

Weather App Documentation

Tampere University – COMP.CS.140 Programming 3

Authors: Lam Dat Minh, Cat Huy Duong

Date: 26/04/2024

Table of Contents

1. Structure of the software (UML diagram)	2
2. Key Java classes' responsibilities	4
3. Software features	5
4. Division of labor	8
5. User manual	9
5.1. Starting the application for the first time:	9
5.2 Look up location and change temperature units:	9
5.3 Favorites and history:	11
5.4 Invalid input	12
6. The use of AI	13
7. Known bugs or missing features	13

1. Structure of the software (UML diagram)

The UML map that represents the structure of the software implemented by our team is illustrated as follows.

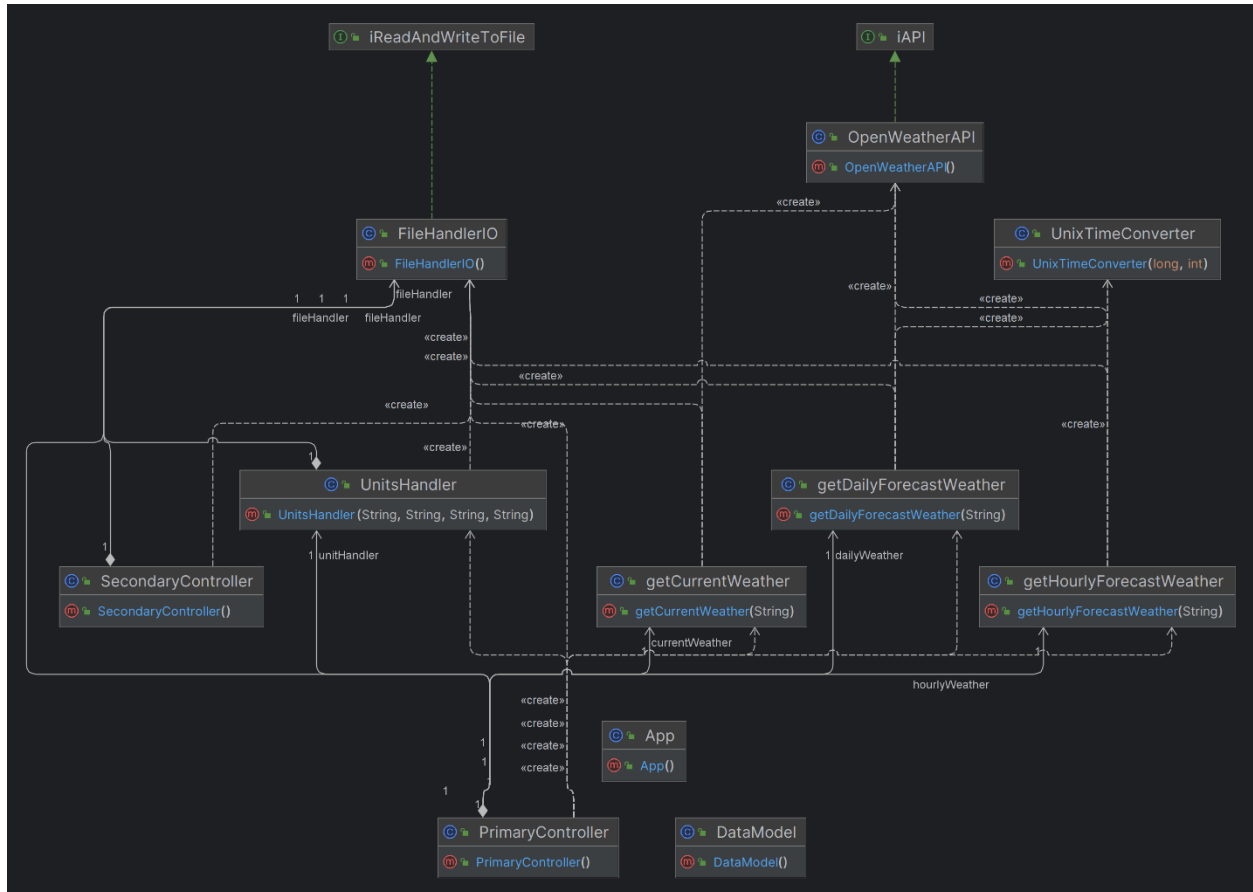


Figure 1: UML diagram of classes (without methods) in the software.

