

# Описание

## Warning

Android SDK 1.1 предназначен для использования на устройствах с Android 8 и более новыми версиями

## Ресурсы

Для разработки мобильных приложений потокового видео и звонков используется Android SDK.

Скачать полную сборку WCS Android SDK с примерами и API документацией: [Release notes](#)

Смотреть API документацию онлайн: [API docs](#)

Скачать исходный код примеров: [GitHub](#)

Исходный код примеров расположен на [Github](#) и будет использоваться в дальнейшем для пояснения работы примеров в данной документации. Например, данная ссылка [line 34](#) указывает на тридцать четвертую строку класса `TwoPlayersActivity.java` примера 2players ревизии с хешем 4ed4c6d77.

Для тестирования готовых приложений, скачайте полную сборку с примерами и установите `...-debug.apk` файл на ваше Android устройство.

## Различия между версиями Android SDK

В версии Android SDK 1.1 обновлена WebRTC библиотека `libjingle_peerconnection.jar`. В связи с этим, минимальная версия Android API, поддерживаемая Android SDK 1.1, увеличена до 26, т.е. приложения на базе Android SDK 1.1 будут работать только в Android 8 и выше. Для поддержки предшествующих версий Android используйте [Android SDK 1.0](#).

При публикации в Google Play для совместимости приложения с устройствами с API ниже и выше 26 можно выгрузить [два APK](#) под каждую версию Android SDK.

## Подготовка примеров для сборки

Если у вас есть опыт разработки под Android, вы можете просто скачать aar-библиотеку и вручную присоединить ее к проекту и сконфигурировать для сборки.

Ниже показано как это сделать автоматически с помощью скрипта `export.sh`:

## 1. Скачайте примеры

```
git clone -b 1.1 git@github.com:flashphoner/wcs-android-sdk-samples.git
```

## 2. Скачайте aar библиотеку и положите ее в папку `export`

Скачайте aar библиотеку и положите ее в папку `export`.

Пример (замените `x` на актуальный номер сборки):

```
wget http://flashphoner.com/downloads/builds/flashphoner_client/wcs-android-sdk/1.0/wcs-android-sdk-1.1.0.x.aar
cp wcs-android-sdk-1.1.0.x.aar export
```

## 3. Запустите скрипт `export.sh`

Перейдите в папку `export` и запустите скрипт `export.sh`, который подготовит конфиги для дальнейшей сборки. Результат будет находиться в папке `output`

```
cd export
./export.sh wcs-android-sdk-1.1.0.x.aar
```

### Warning

Данный шаг очень важен, поскольку исходные тексты примеров одинаковы для Android SDK 1.0 и 1.1. Скрипт `export.sh` автоматически, в зависимости от версии aar библиотеки, устанавливает для сборки примеров минимальную версию Android API.

## 4. Скорректируйте файл `local.properties`

Скорректируйте файл `local.properties` и пропишите пути к Android SDK и NDK

Пример для Linux окружения:

```
ndk.dir=/opt/android-ndk-r12b
sdk.dir=/opt/android-sdk-linux
```

## Сборка примеров с помощью Gradle

Подготовьте примеры для сборки, перейдите в папку `output` и запустите сборку:

```
cd output
gradle build
```

Начиная со сборки Android SDK 1.1.0.55, необходимо использовать Gradle 7 и выше, либо запускать сборку с помощью Gradle wrapper

```
cd output
gradlew build
```

Также для сборки необходимо наличие JDK 11.

