

COLLEGE OF COMPUTING AND ENGINEERING

REAL TIME WEATHER APP SYSTEM

Capstone Project Final Report

Joshua Stephen

Supervised by:

Dr. Simco

Winter 2023

## 

[**Abstract 2**](#_glgqkrz6l64k)

[**1. Introduction 3**](#_whlrietr2aeu)

[1.1 Project Description 3](#_cw9buzxdyj4v)

[1.2 Limitations and Changes 4](#_yls9zc3w6onv)

[1.3 Definition of Terms 4](#_out0oflmtflg)

[**2. Literature Review 6**](#_io59gnvo5vqo)

[**3. Methodology 9**](#_z3rvqpwh3vqf)

[3.1 Architecture 9](#_zbq5gyre537o)

[3.2 Procedures and Resources 10](#_t1qjnkryy13k)

[3.3 Methods 12](#_81wuyv1cymka)

[3.4 Reliability and Validity 14](#_ldrwptywqata)

[**4. Results 15**](#_vio37yb6swbs)

[**5. Conclusions 16**](#_j9j6r9b71ho6)

[**6. References 17**](#_dgyb4ozfg70d)

## 

## Abstract

*Weather forecasting is a large industry with many applications that affect activities ranging from aviation, transportation, and sports to military, agriculture, and power generation. Weather forecasting is something people almost require in their day-to-day lives, and they especially on the go.*

*As such, the main objective of this capstone project is to develop a weather app system in Android which allows the user to accurately and conveniently have access to the current forecast via use of a weather API which takes the user’s location and provides the hourly forecast for the user. It may also aid the user in making plans around the weather, as well as utilizing the data for future systems, such as traffic routing, and flight scheduling.*

*With this project, I aim to display a proficient use of the different skills and approaches I learned throughout my courses. This will range from understanding programming languages, via use of Android Studio and Java, software engineering, in regards to my overall approach in planning and developing this system, proper use of multiple data structures, organization of programming languages, and object oriented design.*

## 1. Introduction

Weather forecasting is the scientific estimate of future weather conditions for a given location and time. These are made through recordings of weather variables which are produced over 24-hours a day across the planet. Technology has played a large role in forecasting weather, as this data is collected and projected through use of different applications and systems.

### 1.1 Project Description

The goal of this capstone project is to develop a weather app in Android that allows for the user to conveniently have access to accurate hourly weather forecasts. This is done to address the problem of users being too busy or occupied, and needing to check the weather in a quick and simple way.

The weather is an important aspect of our day to day lives, regardless of where we live. In order to make plans and work with our schedules, we’d need to know what to expect. So, in order for the app to offer the necessary forecast for its user, it is necessary to consider different aspects of the weather, such as the climate, chance of precipitation, humidity, and UV index.

This system will provide the user with easy access to these factors using a system developed by myself as my capstone project. The way the system will work is that the user will enter the name of their location and using an API, the necessary data will return the different types of weather related data. Furthermore, future iterations of this system will implement a planner system which allows the user to mark down and schedule future events in accordance to the weather, however that will not be shown in this prototype

### 1.2 Limitations and Changes

This project initially was originally to be done as a desktop application through Java, as an executable jar file. However, the portability and practicality of doing it for a mobile platform shifted development to Android Studio. Through Android Studio, The first versions of the application were done in Kotlin before shifting to Java.

Another change was the API that was intended to be used. Initially, OpenWeatherMap would be the API to be used for this application, but through development, due to complications, Open-Meteo was used instead. Likewise, in this current prototype, the humidity and UV index had not been implemented.

At this point in time, a current limitation is the weather not displaying everything properly inside the application resulting from a Client Error with the Volley Library.

### 1.3 Definition of Terms

**Application Programming Interface (API)** - A set of functions and protocols that allow for different software to communicate and share data or functionality with one another.

**Library** - A collection of non-volatile resources used for software development.

**Volley Library** - An HTTP library used for networking in Android apps.

**Network Security Configuration** - An XML file which allows you to specify the security settings of an Android app.

**JSON** - An open file and data format that uses readable text to store and share data objects contained as arrays and value parts.

## 2. Literature Review

Regarding the development and study of weather systems and applications, there have been numerous studies and ideas that have gone into weather forecasting which is important to understand when it comes to this project. A large focus on these previous works outline the importance of portable weather updates as we head more into the cellphone age. As this project is being done using Android studio, that will be focused on in addition to the observations acquired through working on these articles.

