

Сборка примеров при помощи Xcode 10 и выше

Подготовка примеров для сборки

Сборка с использованием локальной копии SDK

Warning

Этот способ является устаревшим и не рекомендуется, начиная со сборки [2.6.97](#)

1. Устанавливаем Cocoapods для сборки зависимостей.

```
sudo gem install cocoapods
```

2. Скачиваем исходный код примеров на Mac.

```
git clone https://github.com/flashphoner/wcs-ios-sdk-samples.git
```

3. Скачиваем архив iOS SDK

```
wget http://flashphoner.com/downloads/builds/flashphoner_client/wcs-ios-sdk/2.6/WCS-iOS-SDK-2.6.x.tar.gz
tar -xvzf WCS-iOS-SDK-2.6.x.tar.gz
```

4. После распаковки iOS SDK получаем фреймворки

- в сборках до [2.6.86](#) два фреймворка

```
FPWCSEApi2.framework
FPWCSEApi2Swift.xcframework
```

- в сборке [2.6.86](#) и новее три фреймворка

```
FPWCSEApi2.framework
FPWCSEApi2Swift.xcframework
WebRTC.xcframework
```

- в сборке [2.6.95](#) и новее Objective C фреймворк поставляется также, как и остальные, в виде XCFramework

```
FPWCSPi2.xcframework
FPWCSPi2Swift.xcframework
WebRTC.xcframework
```

- в сборке [2.6.97](#) и новее структура архива SDK аналогична структуре, доступной в CocoaPods

```
FPWCSPi2
FPWCSPi2Swift
FPWebRTC
```

5. Копируем распакованные фреймворки в папку с примерами

- в сборках до [2.6.97](#)

```
mkdir -p wcs-ios-sdk-samples/Frameworks
cp -R FPWCSPi2.xcframework FPWCSPi2Swift.xcframework WebRTC.xcframework
wcs-ios-sdk-samples/Frameworks
```

- в сборке [2.6.97](#) и новее

```
mkdir -p wcs-ios-sdk-samples/sdk/fp_wcs_api2/Pods
cp -R FPWCSPi2 FPWCSPi2Swift WebRTC wcs-ios-sdk-
samples/sdk/fp_wcs_api2/Pods
cd wcs-ios-sdk-samples
mv Podfile Podfile.public
mv Podfile.local Podfile
```