## Сборка примеров в Android Studio

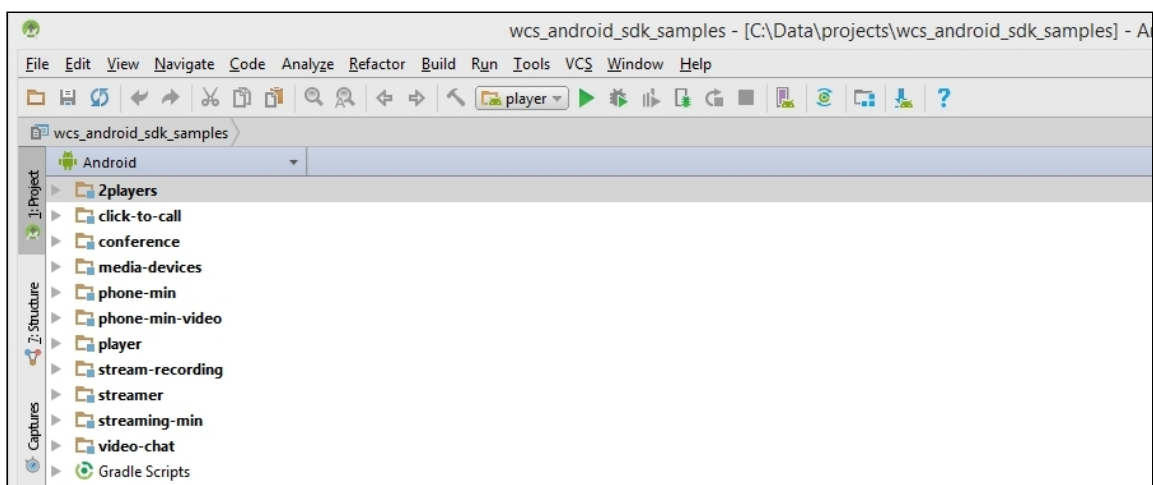
Подготовьте примеры для сборки в папку `output` и переименуйте ее, например, в `wcs_android_sdk_samples`

### 1. Установите необходимые программы

- [Android Studio и Android SDK](#)
- [Android NDK](#)

