



Dialogflow

Índice	
Introducción a DialogFlow.....	2
Agents.....	2
Intents/Objetivos.....	3
Entities.....	4
API de DialogFlow.....	5
Ejercicio.....	5

## Introducción a DialogFlow

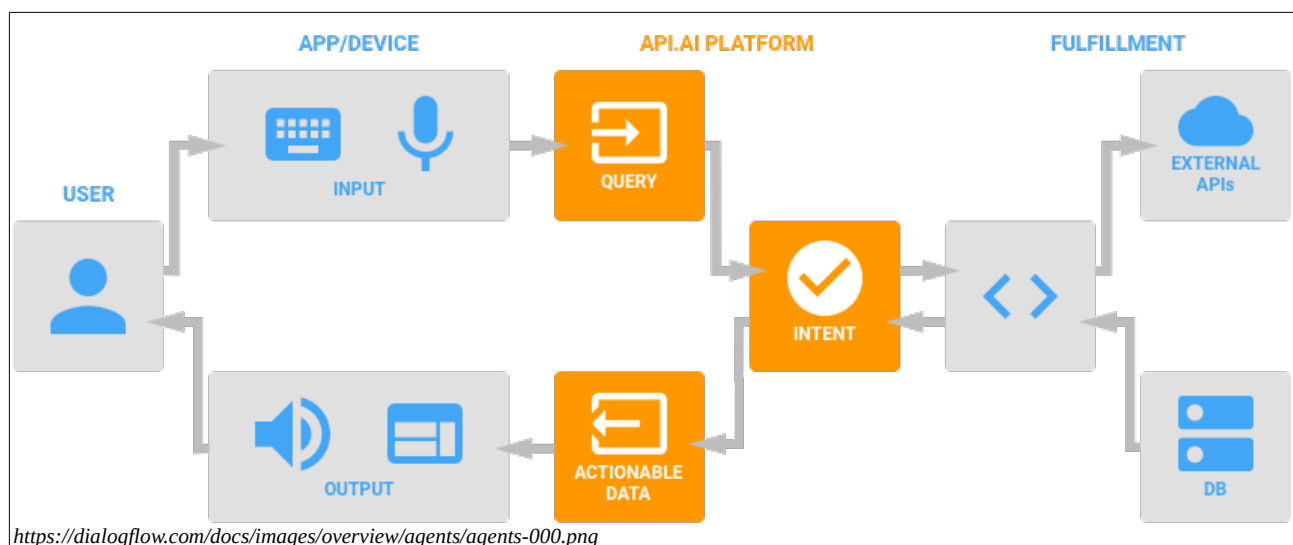
DialogFlow (<http://www.dialogflow.com>) es una herramienta web para la creación de módulos de comprensión del lenguaje natural, tanto hablados como basados en texto. Permite integrar el componente de comprensión con gran cantidad de plataformas como Google Assistant, Twitter, Telegram, Facebook Messenger, Skype, ... Además, dispone de SDK (software development kit) para los lenguajes más utilizados (python, C, java, node.js, ...). Esto facilita el desarrollo de sistemas de diálogo para dominios específicos de forma sencilla.

Inicialmente denominada **Speaktoit**, desde su adquisición por parte de Google la plataforma se denominó **API.ai**, y finalmente, el 10 de Octubre de 2017 ha pasado a llamarse **DialogFlow**.

DialogFlow dispone de una versión gratuita que es más que suficiente para crear sistemas medianamente complejos basados en texto y sistemas más básicos basados en habla. Para poder utilizar DialogFlow se necesita una cuenta de Google. La plataforma está en continuo crecimiento por lo que es posible que esta documentación se quede obsoleta parcialmente en un breve periodo de tiempo.

## Agents

Un módulo de comprensión del lenguaje natural se representa como un Agente (Agent) en DialogFlow. El siguiente esquema muestra la integración de un Agente en un sistema de diálogo.



La función de los Agentes es la interpretación semántica de los turnos del usuario. Estos turnos llegan al agente en forma de texto. Aunque DialogFlow permite la interacción oral, antes de ser procesados, los turnos del usuario deben ser reconocidos utilizando el reconocedor de Google, KALDI o cualquier otro reconocedor automático del habla.