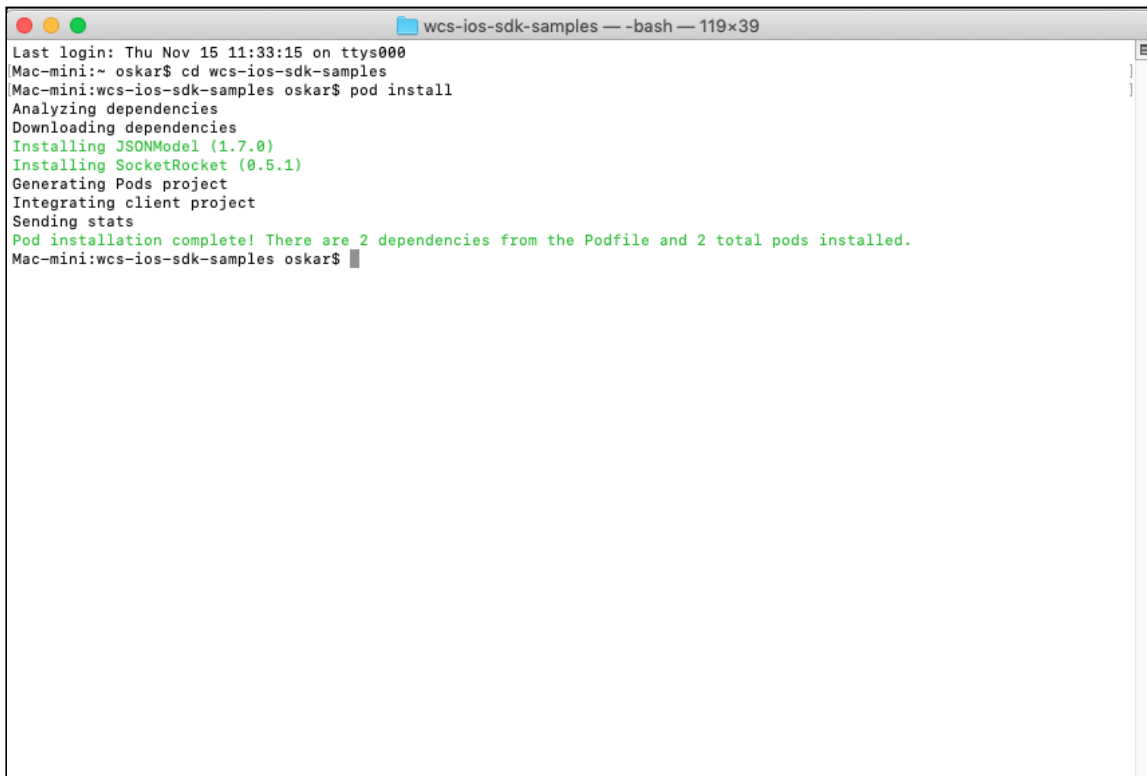
6. Примеры готовы к сборке

В результате получаем папку примеров с фреймворком (iOS SDK). Можно приступить к сборке.

```
bash-3.2$ ls -la
total 64
drwxr-xr-x  14 Flashphoner  staff    448 Jun 17 13:12 .
drwxr-xr-x+  47 Flashphoner  staff   1504 Jun 17 13:07 ..
drwxr-xr-x   5 Flashphoner  staff    160 Jun 17 13:11 Frameworks
-rw-r--r--   1 Flashphoner  staff    245 Apr 22  2019 Info.plist
-rw-r--r--   1 Flashphoner  staff   1567 Apr 13 13:16 Podfile
-rw-r--r--   1 Flashphoner  staff    465 Jun 17 13:12 Podfile.lock
drwxr-xr-x  10 Flashphoner  staff    320 Jun 17 13:12 Pods
-rw-r--r--   1 Flashphoner  staff     27 Apr 22  2019 README.md
-rw-r--r--   1 Flashphoner  staff     35 Apr 22  2019 README.txt
drwxr-xr-x  12 Flashphoner  staff    384 Apr 13 13:16 Swift
drwxr-xr-x  19 Flashphoner  staff    608 Oct 16  2020 WCSEExample
drwxr-xr-x@   6 Flashphoner  staff    192 Jun 17 12:24 WCSEExample.xcodeproj
drwxr-xr-x@   5 Flashphoner  staff    160 Jul  2  2020 WCSEExample.xcworkspace
-rwxr-xr-x   1 Flashphoner  staff  10331 Apr 13 13:16 build_example.sh
```

7. Запускаем Cocoapods

```
pod install
```



```
wcs-ios-sdk-samples — -bash — 119x39
Last login: Thu Nov 15 11:33:15 on ttys000
Mac-mini:~ oskar$ cd wcs-ios-sdk-samples
Mac-mini:wcs-ios-sdk-samples oskar$ pod install
Analyzing dependencies
Downloading dependencies
Installing JSONModel (1.7.0)
Installing SocketRocket (0.5.1)
Generating Pods project
Integrating client project
Sending stats
Pod installation complete! There are 2 dependencies from the Podfile and 2 total pods installed.
Mac-mini:wcs-ios-sdk-samples oskar$
```

Сборка с загрузкой SDK из Cocoapods

✓ Success

Этот способ поддерживается, начиная со сборки [2.6.97](#)

1. Устанавливаем Cocoapods для сборки зависимостей.

```
sudo gem install cocoapods
```

2. Скачиваем исходный код примеров на Mac.

```
git clone https://github.com/flashphoner/wcs-ios-sdk-samples.git
```

3. Запускаем Cocoapods

Этот шаг может занять некоторое время, поскольку фреймворк FPWebRTC сравнительно велик по объему

```
pod install
```

```
Alexanders-MacBook-Pro:iOS-SDK-Samples-2.6 Flashphoner$ pod install
Analyzing dependencies
Downloading dependencies
Installing FPWCSPi2 (2.6.97)
Installing FPWCSPi2Swift (2.6.97)
Installing FPWebRTC (2.6.97)
Installing GPUIImage (0.1.7)
Installing SocketRocket (0.5.1)
Generating Pods project
Integrating client project
Pod installation complete! There are 4 dependencies from the Podfile and 5 total pods installed.
```