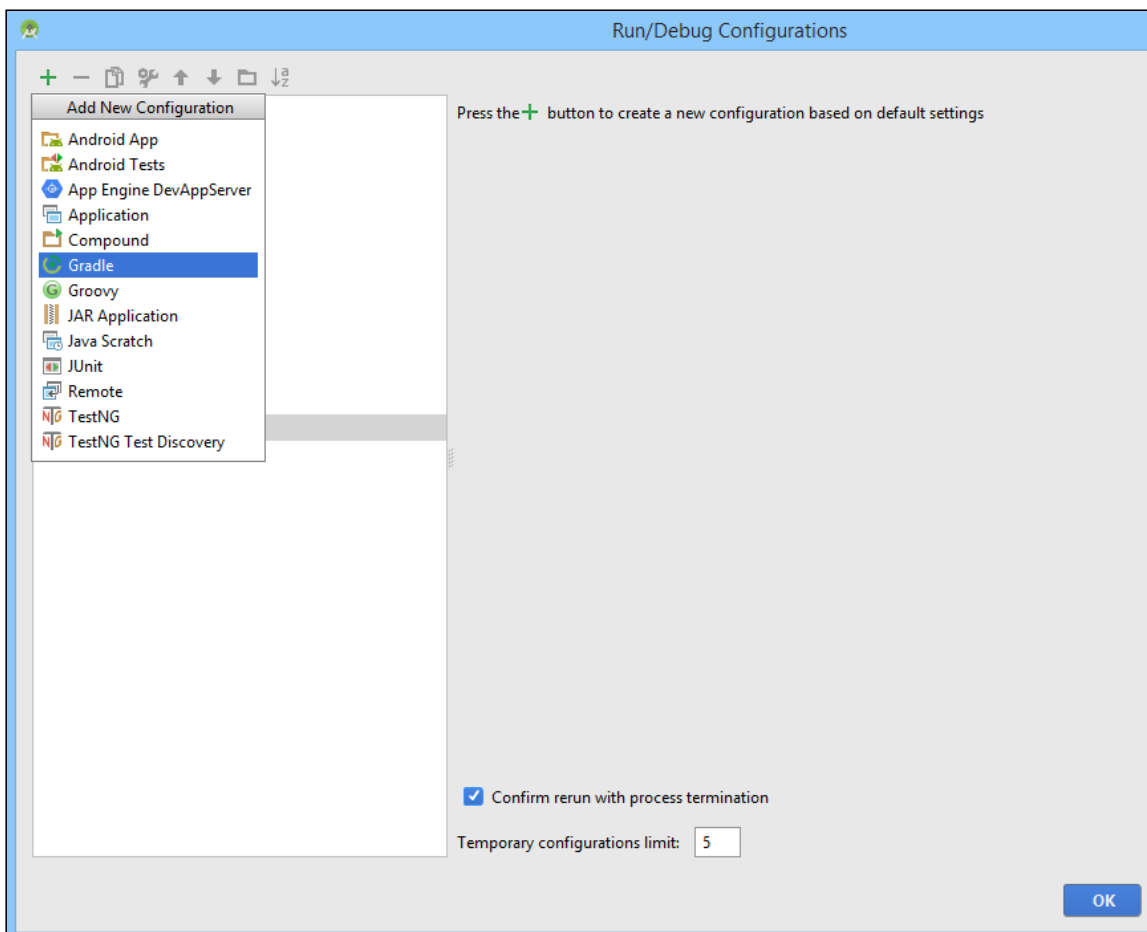
### 2. Откройте проект в Android Studio

Откройте проект из папки `output` в Android Studio



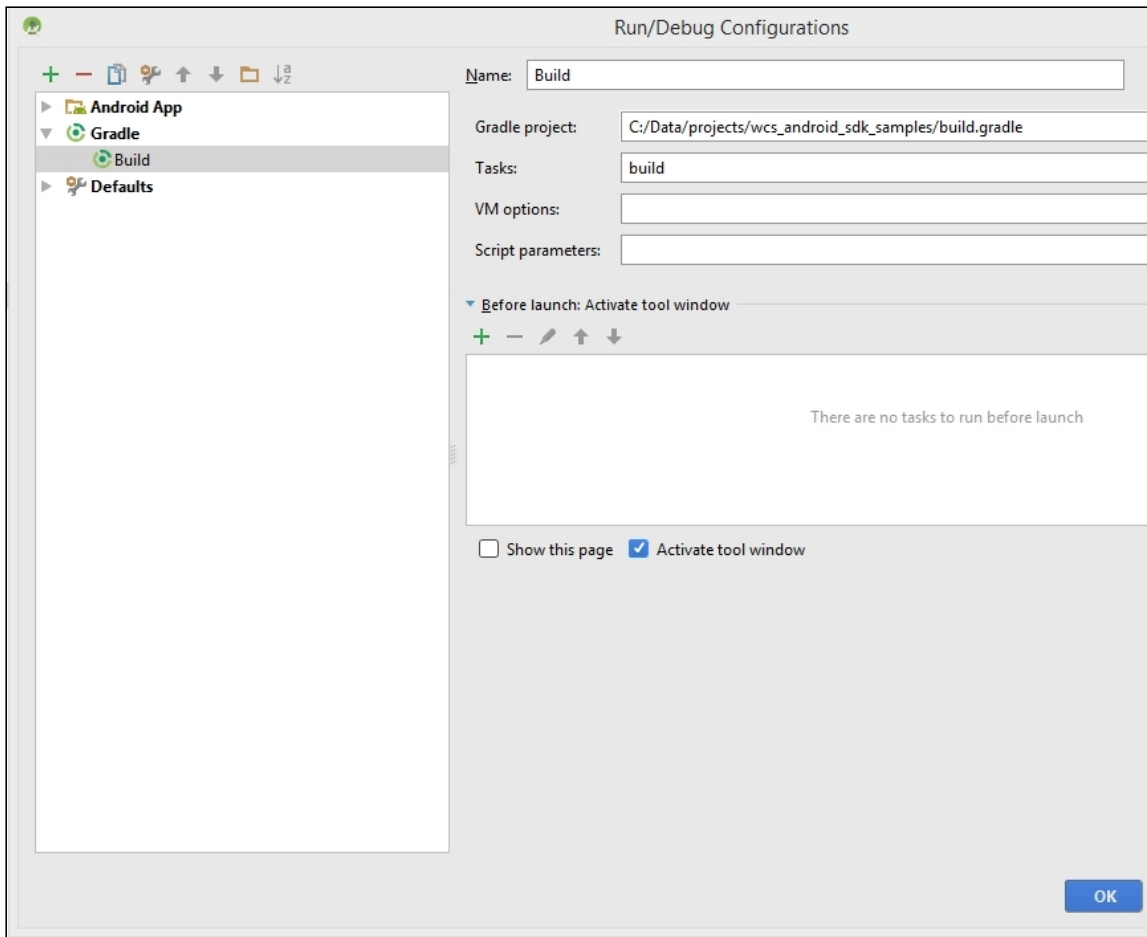
### 3. Добавьте конфигурацию запуска Gradle

