Tampereen yliopisto

Thi Vu, Anh Tran

WEATHER APPLICATION DOCUMENTATION

COMP.CS.140 Programming 3: Interfaces and Techniques

14/12/2023

Table of Contents

1. Introduction

1.1 Purpose

Weather forecast can benefit people in different ways, especially in planning activities. The Weather Forecast Application is designed to provide users with accurate and up-to-date weather information for different locations. The program fetches weather data from OpenWeatherMap and visualizes it through a user-friendly graphical interface.

1.2 Scope

This documentation outlines the functionality, features, and usage of the Weather Forecast Application. It serves as a guide for both users and developers.

1.3 Audience

This documentation is intended for end-users who want to understand how to use the Weather Forecast Application. Additionally, developers may refer to this documentation for insights into the program's design and functionality.

2. User Interface

2.1 Main window view

When opening the app, the default view is shown in Figure 1. The name and description appear. There are two buttons Weather and Forecast for choosing functionality, and a Quit button to exit the app. After choosing functionality, user will be directing to another view.

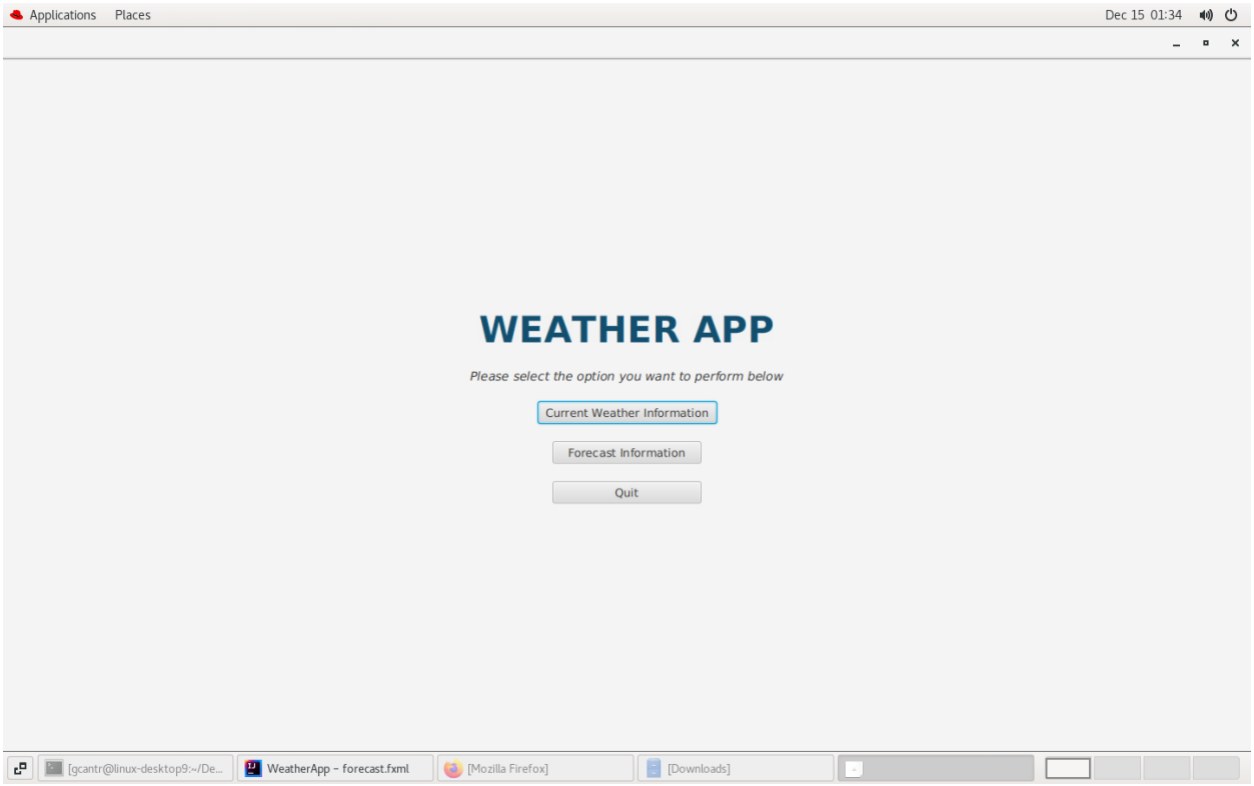

Figure 1. GUI layout for main window view.