Los dos elementos principales de un Agente son los objetivos (**Intents**) y las entidades/atributos (**Entities**).

- Los **Intents** representan los objetivos que el usuario puede querer alcanzar mediante la interacción con el sistema. Por ejemplo, en un sistema para solicitar información meteorológica diferentes Intents podrían ser: *obtener información sobre la temperatura*, *obtener información sobre la probabilidad de precipitaciones*, ... Cada turno del usuario se intentará clasificar en un Intent. Existe la posibilidad de definir Intents “por defecto” que se activaran cuando no se encuentre otro más adecuado.
- Las **Entities** representan la información que el usuario aporta al sistema. En la mayoría de los casos esta información es imprescindible para poder llevar a cabo los objetivos. Cada Entity es un atributo que puede aportar el usuario. Siguiendo con el ejemplo del sistema de información meteorológica, en este caso, Entities evidentes podrían ser, el *lugar* para el que nos interesa conocer la previsión y la *fecha* o *fechas* sobre las que se pregunta. Los valores válidos para las entidades (tipos de datos de los atributos) pueden ser definidos por el diseñador del agente o pueden ser elegidos de entre los predefinidos en DialogFlow. Existen mecanismos para que distintos Intents compartan valores de Entities, de esta forma, en nuestro ejemplo, la fecha y el lugar serían compartidos por los dos Intents.

Cada módulo se vincula automáticamente con un proyecto en Google Cloud Platform. Esto limita indirectamente el número de Agentes que podemos crear con una única cuenta de Google.

## Intents/Objetivos

A cada objetivo del diálogo le corresponderá un Intent. Además, pueden definirse Intents adicionales para la aportación de información, respuestas genéricas, bienvenida, ...

Al definir un objetivo se pueden rellenar varias secciones:

- **User says.** Es la única sección obligatoria al definir un objetivo. En ella se incluyen muestras de intervenciones del usuario cuyo objetivo es el mismo que se está definiendo. Un ejemplo de una muestra *User says* para el objetivo de obtener la previsión de lluvias podría ser: “¿lloverá el próximo jueves en Barcelona?”. Cada ejemplo se debe anotar, el proceso de anotación consiste en asignar Entidades (las entidades se comentan en el punto siguiente) a segmentos del ejemplo. En nuestro caso “el próximo jueves” podría ser una fecha (date) y “Barcelona” una ciudad (geo-city). El sistema realiza un etiquetado automático que puede ser revisado y corregido por el usuario. DialogFlow generalizará las muestras incluidas en *User says* para aumentar la cobertura del *Objetivo*.
- **Events.** Además de por intervenciones del usuario, los *Objetivos* pueden ser activados por *Eventos*. De esta forma se puede forzar la activación de un *Objetivo* independientemente de la intervención del usuario; por ejemplo, para activar el mensaje de bienvenida. En esta sección se definen los eventos que disparan el *Objetivo*.
- **Action.** Define el nombre de la acción asociada al *Objetivo* y los parámetros necesarios para realizar dicha acción. El nombre de la acción y los valores de los parámetros forman parte de la información que es enviada en formato JSON por DialogFlow a la aplicación. Los parámetros permiten asociar trozos de la intervención del usuario con *Entidades*. Dentro de un *Intents*, los parámetros se definen en dos lugares distintos. Por una parte, las frases de ejemplo de “*User says*” se segmentan y esos segmentos se asocian a *Entities* y parámetros. Por otra parte, el apartado dedicado a los parámetros dentro de la sección *Action* permite realizar un ajuste fino de los parámetros de una acción, se puede definir:
  - a) *Required.* Indica si el parámetro es obligatorio para completar el *Objetivo*.
  - b) *Parameter Name.* El nombre del parámetro.

