Coding Challenge: Weather

Below are the details needed to construct a weather based app where users can look up weather for a city.

**Public API**  
**Create a free account** at [openweathermap.org](http://openweathermap.org/). Just takes a few minutes. Full documentation for the service below is on their site, be sure to take a few minutes to understand it.

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

<https://api.openweathermap.org/data/2.5/weather?lat=44.34&lon=10.99&appid=>{API key}

**Built-in geocoding**

**Please use**[**Geocoder API**](https://openweathermap.org/api/geocoding-api)if you need automatic convert city names and zip-codes to geo coordinates and the other way around.

**Please note that**[**API requests by city name**](https://openweathermap.org/current#name)**,**[**zip-codes**](https://openweathermap.org/current#zip)**and**[**city id**](https://openweathermap.org/current#cityid)**have been deprecated. Although they are still available for use, bug fixing and updates are no longer available for this functionality.**

**Built-in API request by city name**

You can call by city name or city name, state code and country code. Please note that searching by states available only for the USA locations.

API call

<https://api.openweathermap.org/data/2.5/weather?q=>{city name}&appid=[{API key}](https://home.openweathermap.org/api_keys)

<https://api.openweathermap.org/data/2.5/weather?q=>{city name},{country code}&appid=[{API key}](https://home.openweathermap.org/api_keys)

<https://api.openweathermap.org/data/2.5/weather?q=>{city name},{state code},{country code}&appid=[{API key}](https://home.openweathermap.org/api_keys)

You will also need **icons from here**:

<http://openweathermap.org/weather-conditions>

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**Requirements**

These requirements are rather high-level and vague. If there are details I have omitted, it is because I will be happy with any of a wide variety of solutions. Don't worry about finding "the" solution.

1. Create a browser or native-app-based application to serve as a basic weather app.
2. Search Screen
   1. Allow customers to enter a US city
   2. Call the [openweathermap.org](http://openweathermap.org/) API and display the information you think a user would be interested in seeing. Be sure to has the app download and display a weather icon.
   3. Have image cache if needed
3. **Auto-load the last city searched** upon app launch.
4. **Ask the User for location access**, If the User gives permission to access the location, then retrieve weather data by default

In order to prevent you from running down rabbit holes that are less important to us, try to prioritize the following:

**What is Important**

* Proper function – requirements met.
* Well-constructed, easy-to-follow, commented code (especially comment hacks or workarounds made in the interest of expediency (i.e. // given more time I would prefer to wrap this in a blah blah blah pattern blah blah )).
* Proper separation of concerns and best-practice coding patterns.
* Defensive code that graciously handles unexpected edge cases.

**What is Less Important**

* UI design – generally, design is handled by a dedicated team in our group.
* Demonstrating technologies or techniques you are not already familiar with (for example, if you aren't comfortable building a single-page app, please don't feel you need to learn how for this).

**iOS:**

* For applications that include CocoaPods with their project code, having the Pods included in the code commits as the source is recommended. (Even though it goes against the CocoaPod's general rules).
* Be sure to use safe area insets
* Using Size class wisely for landscape and portrait
* Make sure to use UIKit, we would love to see a combination of both UIKit and SwiftUI if you desire.

**Android:**

* Make sure you are correctly handing any **necessary permissions**.
* Must have technologies as follows

| **Type** | **Must  have** |
| --- | --- |
| Coding Language | Kotlin  (To demonstrate the use of Java, we'd rather you use a combination of Java and Kotlin) |
| Architecture Pattern | MVVM |
| Network library | Retrofit |
| Unit test | Junit |

* Nice to have technologies as follows.

| **Type** | **Nice to have** |
| --- | --- |
| Network - Concurrency | Rx Java, Kotlin Coroutines |
| Unit test | Espresso or Mockito |
| UI | Jetpack Compose |
| Dependency Injection | Dagger 2 or Hilt or Koin |
| Navigation | Jetpack navigation |