Components:

- View: JavaFX UI with labels and buttons
- Controller: handling user input, updating UI

2.2 Weather view

Initially, users give input of location, then press Search to see result. Current weather view is displayed depending on given location, including current temperature, maximum temperature, minimum temperature, feel-like temperature, humidity, wind speed, rain intensity, weather icon. Users can go back to main view by pressing Return to Main Menu button.

API Links:

Components:

- View: JavaFX UI with location input field and current weather information displayed
- Controller: handling user input, fetching data from API, updating UI
- Model: representing weather data fetched from API

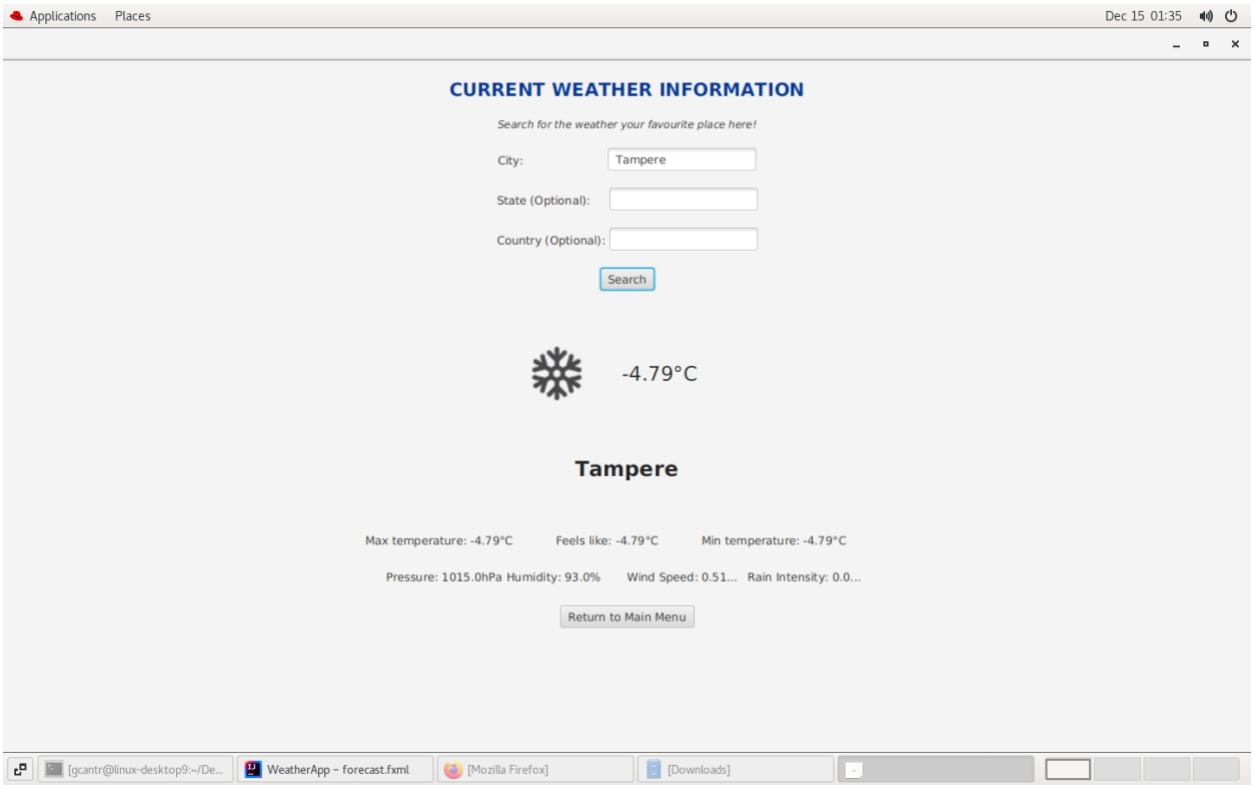

Figure 2. GUI layout for current weather information view.