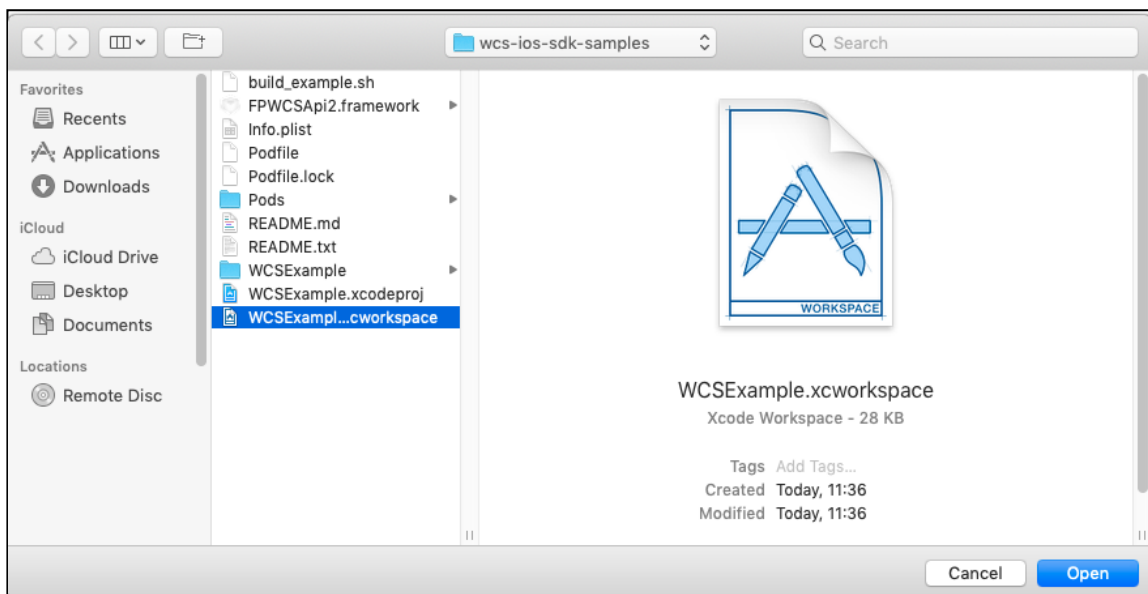
Сборка и запуск примеров в Xcode

1. Открываем workspace в Xcode

После того как все зависимости собрались, открываем workspace в Xcode.

