# Welcome
# Android Listener

GRUBHUB

August 21st 2019

# Voice Actions

Graham Hensley
August 21, 2019

# Lots of reasons to use your voice

When your app supports voice interaction it becomes more versatile.

Your app can be operated hands free.

Your app can be controlled while the user is doing something else (like driving!)

In some states it’s illegal to touch or look at your mobile screen while driving. Help your user stay on the right side of the law.

Lots of voice interactions don’t even require record audio permission

# Lots of ways to use your voice

The Android OS and Goole provide several ways to use voice in your application.

System Voice Actions - Android OS common Intents - “Take a photo”, “Set a timer”

Voice Interaction API - Interact with the user via voice for a confirming action, or to pick an option from a list

Speech Recognition - Voice transcription services handled by google or other OEM provider. Not guaranteed to be on every device

Google Assistant - Google extension of system voice actions to cover food ordering, fitness apps, ride hailing, etc

Accessibility Voice Access - Navigate the app by saying “click [button label]”, “ Go back”, “Go Home”

# System Voice Actions

System Voice Actions allow you to access the Android OS common Intents by using your voice. On Jelly Bean or greater.

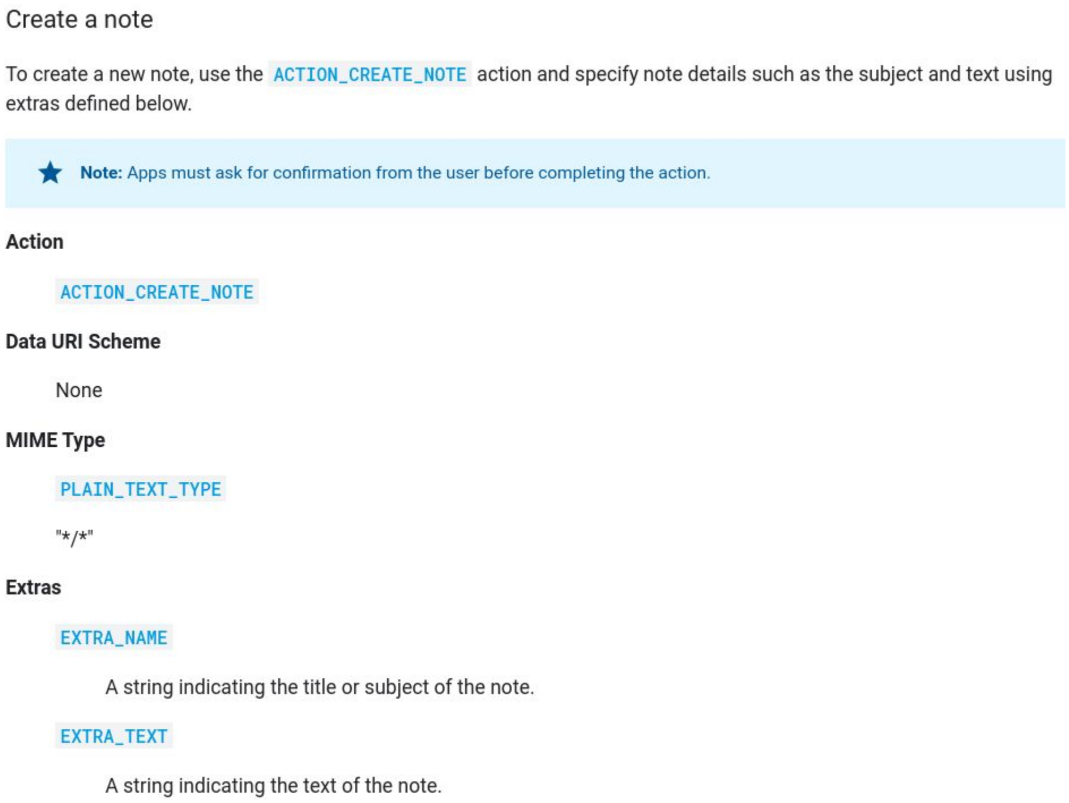
Formerly known as Google Now Voice Actions

Common Intents are a collection of implicit Intents and filters that cover core phone features like setting an alarm, placing a call, stopping or starting music player, etc.

A user may say “Take a note called shopping list, bread, eggs, milk”

Prefered app that implements the note taking common intent will get invoked and data passed in.

# System Voice Actions



### Create a note

To create a new note, use the `ACTION_CREATE_NOTE` action and specify note details such as the subject and text using extras defined below.

