

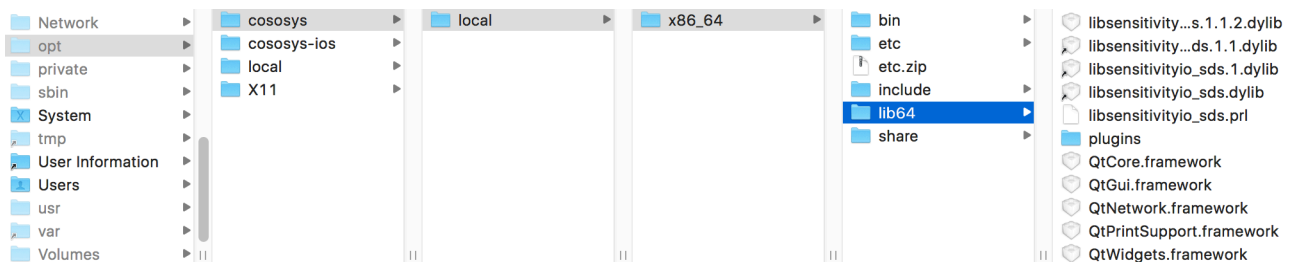
## Overview

This document describes the steps required to set up a macOS/iOS application with sensitivity.io using Xcode.

## Prerequisites

- Xcode supporting at least mac OS 10.7
- sensitivity.io archive for macOS/iOS that needs to be unpacked in `/opt/cososys` or `/opt/cososys-ios` respectively
- Qt with minimum version 5.6.3 for macOS and 5.9.6 for iOS

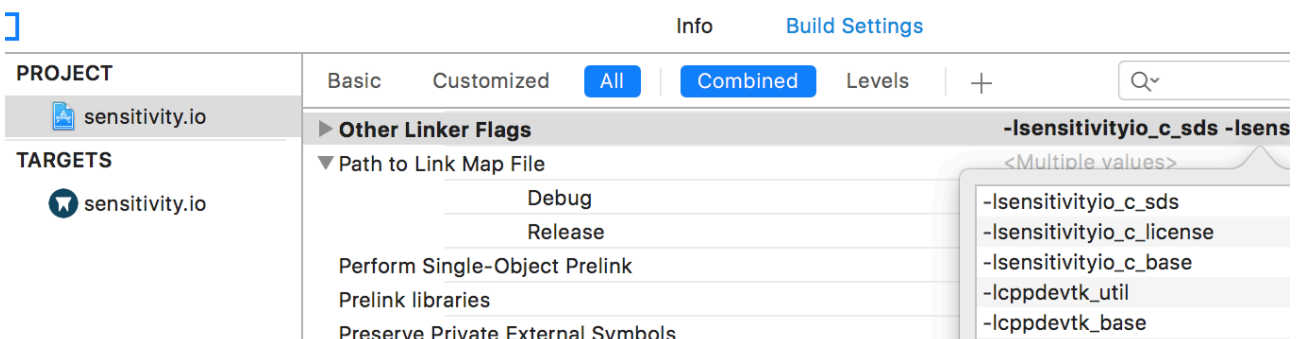
If you don't want to install Qt on your mac, you need to use the `Qt_redist` archive which contains all Qt dependencies required by sensitivity.io. It needs to be unpacked besides all sensitivity.io libraries.



## Project setup

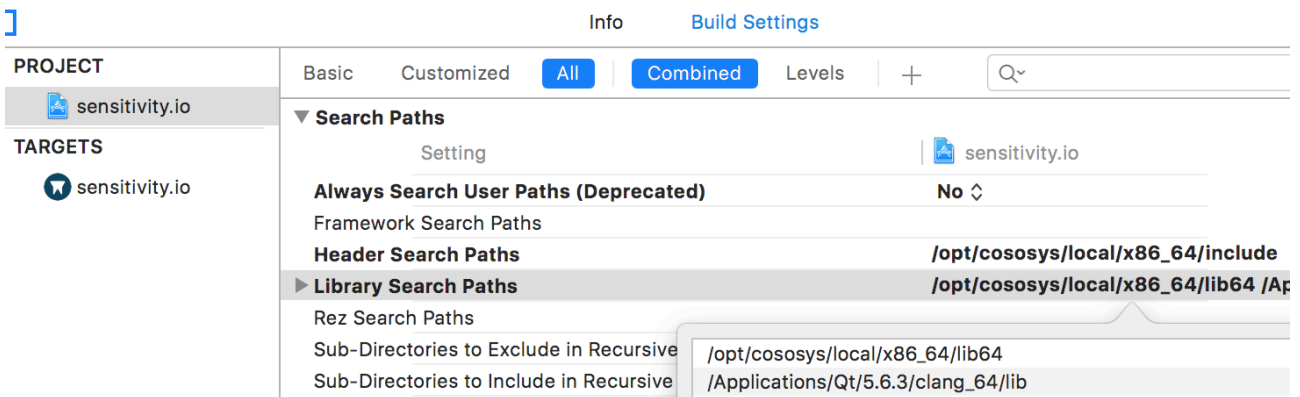
### 1) Linking – Other Linker Flags

Provide all libraries required by sensitivity.io to your *project* Build Settings.