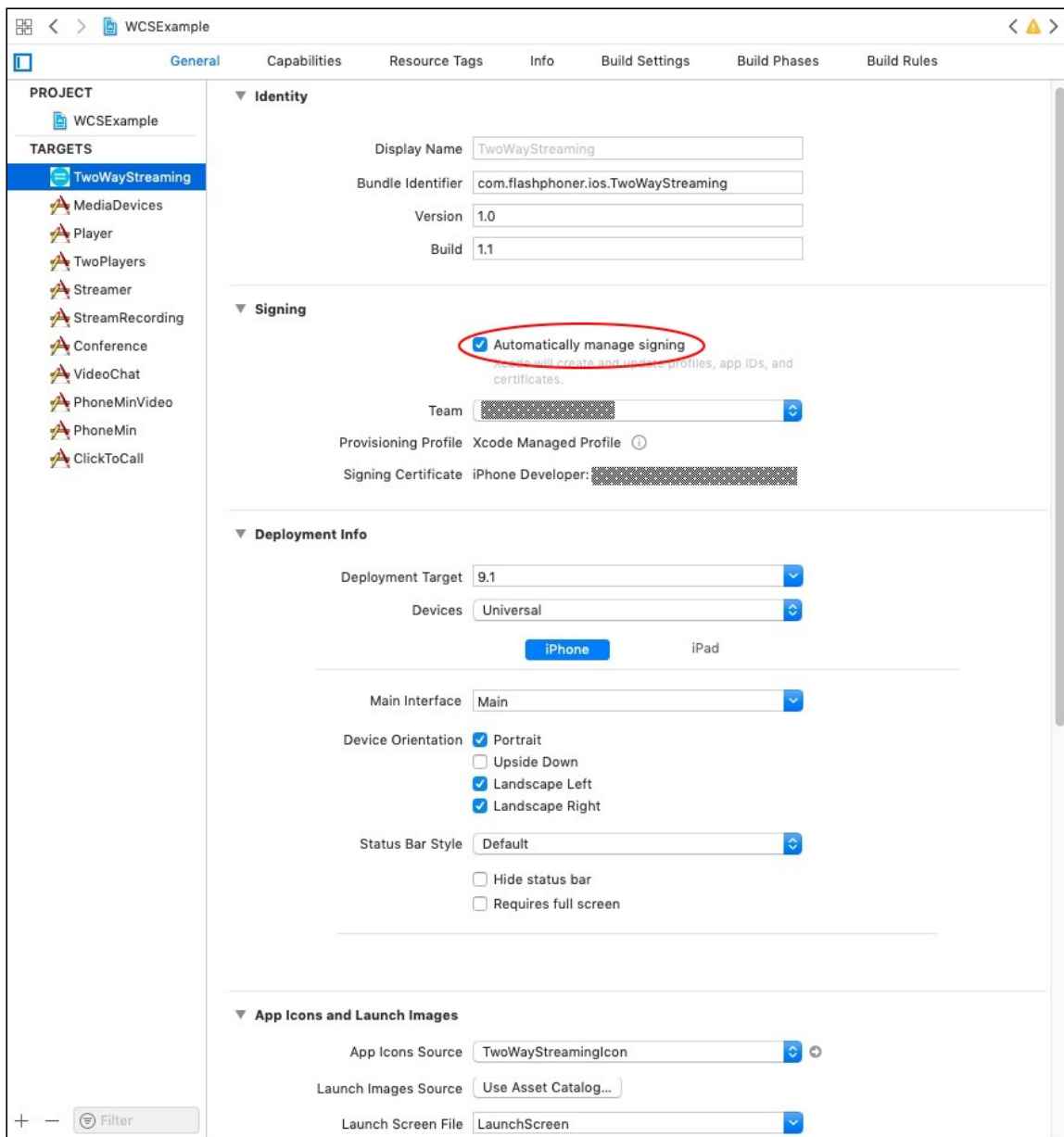
Warning

Открываем именно workspace, а не project-файл. Это нужно чтобы не испортить сборку.



2. Настраиваем управление подписью

На вкладке **General** для каждого примера установить галочку **Automatically manage signing** и указать сертификаты разработчика

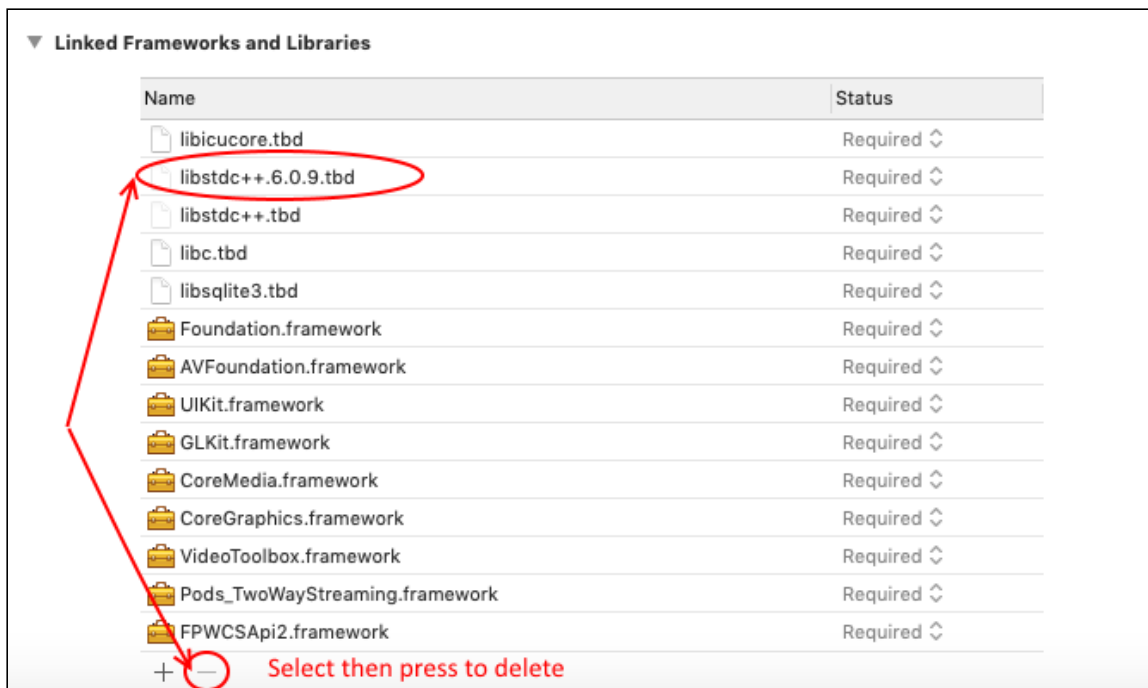


Warning

Начиная со сборки [2.6.10](#), шаги 3-5 не требуются!

