IOS Framework Flagright

In this project we have created a framework for iOS, which can be consumed by any app with the following code standards:-

* We have created a **singleton class** which is available to consume by the calling application.
* We have the **FlagrightDeviceMetricsSDK swift class**, which is responsible for gathering device data which are stored in various private methods and variables.

For example, we have the following **private** methods/functions –

* 1. **func** getContacts() (for fetching the no. of contacts)
  2. **func** getBattery() (for fetching the battery percentage)
  3. **func** carrier() (for fetching the carrier related information)
  4. **func** getIPAddress() (for fetching IP address)
  5. **func** checkAccessibilityEnabled() (for fetching accessibility related information)
  6. **func** totalMemory() (for fetching the memory related information)

* Apart from the above methods, we have private constants in the same class which also store some of the device metrics. For example:-
* **let** deviceID (for fetching the device fingerprint)
* **let** ram (for fetching the device ram)
* **let** language for fetching the device language)
* **let** country (for fetching the country code)
* **let** systemVersion (for fetching the OS version)
* Also there are 2 public methods **emit and `init`,** in the same class. These methods can be **used by developers** in the main application for **fetching the updating device details and making the API call respectively.** Their detailed usage can be found in the **Readme.md** file
* For making the API calls we are using **URLSession** ([link](https://developer.apple.com/documentation/foundation/urlsession)) and the API call is handled by the **NetworkLayer swift class**.
* We have also created some classes that help the FlagrightDeviceMetricsSDk class for data collection:-
* UIDeviceExtension + Jailbreak (for checking jailbreak status)
* UIDeviceExtension (for fetching device model name)
* LocationHandler (for fetching longitude and latitude)
* NetworkLayer (for handling API calls)
* We have also shared the documentation.
* The repo contains the readme file that clearly explains how we can add this framework in our app, along with the required permission that we need to.
* We have also created and shared a **Demo App** that helps the developers in the integrating process.