## 2) Search Paths

Provide sensitivity.io library and header search paths to your *project* Build Settings.

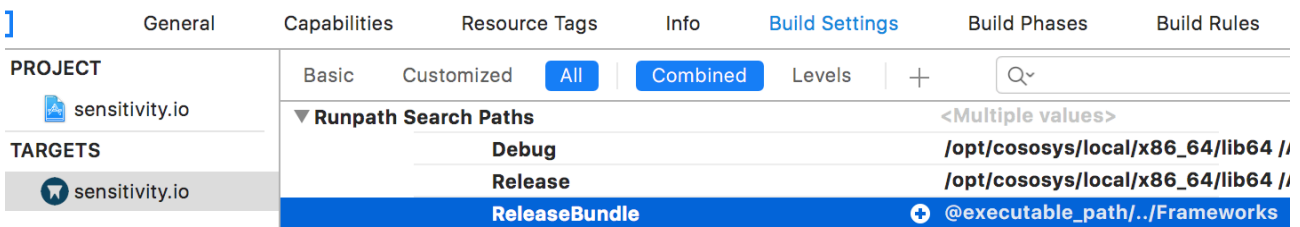


The Qt dependencies will be first checked in sensitivity.io library directory and if not found they will be looked after in the Qt installation path, if it was provided. You can omit it if you're using the sensitivity.io Qt redistributed dependencies.

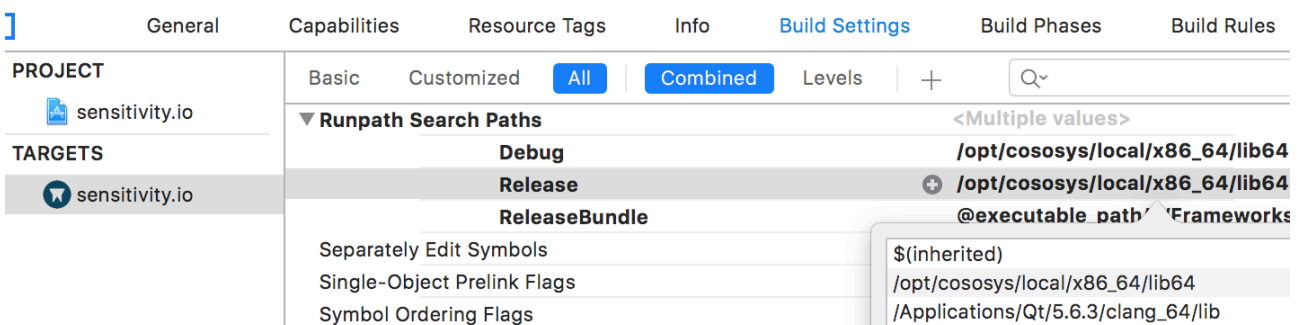
## 3) Linking – Runpath search paths

Add sensitivity.io and Qt to Runpath Search Paths in your *target* Build Settings for macOS.

For the applications distributed through Mac App Store or the ones for which you need to provide all dependencies within your application bundle, use the following setup:



while for the rest use:



In this last case, the sensitivity.io and Qt dependencies need to already be set up on the system you want to run the application on, for it to work as expected.

The next steps are required only for macOS if you want to distribute your application through Mac App Store or if you need to provide all dependencies within your application bundle.