> ★ **Note:** Apps must ask for confirmation from the user before completing the action.

**Action**

`ACTION_CREATE_NOTE`

**Data URI Scheme**

None

**MIME Type**

`PLAIN_TEXT_TYPE`

"*/*"

**Extras**

`EXTRA_NAME`

A string indicating the title or subject of the note.

`EXTRA_TEXT`

A string indicating the text of the note.

# System Voice Actions

Finally, report that you completed or failed to fulfill the users request using the Firebase App Indexing.

```
    AppIndex.AppIndexApi.end(mClient, setAlarmAction);
```

This will allow google to display a card and give call backs to whatever action the user took

All apps can be started with “Start [App Name]”

https://developer.android.com/guide/components/intents-common.html

https://developers.google.com/voice-actions/system/

# Assistant App Actions

The Google assistant supports several new “built-in” voice actions that are an extension of the System voice actions.

New action types include

- Ordering food
- hailing a cab - formerly of Androidwear
- controlling a fitness tracker - formerly of Androidwear
- finance interactions like sending money, and buying stocks

Rather than use Intent Filters in the manifest. Assistant actions rely on an actions.xml file being provided by the app that describe how to handle the action. Actions can be fulfilled by a deep link to a website, app activity, or an inline remote view within the assistant known as a “slice”

# Speech Recognition

Get transcription of user speech in your app. Recognition is handled by OEM or Google.
Not all Android devices support the feature.

API has been around forever. Most recognition functions are part of API 3.

Captured speech can be sent directly to Search API. Though somewhat redundant

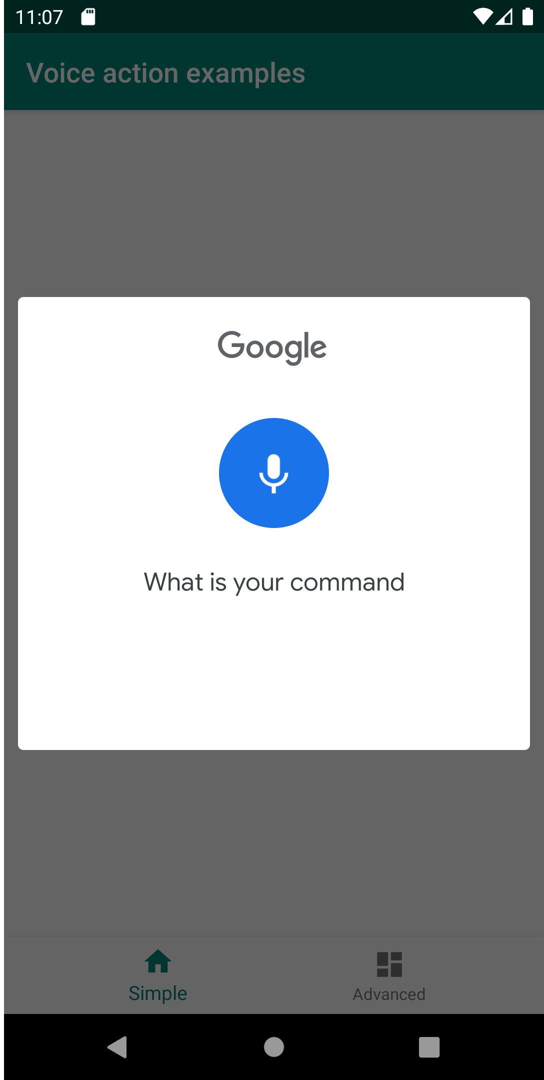
Requests for Speech Recognition are done with a RecognizerIntent
Intent has tons of extras for controlling many parts of the capture experience.
like

- setting partial results as user is talking,
- setting the amount of quiet or idle time is allowed before the speech input is considered done
- setting the language that will be spoken
- setting prompt displayed with Recognition Activity
- Support for offline recognition

# Speech Recognition

```
val intent = Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH)

intent.putExtra(
    RecognizerIntent.EXTRA_LANGUAGE_MODEL,
    RecognizerIntent.LANGUAGE_MODEL_FREE_FORM)

intent.putExtra(RecognizerIntent.EXTRA_PREFER_OFFLINE,
true)

intent.putExtra(RecognizerIntent.EXTRA_PARTIAL_RESULTS,
true)

intent.putExtra(RecognizerIntent.EXTRA_PROMPT,
"What is your command")

startActivityForResult(intent,response_code_basic_voice)
```