With smartphones and the increasing use of mobile weather apps, the need for weather forecasting is becoming more and more common, whether we notice it or not. Through prior research developed by Minh D. Phan, Burrell E. Montz, Scott Curtis, and Thomas M. Rickenbach in 2018 they record three important questions regarding mobile weather apps (MWA):

* Are smartphones the most popular source for weather forecast information among respondents?
* What specific reasons do respondents have for choosing their favorite MWA?
* How do geographic and demographic factors influence MWA use?

To help answer these inquiries, the researchers gathered information from a 2011 Pew Research study regarding the use of cellular phones in the modern day.

“*A 2011 Pew Research Center study found that 95% of the “millennial” generation (ages 18–34) and 85% of all American adults own cellular phones. Today’s college students, who align mostly with the millennial generation, have the highest rate of cell phone use compared to any other generation, with research in 2012 indicating that 62% of undergraduate college students own a smartphone, up from 55% in the previous year*” (Dahlstrom et al. 2012).

This alarming rise in smartphone use naturally translates to the increased use in mobile applications. The most popular applications downloaded from these smartphones tend to be games and entertainment, however, weather apps were a close second, followed by social media platforms.

An overwhelming majority of people in a survey of a sample size of college students admitted to using mobile weather apps for their primary source of weather information. And only 10% use other means such as newspapers or the radio.

When surveying the reasons as to why users preferred their favorite weather applications, it came down to ease of use, understandability, or being preloaded on their device. Which outlines the importance for convenience and usability for these systems.

Alongside the convenience of the systems themselves, another factor that is important to the user is accuracy. In another article regarding weather forecasting, the speaker outlines how accurate forecasting has been one of the biggest challenges around the world for a long time.

*“Unlike traditional methods, modern Weather forecasting involves a combination of computer models, observation (by use of balloons and satellites), and knowledge of trends and patterns”* (Saxena et al. 2013)

When it comes to how weather was recorded, people underwent one of three methods: Synoptic weather forecasting, Numerical methods, or Statistical methods.

Synoptic weather forecasting is the traditional approach that rose until the 1950s, which involved viewing the weather at a common point at a certain time. In order to have an average of the change in pattern to make predictions a series of synoptic charts were made every day.

The second approach, Numerical weather predictions, used computers to make a forecast, and ran on supercomputers that provide predictions on atmosphere variables like precipitation and wind speed. However, this approach wasn’t always accurate or precise.

To improve upon this approach, it would be supplemented with statistical methods. Statistical methods used past records and essentially formed patterns in order to predict future weather events. Many of the weather systems today use this approach

## 3. Methodology

The way I approached this project was through an Agile methodology. In the sense that, I broke it up into different dynamic sprints that I worked on simultaneously with one another. These sprints were the planning phase, the design phase, the development phase, the testing phase, and the review phase. With Agile, factors like testing and fixing errors can occur right away, rather than waiting for the testing phase.

.

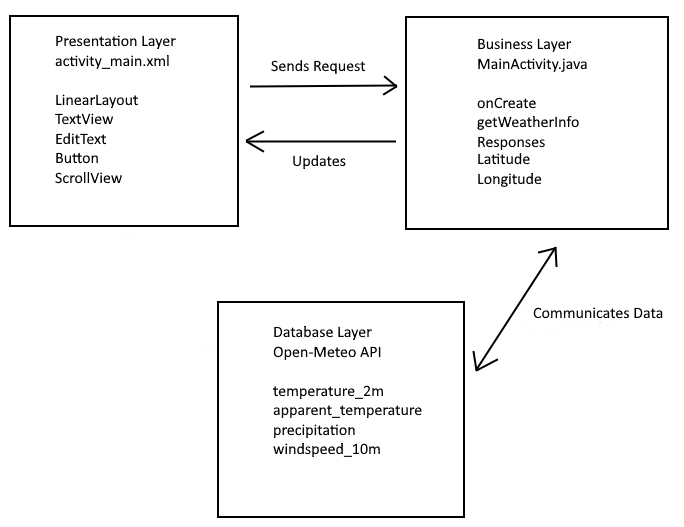
This prototype has one central use case, which is to return the weather details to the user in an efficient and timely manner. The approach I took for this project was developing a system which allows the user to enter their location details, and having the selected API return the requested data.

### 3.1 Architecture

To implement the system, a proper architecture is required. With all the requirements, three layers are needed to keep an overall structure for the system. The **Presentation layer**, which is the information that will be presented to the user. The inputs, buttons, and the weather data displayed. In this system, that would be the *activity\_main.xml*.

The **Business layer**. Which is the layer which holds the code that responds to the inputs requested by the presentation layer. In this system, the business layer is *MainActivity.java*. Validations are performed in this layer, and it also retrieves data from the following layer.

The **Database layer** in this system is the API, its main functionality is storing and sharing information from the source. In this system, the Database is represented through *OpenMeteoAPI*. Once the data is retrieved from the database, it is parsed through the business layer, then displayed through the presentation layer.

