

Weather Cast: Application that provides Real Time Weather

Shruti Dalvi¹, Hitakrit Goplani², Swara Nabar³, Krish Mehta⁴,
Dr. Dashrath Mane⁵

^{1,2,3,4}Department of Computer Engineering, Vivekanand Education Society's Institute of Technology, Chembur, Mumbai.

⁵Assistant Professor, Department of Computer Engineering Vivekanand Education Society's Institute of Technology, Chembur, Mumbai.

Abstract- Weather forecasting has been an important application for predicting the weather changes and accordingly organizing human activities to prevent any loss. The Weathercast application notifies the user about the weather conditions like min-max temperature, humidity, windspeed, AQI (Air Quality Index), UVI (Ultraviolet Index) of any particular location around the globe. This application has been developed using android studio and the soft wares used are XML (for front-end programming), JAVA (for back-end programming). The app focuses on forecasting weather conditions using historical data. This is done by extracting information from OpenWeathermap API. The app has a salient feature of comparing the weather conditions of two locations making the app more user-friendly. The app also updates the user about the current weather news worldwide and thus the app gives a complete insight of the weather condition.

Keywords—temperature, humidity, AQI, UVI, XML, JAVA, OpenWeatherMap API

Date of Submission: 01-09-2022

Date of Acceptance: 12-09-2022

I. INTRODUCTION

Weather Forecasting is the application of science and technology used to predict the meteorological conditions of the atmosphere for a specific location and time. Weather forecasts are made by collecting quantitative data about the current state of the atmosphere at a given place and using meteorology to project how the atmosphere will change. It has enabled many of us to stay notified about the changes in climatic conditions beforehand. Rainfalls, fogs, and snows, as well as hurricanes, tornados, and cyclones, can cause serious damage to the businesses like farming and other fields dependent on weather conditions. Using the weathercast app, we can promptly monitor the weather and predict the most appropriate time for land irrigation and to prevent any loss of profit. It can be said that weather forecasts have been one of the greatest advancements of all time, mothered by innovative technologies and creative thoughts. The furtherance of the weather forecasting is the weathercast application. Our application displays

various weather parameters such as current temperature, wind speed, humidity, AQI, UVI. AQI (Air Quality Index) is a globally accepted standard to identify what is the quality of air and how polluted it is, and the general people can use

necessary measures. Similarly our app will also display the current UVI (Ultraviolet Index) value. UVI is a globally accepted standard to measure the strength of the ultra violet radiations coming from the sun. Our application is beneficial to the general public.

II. LITERATURE SURVEY

The prediction of weather conditions is done by combining historic weather information from nearby cities. This data is combined and is used to train simple machine learning models that can accurately forecast weather for next few days. These simple models can be run on low-cost, low-resource computing systems while still providing fast and reliable forecasts that can be used in our daily lives [1]. The main aim of the application is to provide accurate information to the user. In order to give better data through their APIs, the OpenWeatherMap uses its own technologies to analyse metrics for accuracy and quality. OpenWeatherMap uses their own numerical weather prediction (NWP) model to provide weather data through APIs. This model utilises data from sources such as: weather stations (METAR stations, User's stations, Companies Stations), weather radar data, satellite data, and global NWP models (NOAA

GFS 0.25 and 0.5 grid sizes, NOAA CFS, ECMWF ERA). The OpenWeatherMap gathers and stores data from the abovesources, then processes it using a proprietary set of in-house algorithms to increase the quality and accuracy of the weather conditions. For evaluation, it utilised 371 cities. In addition to numerous other notable cities, this list includes national capitals.

Metrics' behaviour for the OpenWeatherMap's NWP model were examined for

two months. The results revealed that the model's Root Mean Square Error (RMSE) is less than 2 degrees, Mean Absolute Error (MAE) is approximately 0.5 degrees, dependability is between 90% and 100%, and inaccuracy is approximately 1%. (less is better).It is evident that the OpenWeatherMap NWP model yields the best outcome.

