

Geolocation

Fabrice.Kordon@lip6.fr



As an introduction

Huge advantage of mobile devices

- Exploiting location data to self adapt Apps

How it works

- Outside: GPS, network & other sensors
 - ▶ Less precise inside
- Inside (precise): iBeacon (bluetooth) when equipped
 - ▶ Dedicated equipment of the place needed

Since iOS8

- Detailed privacy management

Involved framework

- Core location (handles geolocation only)

Operating geolocation

Two modes

- «when in use» (authorizedWhenInUse)
- «Always» (authorizedAlways)
- Chose one, the other, or both

Continuously updated

Geolocation



Background execution



«Ranging»



Geolocation monitoring

«Region monitoring»



Significative changes



Operating geolocation

Two modes

- «when in use» (authorizedWhenInUse)
- «Always» (authorizedAlways)
- Chose one, the other, or both

Continuously updated

Geolocation

Background execution

«Ranging»

«Always»

«WhenInUse»



Geolocation monitoring

«Region monitoring»

Significative changes



Operating geolocation



Two modes

- «when in use» (authorizedWhenInUse)
- «Always» (authorizedAlways)
- Chose one, the other, or both

Continuously updated

«Fitness» App

Sport activities
geolocation (e.g. jogging)

WhenInUse

«Always»

«WhenInUse»

«Home» App

Switch on at home
Region monitoring
(detect approach)

Always

Operating geolocation



Two modes

- «when
- «Alw
- Chos



Important!

You must declare the way you access
in the info.plist file

`NSLocationUsageDescription`
`NSLocationAlwaysUsageDescription`
`NSLocationWhenInUseUsageDescription`

WhenInUse \subset Always

geolocation (e.g. jogging)