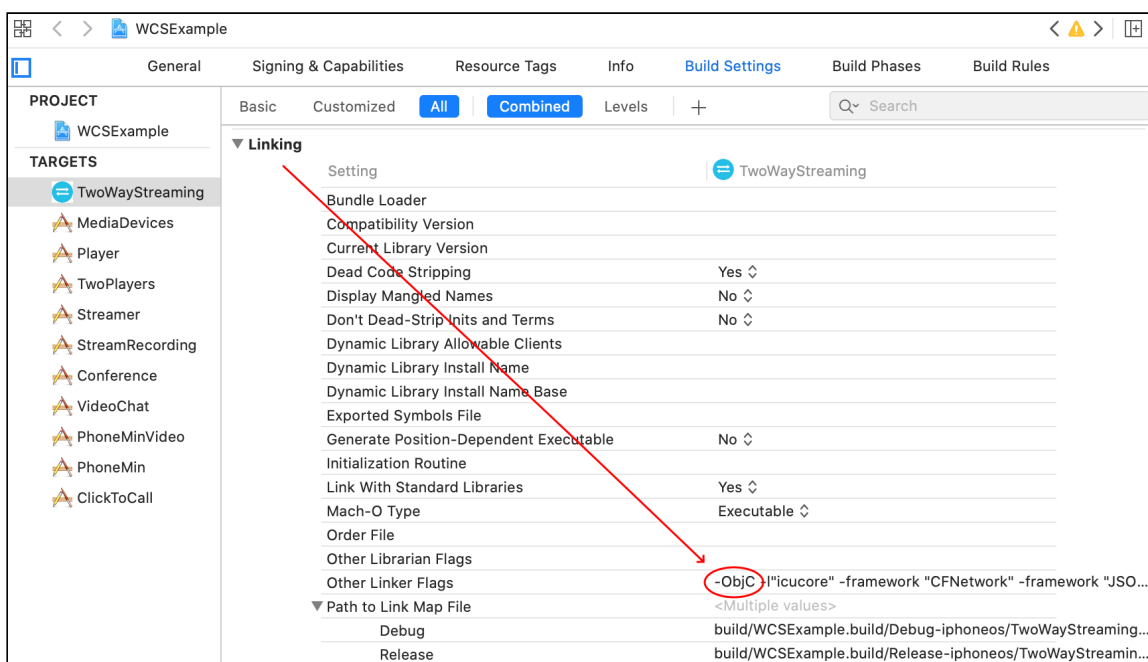
3. Удаление лишних библиотек из списка

Внизу вкладки **General** удаляем библиотеки `libstdc++.6.0.9.tbd` и `libstdc.tbd` (если есть). При необходимости, добавляем библиотеку `libstdc++.tbd`