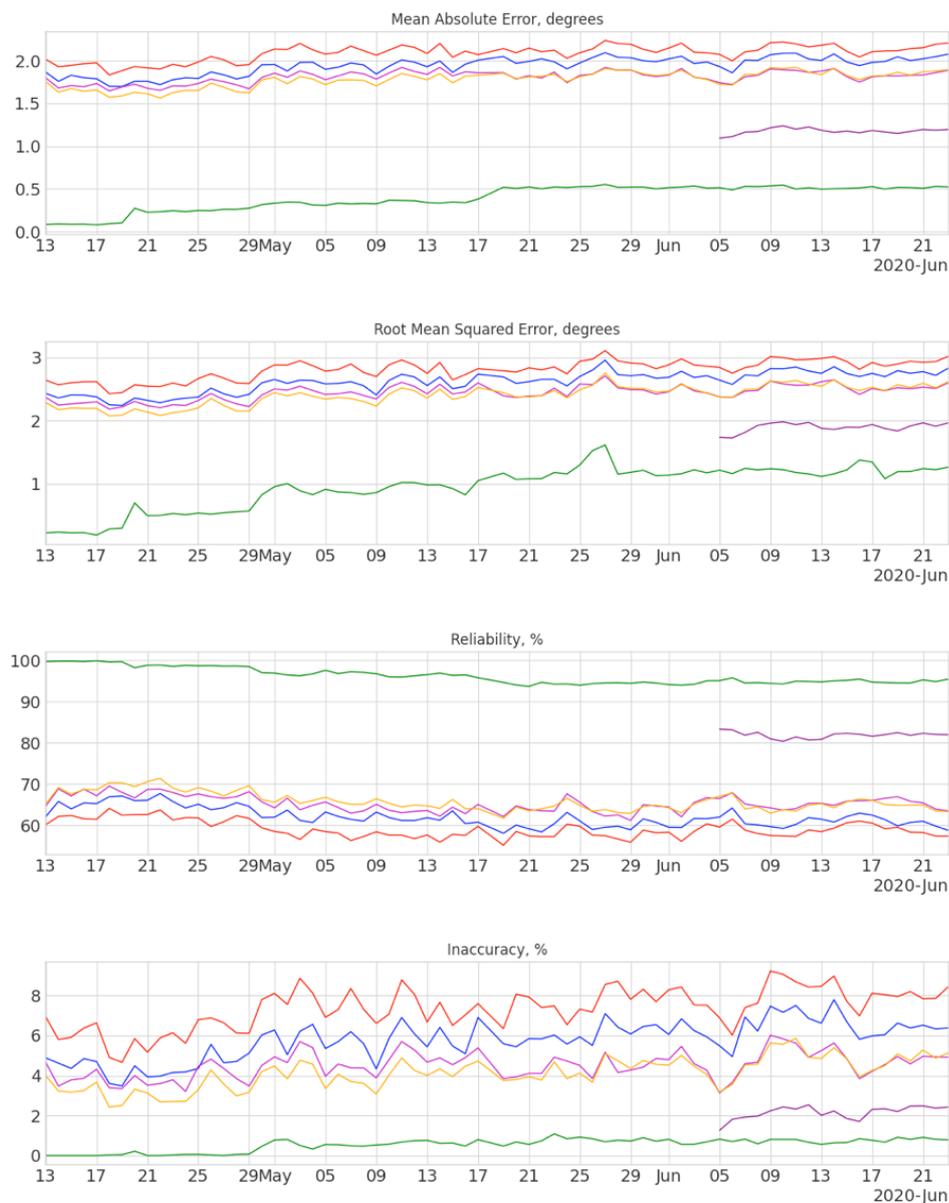


Figure 1: (—) and (—) NOAA GFS05 and GFS025 corrected by our algorithm. Quality metrics for Nowcast. (—) OpenWeather NWP model, (—) weather provider 1, (—) and (—) raw data from NOAA GFS05 and GFS025.[3]

III. ARCHITECTURE AND FRAMEWORK

This Weathercast application has been developed using android studio and the softwares used are XML (for front-end programming), JAVA (for back-end programming). The following conceptual model represents the basic workflow of our app. It is divided into 3 sub-parts i.e. the user, the frontend and the backend.