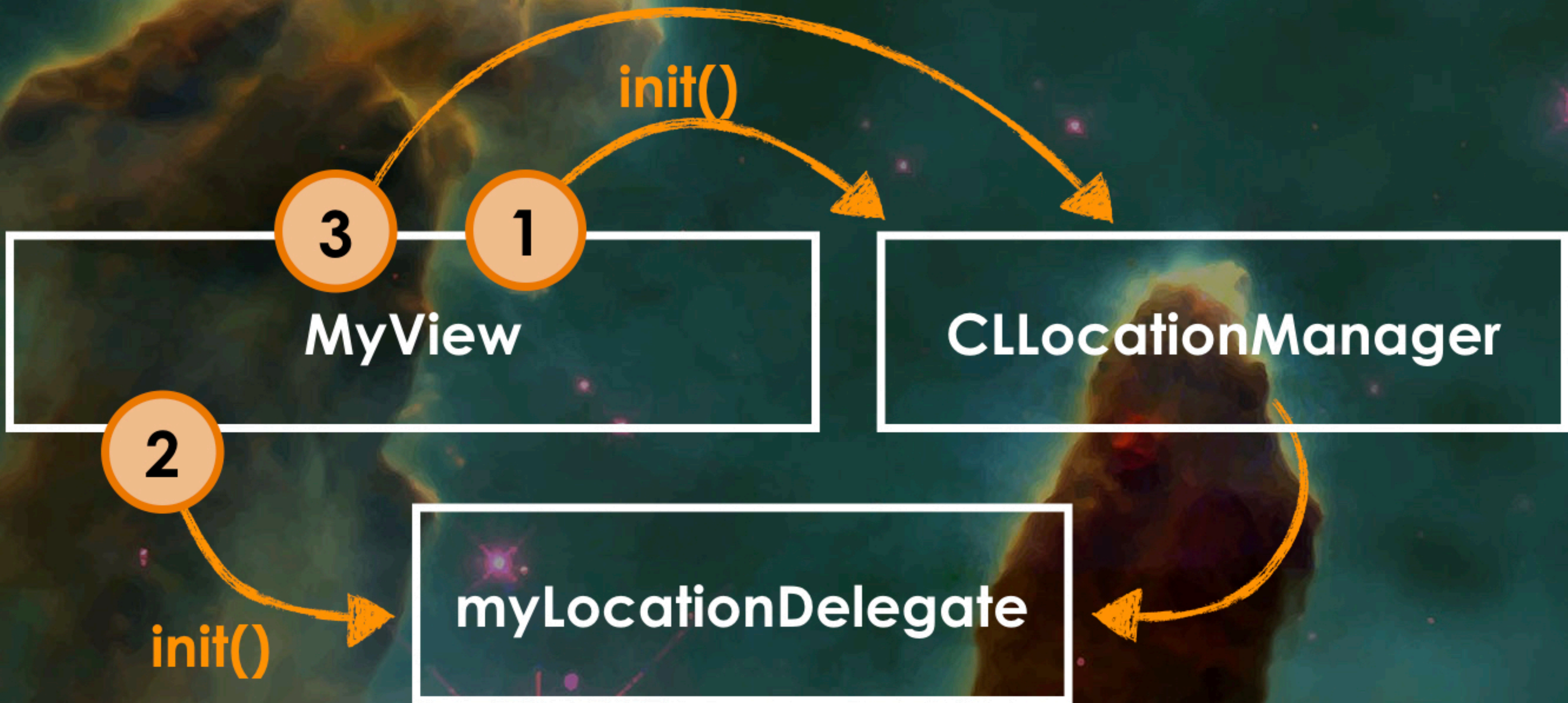
`WhenInUse`

region monitoring
(detect approach)