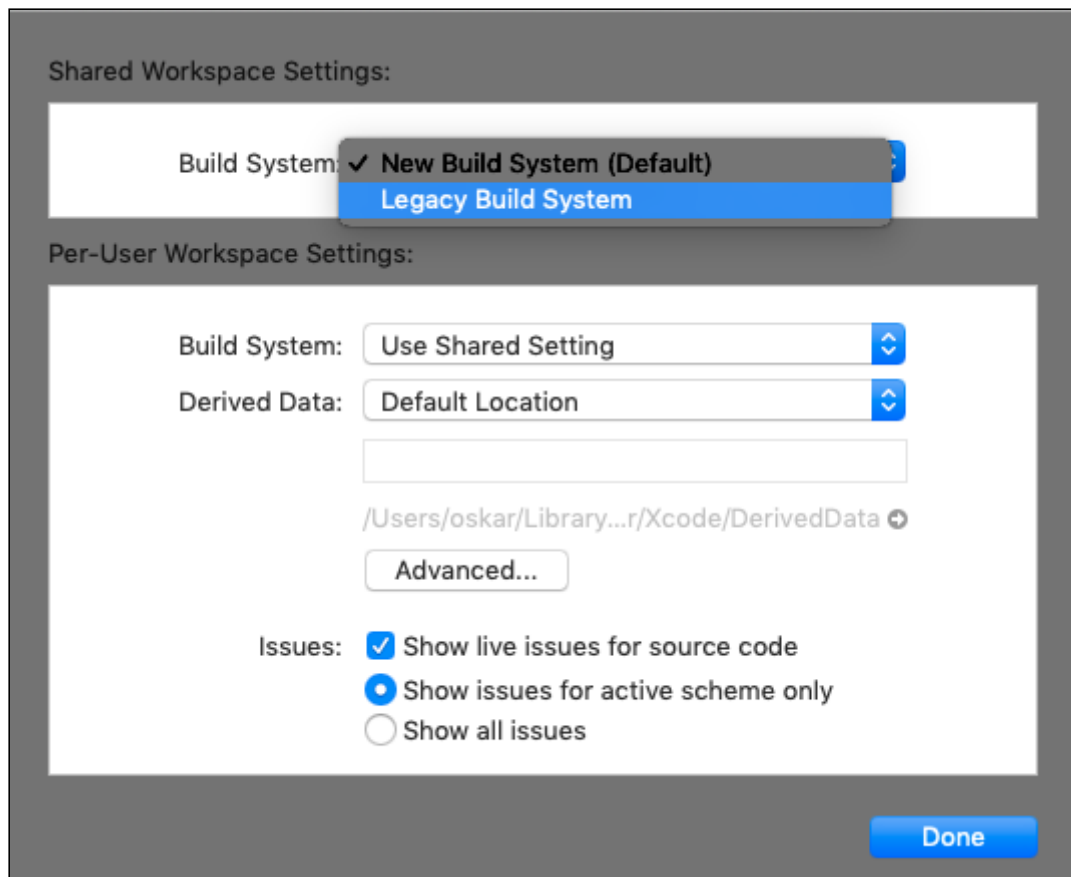
4. Настройка флагов компоновщика

На вкладке **Build settings** в разделе **Linking** добавляем флаг **-ObjC**



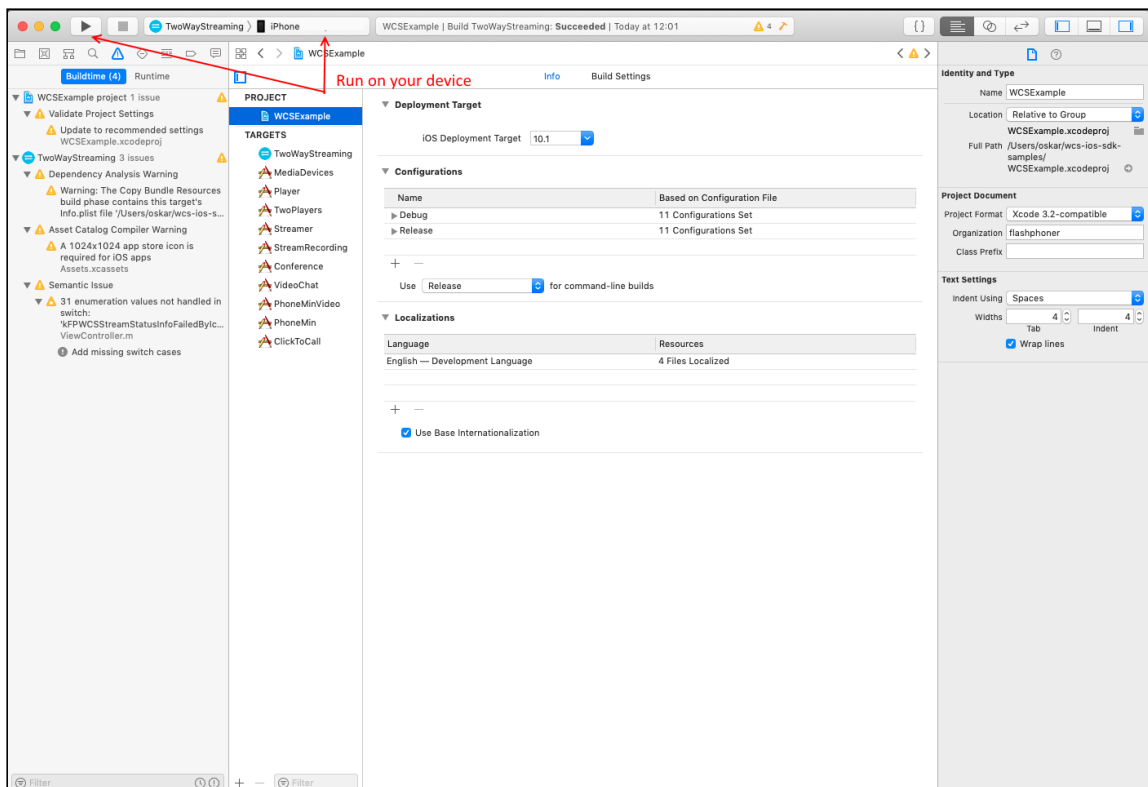
5. Настройка системы сборки

В меню выбираем пункт **File - Workspace settings** и устанавливаем значение **Legacy Build System** для параметра **Build System**



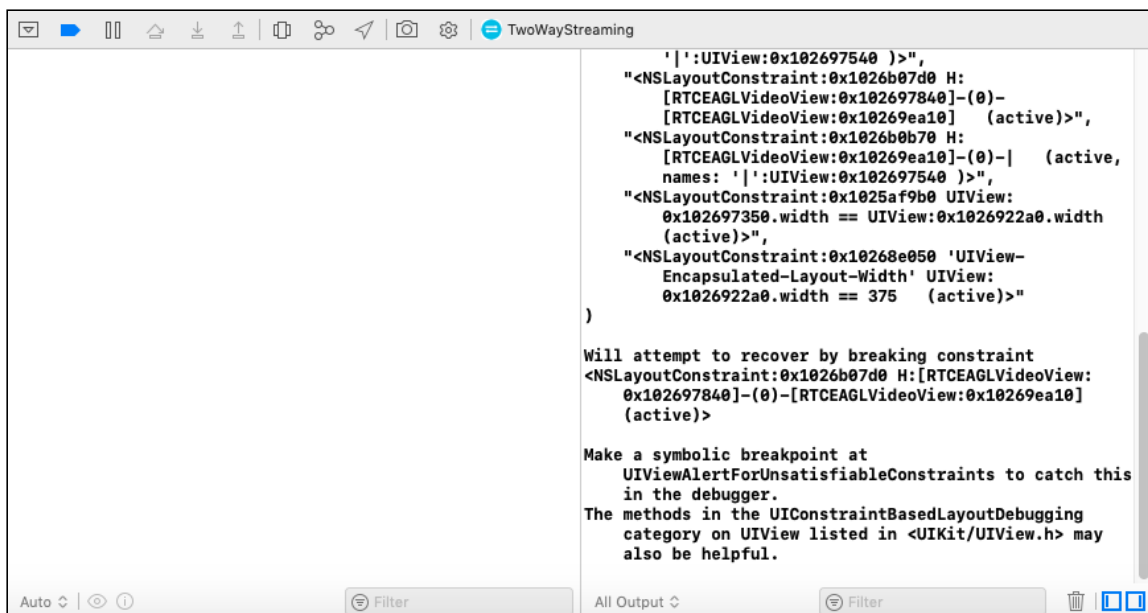
6. Выбор целевого устройства

Для примера `Two Way Streaming` выбираем в целях сборки `Generic iOS Device` и запускаем сборку через пункт меню `Product - Build`. Затем подключаем iPhone или iPad через USB и выбираем его в целях для запуска примера.



7. Просмотр логов приложения

После успешного запуска в нижней части экрана отобразится отладочная информация. Это означает, что пример был корректно установлен на iPhone или iPad и запущен.