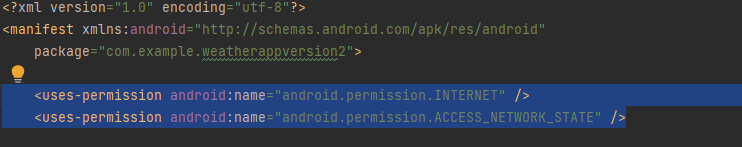


### 3.2 Procedures and Resources

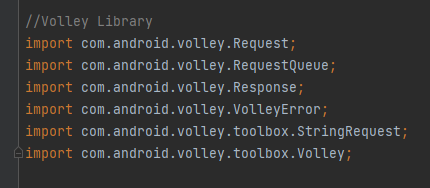
The first step into building this weather app was to find an appropriate Weather API that the application can retrieve data from. The API I used for my system is OpenMeteoAPI, which contains current weather, forecasts, short-term historical data, and weather maps. When the user requests the weather data from their specific location, it should return to them the requested data.

Additionally, I am making use of the Volley library, an HTTP Library for Android. This library was developed by Google as there was an absence of a networking class in Android SDK. Its features consist of being able to request queuing and prioritization, effective request cache and memory management, extensibility and customization of the library to our needs and canceling requests. This is primarily used for making a canned request in order to retrieve a response body at a given URL as a String.

I implement the Volley library by first giving my application network permissions through the AndroidManifest.xml file.



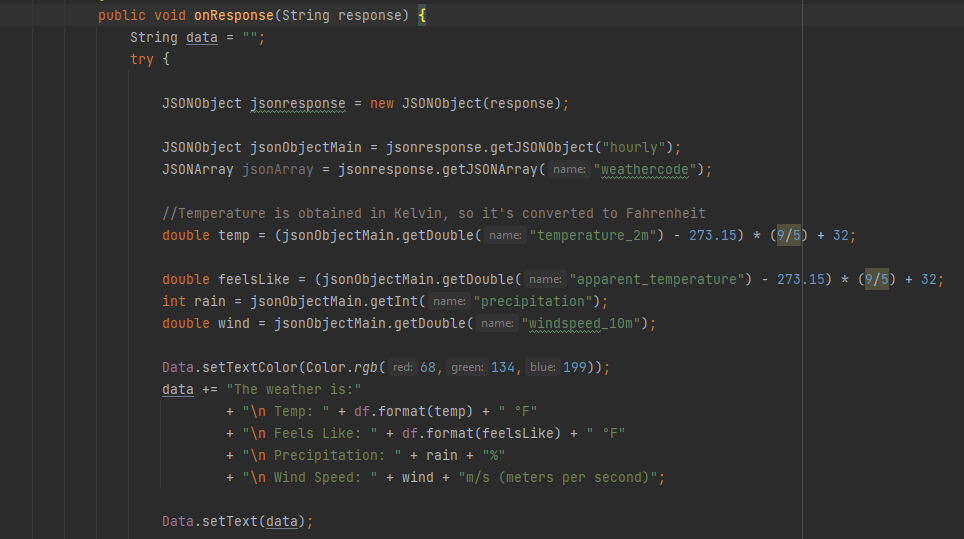
Next, I had to set up the network\_security\_config.xml file inside of a new xml folder in my resource directory. This is used in order to allow requests to be made. After that, I set up imported the library of Version 1.2.1 of Volley



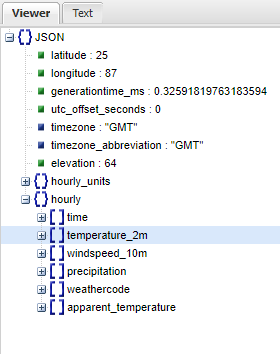
### 3.3 Methods

For my system, the current version of it requires primarily the getWeatherInfo method. This method is the one that retrieves the data from the API and displays it to the user, granted there aren’t any errors. This method requests the user input two floats, the latitude and the longitude, which is then used to search for the location.

Once the latitude and longitude are retrieved through the user’s input, the templink string variable is generated using the inputs from the user. The weather data is then retrieved from that link, and can be viewable more concisely through a JSON viewer. The data is then parsed through the onResponse method, and extracts the necessary data through that. As shown in the code snippet below.



Within the try-catch bracket, a JSONObject is created which retrieves the values that are stored within the variables that share the same name. In this case, that would ne “hourly”, “weathercode”, “temperature\_2m”, “apparent\_temperature”, “precipitation”, and “windspeed\_10m.” These variables are visualized within the JSON Viewer below.