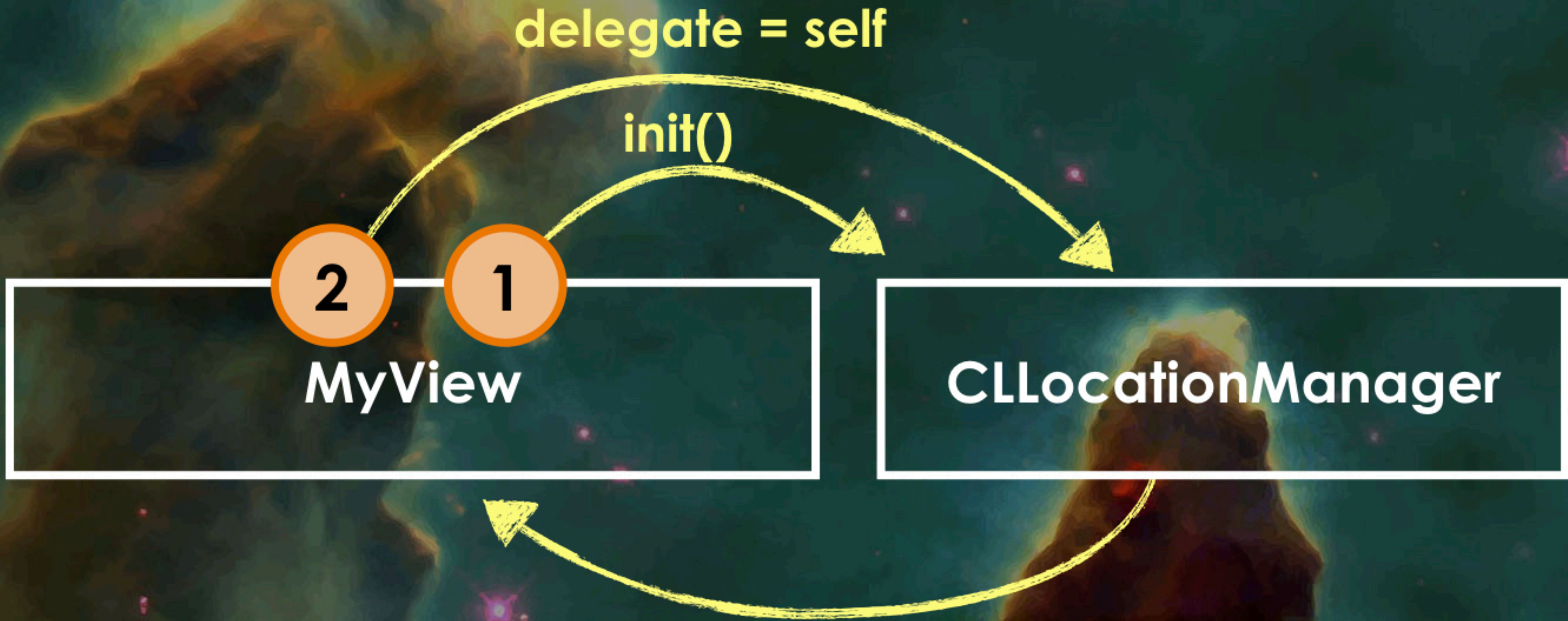
`Always`

Delegation architecture

delegate = myLocationDelegate



Delegation architecture



Practically...

- Delegate is usually `self`

CoreLocation

CLLocationManager management

- Creation (as usual)
- Set the delegate
- Start/Stop Location
 - ▶ `startUpdatingLocation` / `stopUpdatingLocation`
- Start/Stop headings
 - ▶ `startUpdatingHeading` / `stopUpdatingHeading`

CLLocationManagerDelegate protocol

- `locationManager(_:didFailWithError:)`
- `locationManager(_:didUpdateLocations:)`
- `locationManager(_:didUpdateHeading:)`
- etc.



CoreLocation

CLLocationManager management

