

**Android**  
**Text to Speech**  
**Voice Recognition**

---

# Text to Speech

- Synthesizes speech from text
- Immediate playback from mobile device
- Or create a sound file
- Important modality of interaction/communication
- Many applications: Enrich reading, support people with special needs, etc.



# TextToSpeech Overview

- A TextToSpeech instance can only be used to synthesize text once it has **completed** its initialization
- Implement the [TextToSpeech.OnInitListener](#) to be notified of the completion of the initialization.
- When you are done using the TextToSpeech instance, call the [shutdown\(\)](#) method to release the native resources used by the TextToSpeech engine.

# Nested Classes

## Nested Classes

class	<a href="#">TextToSpeech.Engine</a>	Constants and parameter names for controlling text-to-speech.
class	<a href="#">TextToSpeech.EngineInfo</a>	Information about an installed text-to-speech engine.
interface	<a href="#">TextToSpeech.OnInitListener</a>	Interface definition of a callback to be invoked indicating the completion of the TextToSpeech engine initialization.

# Basic Methods

int	<code>setVoice (Voice voice)</code> Sets the text-to-speech voice.
void	<code>shutdown ()</code> Releases the resources used by the TextToSpeech engine.
int	<code>speak (CharSequence text, int queueMode, Bundle params, String utteranceId)</code> Speaks the text using the specified queuing strategy and speech parameters, the text may be spanned with TtsSpans.
int	<code>speak (String text, int queueMode, HashMap&lt;String, String&gt; params)</code> <i>This method was deprecated in API level 21. As of API level 21, replaced by <code>speak (CharSequence, int, Bundle, String)</code>.</i>
int	<code>stop ()</code> Interrupts the current utterance (whether played or rendered to file) and discards other utterances in the queue.

# Speak() method parameters

<i>text</i>	The string of text to be spoken. No longer than <code>getMaxSpeechInputLength()</code> characters.
<i>queueMode</i>	The queuing strategy to use, <code>QUEUE_ADD</code> or <code>QUEUE_FLUSH</code> .
<i>params</i>	Parameters for the request. Can be null. Supported parameter names: <code>KEY_PARAM_STREAM</code> , <code>KEY_PARAM_VOLUME</code> , <code>KEY_PARAM_PAN</code> . Engine specific parameters may be passed in but the parameter keys must be prefixed by the name of the engine they are intended for. For example the keys "com.svox.pico_foo" and "com.svox.pico:bar" will be passed to the engine named "com.svox.pico" if it is being used.
<i>utteranceId</i>	An unique identifier for this request.

# Text2Speech1

Good Morning Students!

SPEAK!



Navigation and system icons including: GPS, ID, Wi-Fi, speaker, volume, and home buttons.

# Speech Recognition

- Very important modality of interaction. Same as TTS, from the opposite direction.
- We can speak to our app.
- Greek not currently supported by Google (End 2015)
- Quite many applications. People with special needs, driving, etc.



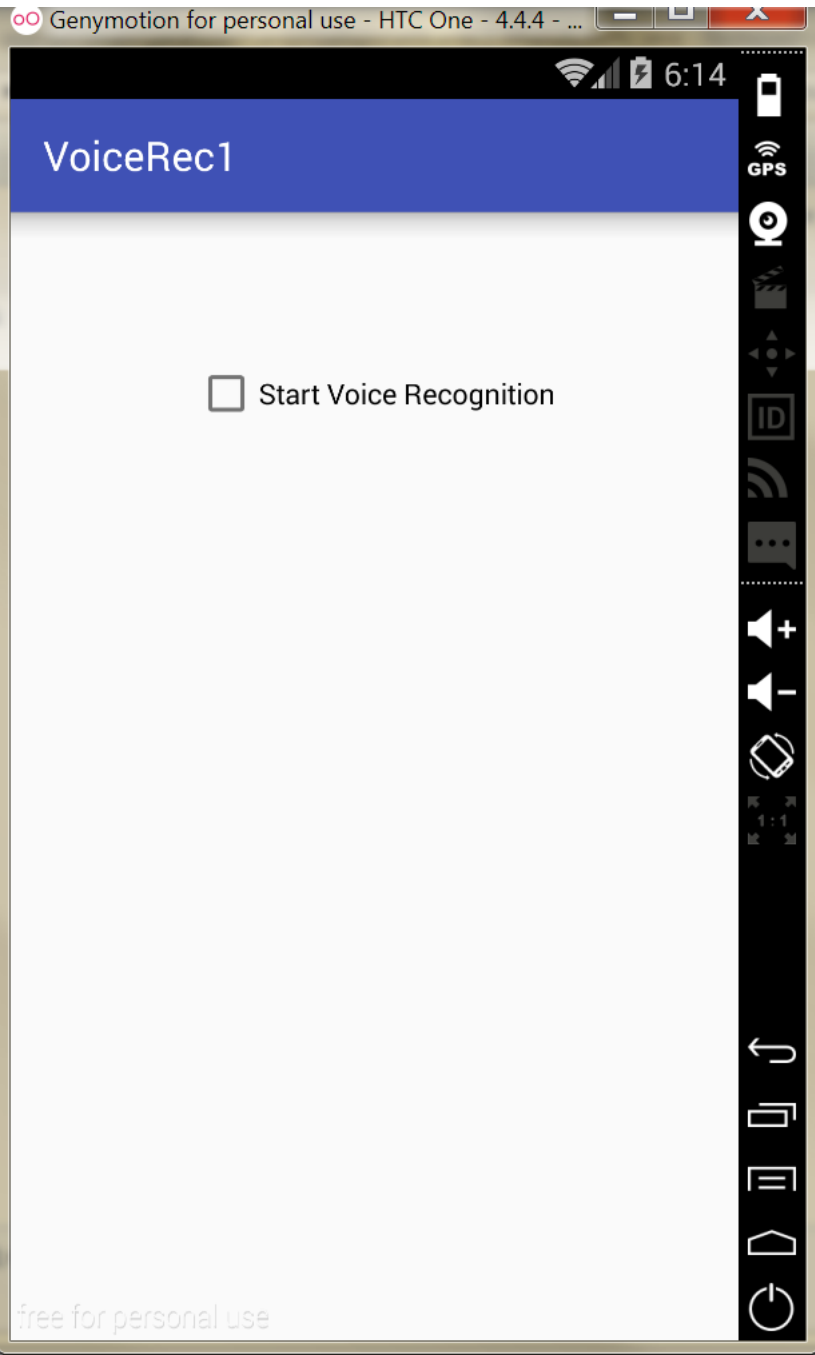


# SpeechRecognizer Overview

- This class provides access to the speech recognition service.
- Do not instantiate this class directly, instead, call [createSpeechRecognizer\(Context\)](#)
- This class's methods must be invoked only from the main application thread.
- The implementation of this API is likely to stream audio to remote servers to perform speech recognition.

# RecognizerIntent Overview

- Support speech recognition through starting an Intent
- ACTION\_RECOGNIZE\_SPEECH
  - Starts an activity that will prompt the user for speech and send it through a speech recognizer.
  - The results will be returned via activity results
  - You must either use [startActivityForResult\(Intent, int\)](#) (,or provide a PendingIntent, to receive recognition results).



# VoiceRec1

Start Voice Recognition