Добавьте конфигурацию запуска Gradle в меню **Run / Debug Configurations**

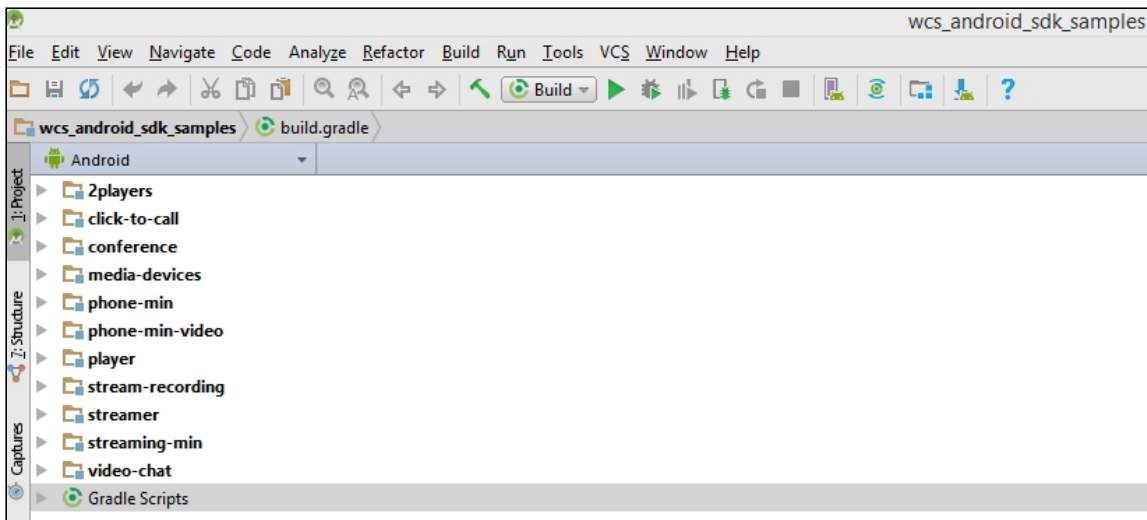


#### 4. Настройте конфигурацию запуска

Для конфигурации запуска укажите файл сборки **build.gradle**, название сборки и назначение **build**



## 5. Запустите сборку примеров



Результатом сборки будут `.apk` файлы в соответствующих директориях `2players/build`, `click-to-call/build`, и т.д. Файл Android SDK находится в проекте по следующему пути: `libs/wcs-android-sdk-1.1.0.x.aar`

## Сборка отдельного примера

Если необходимо собрать отдельный пример, или нет возможности выполнить скрипт экспорта всех примеров, пример для сборки может быть подготовлен вручную следующим образом:

### 1. Скачивание примеры

```
git clone -b 1.1 https://github.com/flashphoner/wcs-android-sdk-samples.git
```

### 2. Копирование нужного примера в отдельный каталог

```
cd wcs-android-sdk-samples
cp streaming-min gradle.properties ~/streaming-min
```

### 3. Скачивание aar библиотеки и размещение в папке `libs` в примере

Download aar library and put it to `libs` subfolder in the example folder

```
wget https://flashphoner.com/downloads/builds/flashphoner_client/wcs-android-
sdk/1.1/wcs-android-sdk-1.1.0.x.aar
mkdir ~/streaming-min/libs
cp wcs-android-sdk-1.1.0.x.aar ~/streaming-min/libs
```

### 4. Добавление секции `buildscript` в файл `build.gradle`

Добавьте в начало файла `build.gradle` в каталоге примера секцию `buildscript`:

```
buildscript {
    repositories {
        jcenter()
        mavenCentral()
        maven { url 'https://maven.google.com' }
        google()
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:7.0.0'
        classpath 'com.github.triplet.gradle:play-publisher:1.1.5'
    }
}
```

### 5. Добавление секции `repositories` в файл `build.gradle`

Добавьте в файл `build.gradle` в каталоге примера секцию `repositories`:

```
repositories {
    jcenter()
    mavenCentral()
    maven { url 'https://maven.google.com' }
    google()
    flatDir{
        dirs 'libs'
    }
}
```

## 6. Замена строки в секции `dependencies` файла `build.gradle`

Замените в секции `dependencies` файла `build.gradle` в каталоге примера строку

```
implementation project(':fp_wcs_api')
```

на

```
implementation 'com.flashphoner.fpwcsapi:wcs-android-sdk-1.1.0.x@aar'
```

где `wcs-android-sdk-1.1.0.x` имя aar файла, скачанного на шаге 3.

