

# FAQs - Modules

## Mapping FAQs

Android Push Notifications

iOS Push Notifications

## Mapping FAQs

ON THIS PAGE

### Why do I not see my map?

#### Do you have the correct configuration values?

*In order for PWMapKit to successfully retrieve building assets from the server, your application identifier, access key and signature key must be correctly provided in the call to `+[MaasCore setApplicationID:accessKey:signatureKey:encryptionKey]`.*

#### Is the building identifier correct?

*The identifier listed in the MaaS portal application should be used in the call to `-[PWMapView loadBuildingWithIdentifier:]`.*

#### Has the application availability been properly set for the given venue?

*The application availability has to be set at the venue level. To check and set the application availability navigate to the venue in MaaS portal and tap the Edit Venue button. You should see a list of applications with check boxes. Ensure that your application checkbox is checked.*

### Why am I unable to create a route?

#### Have you checked the route in the MaaS portal?

*Make sure that your points of interest and waypoints are connected by segments in the portal application.*

#### Are you creating an accessible route?

*If so, then it's possible that one of your endpoints is inaccessible or that the two endpoints are not traversable using only accessible points.*

#### Have you cleared your cache since making changes online?

*The map will cache building resources loaded from the server in order to work in offline situations. If you have recently made changes to the data online, you may need to clear your local cache and reload the building.*

### Why is the user's blue dot not showing up?

### Have you checked to ensure your floor ID mapping is correct?

*The blue dot will not be displayed on the map if it is unable to convert between the floor level reported by your location manager and the Phunware-specific floor identifier. These values are provided to the indoor location manager in a mapping dictionary.*

### Have you registered your location manager with the map view?

*In order for the map to perform location tracking and display a blue dot, you must register an indoor location manager with the map view. This can be done by using a location manager that conforms to the `PWLocationManager` protocol in your call to -*  
`[PWMapView`  
`registerIndoorLocationManager:]`.

## Why is "snap to route" not working?

### Is your route snapping tolerance set too low?

*In areas with excess interference or poorly performing location services, the user's blue dot may not stay snapped to the route as expected. Increasing the route snapping tolerance may solve this issue.*

## Why does it take so long for the blue dot to display after changing floors?

### Is there adequate location services coverage in the area?

*This issue is most likely caused by an excessive delay in receiving an updated location from the location manager. The SDK will immediately switch the floor upon learning of a newly reported location on that floor, so make sure that the problem isn't related to interference or coverage.*

## Why is the map drawing the blue dot on a wall or the route through walls?

There are two instances in which a wall or other immovable obstacle is traversed by a route or blue dot:

### Is the route being drawn through a wall?

*If yes, there are two possibilities: 1. The closest waypoint to a route endpoint is on the opposite side of a wall. In this case, adding additional waypoints to the correct side of the wall will likely solve the problem. 2. There is a segment defined on the server between two points that traverses a wall. In*

*this case, the defined segment should be considered invalid and removed from the server through the portal interface.*

#### **Is the blue dot being drawn on a wall?**

*This is likely due to inadequate indoor positioning information. The blue dot is displayed based on information provided by a location manager and can be erroneous in areas with lots of interference.*

## Why are my points of interest not displayed?

#### **Are zoom levels enabled?**

*When zoom levels are used, a point of interest may not be visible until the user has zoomed in far enough. In addition, it may disappear again after zooming into a certain level. In other words, points may be shown or hidden based on a specified range of zoom levels.*

#### **Have you asked the map to hide them?**

*The map can hide building annotations based on annotation identifier. Once they have been hidden, you must ask the map to show them again in order for them to appear.*

#### **Is the point active in the portal application?**

*Points can be made inactive on the server for maintenance or staging purposes. Any such point will not be loaded on the device at all.*

## Why isn't turn-by-turn automatically switching to the next maneuver?

#### **Is the indoor user tracking mode set to follow or follow with heading?**

*In order for turn-by-turn maneuvers to switch automatically based on the users indoor location the user must be in follow or follow with heading mode.*

## Why isn't turn-by-turn working smoothly with blue dot?

#### **Is there adequate indoor location coverage in your facility?**

*Turn-by-turn maneuver switching fidelity is largely related to indoor location fidelity. If you facility has areas of poor indoor location coverage that will*

## How do I configure push notifications for Android devices for Broadcast Campaigns?

Push notifications in Android rely on Google Cloud Messaging (GCM). When setting up your app, you also need to register with GCM in order to receive push notifications from the M3 backend.

To do so, follow the steps that can be found in Google's official documentation at:

<https://developers.google.com/cloud-messaging/android/client>

1. Create a project on Firebase console
  - a. Choose 'Add Firebase to Android app'
  - b. In the Firebase console, the package name should be the same as application id in build.gradle
2. The Firebase console creates a google-services.json file and downloads it to your default Downloads folder. Drag the downloaded google-services.json file to your Android project (as documented in <https://developers.google.com/cloud-messaging/android/client>)
3. Add the sender id to build.gradle of the app

```
a. defaultConfig {
    resValue 'string', 'GCMSenderId', "\"YOURSENDERIDHERE\""
}
```

4. In the [MAAS portal](#), for the newly created Android app, replace the API Key and sender id with the values for ServerKey and SenderId on the Firebase console (under CloudMessaging section)

- [Android Push Notifications](#)
- [iOS Push Notifications](#)

## How do I configure push notifications for iOS devices for Broadcast Campaigns?

This article is designed to help developers using the messaging SDK to configure push notifications for iOS based applications. Apple has some [excellent documentation](#) available to assist developers. This article assumes a basic knowledge of the developer console for apple developers.

### Step-by-step guide

1. Obtain an apple developer account. You will need an apple developer account, and access

to developer.apple.com. Note that you must have sufficient privileges in your account to manage applications and certificates.

2. Go to "Certificates, Identifiers and Profiles" area and create a new App ID for your app under "Identifiers > App IDs". *Note that for push messages, you need an explicit ID.*
3. Under "App Services" you must enable "Push Notifications"
4. Save your App ID
5. You now need to setup Development / Production SSL Certificates for your application, you do this by editing the App ID, and scrolling down to the Push Notifications area, and selecting "Create Certificate" under the appropriate type of certificate. Note that to debug applications, you MUST use Development SSL certificates. *Note: This is especially useful if you want to debug deep linking or meta-data from a push notification when it arrives to your application in debug mode.*
6. Follow the instructions to create a Certificate Signing Request (CSR) for the certificate.
7. Once complete, download the certificate to your computer, and install it into your Keychain.
8. Once installed, you must export the certificate from your computer, and create a p.12 file. You will need to select a password for this p.12 file - **remember this password, as you will need it later.**
9. Login to MaaS Portal and enter the Applications section under Account
10. Select the appropriate application you created already, and note the section available to configure your application for certificates. Browse and upload your p.12 file, enter the password you chose in the step above, and choose if this certificate is a Production or Development certificate.
11. Save the application in the portal.
12. In XCode - you will need to configure your application with the appropriate bundleID, and provisioning profile depending on if your using a Development or Distribution configuration.
13. Run your application - and your all set!



## Related articles

- [Android Push Notifications](#)
- [iOS Push Notifications](#)

Terms: Use of this software requires review and acceptance of our terms and conditions for developer use located at <http://www.phunware.com/terms/>