- Creation (as ...)
- Set the ...
- Start/S ...
- ▶ startUp ...
- Start/S ...
- ▶ StarUp ...
- CLLocation ...
- location ...
- location ...
- locationManager(_:didUpdateHeading:)
- etc.



Precision

desiredAccuracy property

- kCLLocationAccuracyBestForNavigation,
- kCLLocationAccuracyBest,
- kCLLocationAccuracyNearestTenMeters,
- kCLLocationAccuracyHundredMeters,
- kCLLocationAccuracyKilometer,
- kCLLocationAccuracyThreeKilometers



Geolocation & privacy



You must request authorization

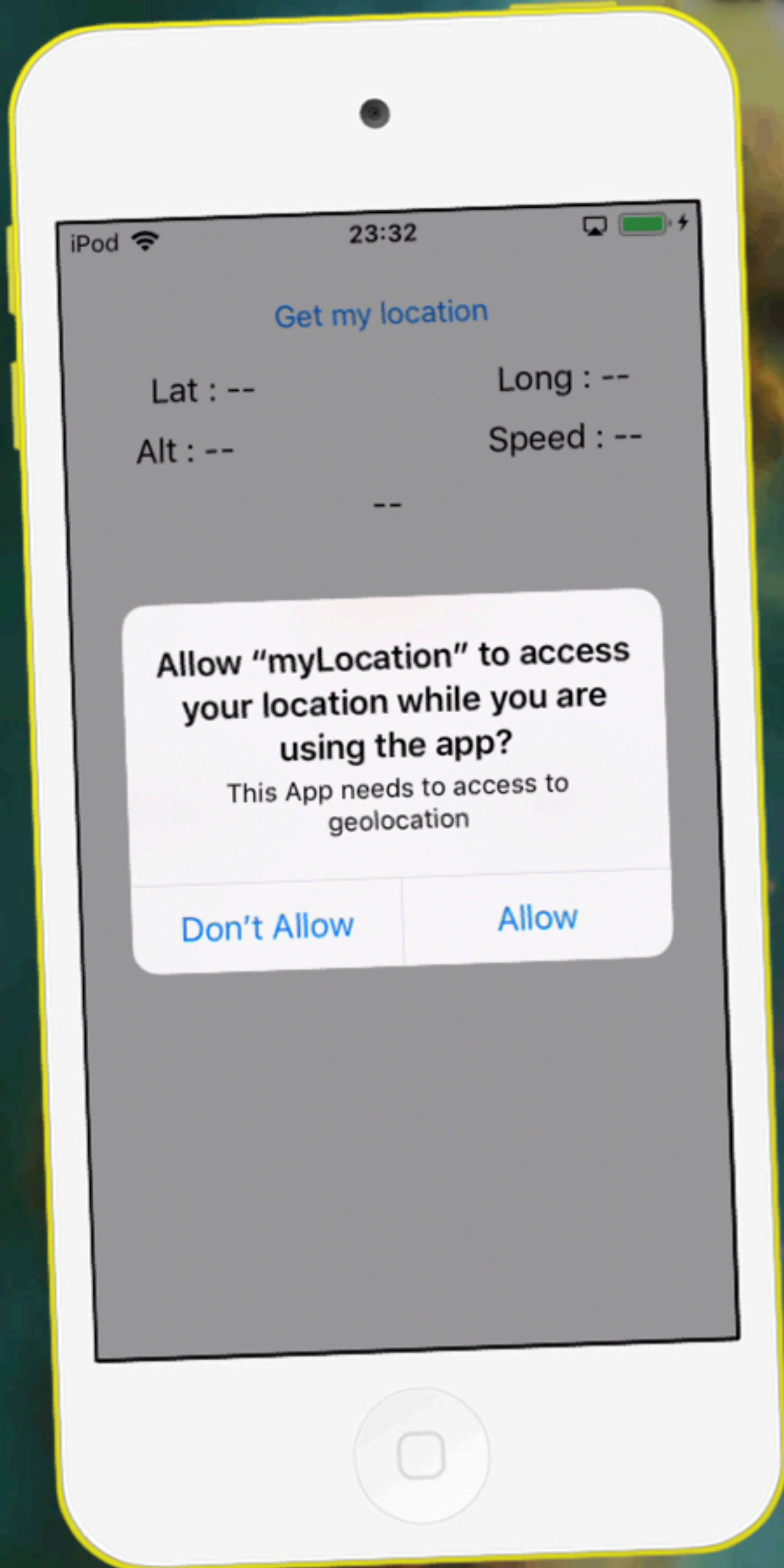
- `locationServicesEnabled`
 - ▶ Relies on preferences settings