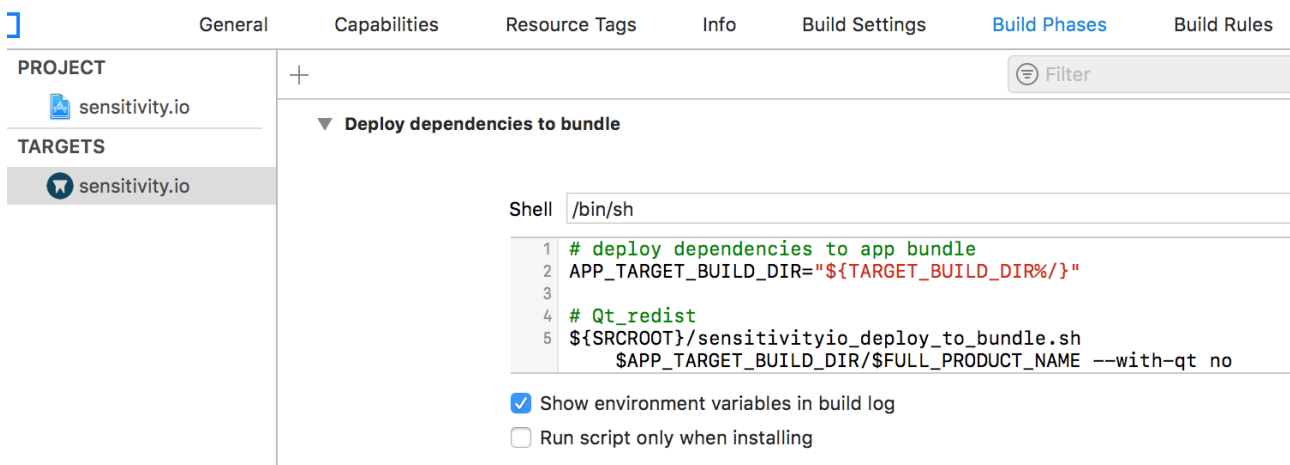
#### 4) Post build step

This step will copy all sensitivity.io and its dependencies in the application bundle.

To do that, add a post build step to your *target* Build Settings by creating a new Run Script Phase and running *sensitivityio\_deploy\_to\_bundle.sh*.

The script requires the application path where to deploy sensitivity.io dependencies and has the following options:

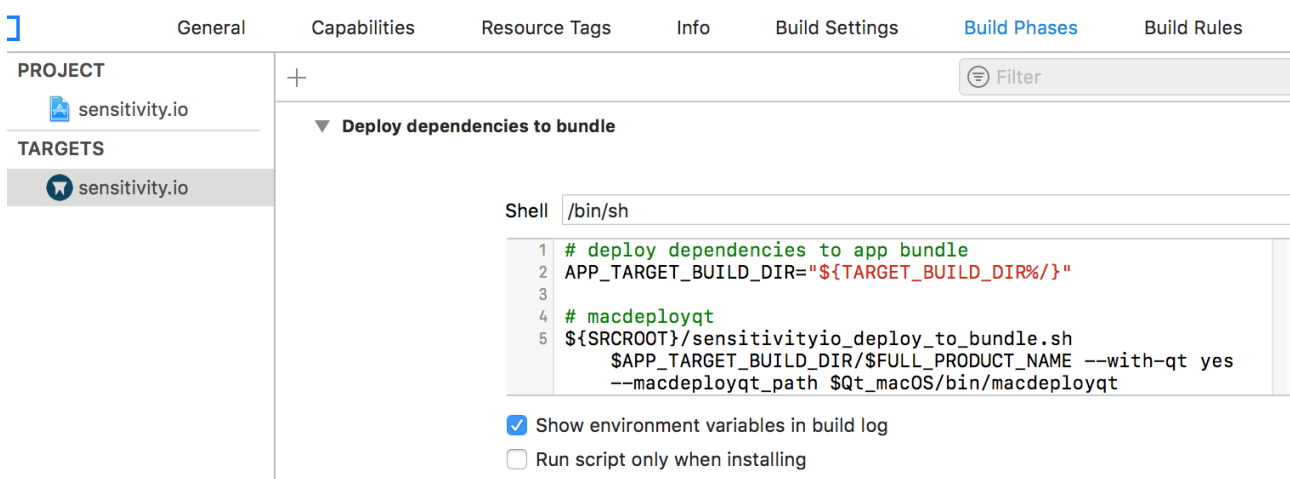
- *--with-qt* with *yes/no* values and which represents the use of installed Qt or Qt\_redist from */opt/cososys*
- *--macdeployqt\_path* represents the path to Qt macdeployqt command and is required when *--with-qt* value is *yes*



This screenshot shows the Xcode Build Phases window for a project named 'sensitivity.io'. The 'Build Phases' tab is selected. Under the 'Post-builds' section, a new phase titled 'Deploy dependencies to bundle' has been added. The shell is set to '/bin/sh'. The script content is as follows:

```
1 # deploy dependencies to app bundle
2 APP_TARGET_BUILD_DIR="${TARGET_BUILD_DIR%/*}"
3
4 # Qt_redist
5 ${SRCROOT}/sensitivityio_deploy_to_bundle.sh
   $APP_TARGET_BUILD_DIR/$FULL_PRODUCT_NAME --with-qt no
```

Below the script, the checkbox 'Show environment variables in build log' is checked, and 'Run script only when installing' is unchecked.



