

«NotifyMe2»

Fabrice.Kordon@lip6.fr



Goal of the example

2



Rework NotifyMe...

- Handle notifications while the App is in foreground
- Embeds actions in the notification



To do so?

- Use the `UNUserNotificationCenterDelegate` protocol

Demo



Changes in AppDelegate

4

```
import UIKit
import UserNotifications // Do not forget it!

@UIApplicationMain
class AppDelegate: UIResponder, UIApplicationDelegate,
                 UNUserNotificationCenterDelegate {

    var window: UIWindow?
```

Changes in AppDelegate

```
func application(_ application: UIApplication,
                 didFinishLaunchingWithOptions
                 launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool {
    let b1 = UNNotificationAction(identifier: "URL_SU",
                                title: "Jump to Sorbonne Université",
                                options: .foreground)
    let b2 = UNNotificationAction(identifier: "URL_LIP6",
                                title: "Jump to LIP6",
                                options: .foreground)

    let actions = UNNotificationCategory(identifier: "actions",
                                        actions: [b1, b2],
                                        intentIdentifiers: ["b1", "b2"],
                                        options: .customDismissAction)
    UNUserNotificationCenter.current().setNotificationCategories([actions])
    let center = UNUserNotificationCenter.current()
    center.delegate = self
}
```

Changes in AppDelegate

```
let rvc = ViewController()
self.window?.rootViewController = rvc
center.requestAuthorization(options:[.badge, .alert, .sound]) { (ok, err) in
    if !ok {
        rvc.deactivate()
        let a = UIAlertController(title: "Problem",
                                message: "Notification are deactivated",
                                preferredStyle: .alert)
        a.addAction(UIAlertAction(title: "OK", style: .default,
                                   handler: nil))
        self.window?.rootViewController?.present(a,
                                                  animated: true,
                                                  completion: nil)
    }
}
window?.makeKeyAndVisible()
return true
}
```

Changes in AppDelegate

```
// UNUserNotificationCenterDelegate protocol

func userNotificationCenter(_ center: UNUserNotificationCenter,
                           didReceive response: UNNotificationResponse,
                           completionHandler: @escaping () -> Void) {
    completionHandler() // Demander si on continue (préférences)
    var url : URL?
    if response.actionIdentifier == "URL_SU" {
        url = URL(string: "https://sorbonne-universite.fr")
    } else if response.actionIdentifier == "URL_LIP6" {
        url = URL(string: "https://lip6.fr")
    }
    if url != nil { // in case no action has been selected
        UIApplication.shared.open(url!, options: [:], completionHandler: nil)
    }
}

func userNotificationCenter(_ center: UNUserNotificationCenter,
                           willPresent notification: UNNotification,
                           completionHandler: @escaping (UNNotificationPresentationOptions) -> Void) {
    completionHandler(.sound) // Check with preferences if we can go on
    let a = UIAlertController(title: notification.request.content.title,
                             message: notification.request.content.body,
                             preferredStyle: .alert)
    a.addAction(UIAlertAction(title: "OK", style: .default, handler: nil))
    self.window?.rootViewController?.present(a, animated: true, completion: nil)
}
```

Changes in AppDelegate

4

```
// No change then  
func applicationWillResignActive(_ application: UIApplication) {}  
func applicationDidEnterBackground(_ application: UIApplication) {}  
func applicationWillEnterForeground(_ application: UIApplication) {}  
func applicationDidBecomeActive(_ application: UIApplication) {}  
func applicationWillTerminate(_ application: UIApplication) {}
```

```
}
```


Changes in the ViewController

```
func activateNotification (delay : Bool) {
    let centre = UNUserNotificationCenter.current()
    let n = UNMutableNotificationContent()
    n.title = "Listen everybody!"
    if answer?.text != nil {
        n.body = (answer?.text!)!
    }
    n.categoryIdentifier = "NotifyMe-Alert"
    n.sound = UNNotificationSound.default
    n.badge = NSNumber(value: UIApplication.shared.applicationIconBadgeNumber + 1)
    n.categoryIdentifier = "actions"
    var t : UNTimeIntervalNotificationTrigger?
    if delay {
        t = UNTimeIntervalNotificationTrigger(timeInterval: 5, repeats: false)
    } else {
        t = UNTimeIntervalNotificationTrigger(timeInterval: 0.1, repeats: false)
    }
    // UUID -> to retrieve unique identifiers (otherwise, messages are lost)
    let nr = UNNotificationRequest(identifier: UUID().uuidString,
                                   content: n, trigger: t!)
    centre.add(nr, withCompletionHandler: nil)
}
```

As a conclusion...

You now know almost everything on notifications

- Apple Push Notification Service (APNS) is missing
 - ▶ But it requires big energy (provisioning profiles)
 - ▶ But principles remain the same

Anyway, this allows enriched Apps

- Do not hesitate to use it!