You must check availability

- `headingAvailable`
 - ▶ Depends on the device's capabilities

Geolocation & privacy



You must request authorization



`locationServicesEnabled`

▶ Relies on preferences settings



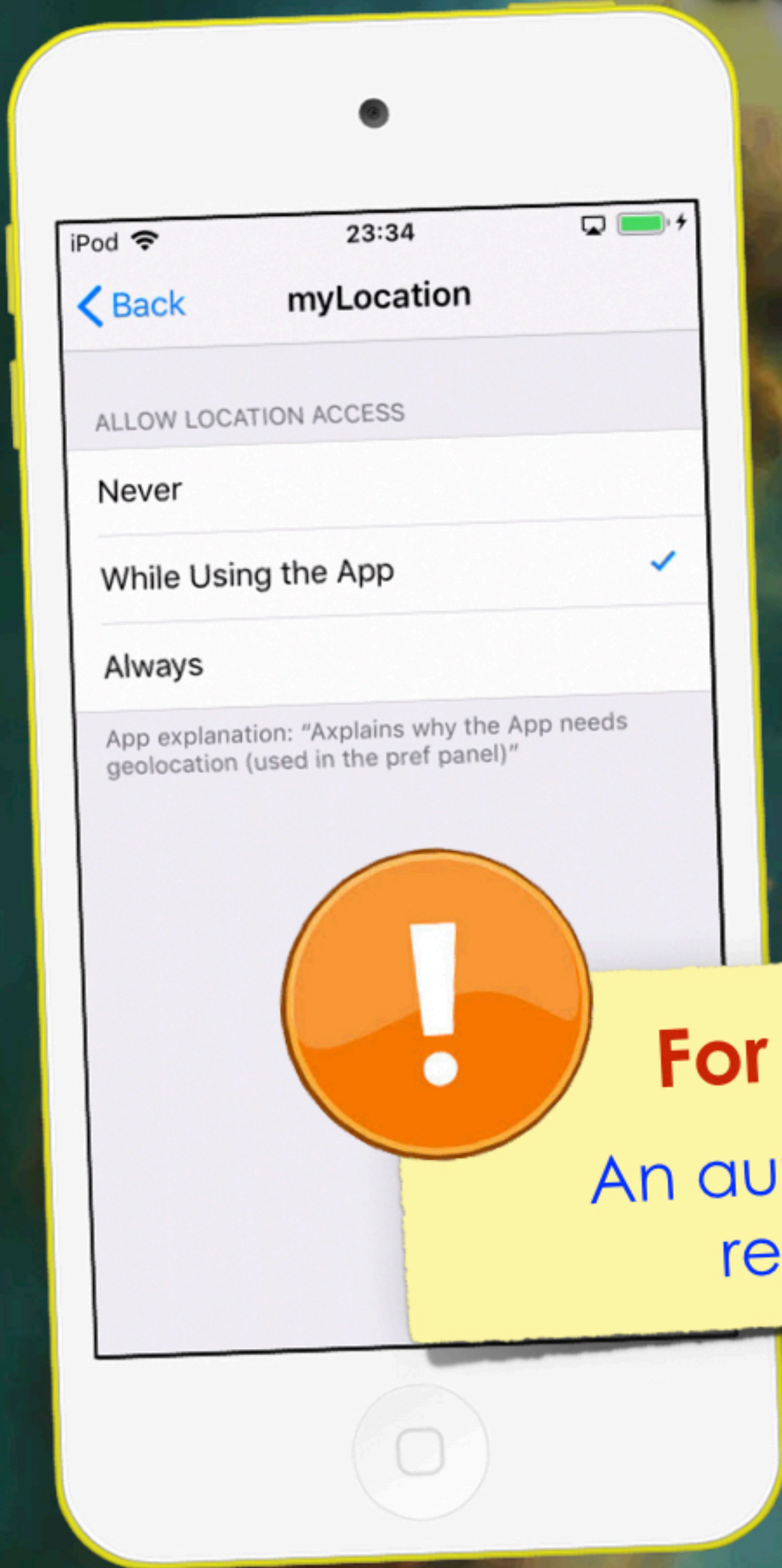
You must check availability



`headingAvailable`

▶ Depends on the device's capabilities

Geolocation & privacy



 **You must request authorization**

- `locationServicesEnabled`
 - ▶ Relies on preferences settings

 **You must check availability**

- `headingAvailable`
 - ▶ Depends on the device's capabilities



For «Always» mode

An authorization reminder is requested after 48h

CLLocation & CLHeadings

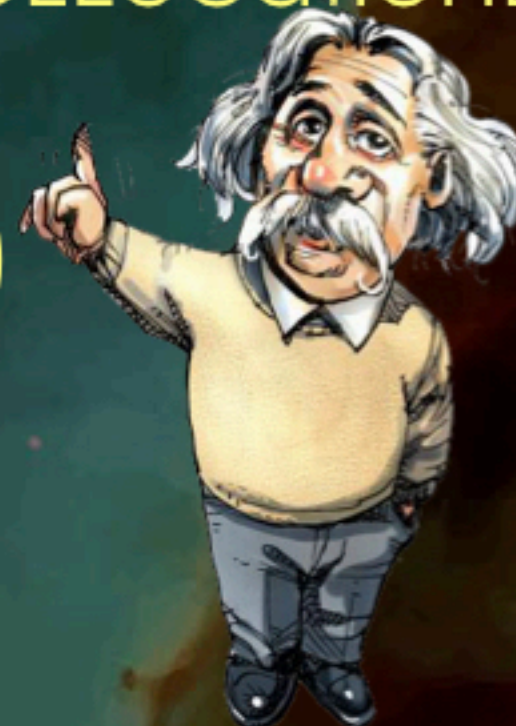
7

CLLocation = location data

- coordinate (CLLocationCoordinate2D)
- altitude (CLLocationDistance)
- timestamp (Date)
- speed (CLLocationSpeed)
- course (CLLocationDirection)
 - ▶ Direction the device is traveling

CLHeadings = orientation

- magneticHeading & trueHeading (CLLocationDirection)
 - ▶ Relative to magnetic of «true» north
- x, y, z (CLHeadingComponentValue)
 - ▶ Electromagnetic data (μ Tesla)



Do not forget!



