Beacons in iOS

Overview

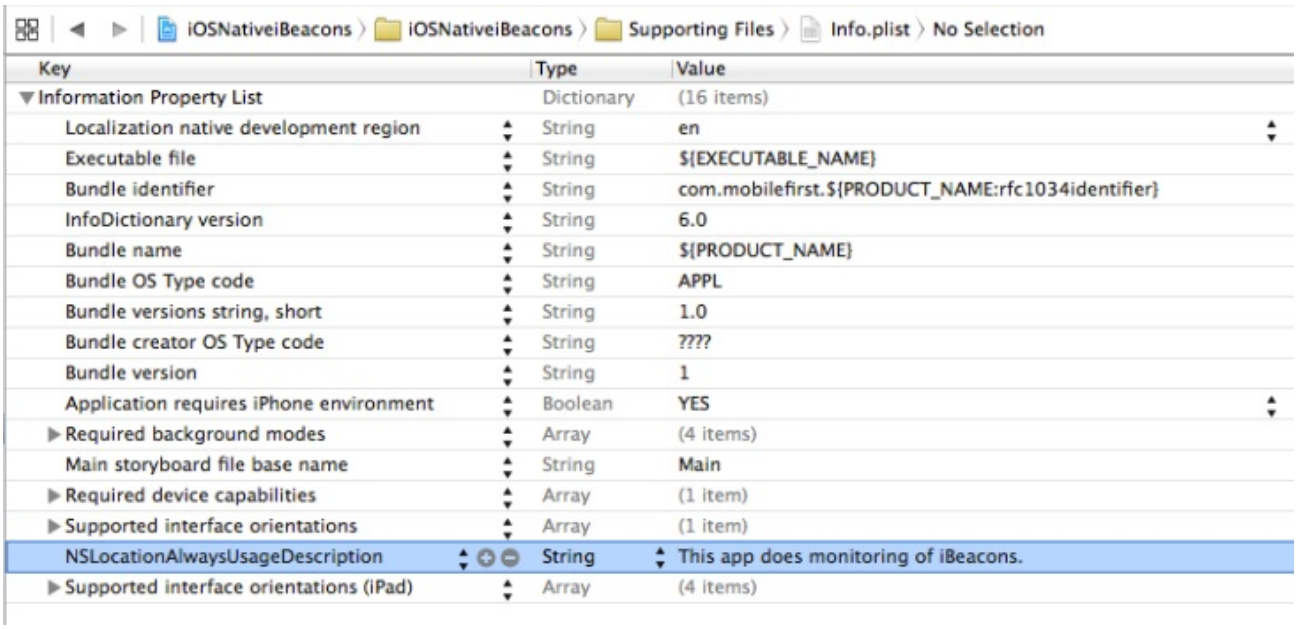
This tutorial discusses the client-side API for working with beacons in iOS.
This tutorial is a continuation of Working with Beacons (../) (introduction, setup, and server-side API).

The following topics are covered:

- Configuring you application
- Simplified APIs for ranging/monitoring iBeacons
- Running the sample application

Configuring your application

1. On iOS 8 and later, to enable Location services, add `NSLocationAlwaysUsageDescription` in the iOS project `Info.plist`.



2. Copy the `iOSNativeiBeaconsLibrary` folder of the iOSNativeiBeacons project into your native iOS application. That folder contains the library that provides the simplified set of APIs for ranging/monitoring iBeacons.
   To request permission to use location services and local/push notifications, write the following code:

```
- (void) viewDidLoad {
 [super viewDidLoad];
 ...
 [[WLBeaconsLocationManager sharedInstance] requestPermissionToUseLocationServices];
 [[WLBeaconsLocationManager sharedInstance] requestPermissionToUseNotifications];
}
```