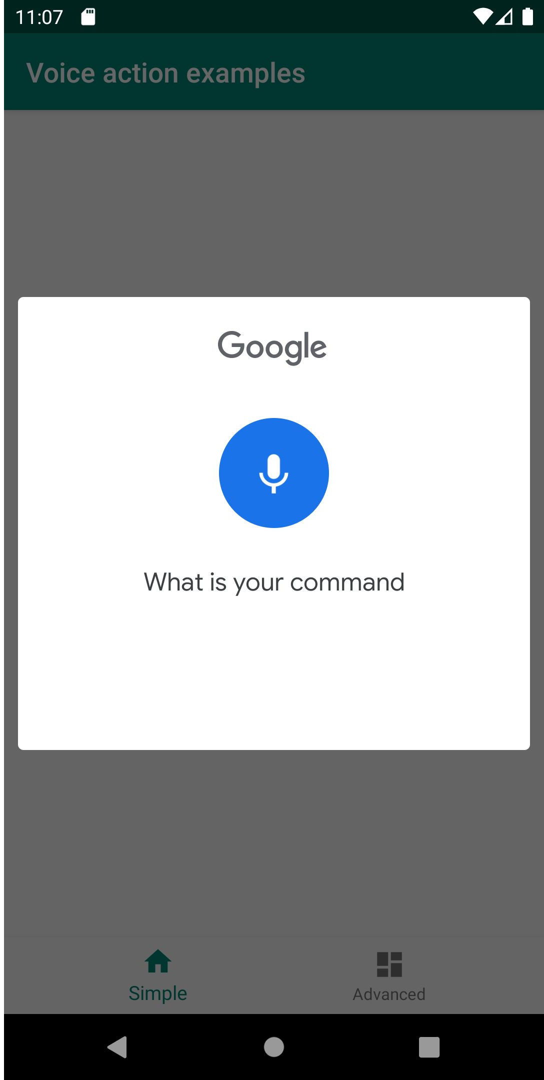
GRUBHUB™



What is your command

# Speech Recognition

```
override fun onActivityResult(requestCode: Int, resultCode: Int, data:
Intent?) {

    if (requestCode == response_code_basic_voice
          && resultCode == Activity.RESULT_OK) {

        val results = data!!
               .getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS)
    }
}
```