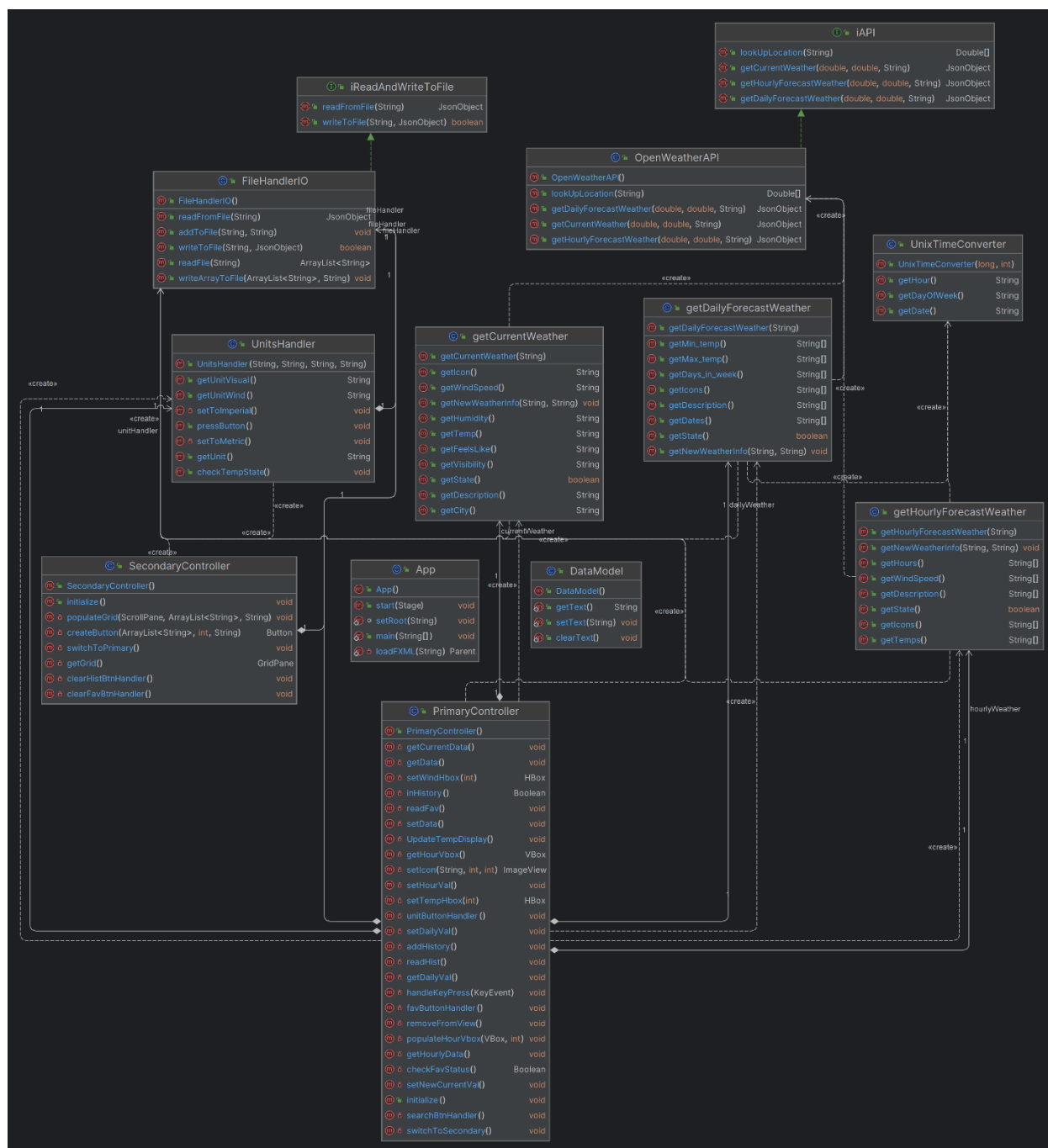


Figure 2: UML diagram of classes (with methods) in the software

2. Key Java classes' responsibilities

All key classes implemented in the software are presented in the table below.

