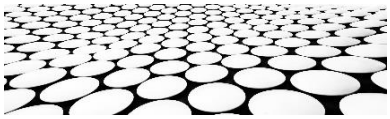
SEG3125 User Interface Design and Analysis



MODULE 10 – TUTORIAL/LAB

Building a Service Business Chatbot with Dialogflow

GOALS

The goal of Lab 10 is to get a bit of hands-on experience in building a chatbot that could be used as an alternative to a traditional (WIMP style) UI for a service business. We will go back to your Lab 4 and 5 in which you had to build a UI for a hair salon or a bike repair shop.

You will now build a simple chatbot to obtain the information necessary to book an appointment. This information is the information you had in your traditional menus (e.g. name of client, phone number, type of hair cut desired, day/time of appointment, professional preferred).

The WIMP interface is now replaced by a dialog. The client could say: "I want a long hair-cut with Johan on Tuesday July 22" and the chatbot will have learned to classify this sentence to the intent "Get-Service" and it would also detect 3 entities (parameters) for doing the following slot filling:

- Intent: Get Haircut
  - Hairstyle: "long hair-cut"
  - Hairdresser: "Johan"
  - Date: "Tuesday July 22"

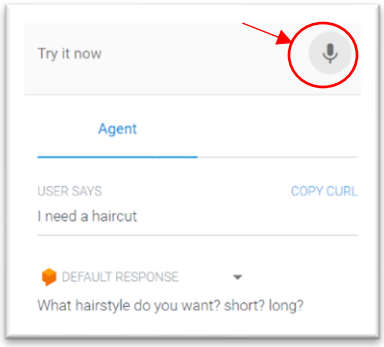
Of course, there would be a lot to do after this step, but we won't go further in this lab. We will not link our chatbot to any real action, nor will we make it very complex. This laboratory is meant as an exploration of the platform *Dialogflow* and should help you better understand the ideas of intents and entity detection that I presented in the videos.

Why Dialogflow? Well, I chose *Dialogflow* for this laboratory, as it is the best "match" in its view of conversational agents to what I covered in the videos and it uses the same terminology (intent detection, entity extraction, etc).

It also offers good integration with JavaScript based backend, which again is compatible to the focus on JavaScript we adopted this semester. We will NOT do any integration in this lab, but it is a possibility for you if you wish to further explore conversational agents.

Also, Dialogflow is used by a large community, making it somewhat "stable", or at least insures some future to it. It also means that there is community support, as many users post information to help each other.

And one last feature is that Dialogflow contains a voice recognition module, so when you try your dialog, you can use the little microphone, as you see in this screenshot of the top right of the Dialogflow console.



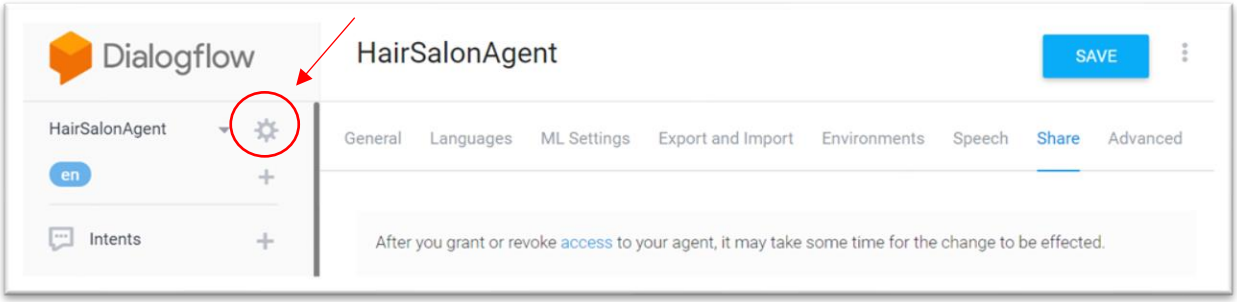
Unfortunately, a small downside is that even if you can use Dialogflow for free during the trial period, as it is software on the Google Cloud platform, you are asked for information to register, including a credit card number. We assure you that nothing will be debited, but ... if you are not comfortable with that, write me an email, and I will then suggest a theoretical alternative laboratory.

