

Overview

- [Resources](#)
- [iOS SDK builds differences](#)
- [Examples building](#)
- [Known issues](#)

Resources

Download	API docs	Examples source code
2,5	2.5	2.5
2.6	2.6	2.6

The source code of the examples is located at [Github](#) and is used to comment the examples in the present documentation. For example, this link [line 78](#) points to the seventy eighth line in the ViewController.m file of the Media Devices example the revision with the hash of 79a318b6.

iOS SDK builds differences

1. iOS SDK 2.6, since build [2.6.19](#), includes two frameworks: Objective C and Swift, and since [2.6.95](#) Objective C framework is shipped as XCFramework.
2. Due to WebRTC library update, iOS SDK builds since [2.6.86](#) will work only on arm64 hardware (iPhone 5s and newer).
3. Version 2.5 examples cannot be build with version 2.6 due to changes in video capturing to allow custom video capturer implementation. Links in the table above point to compatible source code for each SDK version.

Examples building

[Here](#) is described how to build examples using Xcode before 10. Examples building procedure with Xcode 10 and newer slightly differs, and described [here](#). Since iOS SDK build [2.6.114](#), examples may be built with SPM packages, the procedure is described [here](#).

Known issues

1. To play WebRTC stream correctly in iOS application in fullscreen mode it is necessary to follow client device aspect ratio

Symptoms: when stream is published with aspect ratio 4:3 (640x480 for example), video looks stretched in iOS application in fullscreen mode.

Solution: stream should be published with aspect ratio 16:9 (1280x720 for example)

2. Build [2.6.37](#) is compatible only with WCS builds since [5.2.935](#)

Symptoms: streaming works in example application, but buttons state and status messages remain unchanged, there can be messages in application logs like

```
Error converting audioState JSON to FPWCSApi2MediaState
Error converting videoState JSON to FPWCSApi2MediaState
```

Solution: update WCS build to [5.2.935](#) or newer

3. It is recommended to use Rosetta when building applications in Xcode for iOS Simulator on Mac with M1 CPU.