Mobile Speech Recognizer

by Piotr Zmudzinski

ptr.zmudzinski@gmail.com

About

Do you want to ask user for his player name so he can actually tell it instead of making him type? You are in good place…

This plugin uses OS components for speech recognition and send it to your Unity scripts as String objects.

Plugin supports:

- **Android >= 3.0**
- **iOS >= 10.0**

You have to prepare fallback code to cover cases when user doesn't have access to speech recognition or when it fails. Keep in mind that both iOS and Android might use Internet connection for speech detection, which means it might fail in case there's no active connection.

Plugin doesn't work in Editor!
You have to run your app on real iOS or Android device.

Quick Start

Open example scene!

Go to *KKSpeechRecognizer/Example* folder inside Unity and open *ExampleScene*:

It shows basic usage of a plugin, which is:

1. Detecting if speech recognition exists on user's device (keep in mind that it won't be available on e.g. iOS 9 or old Android phones)
2. If it exists, and user clicks on "Start Recording" button it listens for recognized text and displays it on a screen.
3. On Android, speech recognition automatically detects when user finishes speaking, but on iOS we have to wait for user clicking "Stop Recording" to finish whole process (i.e. get final results).

Before running it on Android or iOS device you have to…

Setup permissions!

iOS

After generating Xcode project (keep in mind that you have to use **Xcode 8 or higher**) you have to add two permissions keys to your project:

- **NSMicrophoneUsageDescription**

explanation from Apple docs:

> This key lets you describe the reason your app accesses any of the the device's microphones. When the system prompts the user to allow access, this string is displayed as part of the alert.

- **NSSpeechRecognitionUsageDescription**

explanation from Apple docs:

> This key lets you describe the reason your app sends user data to Apple's speech recognition servers. When the system prompts the user to allow access, this string is displayed as part of the alert.

You can do it in two ways:

- **Automatic (default and recommended)**

Open *KKSetSpeechRecognitionPermissionsoniOS* script from Editor folder inside your project. Change values of those texts:

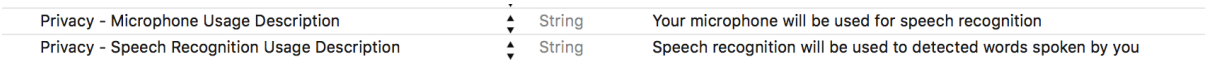
```
public static string microphoneUsageDescription = "Put something here about microphone usage";
public static string speechRecognitionUsageDescription = "Put something here about speech recognition usage";
```

Generate your Xcode project. Your are ready.

- **Manually**

Disable *KKSetSpeechRecognitionPermissionsoniOS* script by either removing it or setting shouldRun variable to false.
After generating your Xcode project open *Info.plist* file and right-click on it and click *"Add row"*. Then enter two required two keys with descriptions.

In the end it should look like it:



Android

You must add those two permissions to your AndroidManifest.xml:

```
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.INTERNET" />
```

If you don't know how to do that check that link: http://answers.unity3d.com/questions/525838/help-about-adding-permissions-on-android.html

Now you are ready to play with the app. Press Build&Run and…

Play with speech recognition app!

Click on "Start Recording" button and start talking. You should see live results in text above. Notice that on Android it automatically detects when you stop talking, but it rely on user when it comes to iOS.

API Overview