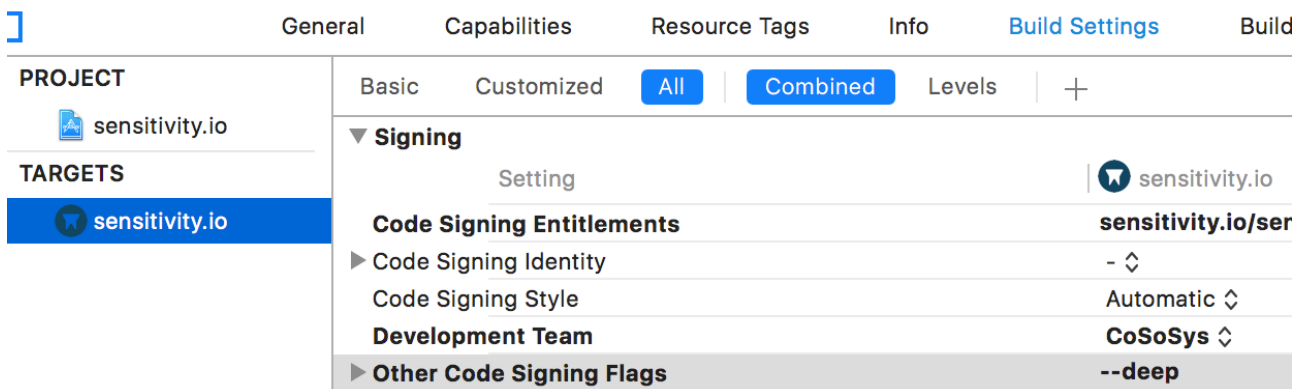
This screenshot shows the Xcode Build Phases window for the same project, but with the macOS target selected. The 'Deploy dependencies to bundle' phase is configured with the following script:

```
1 # deploy dependencies to app bundle
2 APP_TARGET_BUILD_DIR="${TARGET_BUILD_DIR%/*}"
3
4 # macdeployqt
5 ${SRCROOT}/sensitivityio_deploy_to_bundle.sh
   $APP_TARGET_BUILD_DIR/$FULL_PRODUCT_NAME --with-qt yes
   --macdeployqt_path $Qt_macOS/bin/macdeployqt
```

The 'Show environment variables in build log' checkbox is checked, and 'Run script only when installing' is unchecked.

## 5) Signing

Add `--deep` flag to Other Code Signing Flags in your *target* Build Settings.



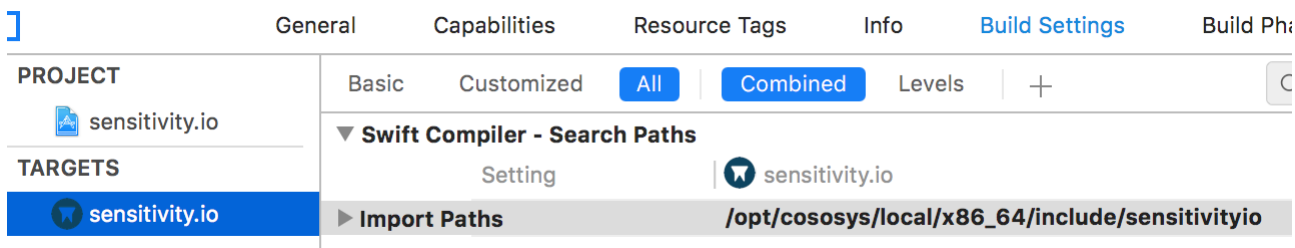
The image shows the Xcode Build Settings interface for a project named 'sensitivity.io'. The 'Build Settings' tab is selected. Under the 'Signing' section, the 'Other Code Signing Flags' is set to '--deep'. The 'Code Signing Entitlements' are set to 'sensitivity.io/ser'. The 'Code Signing Identity' is set to '-', the 'Code Signing Style' is set to 'Automatic', and the 'Development Team' is set to 'CoSoSys'.

Setting	Value
Code Signing Entitlements	sensitivity.io/ser
Code Signing Identity	-
Code Signing Style	Automatic
Development Team	CoSoSys
Other Code Signing Flags	--deep

## Use sensitivity.io Objective-C libraries with Swift

Provide the path to the libraries module.modulemap files to Import Paths in Swift Compiler - Search Paths from your *target* Build Settings.

The module.modulemap file of each library is placed alongside its header files.



The image shows the Xcode Build Settings interface for a project named 'sensitivity.io'. The 'Build Settings' tab is selected. Under the 'Swift Compiler - Search Paths' section, the 'Import Paths' is set to '/opt/cososys/local/x86\_64/include/sensitivityio'.

Setting	Value
Import Paths	/opt/cososys/local/x86_64/include/sensitivityio