You must

📍 Include CoreLocation

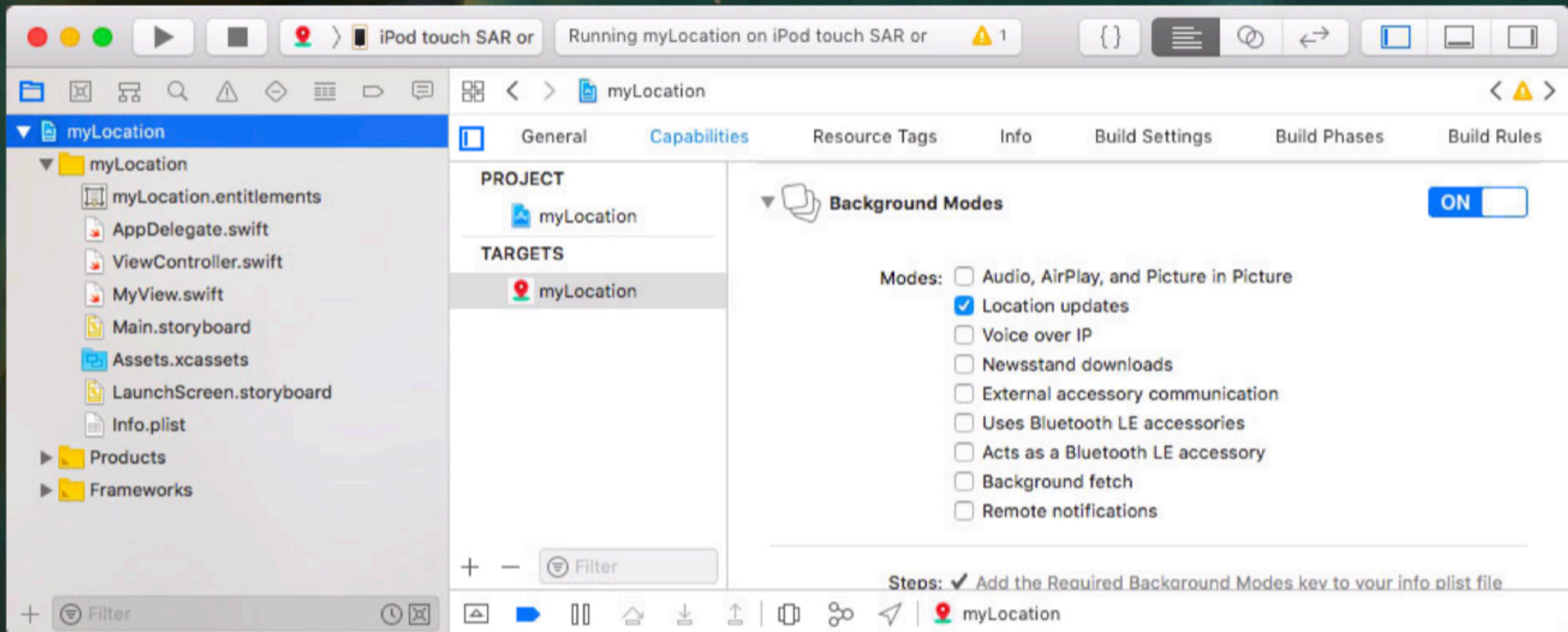
```
1 //
2 // ViewController.swift
3 // WhereAmI
4 //
5 // Created by Fabrice Kordon on 15/10/2018.
6 // Copyright © 2018 Sorbonne Université. All rights reserved.
7 //
8
9 import UIKit
10 import CoreLocation // do not forget
11
12 class ViewController: UIViewController {
13
14     private let v = MyView(frame: UIScreen.main.bounds)
15
16     override func viewDidLoad() {
```

Do not forget!