Simplified APIs for ranging/monitoring iBeacons

Loading beacons and triggers from the server

To load information pertaining to beacons, triggers, and their associations from the server and storing it in the application JSONStore, write the following code:

```
[[WLBeaconsAndTriggersJSONStoreManager sharedInstance] loadBeaconsAndTriggersFromAda
pter:@"BeaconsAdapter" withProcedure:@"getBeaconsAndTriggers" withCompletionHandler:^(bo
ol success, NSString *error) {
 if (success) {
  [self showBeaconsAndTriggers];
 } else {
  [self updateView:error];
 }
}];
```

Triggering actions

To set the monitoring/ranging of iBeacons and to fire trigger actions, write the following code:

```
/>
[[WLBeaconsLocationManager sharedInstance] startMonitoring];
```

To let the user opt out of iBeacon monitoring, write the following code:

```
[[WLBeaconsLocationManager sharedInstance] stopMonitoring];
 // Optionally reset monioring/ranging state
 [[WLBeaconsLocationManager sharedInstance] resetMonitoringRangingState];
```

Turning the device into an iBeacon

To turn a device into an iBeacon, write the following code:

```
NSString *uuidString = @"3d402cf0-3691-4bd9-97ff-0b0a93a160ef";
int major = 1;
int minor = 4417;

[[WLBeaconsLocationManager sharedInstance] turnIntoiBeaconWithUuid:uuidString withMajor:ma
jor withMinor:minor withStatusHandler:^(NSString *turnIntoiBeaconStatusDetails) {
 [self updateView:turnIntoiBeaconStatusDetails];<br />
}];
```

Generally, use this feature for testing and development purposes. The UUID string must match the

UUID of the beacons that are registered on server.

Increasing the application running time

To increase the running time of the application when it runs in background (as part of iBeacon monitoring), write the following code:

```
- (void)applicationDidEnterBackground:(UIApplication *)application {
 ...
 [[WLBeaconsLocationManager sharedInstance] notifyApplicationDidEnterBackground];
}
- (void)applicationDidBecomeActive:(UIApplication *)application {
 ...
 [[WLBeaconsLocationManager sharedInstance] notifyApplicationDidBecomeActive];
}
```

This code increases the running time of the application up to 3 minutes from the default 10 seconds.

Handling local notifications

To handle local notifications that originate from the application, write the following code:

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)l
aunchOptions {
 ...
 UILocalNotification *localNotif = [launchOptions objectForKey:UIApplicationLaunchOptionsLocalN
otificationKey];
 if (localNotif) {
  application.applicationIconBadgeNumber = application.applicationIconBadgeNumber - 1;
  [self processLocalNotification:localNotif];
 }
 return YES;
}

- (void)application:(UIApplication *)application didReceiveLocalNotification:(UILocalNotification *)n
otification {
 application.applicationIconBadgeNumber = application.applicationIconBadgeNumber - 1;
 [self processLocalNotification:notification];
}