2.3 Forecast view

Forecast view shows hourly forecasts and forecasts of 4 consecutive days depending on given location, which are shown in Figure 3 and 4. Day, highest and lowest temperature, weather image can be found in daily forecast. For hourly forecast, users can see temperature, feel-like temperature, wind speed, humidity, weather image corresponding to different hours in a day. User can go to the main view by pressing Back button.

Components:

- View: Java UI with location input and table for displaying hourly forecast and daily forecast
- Controller: handling user input, communicating with API, updating UI
- Model: representing forecast model fetched from API

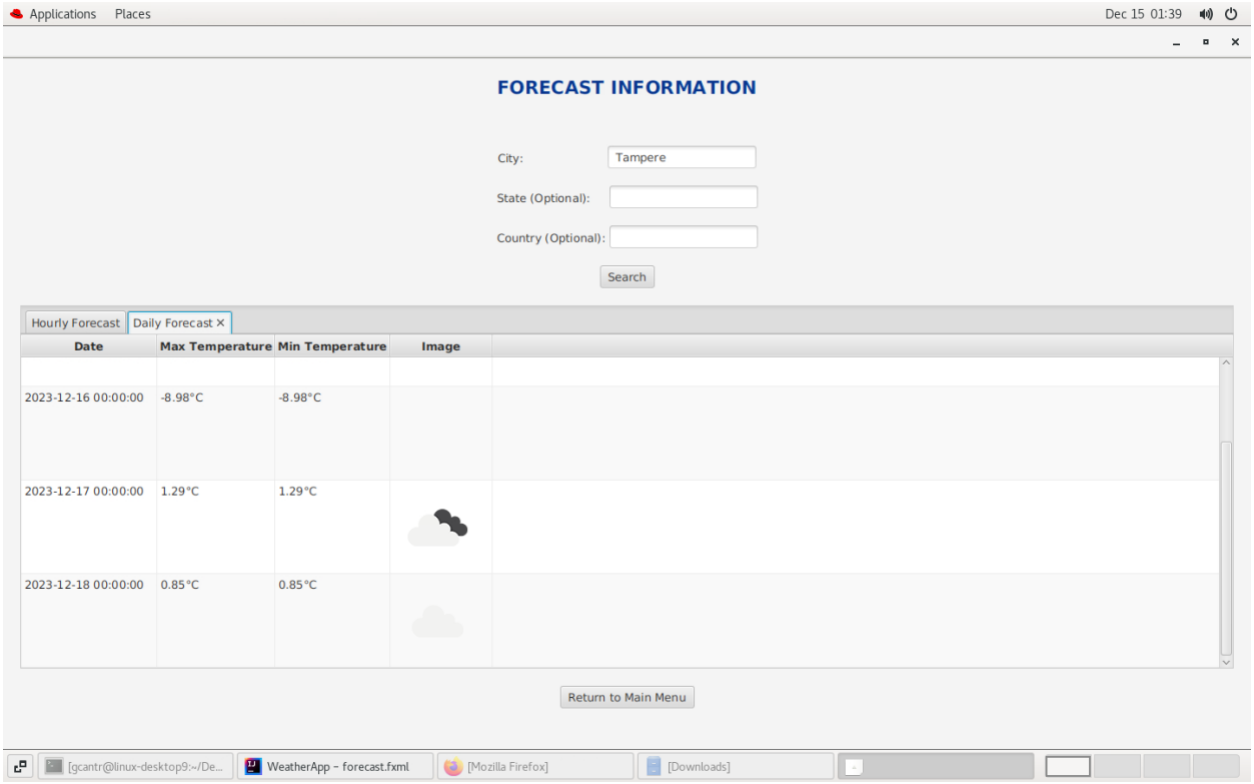

Figure 3. GUI layout for daily forecast view.