- c) *Entity*. Entidad (tipo de datos) sobre la que se define el parámetro, tipo de datos.
  - d) *Value*. Valor por defecto del parámetro, pueden ser constantes o referencias a otros valores.
  - e) *Is List*. Indica si el parámetro puede tener más de un valor. En caso de ser falso, los nuevos valores sustituirán a los anteriores.
  - f) *Prompts*. Define el mensaje que el *Agente* responderá cuando la frase del usuario no tenga valor para un parámetro obligatorio. Se pueden definir varios mensajes asociados al mismo parámetro.
- **Response**. El apartado *Response* define la respuesta que proporcionará el *Agente* cuando se active el *Objetivo*. Se pueden añadir hasta 10 respuestas distintas para dar mayor naturalidad al *Agente*. La respuesta por defecto siempre es textual, pero dependiendo de la plataforma para la que se desarrolle el bot (Google Assistant, Skype, Telegram, Viber, ...) es posible añadir otros elementos adicionales como imágenes o tarjetas. En el texto de la respuesta se pueden añadir referencias a parámetros, tanto a los parámetros del propio *Objetivo* (referenciados como *\$nombre\_parámetro*), como a parámetros definidos en otros *Objetivos* y accesibles mediante el uso de contextos compartidos (referenciados mediante *\$nombre\_contexto.nombre\_parámetro*). Incluir los valores de los parámetros en las respuestas que da el agente permite informar al usuario de lo que conoce el sistema. Pero se debe tener en cuenta que no se pueden utilizar respuestas que hagan referencias a parámetros que no tienen valores definidos. Por tanto, en todas los *Objetivos* se debe incluir alguna respuesta que no tenga parámetros o que sólo tenga parámetros obligatorios.
  - **Contexts**. Definen los contextos del usuario, esto permite tratar en un mismo *Objetivo* situaciones distintas. Existen tanto contextos de salida (*output contexts*) como contextos de entrada (*input contexts*). Los contextos de salida se activan cuando la intervención del usuario coincide con el *Objetivo* en los que se define el contexto. Los contextos de entrada impiden que se active un *Objetivo* a menos que nos encontremos en el contexto de entrada definido para ese *Objetivo*. También es posible forzar un contexto cuando se envía una petición de usuario a **DialogFlow**.

## Entities

Todos los parámetros tienen una entidad asociada. Las entidades se pueden ver como tipos de datos sobre los que se definen los parámetros de los *Intents*.

Existen tres tipos de entidades:

- Entidades del sistema: **DialogFlow** incluye un gran número de entidades predefinidas, para fechas, horas, números de teléfono, colores, países, direcciones, ... El número y la funcionalidad de las entidades disponibles depende del idioma. Se pueden consultar las entidades predefinidas en: <https://dialogflow.com/docs/reference/system-entities>
- Entidades definidas por el desarrollador del bot: el diseñador del agente puede definir sus propias entidades para ser utilizadas en el bot. Los valores de las entidades se pueden introducir desde la interfaz web de **DialogFlow** o mediante ficheros en formato CSV o JSON. También es posible definir entidades desde llamadas al API de **DialogFlow**.
- Entidades de usuario: las entidades de usuario se definen para cada sesión, esto permite personalizar las entidades para cada usuario. La definición de entidades de usuario se debe realizar necesariamente mediante llamadas al API. Más información sobre la creación de entidades de usuario en <https://dialogflow.com/docs/reference/agent/userentities>

## API de DialogFlow

Existen SDKs para múltiples lenguajes que permiten acceder al API de **DialogFlow** (<https://dialogflow.com/docs/sdks>). Aunque existen dos versiones del API, nosotros utilizaremos la v1 desde python mediante la librería *apiai* (<https://github.com/dialogflow/dialogflow-python-client>).

## Ejercicio

Como ejercicio proponemos el desarrollo de un bot conversacional para una tarea sencilla. Puedes utilizar como base el ejemplo de la pizzería y modificar lo que necesites o puedes empezar desde el principio con un sistema nuevo. Si conoces algún sitio web que admita peticiones GET para proporcionar información (como horarios de cines, previsiones meteorológicas, últimas noticias, ...), puede ser una buena idea diseñar un bot para acceder a él.