

- ➔ All you need to import to get up and running is the Assets/Plugins/Android/ttsmanager/ttsmanager.jar/ script and the TTSTManager script in the Assets/C# TTSTManager Script/ Folder
- ➔ All Implementation of Text-To-Speech (TTS) functionality can be accessed via the Class TTSTManager's static methods.
- ➔ This TTS Library is an android plugin, it does not work on IOS and Windows. You must build your app onto an android phone to test out the libraries.

API

Methods	Description	Example
BootUpTTS ()	This Method Must be Invoked at the start to start up the TTS Engine.	TTSTManager.BootUpTTS()
SetSpeechRate (float value)	Sets the rate at which the TTS Engine speaks the words. The TTS Engine speed is 1 by default. Setting a lower value than 1 will lower the speech rate and a higher value than 1 will increase the speech rate.	TTSTManager.SetSpeechRate(0.8f); //Sets the Speech Rate of the TTS Engine to 0.8f, slower than the default rate of 1.
SetPitch (float value)	Sets the pitch of the TTS Engine. The TTS Engine pitch is 1 by default. Setting a lower value than 1 will lower the pitch and a higher value than 1 will increase the pitch.	TTSTManager.SetPitch(0.8f);
GetLanguageAvailability (string locale)	Checks to see if the users phone has a specific language's TTS Support. Returns true if the specified language is supported, else returns false.	TTSTManager.GetLanguageAvailability("ja"); //Check to see if the user's phone has support for the Japanese Language. TTSTManager.GetLanguageAvailability("th"); //Check to see if the user's phone has support for the Thai Language.
Speak (string sentence)	Generic method for accessing the TTS Engine to play words. Take note that the default language used by the TTS is Japanese. Use the SetLocale() Method to Change the language used.	TTSTManager.SetLocale("en"); TTSTManager.Speak("Hello World!"); // The first method changes the TTSTManager to implement the English language // Second method calls the Speak method to say the words "Hello World!".

DownloadTTSDData ()	Creates an Intent which allows users to download TTS Languages. This Method does not facilitate the download of specific languages. The user will have to select the desired TTS Language to download when the download screen appears.	TTSManager.DownloadTTSDData();
StopTTS ()	Stops the Running of the TTS and free up resources.	TTSManager.StopTTS();
GetAvailableLocales ()	Returns a list of all the available locales/languages as a string array	TTSManager.GetAvailableLocales()
SetLocale (string locale, string script = "", string region = "")	Changes the Implementing Language of the TTS to the desired language. For Normal use, passing the locale argument to the method is sufficient. Returns true if the locale was successfully implemented, else returns false.	TTSManager.SetLocale("en"); //English TTSManager.SetLocale("th"); //Thai TTSManager.SetLocale("de"); //German TTSManager.SetLocale("ja"); //Japanese
GetLocale ()	Returns the Locale currently being implemented by the TTS as a string.	TTSManager.GetLocale();
IsBootedUp ()	This method checks if the TTSManager has been started up. Returns true if it has been started up, else returns false. Use BootUpTTS () to start up the TTSManager.	TTSManager.IsBootedUp();
IsUttering ()	Checks to see if the TTS is currently speaking a word/sentence. Returns true if the TTS is currently speaking, else returns false.	TTSManager.IsUttering();
SetEngineByPackageName ()	Sets the TTS Engine to be used by the TTS. The default TTS Engine used is com.google.android.tts	TTSManager.SetEngineByPackageName("com.google.android.tts"); //Sets the TTS to use the com.google.android.tts library
StopSpeaking()	Stops the TTS if it is currently speaking a sentence. Can be used to stop a sentence that has been spoken midway.	TTSManager.StopSpeaking();

<https://stackoverflow.com/questions/7973023/what-is-the-list-of-supported-languages-locales-on-android>