Table 1: Class name and description

No.	Name	Description
1	App	Main class of the weather application.
2	DataModel	Class used for communicating between primary and secondary controller.
3	PrimaryController	Class handles the primary screen used for displaying relevant weather information. The class includes logic for handling the user interface.
4	SecondaryController	Class handle UI for favorites and history.
5	FileHandlerIO	A utility class for reading from and writing to files, particularly focusing on JSON and text file formats. It provides methods for reading JSON data from a file, writing JSON data to a file, appending city names to a text file, reading city names from a text file, and writing city names to a text file.
6	UnitsHandler	This class handles switching units of measurement and maintains the state of the unit system. It provides methods to switch between metric and imperial units, as well as methods to read and update the unit state from a file.
7	OpenWeatherAPI	A class that provides methods for accessing weather data using the OpenWeatherMap API.

		It allows users to look up location coordinates based on a location name, retrieve current weather data, daily forecast data, and hourly forecast data for a given location.
8	UnixTimeConverter	A utility class for converting Unix timestamps to human-readable date and time formats, considering a specified offset.
9	getCurrentWeather	A class responsible for fetching and managing current weather information from the OpenWeatherAPI, storing it in a JSON file, and providing methods to retrieve specific weather data.
10	getDailyForecastWeather	A class responsible for fetching and managing daily forecast weather information from the OpenWeatherAPI, storing it in a JSON file, and providing methods to retrieve specific weather data for the next seven days.
11	getHourlyForecastWeather	A class responsible for fetching and managing hourly forecast weather information from the OpenWeatherAPI, storing it in a JSON file, and providing methods to retrieve specific weather data for the next twelve hours.

3. Software features

Our software has two different scenes.

The first (main) scene is created for users to search for a specific location and save it as their favorite location. It shows weather data of the location searched including current weather, 7-day weather forecast and 12-hour weather forecast. Users can change the unit's system displayed (metric or imperial) with a button.

The second scene can be accessed from the main scene with a button. All the searched and saved as favorite locations are displayed here. Pressing on any displayed location in this scene will change the software back to the main scene to search for it automatically. Users can also go back to the main scene with a button in the second scene.

All default features are presented in the table below.

Table 2: Default features of the program.

No.	Feature name	Feature description
1	Search for a location	Allows the user to search for different locations.
2	Display detailed current and forecast weather information	<p>The program displays main weather, main temperature, feels-like temperature, visibility, humidity, and wind speed in the current weather section.</p> <p>Day in week, date, main weather, minimum and maximum temperature are displayed in the 7-day forecast section.</p> <p>Local time, main weather, description, temperature, and wind speed are displayed in the 12-hour forecast section.</p>
3	Notify location search error	If the location cannot be found, the program will display “city not found”.
4	Add and remove locations as favorite	The program allows saving and removing locations as favorites. Can remove each individual locations or remove all saved locations.

5	Custom weather icons	The program uses a custom set of icons (not those provided by OpenWeatherMap).
6	Unit tests for functionality	Unit tests have been implemented for the program.
7	Automatically save and reload state of the program	The state of the program is saved after searching for a location including all weather information and units of measurement displayed.

Extra features are shown in the table below.

Table 3: Extra features of the program.

No.	Feature name	Feature description
1	Support multiple systems of units of measurement.	Provides some data in various units, mainly SI and imperial. The program can show data using both systems, as the data retrieved from the service allows.
2	Location search history	Recently searched locations are kept in memory and displayed to the user to make it easier to go back to them later. Search history is also saved in a file and restored on program restart. Can remove all search locations.
3	Unit tests using continuous integration (CI)	Unit tests have been implemented using continuous integration in the remote repository.

