

LetsGetChecked - iOS Tech Task

Create an app that does the following:

The app should make an api request using the following weather api.

Details can be found here: <https://www.weatherapi.com/>

There are links on the home page for “Sign Up” & “View Docs”.

You will need to sign up for a free account to retrieve an api key - once retrieved you can use the api key to make various weather related requests.

An example of some of the request endpoints:

- <https://api.weatherapi.com/v1/current.json?Key={api-key}&q=Dublin>
Http Get request which returns current weather related data in Json format for a specified query location.
- <https://api.weatherapi.com/v1/forecast.json?Key={api-key}&q=Dublin&dt={yyyy-MM-dd}>
Http Get request which returns weather forecast data in Json format for a specified query location and for a specified date.

The weatherapi service provides various other api endpoints and various other request parameters. The Swagger link below can be used to investigate the various available weather api. <https://app.swaggerhub.com/apis-docs/WeatherAPI.com/WeatherAPI/1.0.2#/>

Description

The app should allow the user to enter a location. Once entered the user can hit a search button which makes the “Current” Weather api call. The query should be configured to return Json (not xml). Parse the response to display the weather for the current location.

Display at least 3 elements of the above data on an action sheet.

It should have a cancel button to dismiss the view so that the user can interact with the search box again.

Every time a search is made with the api, store the query key word in a table below the search box to allow the user to tap on them to search for the weather details again.

It should also meet the following requirements:

- A universal app.
- Use UIKit with auto layout (not SwiftUI)
- Written in Swift.
- Include unit tests (at least one) with a test target included in the workspace.

- Available on a git repo of your choice for me to access (eg. Github, Bitbucket).
- Set up the project to use localisation. Add one translation for one other language.
- Create a networking layer to handle API requests (success and failures) in a generic way.
- Support iOS 15 minimum.
- If you use an external dependency please include an explanation of what the library does and why you choose it.
- Please provide a write up on any architecture choices you made in the app.

This is the minimum required. Have fun and feel free to impress.