SUBMISSION DEADLINE

- Tuesday, July 28th, 11:30pm

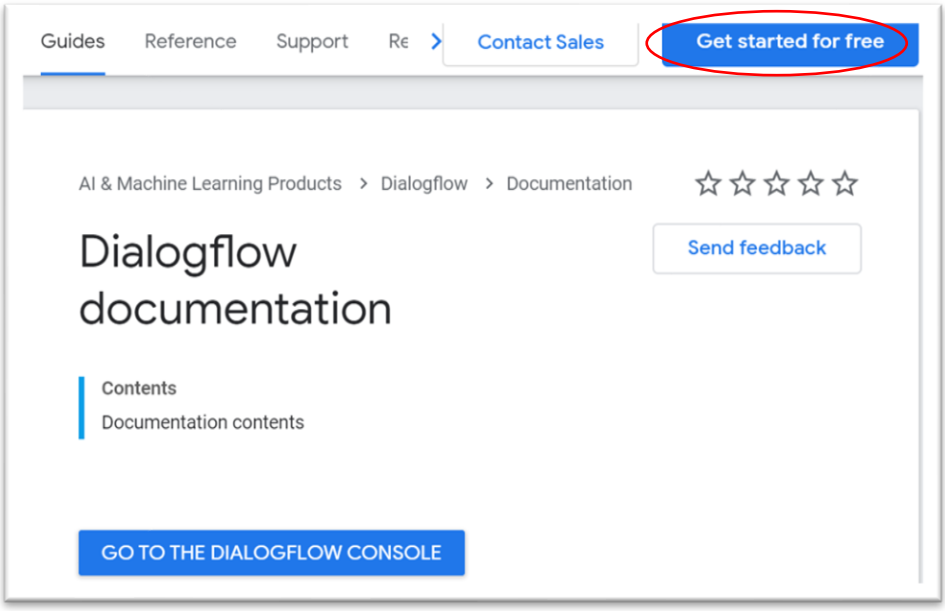
SUBMISSION METHOD

- In Brightspace, the Module 10 checklist contains a link for your submission.
- Submit a link to a video demonstrating your chatbot. Be specific and concise. Your video should not be more than 10 minutes long.
- ALSO, in Dialogflow, if you click on the parameter icon next to your agent's name, you will have access to a menu in which there is an option Share. Please invite your TA as a Reviewer. Make sure you invite the TA you are assigned to.



INSTRUCTIONS / TUTORIALS

Dialogflow is part of Google Cloud AI & Machine Learning Products, if you follow this link: https://cloud.google.com/dialogflow/docs/ you will see the screen below.

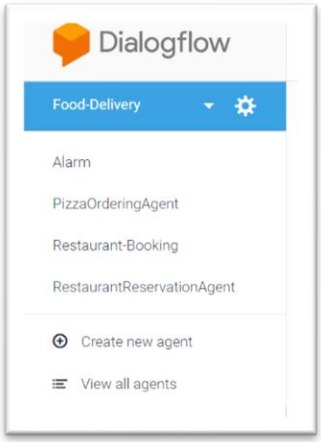


Step 1 - Get access to Dialogflow

First, you need to sign-in into DialogFlow. You see the "Get started for free". **Unfortunately, you cannot use your U of O account because you will not be able to save your agents.** I suggest you use your personal Gmail account, or if you don't want to use your regular account, you can create a new Gmail account with which you will use to test this software.

Step 2 - Learn about Dialogflow underlying model

If you scroll down the page you will see a link to Introduction videos. I recommend the first two episodes of the **Basics of Dialogflow** series. The first one (3 min) talks about intent matching, and the second one (5 min) talks about identify entities. You will need both for this lab. The third video is also interesting, but we will not go into non-linear dialogs in this lab. You could watch the beginning up to when it talks about mandatory slots to fill.



Build an agent

Now if you go in the QuickStart section, you will see a tutorial on how to **build an agent**. This is REALLY the place to start.

The first step tells you to go to Dialogflow console. Then, in the console, on the left, there is a list of agents (if you have created some already) and the possibility to build a new agent.

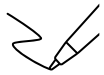
When you create your agent, give it the name your want, BUT make sure to include your name in the name of the agent (e.g. HairSalon-CarolineBarriere for me) because you will eventually share your agent with the TA and he must be able to quickly identify yours among the list he will have.

Once your agent is created, you will be able to see (and create) intents. First you should deal with the Welcome Intent and Fallback Intent. These are provided as defaults in Dialogflow as they would be useful to all dialogs. The Welcome Intent is to recognize when the user says “hi” or “hello”, and the FallBack intent is when a user says something that the Agent does not understand. You should customize those.

The tutorial then talks about creating new intents, and this is where the chatbot will become even more specific to your business (hair salon/bike repair).

The tutorial will then talk about parameters and entities. It will even talk about the predefined system entities and the ones you can define yourself. You need all this information for your lab.

Continue exploring. Everything you need to do this lab is provided in this section on **Build an agent**.

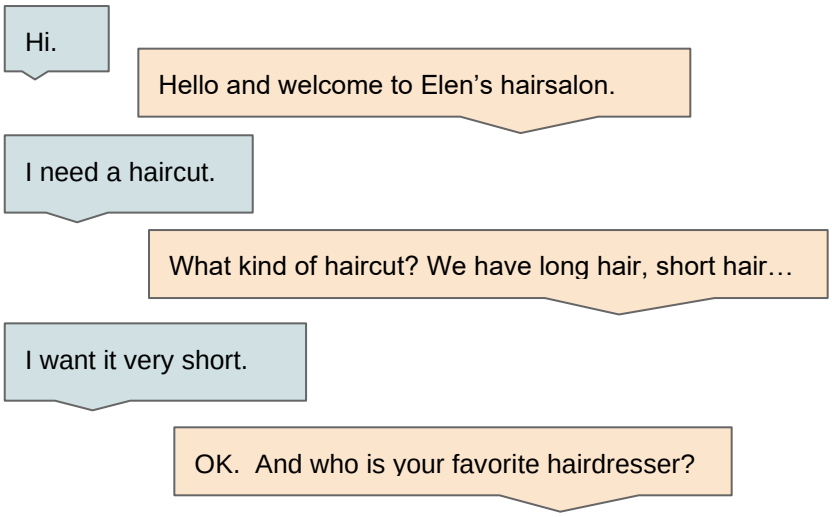


DESIGN

In general, when we build a chatbot, a large part of the design is what we call "Conversational Design", meaning how we want the conversation to evolve (or "flow" hence the name of the platform).