Arabic, Egypt (ar_EG)
Arabic, Israel (ar_IL)
Bulgarian, Bulgaria (bg_BG)
Catalan, Spain (ca_ES)
Chinese, PRC (zh_CN)
Chinese, Taiwan (zh_TW)
Croatian, Croatia (hr_HR)
Czech, Czech Republic (cs_CZ)
Danish, Denmark (da_DK)
Dutch, Belgium (nl_BE)
Dutch, Netherlands (nl_NL)
English, Australia (en_AU)
English, Britain (en_GB)
English, Canada (en_CA)
English, India (en_IN)
English, Ireland (en_IE)
English, New Zealand (en_NZ)
English, Singapore (en_SG)
English, South Africa (en_ZA)
English, US (en_US)
Finnish, Finland (fi_FI)
French, Belgium (fr_BE)
French, Canada (fr_CA)
French, France (fr_FR)
French, Switzerland (fr_CH)
German, Austria (de_AT)
German, Germany (de_DE)
German, Liechtenstein (de_LI)
German, Switzerland (de_CH)
Greek, Greece (el_GR)
Hebrew, Israel (he_IL)
Hindi, India (hi_IN)
Hungarian, Hungary (hu_HU)
Indonesian, Indonesia (id_ID)
Italian, Italy (it_IT)
Italian, Switzerland (it_CH)
Japanese (ja_JP)
Korean (ko_KR)
Latvian, Latvia (lv_LV)
Lithuanian, Lithuania (lt_LT)
Norwegian, Norway (nb_NO)
Polish (pl_PL)
Portuguese, Brazil (pt_BR)
Portuguese, Portugal (pt_PT)
Romanian, Romania (ro_RO)
Russian (ru_RU)
Serbian (sr_RS)
Slovak, Slovakia (sk_SK)
Slovenian, Slovenia (sl_SI)
Spanish (es_ES)
Spanish, US (es_US)

Swedish, Sweden (sv_SE)
Tagalog, Philippines (tl_PH)
Thai, Thailand (th_TH)
Turkish, Turkey (tr_TR)
Ukrainian, Ukraine (uk_UA)
Vietnamese, Vietnam (vi_VN)

Optional Links

<http://hyperionics.com/TtsSetup/eng/TtsInfo.html>

<https://stackoverflow.com/questions/12449801/ivona-tts-and-how-to-get-programmatically-list-of-all-installed-voices>

Suggestions

In the example scripts, the Speak method was split up into two parts,

Using System.Threading.Tasks;

Static Task SpeechTask;

```
public void Speak() {  
    if (TTSManager.IsUttering())  
    {  
        TTSManager.StopSpeaking();  
    }  
    SpeakText();  
}  
  
async Task SpeakText() {  
    if (!TTSManager.IsBootedUp())  
    {  
        TTSManager.BootUpTTS();  
        return;  
    }  
    TTSManager.Speak(BoundaryClass.ReturnInputFieldComponent().text);  
}
```

Like so. Doing so allowed for the termination of a sentence that has been spoken midway.

For the option of changing languages, A async method was also employed to facilitate mid-sentence termination.

```

static readonly string[,] languagelocaledict = new string[4,2] { { "japanese", "ja" },
                                                                    {"english", "en"},
                                                                    {"thai", "th"},
                                                                    {"german", "de"}
};

public async void ChangeTTSLanguage(string language) {

    if (!TTSManager.IsBootedUp()) {
        TTSManager.BootUpTTS();
    }
    else if (TTSManager.IsUttering())
    {
        TTSManager.StopSpeaking();
    }

    string locale = null;

    for (var i=0;i<languagelocaledict.Length;i++) {
        if (languagelocaledict[i,0] == language.ToLower()) {
            locale = languagelocaledict[i,1];
            break;
        }
    }

    if (locale==null) {
        return;
    }

    TTSManager.SetLocale(locale); // Changes the language used by the TTSManager

    // The Code below are used in the Example Script to update the UI View //
    SetLocaleOnView(locale);
    SetLocalAvailabilityOnView(locale);
    BoundaryClass.PopulateScrollView(locale);

}

```