4. Division of labor

Our team consists of two members. Detailed information about each member is shown below.

Table 4: Team members' information.

Full name	University email
Dat Minh Lam	datminh.lam@tuni.fi
Huy Duong Cat	duong.cat@tuni.fi

- Dat Minh Lam:

- Implement all classes related to fetching weather data from OpenWeatherAPI and file handling.
- Implement all unit tests related to fetching weather data from OpenWeatherAPI and file handling.
- Write the documentation for the software.

- Huy Duong Cat:

- Design the UI of the software.
- Implement all classes related to displaying UI and logic of main scene.
- Gather custom icon resources.
- Write the documentation for the software.

5. User manual

5.1. Starting the application for the first time:

When the application runs for the first time, the location will automatically set to a default location i.e. Hanoi and display all the related information of the location:



Figure 5: Application starts for the first time.

5.2 Look up location and change temperature units:

Information of a city can be lookup by entering the city name then presses enter or click the search button:



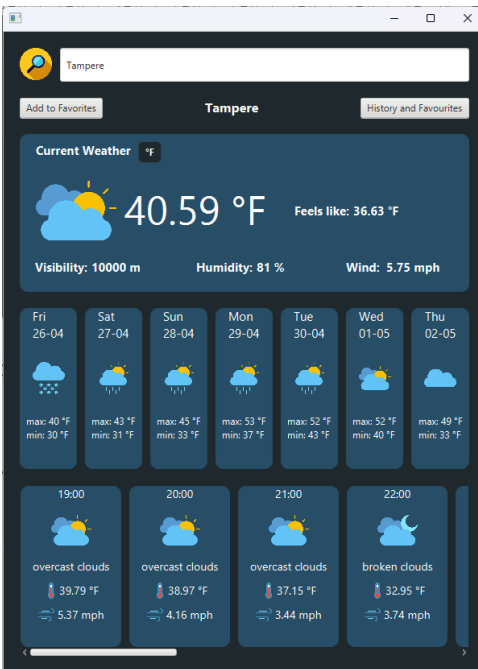
a) Entering a city name



b) Clicking the search button or press the enter key.

Figure 6: Looking up a location.

User can switch between the metric and imperial system by pressing the unit button (°C):



