# CHEATSHEET

## USING AT&T API IN YOUR UNITY3D GAME

## Steps to use AT&T Speech API in your game:

With just 5 lines of code, you can start using AT&T Speech API in your Unity3D game.

1. Copy ATT_MSSDK.dll into the source code. This library is available in the package at: http://developer.att.com/sdks

2. Create an instance of RequestFactory class, by passing API Key, Secret, Endpoint and Scope information.

```
        List<RequestFactory.ScopeTypes> scopes = new
        List<RequestFactory.ScopeTypes>();
        scopes.Add(RequestFactory.ScopeTypes.Speech);
        RequestFactory requestFactory = new RequestFactory(endpoint, apiKey,
secretKey, scopes, null, null);
```

API Key – API Key of the registered application in AT&T Developer Portal. See here, to get an API Key.

Secret Key – Secret Key of the registered application. Endpoint – The API gateway end point.

Scopes – List of scopes registered by the application.

3. Capture the audio using Unity 3D functions.

```
        audio.clip = Microphone.Start(null, false, 5,
        8000); yield return new WaitForSeconds(5);
        Microphone.End(null);
```

4. Pass the captured audio to create a wav file. (NOTE: Speech API accepts other formats as well. Please refer Speech API documentation on AT&T Developer Portal.

```
        float[] clipData = new float[audio.samples *
        audio.channels]; audio.GetData(clipData, 0);
        WaveGen.WaveFormatChunk format = new
        WaveGen().MakeFormat(audio); string filename =
        "recordedSpeech.wav";
        FileStream              stream              =
        File.OpenWrite(filename);                 new
        WaveGen().Write(clipData,  format,  stream);
        stream.Close();
```

WaveGen is a custom class, to create WAV file header.

5. Call the SpeechToText method of requestFactory instance.

```
        SpeechResponse response = requestFactory.SpeechToText(filename);
```

6. Capture the response from the API gateway

```
        string transcribedText = response.Recognition.NBest[0].ResultText;
```

7. Use the transcribedText to perform your own game activity.