Initially, when the user loads the app he gets a dialog box requesting his/her location. After this he is redirected to the main screen i.e. the frontend of our app.

The frontend is coded using XML. The display screen shows various tabs like Temperature tab, Humidity and Windspeed tab, AQI and UVI tab and a sidebar with features like Forecast, News and Compare, an About tab and also a Settings tab for changing app language. On clicking on any one of these tabs the users location is taken as a parameter and a request is send to the backend. The parameter can also be any random city name which can be

typed by the user in the search-box given in every tab.

The backend is coded using Java. An Application Programming Interface or API is a set of code that enables data transmission between different machines (for e.g. - Between a web server and a web app). We have used OpenWeathermap API for getting values of different weather parameters like temperature, wind speed, humidity, AQI, UVI, etc. The API gives results in JSON format which are then parsed using the JSONObject Request package in java. The only parameter we need to give our API is 'city name'. To get city name we first request the user for location access. If the user grants this permission, using the GeoCoder package in java we get the position of the user in terms of latitude and longitude. If the user denies this permission or has location services turned off then his/her location will be set to 'Mumbai' which we have selected as a default location. Then we use this information while making the API call and hence get a personalized response.

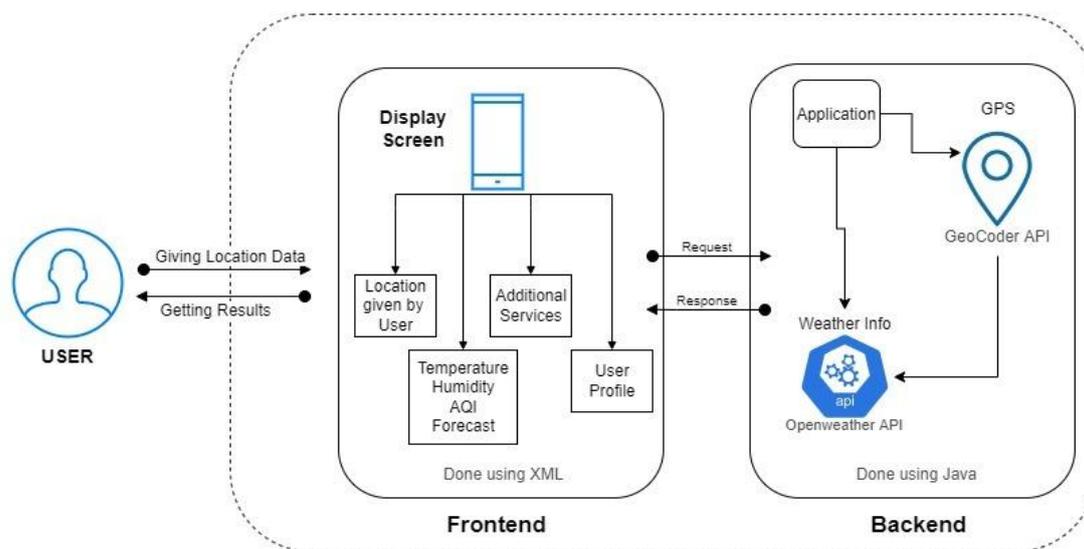


Figure2: Architecture of the Weathercast app

IV. METHODOLOGY

First the app asks the user his/her location. This can be done automatically by the app if the user grants permission to access their location using GPS. The home screen gives multiple options to the user like :

1. Getting weather information of their current location.
2. Giving a weekly forecast of the weather in the desired location of the user.
3. Giving the user various services like weather news and comparing weather of 2 locations.

On choosing any of these options the system will take the users location as a parameter and show the weather with the help of an API (OpenWeathermap API).

The API gives various information about the temperature, humidity, Air Quality Index, UV index, etc. and returns it to the application.

The app also gives weekly forecast to help the user know beforehand regarding the weather they might experience in the future.

In the end of the session the app stores some user information (for e.g.: The locations the user usually checks the weather of).