a) Changing the unit system



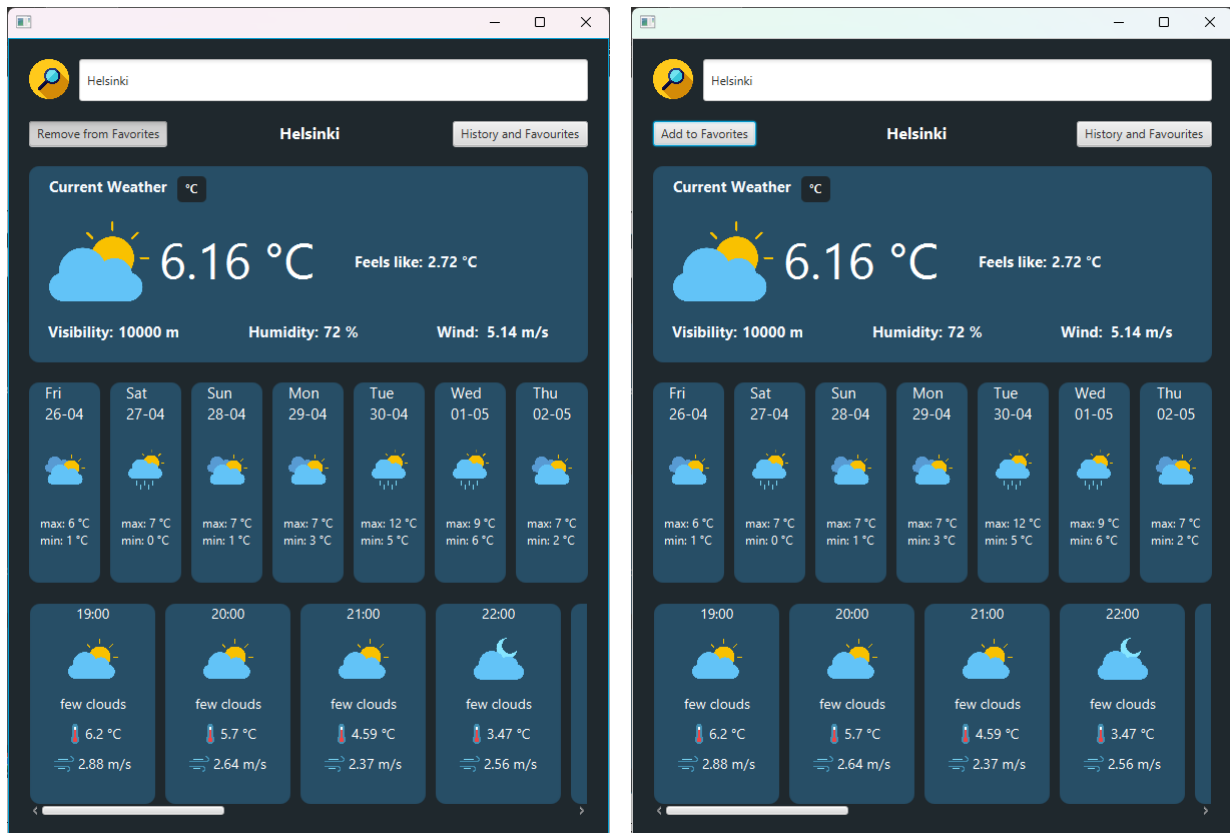
b) Looking up a new location

Figure 7: Changing the unit system and updating the location.

The unit of measurement state will be retained even if the user looks for a new location or closes the application.

5.3 Favorites and history:

User can add or remove cities to favorites by pressing the button (**Add to Favorites**) or (**Remove from Favorites**):



a) After add city to favorites

b) After remove from favorites

Figure 8: Application states

The application store information for users' favorite's location and history which can be lookup by pressing the button ([History and Favourites](#)):