## 7. Пример файла `build.gradle`

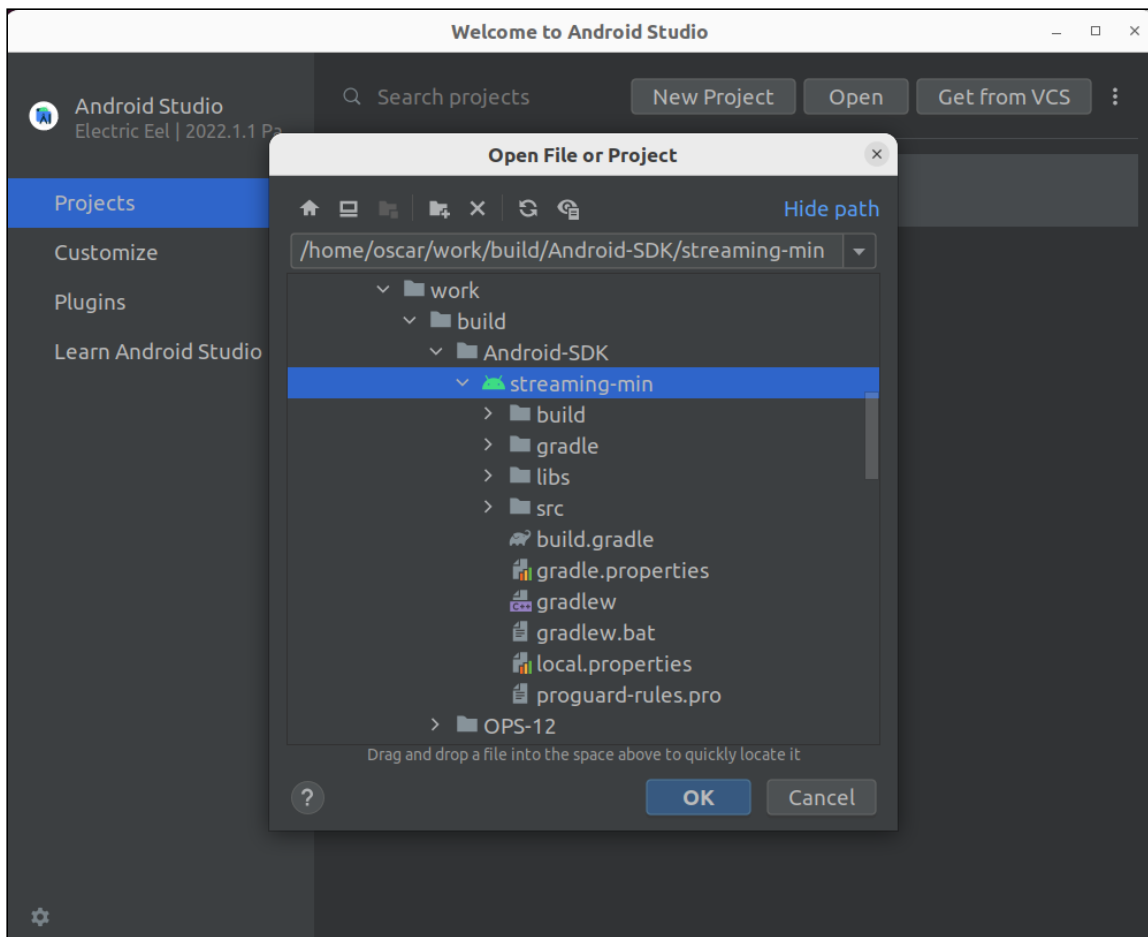
build.gradle Expand source

 **build.gradle**



## 8. Сборка примера в Android Studio

Для сборки в Android Studio откройте каталог примера. Android Studio выполнит Gradle sync и установит необходимую версию gradle



## 9. Сборка примера с помощью Gradle

Для сборки с помощью Gradle установите Gradle wrapper и выполните команду

```
gradlew build
```

## Известные ограничения

1. Невозможно управлять уровнем громкости микрофона при публикации потока в Android SDK.