Overview

- Resources
- Differences between Android SDK versions
- Preparing examples for building
- Building examples with Gradle
- Building examples in Android Studio
- Known issues

Resources

Use Android SDK to develop mobile applications for streaming video and calls.

Download full WCS Android SDK build including examples and API documentation
https://docs.flashphoner.com/display/ANDROIDSDK1EN/Android+SDK+release+notes

Read API documentation online
http://flashphoner.com/docs/api/WCS5/client/android-sdk/latest

Download the source code of the examples
https://github.com/flashphoner/wcs-android-sdk-samples

The source code of the examples is located at Github and is used to comment the examples in the present documentation.
For instance, this link line 34 refers to the thirty fourth line in the TwoPlayersActivity.java class of the 2players example, the revision with the hash of 4ed4c6d77.

To test compiled applications, download The full build with the examples And install the .apk file to your Android device.

Differences between Android SDK versions

In Android SDK 1.1 WebRTC librarylibjingle_peerconnection.jar is updated to actual version. So Android SDK 1.1 requires Android API 26, i.e. application build with Android SDK 1.1 will run on Android 8 and higher. UseAndroid SDK 1.0to support previous Android versions.

When publishing in Google Play, two APKs can be deployed - one for each of the Android SDK versions - for compatibility with devices with API lower and higher than 26.

Preparing examples for building

If you have some experience in developing Android apps, you can simply download the aar-library and link it to the project manually, then configure building.

Below is how to do this automatically using the export.sh script:

1. Download the examples

```
git clone -b 1.0 git@github.com:flashphoner/wcs-android-sdk-samples.git
```

2. Download the aar library and put it to the 'export' folder

Example:

```
wget http://flashphoner.com/downloads/builds/flashphoner_client/wcs-android-sdk/1.0/wcs-android-sdk-1.0.1.x.aar
cp wcs-android-sdk-1.0.1.x.aar export
```

3. Open the 'export' folder and execute the export.sh script that will prepare configs for further building. The result is placed into the 'output' folder.

```
cd export
./export.sh wcs-android-sdk-1.0.1.x.aar
```

> ⊘ This step is very important because application examples source code is the same for different versions of Android SDK. export.sh script automatically sets miniman required Android API version for building examples depending on Android SDK version

4. Edit the local.properties file and specify paths to Android SDK and NDK

Example:

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

Building examples with Gradle

Prepare examples for building, then go to the 'output' folder and start building:
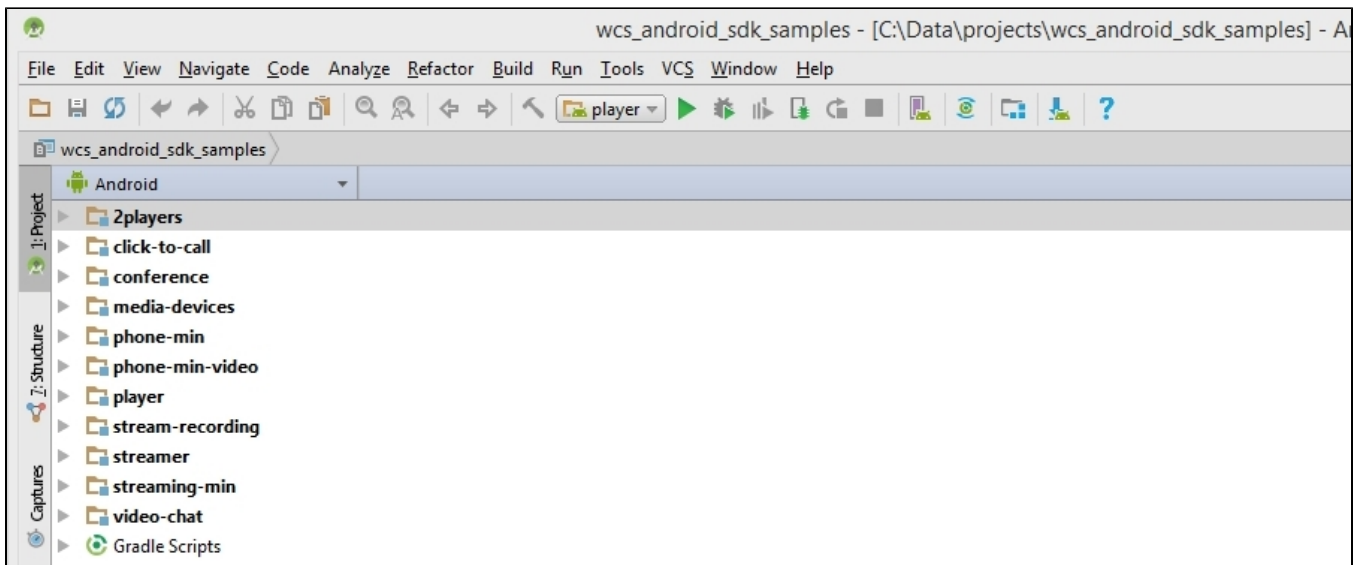
```
cd output
gradle build
```