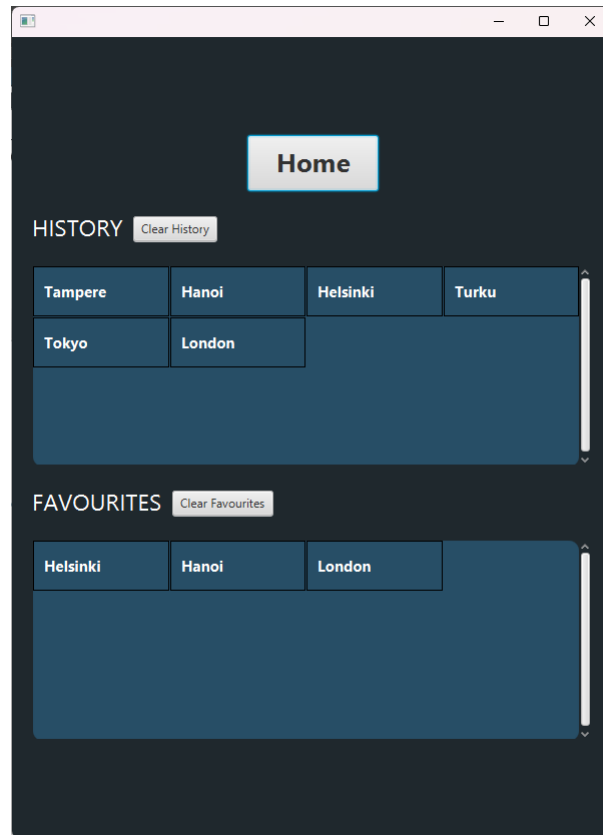


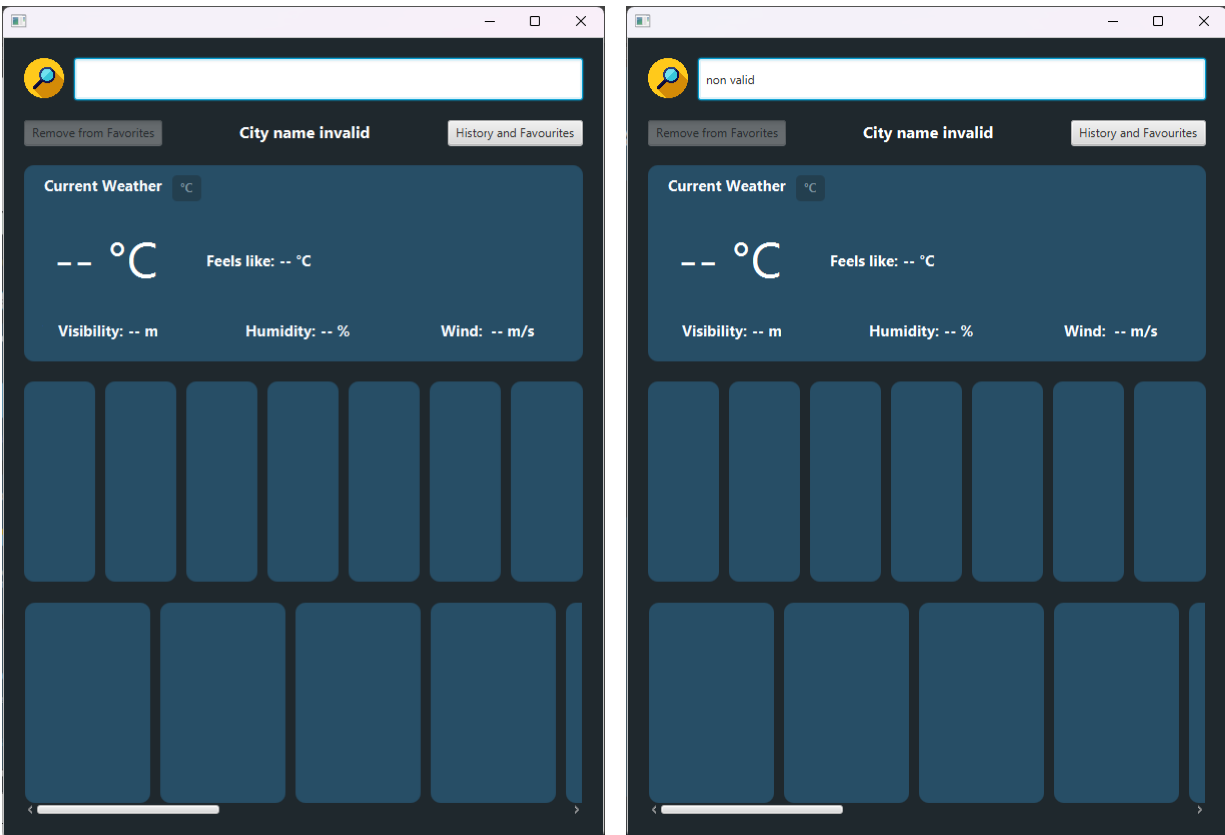
Figure 9: Application's favorites and history window

User can select a city by clicking a box containing that city and the program will switch back to the main screen and display all the data of said city.

The users also have the option to clear the history and favorites by clicking the corresponding buttons. The user can also click the "HOME" button to return to the main screen.

5.4 Invalid input

When the users enter an invalid input, the application will clear the main screen and let the user know:



a) When enter with the text field blank

b) When enter an invalid name

Figure 10: The “error screen”

The “error screen” also disables the temperature unit button and the add to favorites button. To resolve the error screen, the user needs to input a valid city name.

6. The use of AI

Our team used AI as a medium to look up information of the syntax required for our original software design. It is also used to support writing comments for Javadoc.

7. Known bugs or missing features

Features for the minimum and intermediate requirements have been fully implemented.

There have been no bugs detected in the current version.