8. Приложение запущено

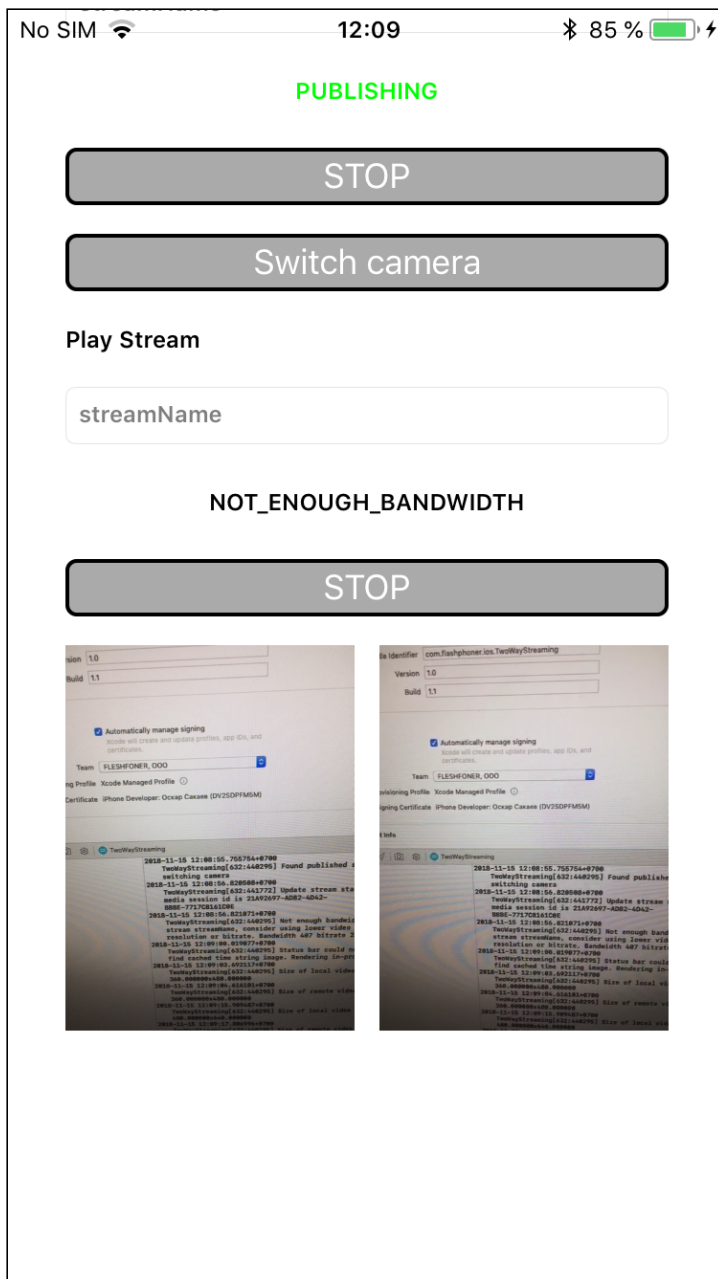
На iPhone появится интерфейс приложения, который можно начинать тестировать с WCS-сервером

The screenshot shows an iPhone screen with the following elements:

- Status Bar:** Top of the screen showing "No SIM", signal strength, time "12:07", Bluetooth icon, "85 %", battery icon, and a lightning bolt icon.
- URL Field:** A text input field containing the URL "wss://wcs5-eu.flashphoner.com:8443/".
- Status:** The text "NO STATUS" is displayed below the URL field.
- CONNECT Button:** A large, grey, rounded rectangular button with the text "CONNECT" in white.
- Publish Stream Section:**
 - Label:** "Publish Stream" in bold.
 - streamName Field:** A text input field containing the placeholder text "streamName".
 - Status:** The text "NO STATUS" is displayed below the streamName field.
 - PUBLISH Button:** A large, grey, rounded rectangular button with the text "PUBLISH" in white.
 - Switch camera Button:** A large, grey, rounded rectangular button with the text "Switch camera" in white.
- Play Stream Section:**
 - Label:** "Play Stream" in bold.
 - streamName Field:** A text input field containing the placeholder text "streamName".
 - Status:** The text "NO STATUS" is displayed below the streamName field.
 - PLAY Button:** A large, grey, rounded rectangular button with the text "PLAY" in white.
- Video Viewers:** Two black rectangular boxes at the bottom of the screen, intended for video playback.

9. Тестирование стриминга

Соединяемся с сервером и отправляем видеопоток с веб-камеры на iPhone.



Таким образом, мы в Xcode 10.1 на Mac OS Mojave собрали пример **Two Way Streaming** из исходного кода, с использованием iOS SDK (FPWCSPi2.framework) и запустили этот пример на iPhone6. Пример продемонстрировал успешный стриминг видео через Web Call Server 5.