Building examples in Android Studio

Prepare examples for building and put them to the 'output' folder, then rename it to 'wcs_android_sdk_samples', for instance.
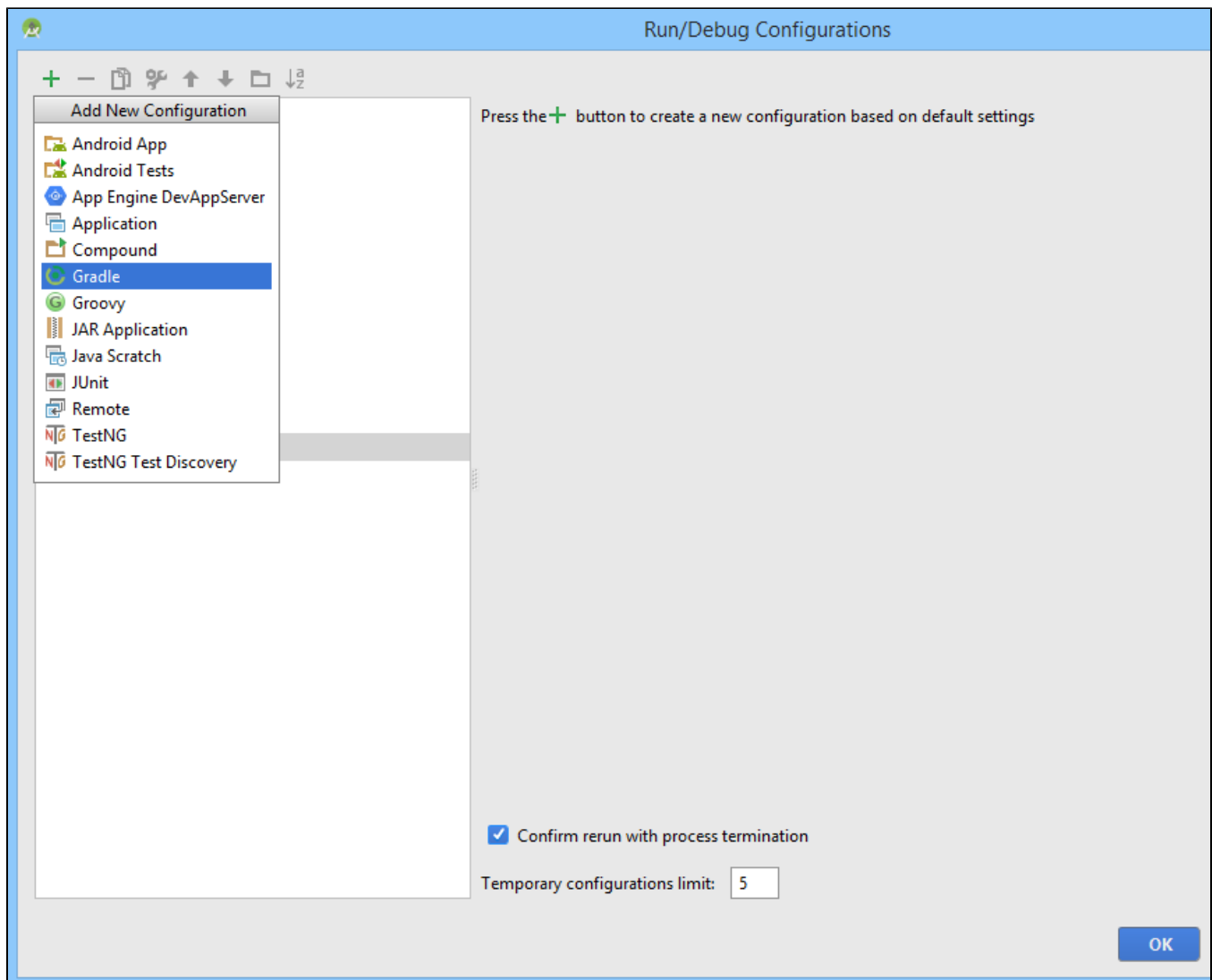
1. Install the following programs:

