[Group3207]

Project document.

Project Name:	Team Members:
WeatherApp	Ayman Asad Khan | An Cao | Areeba Nadeem

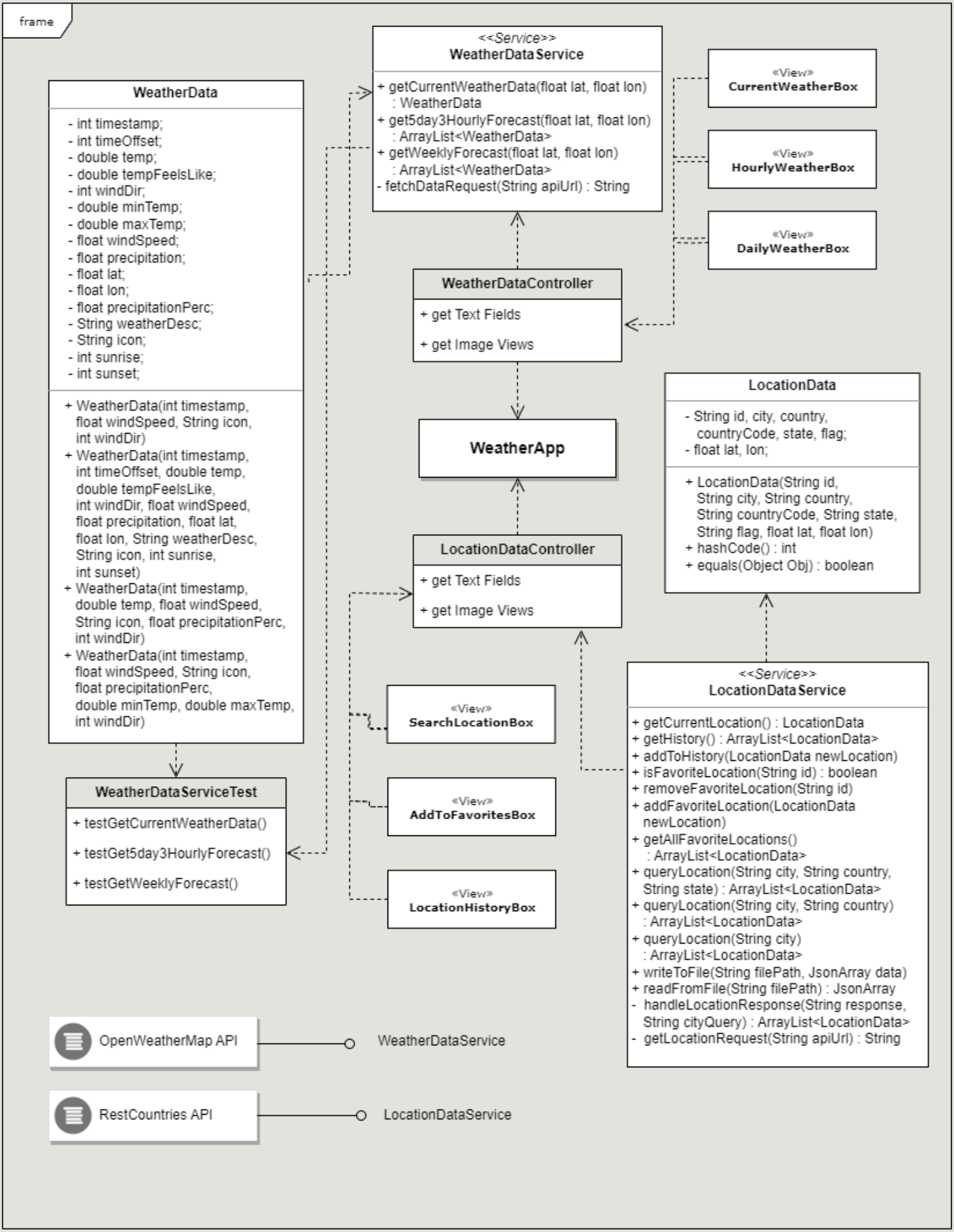
Project Description:

The Java based desktop Weather App project aims to create a functional program that retrieves weather data from OpenWeatherMap and presents it through a JavaFX graphical user interface (GUI). This application assists users in accessing weather forecasts for different locations, saving favorite locations, visualizing and displaying weather information based on time intervals. The Model-View-Controller (MVC) architecture organized the software development into three interconnected components: Model, View, and Controller.

In the Weather App, the Models are responsible for fetching data, handling data storage, retrieval, and manipulation to support the application's functionalities. Views represent the presentation layer of the application - the GUI. It displays information to the user and captures their interactions. And the Controllers act as an intermediary between the Model and the View, handling user inputs and responding accordingly by manipulating the Model's data or updating the View.

Objectives:	Action Plan:
• **Enable location searches and display current weather**	MainView presents current weather and search button
• **Implement at least one of the provided interfaces**	iReadAndWriteToFile implemented in LocationDataService
• **Develop a unique graphical user interface.**	Custom JavaFX User Interface rendered
• **Allow saving locations as favorites and restore program state on restart**	Search history and program state maintained
• **Present a detailed forecast (hourly and daily aggregate).**	Visualization of 3 Hourly and Daily weather forecasts
• **Use custom weather icons, more than what OpenWeatherMap provides.**	//
• **Handle errors during file processing.**	//
• **Implement unit tests for the program.**	JUnit Tests implemented for Models and Services
• **Support for Multiple Systems of Units.**	Conversion to Imperial and Metric Temp Unit Systems
• **Create an additional feature not listed in the requirements but requiring coding effort.**	//

UML Class Diagram



Key Classes & Responsibilities

Project Functionalities

Division Of Work