

```
// Pede permissão para mandar notificações
```

③ OK! Tudo certo então? Posso continuar fazendo minhas coisas?

Calma lá! Ainda tem alguns passos, como definir como serão nossas notificações, e em qual momento elas devem ser mandadas. Pra isso, devemos colocar o delegate do `UNUserNotificationCenter`. Então, também dentro do `didFinishLaunchingWithOptions`, colocamos:

```
UNUserNotificationCenter.current().delegate = self
```

Ótimo! Agora vamos criar uma extensão do `AppDelegate` para conformar com o `UNUserNotificationCenterDelegate`

```
// Essa função é para quando recebemos notificação
```

```
func userNotificationCenter(_ center: UNUserNotificationCenter, willPresent notification: UNNotification, withCompletionHandler completionHandler: @escaping (UNNotificationPresentationOptions) -> Void) {
```

```
    // show the notification alert (banner), and with sound
    completionHandler([.alert, .sound, .badge])
}
```

```
// Essa função é chamada quando o usuário clica na notificação
```

```
func userNotificationCenter(_ center: UNUserNotificationCenter, didReceive response: UNNotificationResponse, withCompletionHandler completionHandler: @escaping () -> Void) {
    completionHandler()
}
```

```
        tificationCenter.current().requestAuthorization(options:
[.alert, .badge, .sound], completionHandler: { authorized, error in
    if authorized {
        DispatchQueue.main.async(execute: {
            application.registerForRemoteNotifications()
        })
    }
})
```