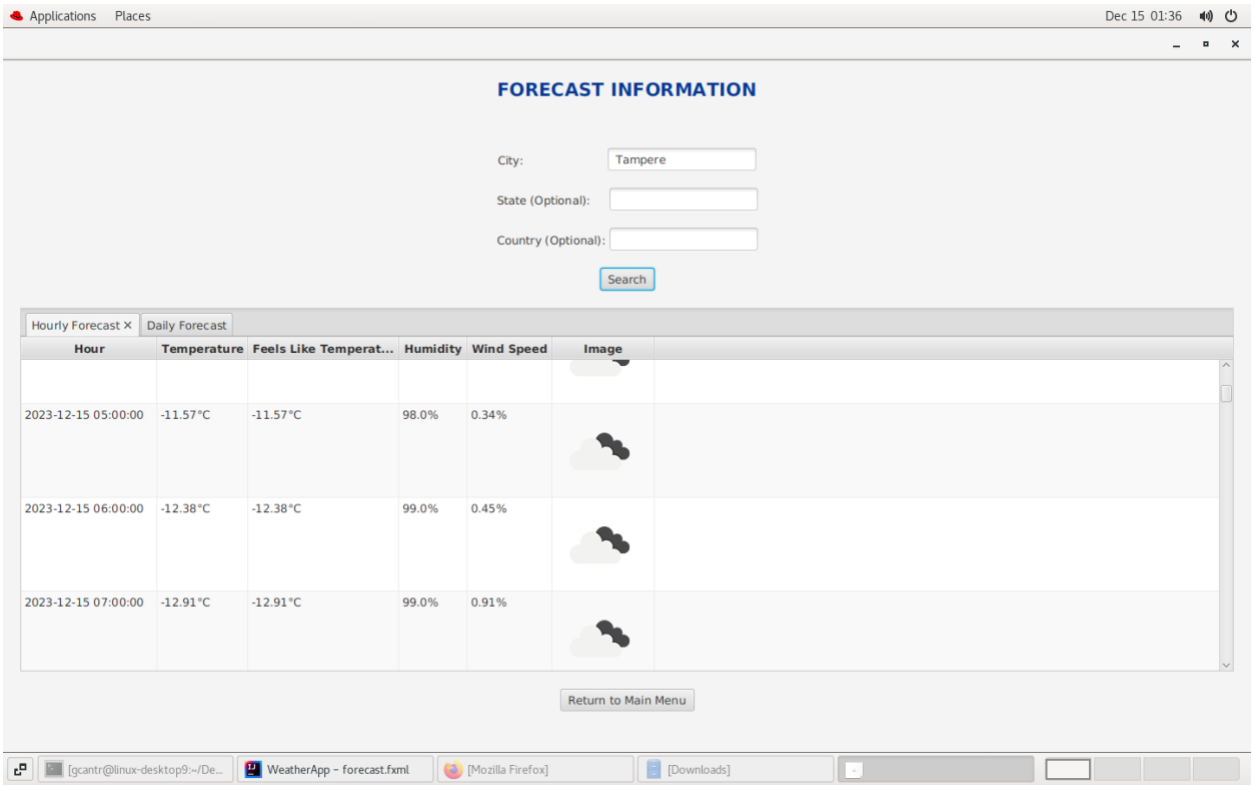

Figure 4. GUI layout for hourly forecast view.

3. UML Diagram

UML diagram for functions can be found in Figure 5.