- (void)processLocalNotification:(UILocalNotification *)notification {
 NSString *alertMessage = [notification.userInfo objectForKey:@"alertMessage"];
 UIAlertView *alert = [[UIAlertView alloc] initWithTitle:@"Received local notification" message:alert
Message delegate:nil cancelButtonTitle:@"OK" otherButtonTitles:nil, nil];
 [alert show];
}
```

Running the sample application

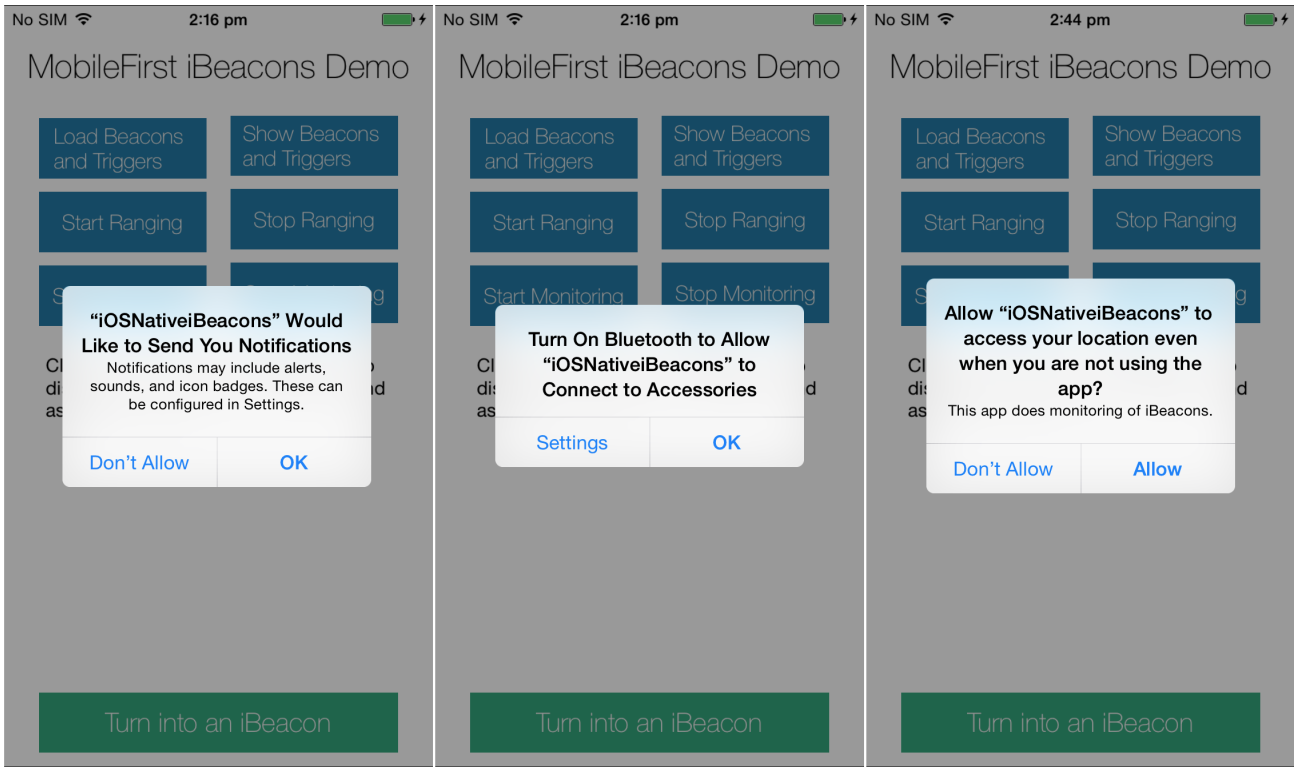
The sample is in two parts:

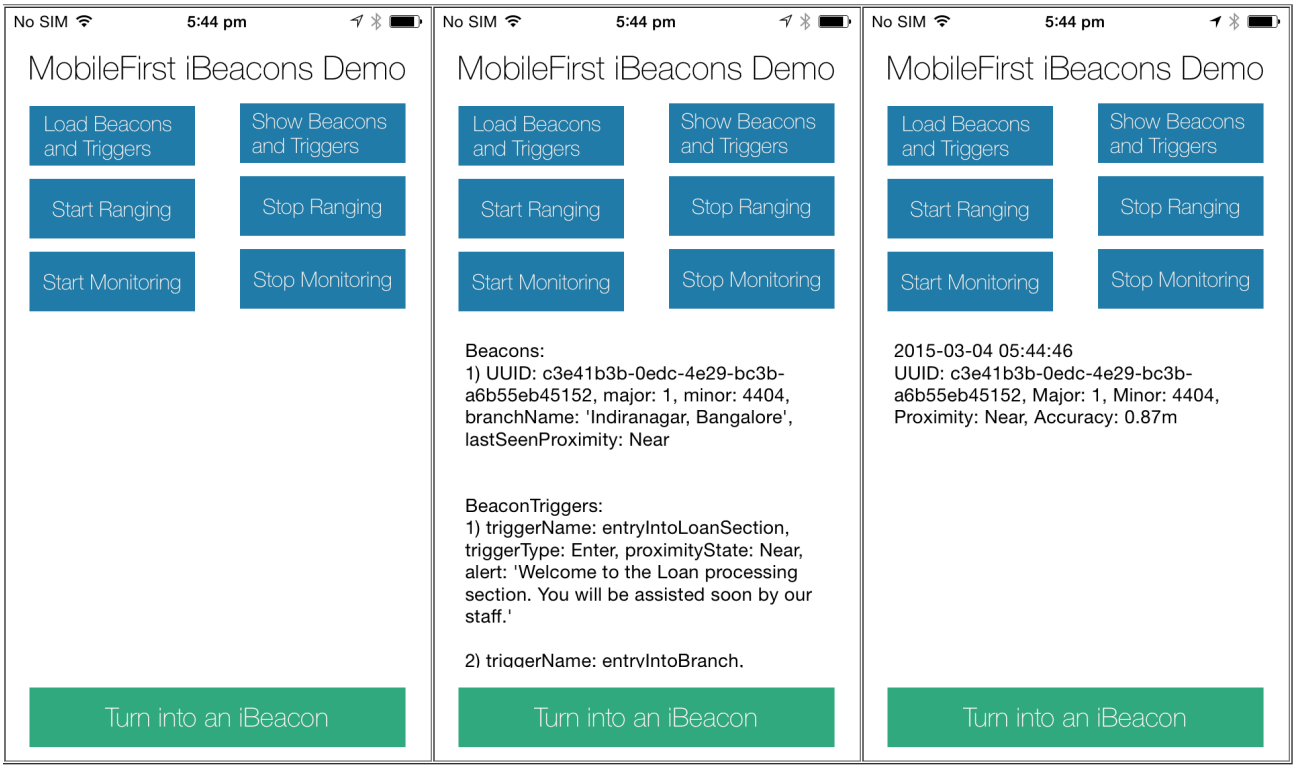
- The Beacons (https://github.com/MobileFirst-Platform-Developer-Center/Beacons/tree/release71) MobileFirst project contains a **MobileFirst native API** that you can deploy to your MobileFirst Server instance.
- The BeaconsObjC (https://github.com/MobileFirst-Platform-Developer-Center/BeaconsObjC/tree/release71) native project contains **a native library** that provides a simplified set of APIs for ranging/monitoring iBeacons and a **native iOS application** that demonstrates the usage of those APIs.