But in this laboratory 10, we will not go very far in that design. I will provide a set of requirements in the Coding Section. Nevertheless, we do want to plan for a short conversation in which the important information would be obtained to schedule an appointment (but not validate it).

For example:



And so on… See how the Welcome message (Hello and welcome to Ellen's hair salon) is personalized to the hair salon. You do want various ways of saying Welcome though, so it seems a bit more human-like. That Welcome message is the *Text Response* to the *Welcome Intent*, which is detected first. And how was that detected? By the fact that you would provide *training sentences* (hi, hello, howdy) that correspond to that intent.

So each intent comes with training sentences and with an optional text response. Often an intent, such as *GetService* for example, could be detected from the sentence "I need a haircut", but then, that intent requires mandatory parameters (e.g. haircut style, name of hairdresser, etc) which we set in Dialogflow and for which we provide a Prompt (e.g. What kind of haircut?).

The haircut style would be an Entity, defined by a list of possible values (like an enumeration in programming). To avoid having the user going all over the place, it is good conversational design to include elements of the list in the prompt (as you see above, "we have long hair, short hair, …").

STARTING POINT

I do not provide starting code for this lab. Dialogflow does have Default Intents, such as the Welcome and Fallback intents since those would happen in almost all dialogs. So, no need to start from scratch there. You will see that there are already many training sentences as paraphrases of "hello".

Also you can browse through the "Prebuild agents" shown in Dialogflow. These are agents built for all kinds of situations. But it's easy to get lost in all those examples…

CODING

There is no coding per se in this lab, but rather the "programming" of your HairSalonAgent or your BikeRepairAgent is in terms of:

- intents to recognize and for which you provide training sentences (paraphrases containing some or none of the entities required)
- required entities, which are associated with intents and for which you provide prompts
- intents to respond to and for which you should give text responses

Here are the requirements:

- A personalized Welcome intent. This means you modified the Text Responses to have 3 ways of saying Hello/Welcome for your business. (e.g. "Hi and welcome to John's Salon" is one of them).
- A personalized Fallback intent. This means you modify the Text Responses to tell the user something else than "Sorry I don't understand". More productively give user examples of sentences that the Agent understands.
- A ThankYou intent, for which you can define 5 training sentences and 3 possible text responses. This intent correspond to when the user says "thank you".
- A NeedService intent, for which you define some parameters (haircut style/bike repair type, professional wanted, date, time, client's name, client's phone number) as mandatory. Provide at least 15 sentences for this intent with various entities in them or not.
- Define the parameters used in the NeedService intent. Some will be of predefined types (e.g. @sys.date-time, @sys.phone-number) but others you will have to define as entities.
    - For any predefined types, you can assume that Dialogflow already has training examples. But, for the entities that you define, you need to provide training sentences.
    - Define at least 3 prompts for each parameter required.

There will be no validation of dates/times that are correct/incorrect given the chosen professional. There will also be no action performed after this dialog. So we are only doing the gathering of information.

The number of training phrases and text responses given above is the minimum number required for your laboratory. But under the hood, in Dialogflow, there is "machine learning"… and therefore the more training examples you give to Dialogflow for each of your intentions, the better the chatbot will be able to correctly classify the user's intentions and extract the required parameters.

☆☆☆ EVALUATION

- Lab 10 is worth 3.5% and will be marked on 10.
- The TA will watch your video (max 10 minutes) in which you show:

  1. First of all, you simulate a short linear conversation for making an appointment. Your conversation should contain:

     - Exchange for Hello / Welcome (*Welcome Intent*)
     - Something the user says the chatbot does not understand to show the response to the *FallBack intent*.
     - An appointment request with partial info (*GetService intent*)
     - Requests for information to do slot filling by the chatbot
     - A thank you from the user to whom the chatbot responds (*ThankYou Intent*)

  2. Show the list of intents and show for each intent a set of training sentences.

  3. Show your parameters and their types (the entities you had to define or the predefined types), as well as the prompts used to get their values.

- You will also share your agent with the TA who will be able to look at what you did.
- Any delay beyond the deadline will have a penalty of 10% per day.

QUESTIONS

- You can ask your questions in the Module 10 discussion forum on Brightspace.
- There is a consultation schedule for the laboratories in the organization section of the course on Brightspace. That will tell you when a teaching assistant is available on the forum or by zoom.
- You can also send your questions directly to the TA you are assigned to. Refer to the lab consultation schedule to see which TA you are assigned to.