You must

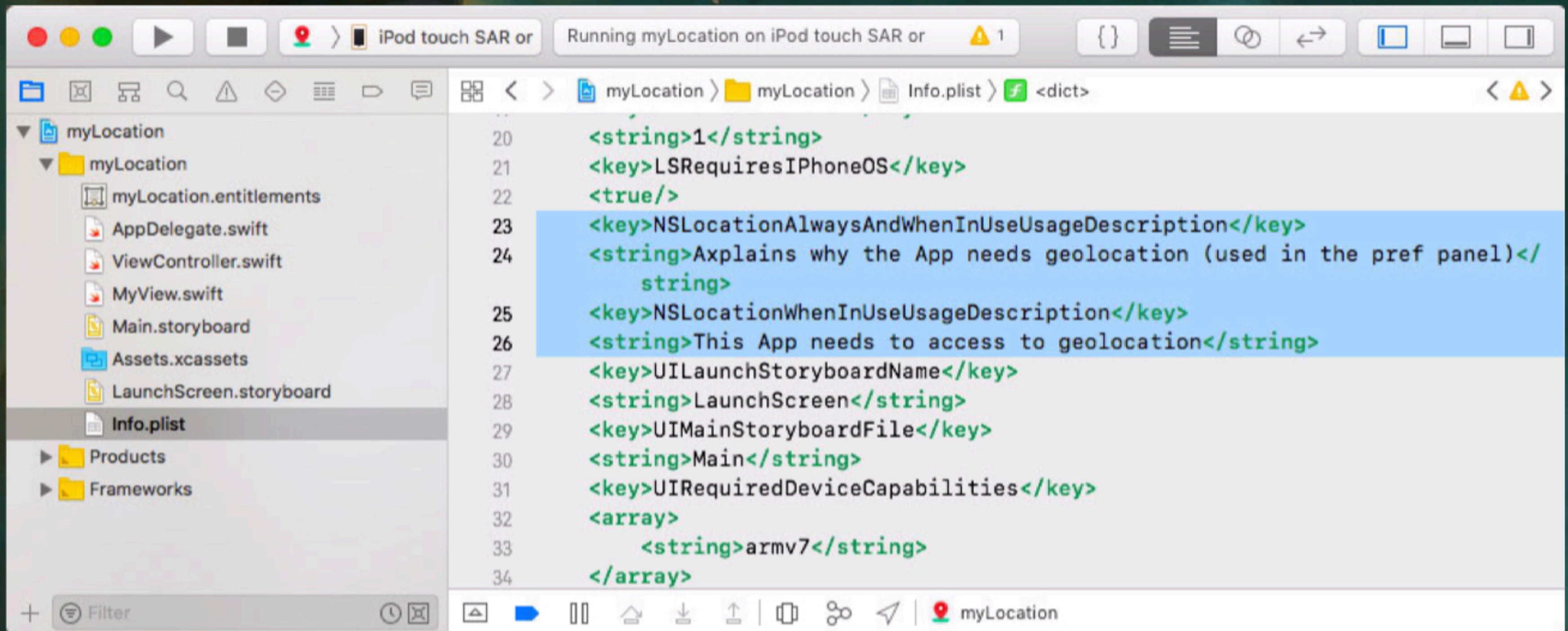
- Include CoreLocation
- Specify (if necessary) that you use location in background



Do not forget!

You must

- Include CoreLocation
- Specify (if necessary) that you use location in background
- Set-up description in info.plist