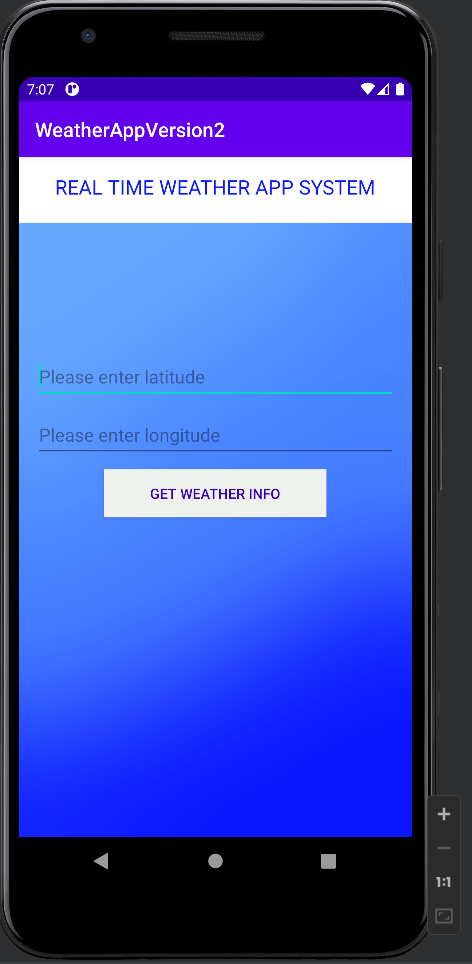


### 3.4 Reliability and Validity

Due to the errors derived from Volley this methodology, as it is, is not entirely reliable, however, should the weather data display properly, there isn’t anything that should stop it from functioning properly in any environment over any durations of time. However, through debugging and testing, I believe it is valid, as the logic does meet the requirements it needs to.

## 4. Results

Through testing and debugging, my application runs with no noticeable errors but it doesn’t return the required information as it should. It gets the link and requests information from the API, which then generates the weather forecast that can then be parsed via use of the JSON functions, however, it does not display it properly within the application.



The application also doesn’t have the user request the weather data in a convenient and intuitive way, as limitations throughout development resulted in the inability to simply enter the location name, and have the API generate the weather data through that alone.

## 5. Conclusions

Planning, developing, and writing this system was a rewarding experience as it allowed me to fully exercise what I had learned through my computer science courses on a realistic level. It also allowed me to gain new knowledge and experience that I’d only receive through the workfield, and got me better familiar with technologies and systems that I otherwise didn’t have a chance to touch on in depth in prior classes.

As simple as the functionality may be, this project was challenging in multiple ways, as this was the first time I’ve developed an app that made use of tools such as a weather API, a JSON viewer, and the Volley Library. Not to mention my prior knowledge in Android Studio was rather limited.

As for future research, as mentioned before, I want to further develop this application to allow it to be more convenient towards the user. Likewise, as most of the work in this system was dedicated towards the backend, I’d want to improve my skills in frontend development and further improve upon the system’s design and UI.

Furthermore, I want to apply my research towards other areas, such as applications that deploys the current state of traffic, or an application that records the gas prices around a specified area.

## 6. References

Abhishek Saxena, Neeta Verma and Dr K. C. Tripathi, 2013: A Review Study of Weather Forecasting Using Artificial Neural Network Approach - IJERT. (n.d.). Retrieved April 25, 2023, from https://www.ijert.org/research/a-review-study-of-weather-forecasting-using-artificial-neural-network-approach-IJERTV2IS110623.pdf

Dahlstrom, E., C. Dziuban, and J. Walker, 2012: ECAR study of undergraduate students and information technology, 2012. EDUCAUSE Center for Applied Research Rep., 38 pp., https://library.educause.edu/∼/media/files/library/2012/9/ers1208.pdf?la=en.

Phan, M. D., Montz, B. E., Curtis, S., &amp; Rickenbach, T. M. (2018, November 1). Weather on the go: An assessment of smartphone mobile weather application use among college students. AMETSOC. Retrieved April 24, 2023, from https://journals.ametsoc.org/view/journals/bams/99/11/bams-d-18-0020.1.xml

****

**Certification of Authorship**

Submitted to (Professor’s Name): Dr. Simco\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student’s Name: Joshua Stephen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date of Submission: April 24th, 2023\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Purpose and Title of Submission: Capstone Project Final Report\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Certification of Authorship: I hereby certify that I am the author of this document and that any assistance I received in its preparation is fully acknowledged and disclosed in the document. I have also cited all sources from which I obtained data, ideas, or words that are copied directly or paraphrased in the document. Sources are properly credited according to accepted standards for professional publications. I also certify that this paper was prepared by me for this purpose.

Student's Signature: Joshua Stephen\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_