iOS SDK Distribution guide

Creating XCFramework

This document describes modern way of distributing frameworks in form of `xcframework`, i.e. frameworks containing several architectures within their subfolders. To create XCFramework follow the steps bellow:

1. Clone project folders from repository:

```
git clone <path-to-repo>
```

2. Go to project folder and enter to SDK subfolder:

```
cd AcousticContentSDK
```

3. Run script:

```
./make_xcframework.sh
```

4. Wait when creating process complete.
5. If process was successfull then the results will be placed in `outputs` sub-folder of project's folder.
6. Structure of `outputs` folder:
    - `outputs/platform_device.xcarchive` - archive containing framework built for devices and production.
    - `outputs/platform_simulator.xcarchive` - archive containing framework built for simulators.
    - `outputs/AcousticContentSDK.xcframework` - XCFramework containing both architectures.
7. Now `AcousticContentSDK.xcframework` can be drag-and-dropped to hosting application.

Additional `xcodebuild` related notes

1. You may need to update `make_xcframework.sh` script with installed iOS Simulator version. To find available `destination` options run:

```
xcodebuild -scheme AcousticContentSDK -showdestinations
```

The output could look like the following:

```
    Available destinations for the "AcousticContentSDK" scheme:
    { platform:macOS, arch:x86_64, variant:Mac Catalyst, id:6884F8A1-
1593-51E4-99C8-4EE4D9652047 }
    { platform:iOS Simulator, id:CFE4D65C-6F61-4147-98AE-072605891B7A,
OS:13.3, name:iPhone 11 }
    ...

Ineligible destinations for the "AcousticContentSDK" scheme:
    { platform:iOS, id:dvtdevice-DVTiPhonePlaceholder-
iphoneos:placeholder, name:Generic iOS Device }
```

2. To create `destination` parameter value please use the data above to create a string with follwoing format:

```
-destination 'platform=iOS Simulator,OS=13.3,name=iPhone 11'
```

> Please pay attention to `=` instead of `:`, commas `,` and no spaces after comma