- Android Studio и Android SDK
- Android NDK

2. Open the project from the 'output' folder in Android Studio

4. Add Gradle run configuration to the Run / Debug Configurations menu

Run/Debug Configurations

Add New Configuration

- Android App
- Android Tests
- App Engine DevAppServer
- Application
- Compound
- Gradle
- Groovy
- JAR Application
- Java Scratch
- JUnit
- Remote
- TestNG
- TestNG Test Discovery

Press the + button to create a new configuration based on default settings

☑ Confirm rerun with process termination

Temporary configurations limit: 5

OK

5. To configure the run, specify the build.gradle file, the name of the build and set the purpose to: build
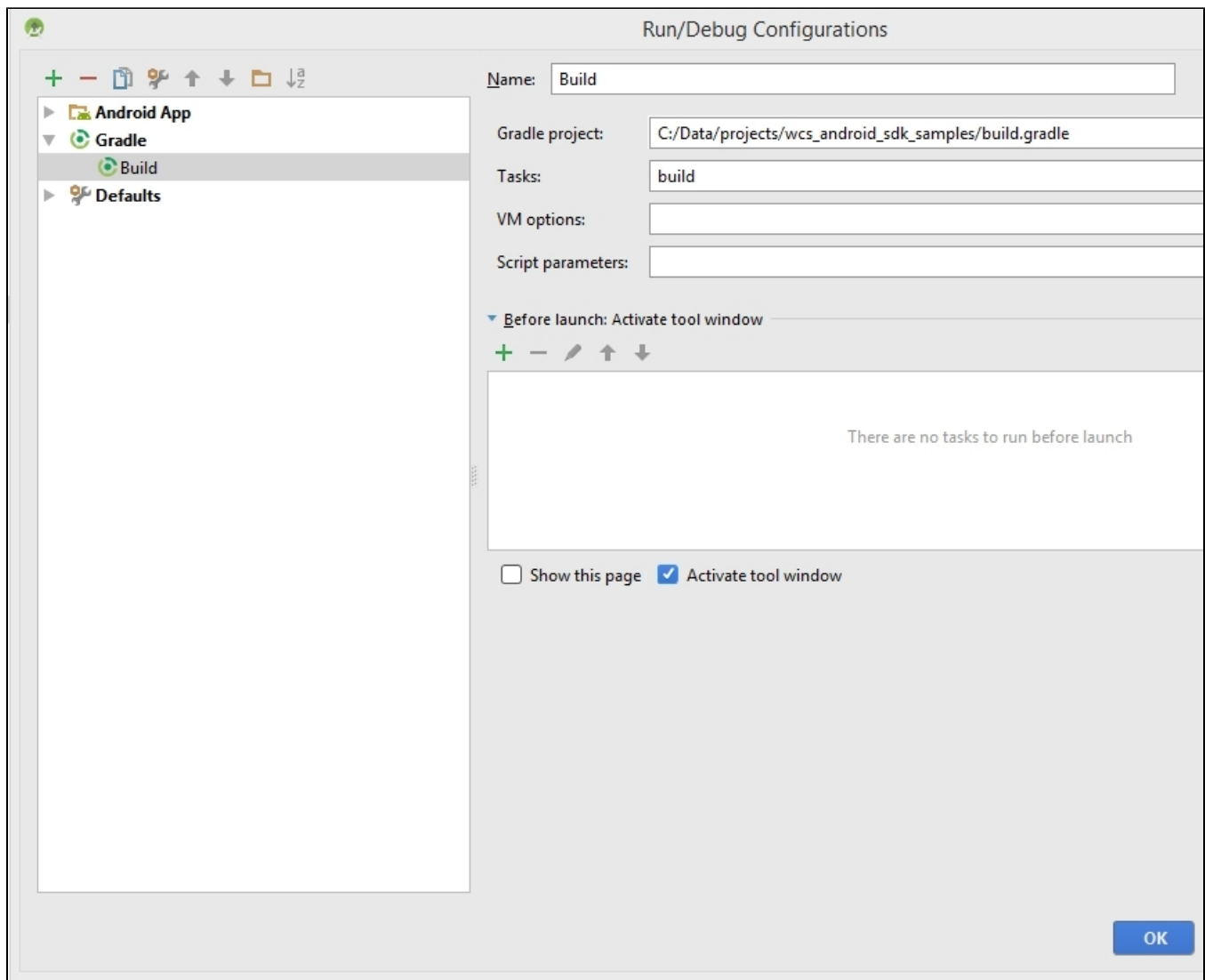
6. Run building of examples.
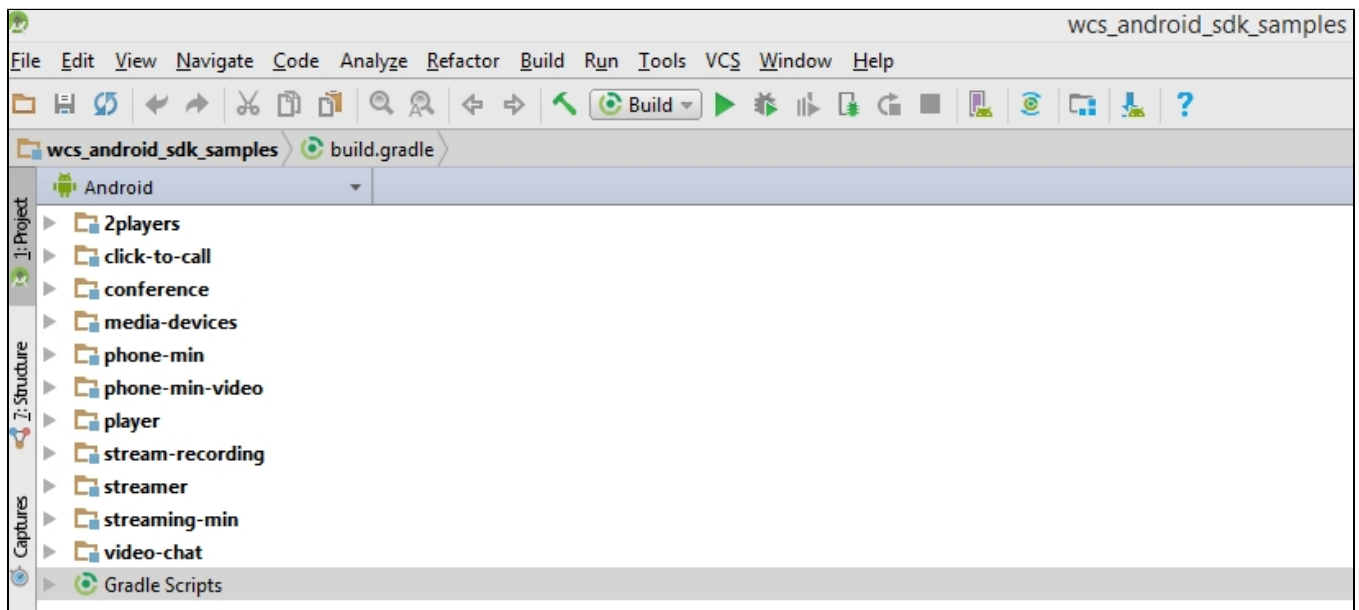
The building result is .apk files located in the corresponding folders: 2players/build, click-to-call/build, and so on.
The Android SDK file is located in the following path in the project: libs/wcs-android-sdk-1.0.1.2.aar

Known issues

1. It is impossible now to set microphone gain in Android SDK while publishing stream.