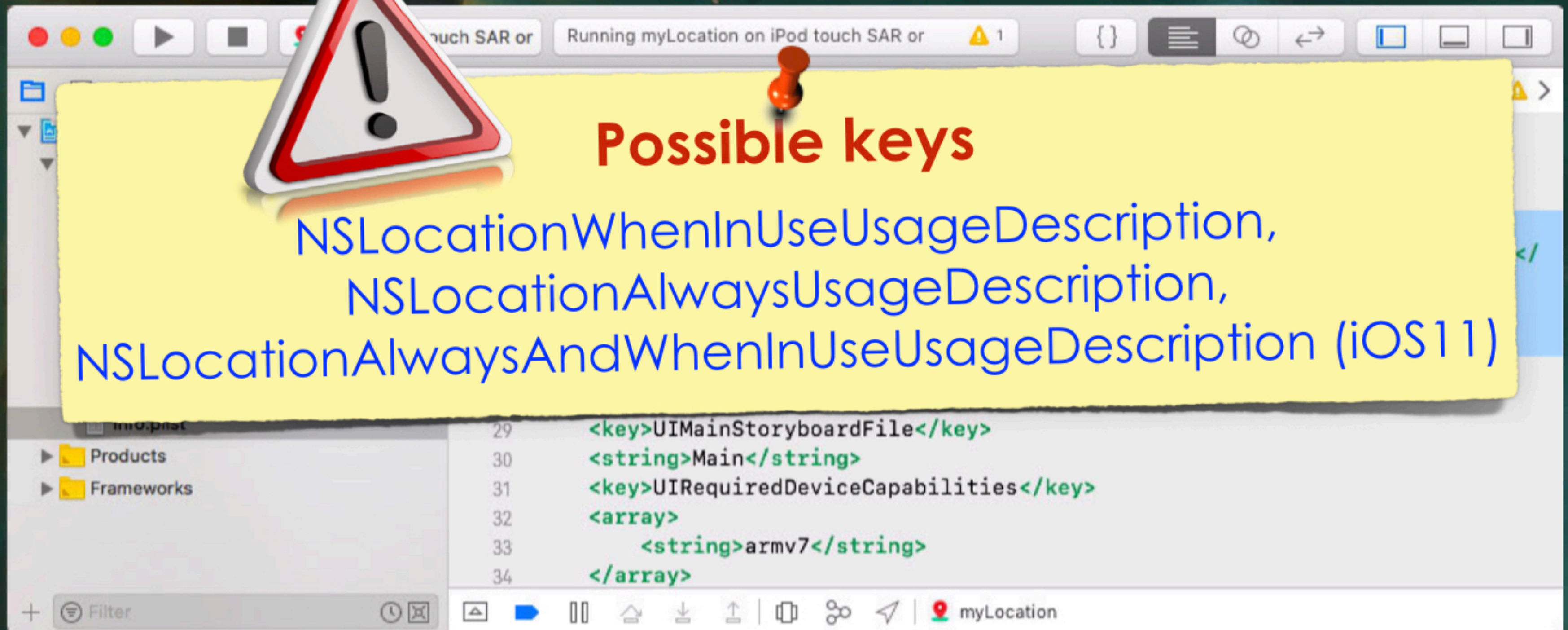


```
20 <string>1</string>
21 <key>LSRequiresiPhoneOS</key>
22 <true/>
23 <key>NSLocationAlwaysAndWhenInUseUsageDescription</key>
24 <string>Explains why the App needs geolocation (used in the pref panel)</string>
25 <key>NSLocationWhenInUseUsageDescription</key>
26 <string>This App needs to access to geolocation</string>
27 <key>UILaunchStoryboardName</key>
28 <string>LaunchScreen</string>
29 <key>UIMainStoryboardFile</key>
30 <string>Main</string>
31 <key>UIRequiredDeviceCapabilities</key>
32 <array>
33 <string>armv7</string>
34 </array>
```

Do not forget!

You must

- Include CoreLocation
- Specify (if necessary) that you use location in background
- Set-up description in info.plist



Possible keys

NSLocationWhenInUseUsageDescription,
NSLocationAlwaysUsageDescription,
NSLocationAlwaysAndWhenInUseUsageDescription (iOS 11)

```
29 <key>UIMainStoryboardFile</key>
30 <string>Main</string>
31 <key>UIRequiredDeviceCapabilities</key>
32 <array>
33 <string>armv7</string>
34 </array>
```

As a conclusion...

Reality is (a bit) more complex...

- Geolocation management is quite subtle
 - ▶ You will need a bit of practice
- But you have the bases
 - ▶ Be aware of traps
-  recommended



Remind that you must require authorization

- Not too early (Apple's recommendation)
- But prior to to the startUpdating<X>