

Mobile assignment PS

The goal of this assignment is to evaluate your problem-solving skills, UX judgement and code quality.

Weather, everybody wants to know how it is going to be during the week. Will it be rainy, windy, or sunny? Luckily for us, in the information age, there are open APIs to retrieve information about it.

For this assignment you will be using the API from: <http://openweathermap.org/api>. The API key is provided at the end of the statement, or you can request your own by registering on the website for free.

Your app should at least contain the following screens:

- **Home screen:**
 - Showing a list of locations that the user has bookmarked previously.
 - Show a way to remove locations from the list.
 - Add locations by placing a pin on map.
- **City screen:** once the user clicks on a bookmarked city this screen will appear. On this screen the user should be able to see:
 - Today's forecast, including temperature, humidity, rain chances and wind information
 - Help screen: The help screen should be done using a webview, and contain information of how to use the app, gestures available if any, etc.

The following bonus points can be implemented:

- **Settings page:** where the user can select some preferences like unit system (metric/imperial), any other user setting you consider relevant, e.g., reset cities bookmarked.
- On the city screen: show the 5-days forecast, including: temperature, humidity, rain chances and wind information.
- **On the home screen, implement a list of known locations with search capabilities.**

How navigation occurs, or how elements are placed on the screen is open for interpretation and creativity.

Additionally, the following requirements must be met:

- Alpha/beta versions of the IDE are forbidden, you must work with the stable version of the IDE.
- The API must be consumed in JSON format.
- **The UI must be responsive (landscape and portrait orientations, and tablet resolutions must be supported).**

- The code must be published on [GitHub](#) or Bitbucket. We want to see the progress evolution.

For Android:

- Language must be Java.
- The coordinator layout must be used at least in one of the screens.
- UI must be implemented using 1 activity with multiple fragments.
- Only 3rd party libraries allowed are: GSON or Jackson.
- Compatible with Android 4.1+.
- Unit tests must be present.

For iOS:

- Language can be Objective-C or Swift (latest version).
- [UICollectionView](#) or [UIStackView](#) must be used at least in one of the screens.
- Compatible with iOS 9+ ◦ 3rd party libraries are forbidden.
- [Unit tests](#) must be present.
- [Error handling and error recovery mechanisms](#) must be present.

API information:

API Key: c6e381d8c7ff98f0fee43775817cf6ad

Today's forecast:

Information: <http://openweathermap.org/current>

Example of use:

<http://api.openweathermap.org/data/2.5/weather?lat=0&lon=0&appid=c6e381d8c7ff98f0fee43775817cf6ad&units=metric>

5-days forecast:

Information: <http://openweathermap.org/forecast5>

Example of use:

<http://api.openweathermap.org/data/2.5/forecast?lat=0&lon=0&appid=c6e381d8c7ff98f0fee43775817cf6ad&units=metric>