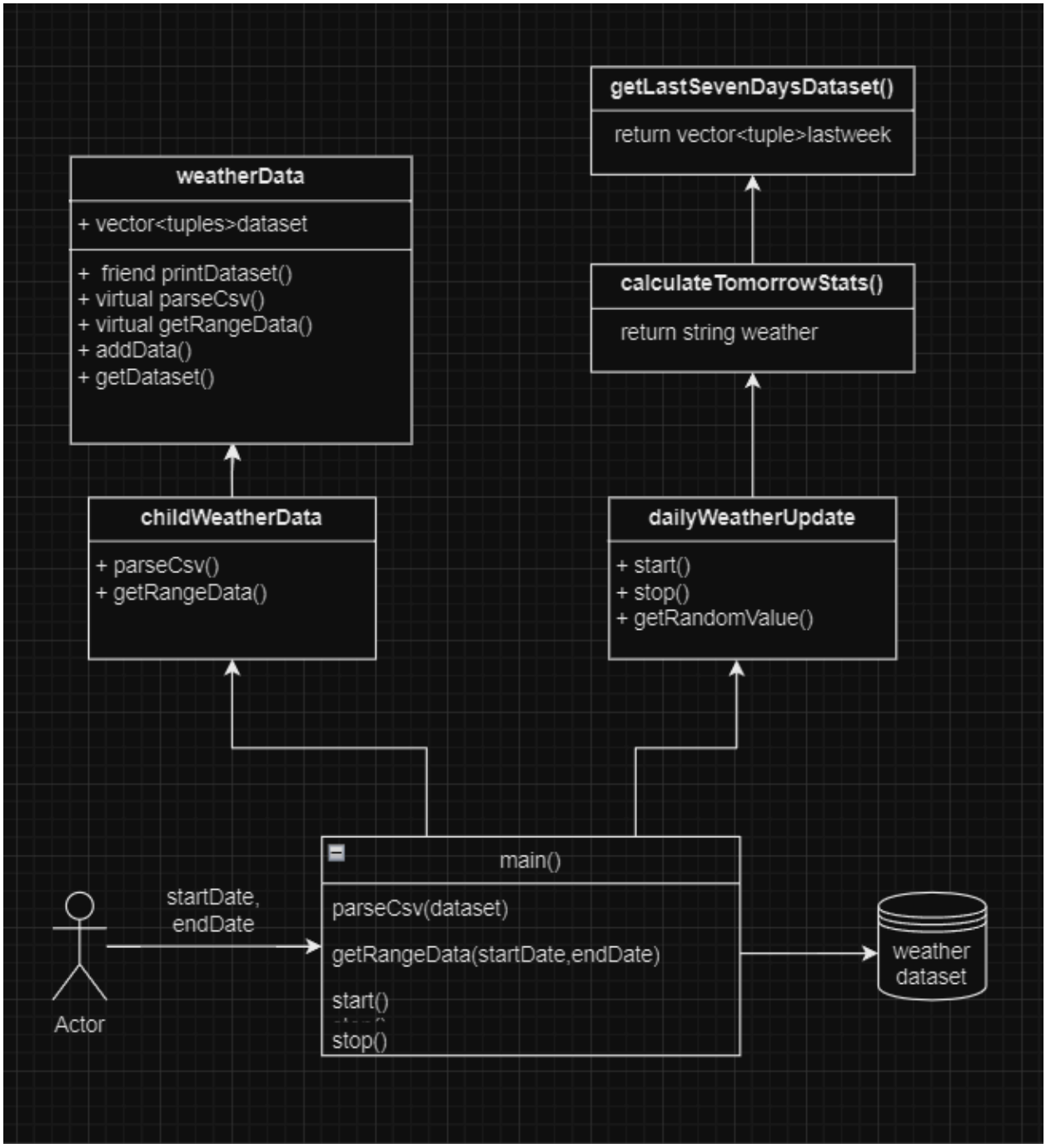
Weather Data Analyzer



This is a simple weather data analysis application that's built using a single main C++ file. The main.cpp file consists of two classes weatherData and childWeatherData.

The weatherData class acts as an abstract class that encapsulates the logic for constructing a container made using a vector of tuples that stores the dataset elements(date, precipitation, max_temp, min_temp, wind, weather). Inside the weatherData we also define a friend function(printDataset()) and two virtual functions(parseCsv(),getRangeData()) which can later be defined in a child class through the concept of polymorphism.
A setter and getter method called addData() and getDataset() respectively have also been implemented to add csv data into the container and retrieve from it.

The child class childWeatherData uses the concepts of inheritance and polymorphism to modify the parent class methods and implements the logic for parsing and performing statistical calculations on the data.
The parsing of dataset is done with the help of an csv.h library file(https://github.com/ben-strasser/fast-cpp-csv-parser/blob/master/csv.h) that has predefined function for easy traversal of csv files and extraction of relevant data.
The data is stored into the dataset container via addData(). The calculations are done by initializing variables with a base value and comparing and aggregating with each element of the dataset while traversal, for comparison to find max and min temperature and calculating averages.

For getRangeData() we take in arguments from users for start and end dates. And use a loop to traverse the dataset container within specified range and perform calculation for that duration in the same way as mentioned above.

A friend function printDataset() is used to print to console the entire dataset if explicitly called from main()(Commented out).

A daemon thread class dailyWeatherUpdate is used which runs in the background that can simulate daily weather parameter changes by calling the getRandomValue() within the same class. We call the getLastSevenDayDatasets() to get the last week's data. Two helper function tmToTimeT(), stringToTm() are used to convert date in string format to date format. Next calculate tomorrowStats() is called which defines the logic for tomorrow's weather condition by comparing the current stats and last week's average report. The newly generated data is added to the dataset container for enhancing the dataset.

From the main() we are using smart pointers to create a pointer of type weatherData which is pointing to childWeatherData so that polymorphism is enabled.
The start and end dates are received in main() and the creation of dailyWetherUpdate object is also from main(),which runs as background thread without affecting main thread and stops only when the enter key is pressed by user.