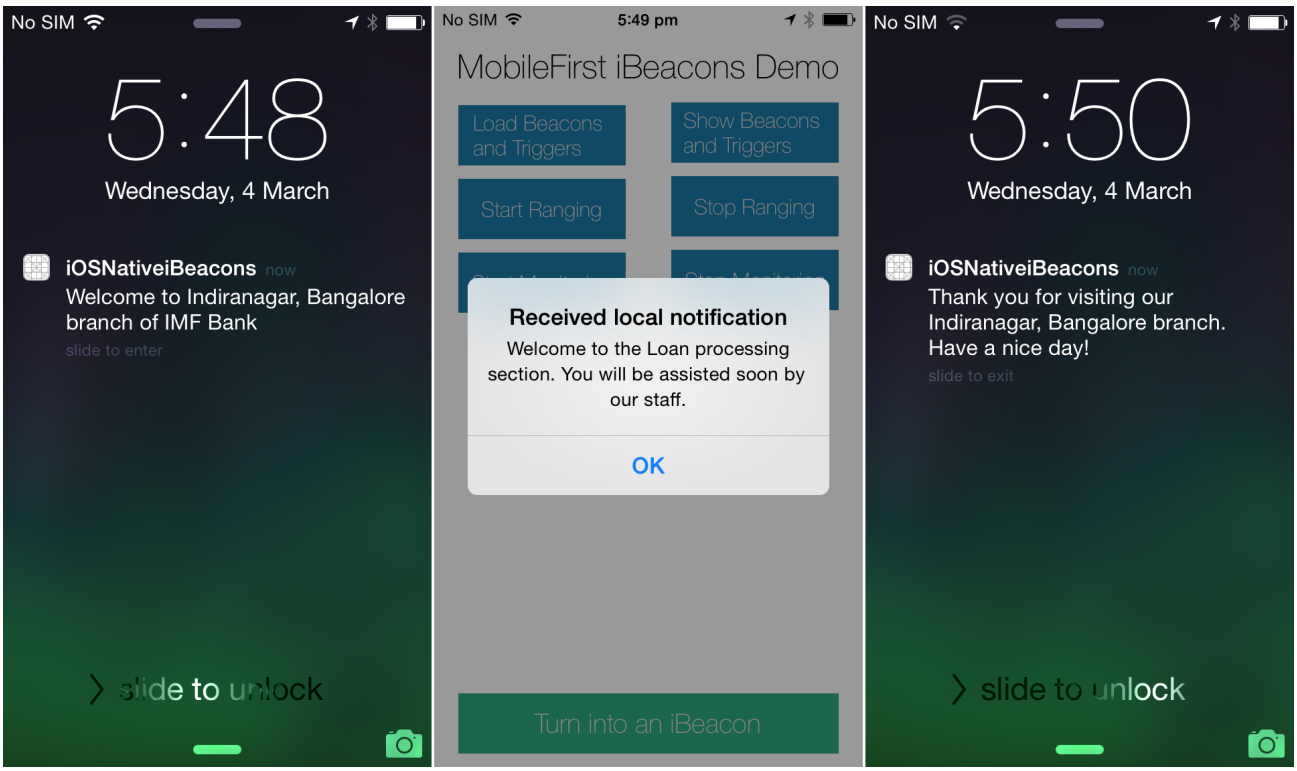
Make sure to update the `worklight.plist` file in iOSNativePush with the relevant server settings.

The sample showcases a simple banking scenario where actions are triggered based on proximity:

- When the user enters a Bank branch, a notification action is triggered with a Welcome message.
- When the user enters the loan processing section, an alert is triggered with a message stating that a bank personnel will assist the user soon.
- When the user exits the Bank branch, a notification action is triggered with a message.




No SIM
5:44 pm
MobileFirst iBeacons Demo
Load Beacons and Triggers
Show Beacons and Triggers
Start Ranging
Stop Ranging
Start Monitoring
Stop Monitoring
Turn into an iBeacon
Beacons:
1) UUID: c3e41b3b-0edc-4e29-bc3b-a6b55eb45152, major: 1, minor: 4404, branchName: 'Indiranagar, Bangalore', lastSeenProximity: Near
BeaconTriggers:
1) triggerName: entryIntoLoanSection, triggerType: Enter, proximityState: Near, alert: 'Welcome to the Loan processing section. You will be assisted soon by our staff.'
2) triggerName: entryIntoBranch,
2015-03-04 05:44:46
UUID: c3e41b3b-0edc-4e29-bc3b-a6b55eb45152, Major: 1, Minor: 4404, Proximity: Near, Accuracy: 0.87m


No SIM
5:48
Wednesday, 4 March
iOSNativeiBeacons now
Welcome to Indiranagar, Bangalore branch of IMF Bank
slide to enter
slide to unlock
No SIM
5:49 pm
MobileFirst iBeacons Demo
Load Beacons and Triggers
Show Beacons and Triggers
Start Ranging
Stop Ranging
Received local notification
Welcome to the Loan processing section. You will be assisted soon by our staff.
OK
Turn into an iBeacon
No SIM
5:50
Wednesday, 4 March
iOSNativeiBeacons now
Thank you for visiting our Indiranagar, Bangalore branch. Have a nice day!
slide to exit
slide to unlock