Our app can automatically detect the users location i.e. his/her longitude and latitude and find out the city name using these coordinates.

This is done by using the LocationListener, LocationManager and Geocoder packages.

Before using any of these classes we need to make sure location services i.e. GPS is turned on, on the users device. We do this using the `isLocationEnabled` method in the `LocationManagerCompat` class.

`LocationListener` class keeps updating the device location at an interval defined by the developer. We implement our `MainActivity` class with `LocationListener` and use its `onLocationChanged` method to get regular updates of our location.

The `LocationManager` uses system services to find out the device coordinates. It uses the `LocationListener` to keep updating the location.

Geocoding is the process of transforming a street address or other description of a location into a (latitude, longitude) coordinate. Reverse geocoding is the process of transforming a (latitude, longitude) coordinate into a (partial) address. The `Geocoder` class then uses these coordinates to locate the city corresponding to it.

Multilingual apps help the product reach larger audiences and can result in a greater user-friendly experience. Keeping this in mind we decided to add Hindi language support in our app. Adding multiple languages in XML can be done by using the string resource feature. A string resource is basically declaring a variable for hard-coded text. For eg- We set a variable 'SearchBar' for the text 'Enter city

name' which is shown in the search bar. All variables like these are stored in `strings.xml` file located in the `res/values/strings` folder.

In order to add a different language, we create another `strings.xml` file in which we create the same variables as in the first file but instead we now store the translation of the text in these variables. For eg - We again create a variable 'SearchBar' and now store the text 'शहरकानामदर्जकरे' in it. Now to change the language we use the `Locale` and `Configuration` classes. `Configuration` class describes all device configuration information that can impact the resources the application retrieves. A `Locale` object represents a specific geographical, political, or cultural region. The `setDefault` method in `Locale` class takes in a string as a parameter (e.g. - 'en' or 'hi' which correspond to the `strings.xml` file for English and Hindi respectively).

After calling the `setDefault` method we recreate our `MainActivity` which redirects to our Home page which is now in the language selected by the user.

One of our features that can be accessed from the sidebar is the 'Compare Weather' feature.

The user is asked to enter two cities. On clicking the compare button, a comparison between temperature, wind speed, humidity, aqi and uvi values.

This feature basically calls our API (`OpenWeathermap`) for both the cities entered by the user and then displays the results to the user.

This feature is something that differentiates our app from other weather apps available in the market. One of its usages can be for travelling purposes. The user can compare the weather of his current location with the target location and then make plans accordingly.