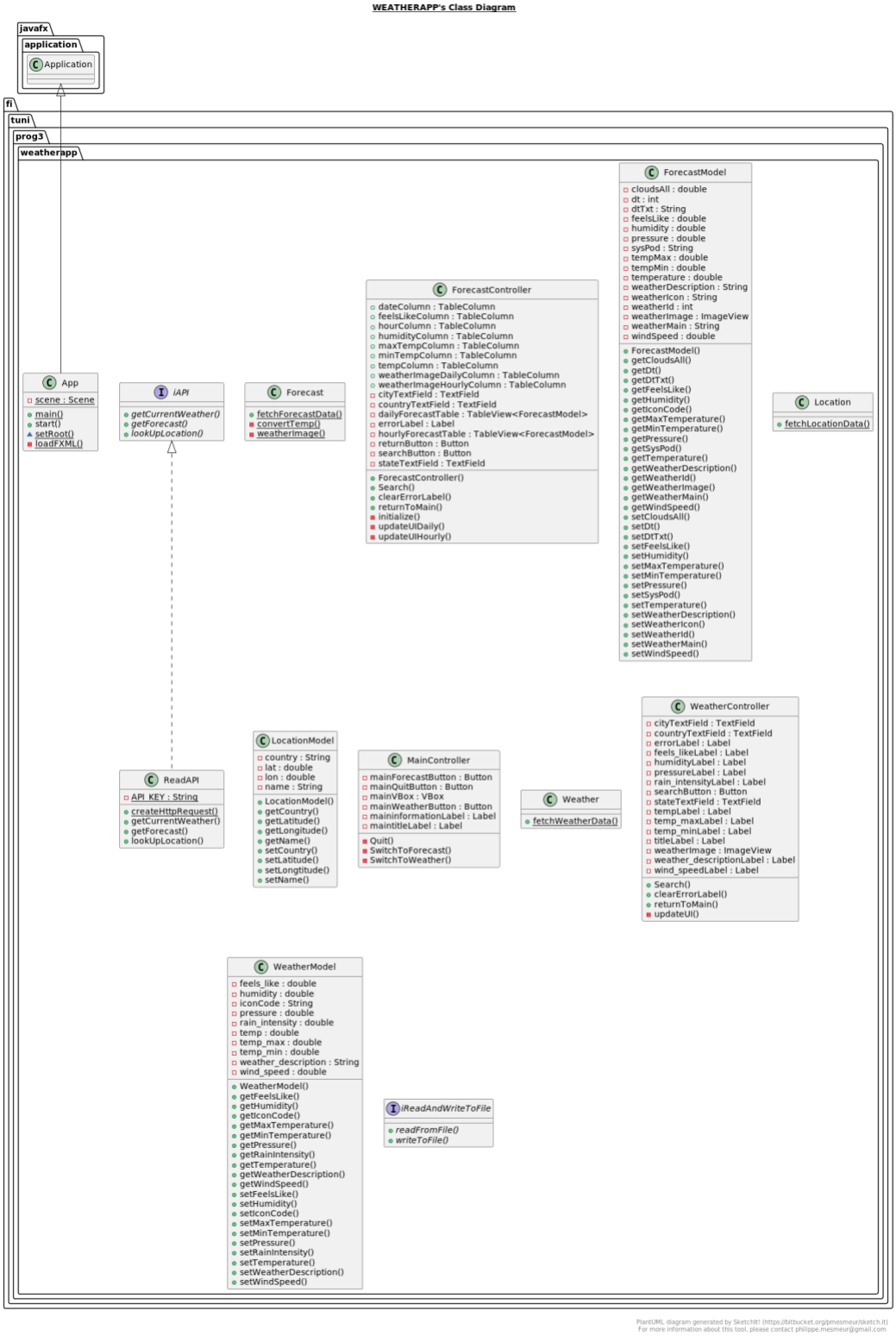

Figure 5. UML diagram for Weather Forecast App.

4. API Usage

Application Programming Interface (API) works as a connection between different software applications, allowing them to communicate and share data. In this program, the API is used to fetch the data from OpenWeatherMap, which is a popular online weather service that provides real-time weather information such as temperatures, wind speed, rain intensity,... and enables the program to access weather-related data, such as current conditions, forecasts, and historical information. The fetching data can be available in JSON and XML formats. In the context of our program, the JSON is utilized.

For the OpenWeatherMap, the data can be fetched using API and a HTTP request. Specifically, the API is used to obtain the current weather data, the daily and hourly forecast data, and transforming locations' name to their respective coordinates (latitude & longitude).

5. Acknowledgements

The Weather Forecast Application acknowledges the use of OpenWeatherMap for providing free weather data.

6. Open issue and further development

The fundamental functionalities of the app are done and they work. However, current user interface is rather basic, which can be improved to be more aesthetic, more appealing to users. Further development can focus on displaying statistical data such as graphs for weather and forecast in a long term.