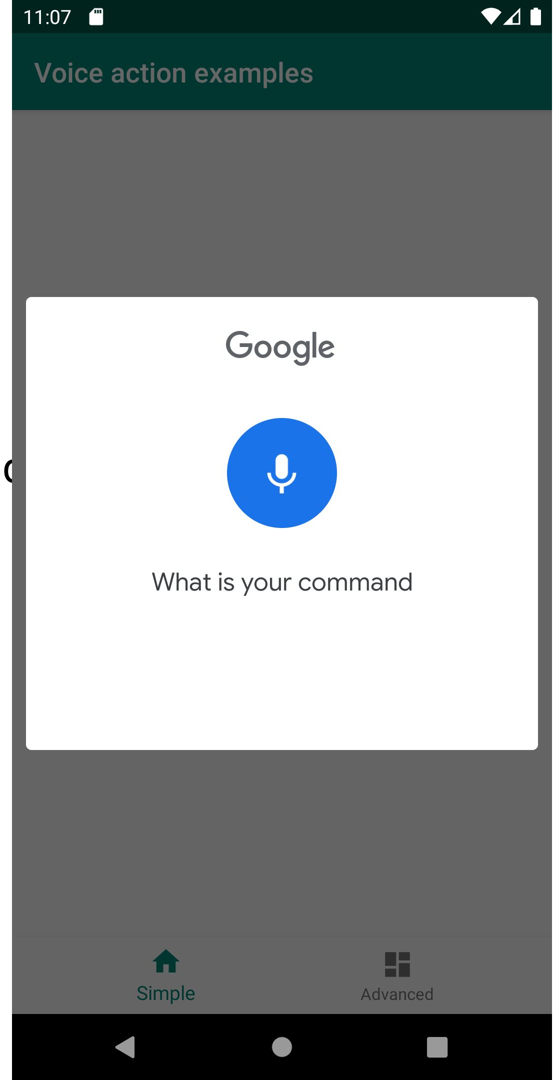
What is your command

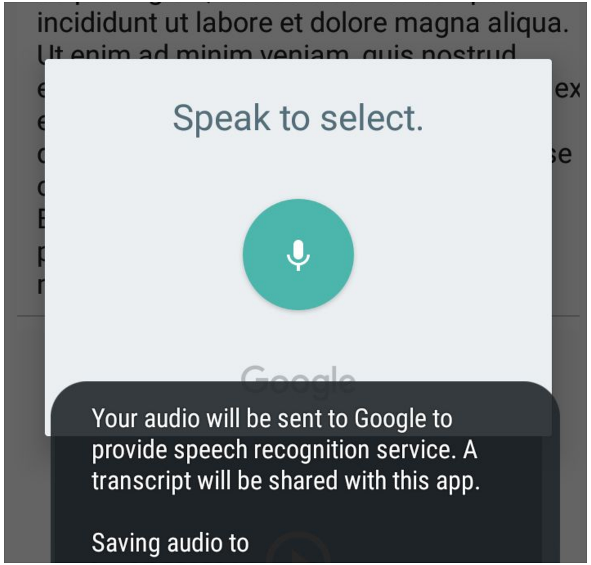
GRUBHUB™

# Speech Recognition

This basic implementation doesn’t require RECORD_AUDIO permission, but comes with some visual drawbacks.

Audio tone always plays on start and end of audio recording.




incididunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrud

Your audio will be sent to Google to provide speech recognition service. A transcript will be shared with this app.

Saving audio to

# Advanced Speech Recognition

By implementing our own RecognitionListener we can better control the experience around speech input.

To user your own Listener however you must request the RECORD_AUDIO.

By implementing your own Listener you will have better control over starting and stopping of voice input, as well as better support for partial transcripts as the user is talking.

With your own Listener you will not be forced to show your audio is going to Google toast, or any sort of modal. Recognition will appear to becoming from within your own app even though it is the same speech recognition service.

RecognitionListener also provides callbacks with metrics to help visualize the users speech as they are talking

# Advanced Speech Recognition

override fun **onReadyForSpeech**(argsIn: Bundle?) - the speech recognizer is ready

override fun **onBeginningOfSpeech**() - the user has started speaking
override fun **onEndOfSpeech**() - the user has stopped talking
override fun **onRmsChanged**(rmsdB: Float) - root mean squared of the current
    decibel level of speech
override fun **onBufferReceived**(buffer: ByteArray?) - sequence of 16 bit ints
    describing the audio input - not used anymore

override fun **onEvent**(p0: Int, p1: Bundle?) - reserved for future events

override fun **onError**(erroCode: Int) - error occurred while transcribing
override fun **onPartialResults**(results: Bundle?) - bundle with transcribed text
so
    far and meta (confidence score)
override fun **onResults**(results: Bundle?) - bundle with transcribed text  so far
    and meta (confidence score)

# Advanced Speech Recognition

To use your listener, first acquire a Speech Recognizer

```
        recognizer = SpeechRecognizer.createSpeechRecognizer(this)
```

Set your listener

```
        recognizer.setRecognitionListener(this)
```

Start transcribing

```
        recognizer.startListening(transcribeIntent)
```

Stop transcribing

```
        recognizer.stopListening()
```

Destroy your Recognizer

```
        recognizer.destroy
```

# Demo time!

# About our Host

## What is Grubhub?

Grubhub is the nation's leading online and mobile food ordering and delivery marketplace dedicated to connecting hungry diners with local takeout restaurants. The company's online and mobile ordering platforms allow diners to order from more than 105,000 takeout restaurants in over 2,000 U.S. cities and London. The Grubhub portfolio of brands includes Grubhub, Seamless, LevelUp, Tapingo, Eat24, AllMenus and MenuPages.

## Company Statistics

- Provided more than $5 billion in gross food sales to local takeout restaurants in 2018
- Processes 467,500 daily orders
- Serves 17.7 million active diners
- Sent more than $1 billion in total tips to drivers

**Tech Blog**: https://bytes.grubhub.com/

GRUBHUB™

# Join the Team!

**GrubHub is currently looking for talented people across a wide rage of studies. Marketing, finance, operations, and technology.**

**Tech needed in Chicago :**

Android Software Engineer II - Mobile

Sr. Software Engineer

Team Lead / Tech Lead

Android Senior Software Engineer - Diner

**More info: https://careers.grubhub.com**

# Thanks

Graham Hensley
Team Lead - Software Engineering
ghensley@grubhub.com