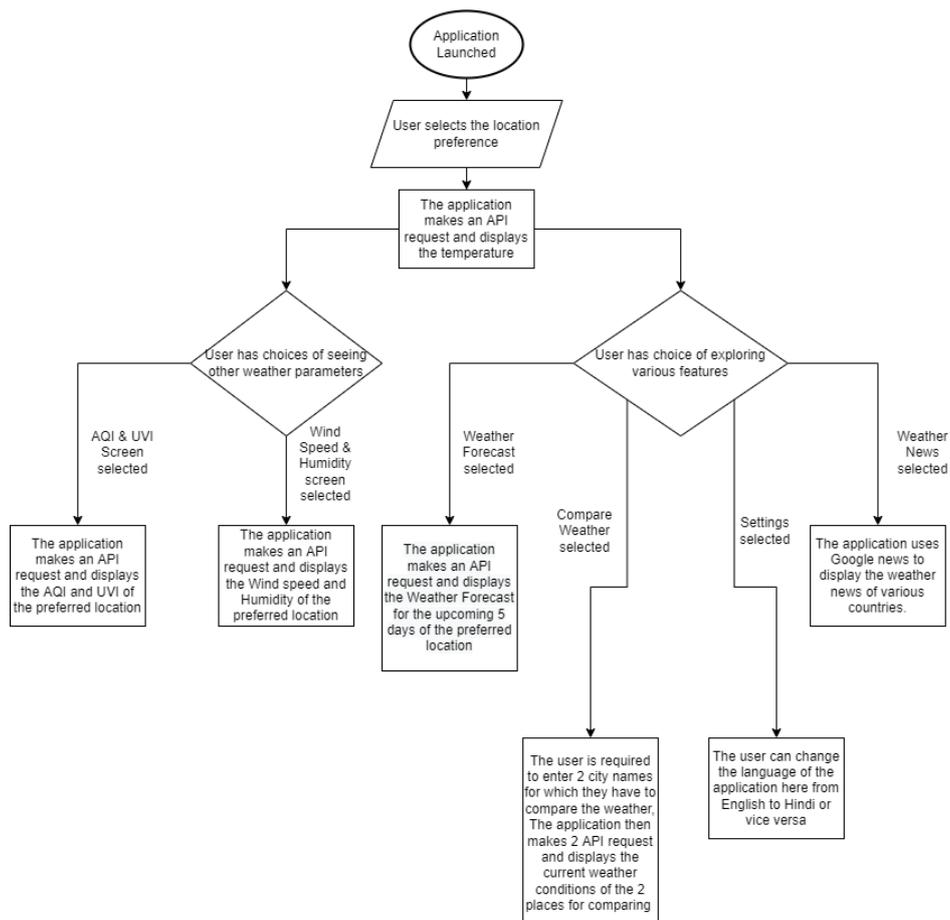


Figure 3: General flow of the application

V. RESULT

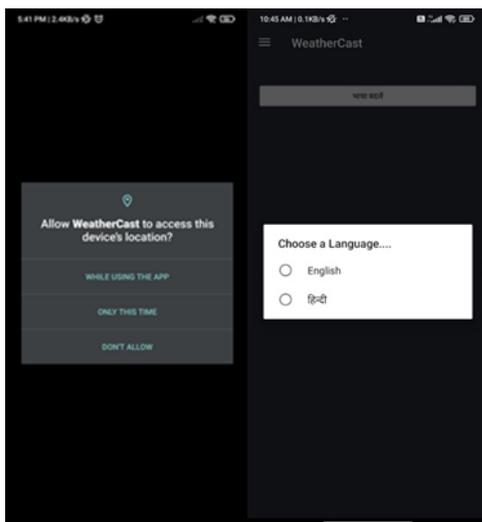


Figure 4: Initial screen on opening app for first time, asking users to grant location access.

Figure 5: In settings tab there is a language change option between Hindi and English.

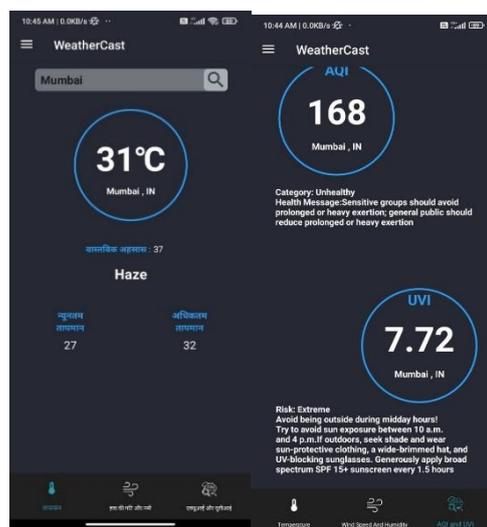


Figure 6: Main Screen showing temperature, after changing the application language to Hindi.

Figure 7: AQI and UVI screen, displaying the category, risk and health message for AQI and UVI.

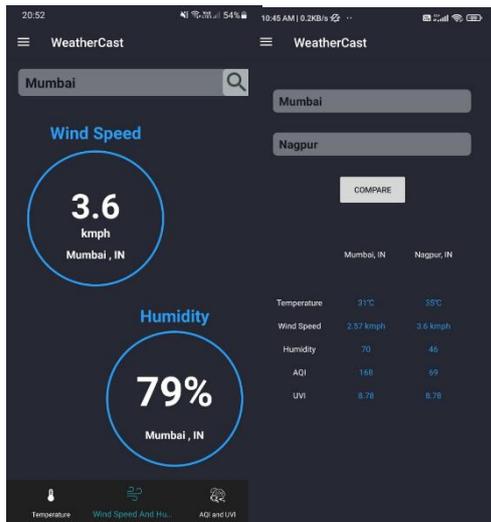


Figure 8: Wind speed and Humidity Screen, displaying Wind speed in kmph for comparison of chosen location.

Figure 9: In Compare Weather tab, there is Temperature, Wind speed, humidity, AQI, UVI between two cities.

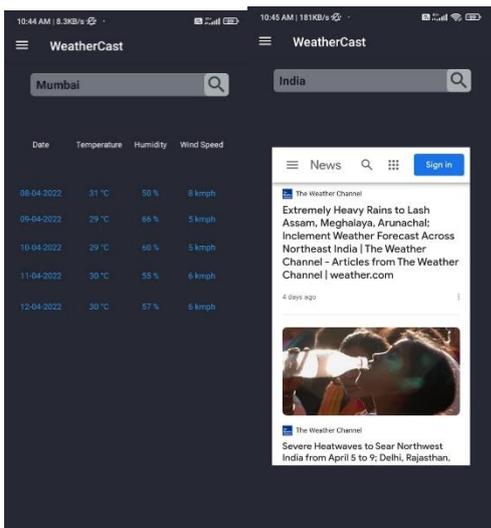


Figure 10: Weather Forecast tab, displaying values of Temperature, Humidity, Windspeed for next 5 days.

Figure 11: Weather News tab, showing weather news articles for India.

VI. CONCLUSION

In this paper we have described the working and features of the WeatherCast mobile application which is used to access forecast weather information. All the steps followed for the development of this application have been discussed. The values of all the weather parameters are displayed using OpenWeathermap API'S. It is observed that there are rapid changes in the weather conditions so dynamic monitoring of these weather parameters is essential. This application can be very helpful for the farmers to use weather forecast and

accordingly plan their activities. We can obtain more accurate results in the future which can be achieved by using concepts of Machine Learning and Artificial Intelligence.

REFERENCES

- [1]. Bochenek, B.; Ustrnul, Z. Machine Learning in Weather Prediction and Climate Analyses— Applications and Perspectives. *Atmosphere* 2022, 13, 180.
- [2]. Anant Sharma, Ankit Yadav, Bhanu Kumar, Dr. Sunil Gupta. *Weather Forecast Application-IJIRT-Volume 6 Issue 11, April 2020.*
- [3]. Scott C. James Yushan Zhang, Fearghal O'Donncha, "A machine learning framework to forecast wave conditions", *Coastal Engineering Volume 137, July 2018, Pages 1-10.*
- [4]. Steven Dewitte , *Artificial Intelligence Revolutionises Weather Forecast, Climate Monitoring and Decadal Prediction*", published on 13 August 2021.
- [5]. WengianZhang, YomgviChen, "Mathematical Models of Multifactorial Decision and Weather Forecast", *IFAC Proceedings Volume 16, Issue 13, July 1983, Pages 265-269.*
- [6]. United States Environmental Protection, "UV Index Scale: UV index Scale to help you avoid harmful exposure to UV radiation" Available: <https://www.epa.gov/sunsafety/uv-index-scale-0> [Accessed: Oct 08, 2021]."
- [7]. World Health Organization: UVIGuide WHO PDF, "Global solar UV Index practical guide." Available: <https://www.who.int/uv/publications/en/UVI-Guide.pdf> [Accessed: Oct 16,2021]."
- [8]. United States Environmental Protection Agency, "UVIGuide-EPA: Guide to the UVI Index." Available:- <https://www.epa.gov/sites/default/files/documents/uviguide.pdf> [Accessed: Oct 16,2021]."
- [9]. World Health Organisation, "Heat and Health: Impact on health due to climate change Available: <https://www.who.int/news-room/factsheets/detail/climate-change-heat-and-health> [Accessed: Oct 08, 2021]."
- [10]. <https://openweathermap.org/accuracy-and-quality>
- [11]. <https://github.com/public-apis/public-apis#weather>