

Topic 8

Deploy Chatbots with Advanced Concepts

AI HUMAN INTERFACE

Learning Outcomes

- ❑ Understand how to deploy a simple chatbot on a real website
- ❑ Understand how to maintain the conversation state with context variables

Deploying the Chatbot

AI HUMAN INTERFACE

Assistant + Skills

- We built dialog skills in previous topics (intents, entities, response).
- An assistant is a chatbot that can have one or more skills with it.
- It is time to: create the actual assistant, link the dialog skills to it, and deploy the chatbot to make it publicly accessible.

What We Have Now (Recap)

- We have created a basic, but functioning chatbot.
- The chatbot can detect user intents, identify entities from user queries, and respond properly to users.
- But, it is only available from the “Try It Out” panel in the Dialogflow platform.



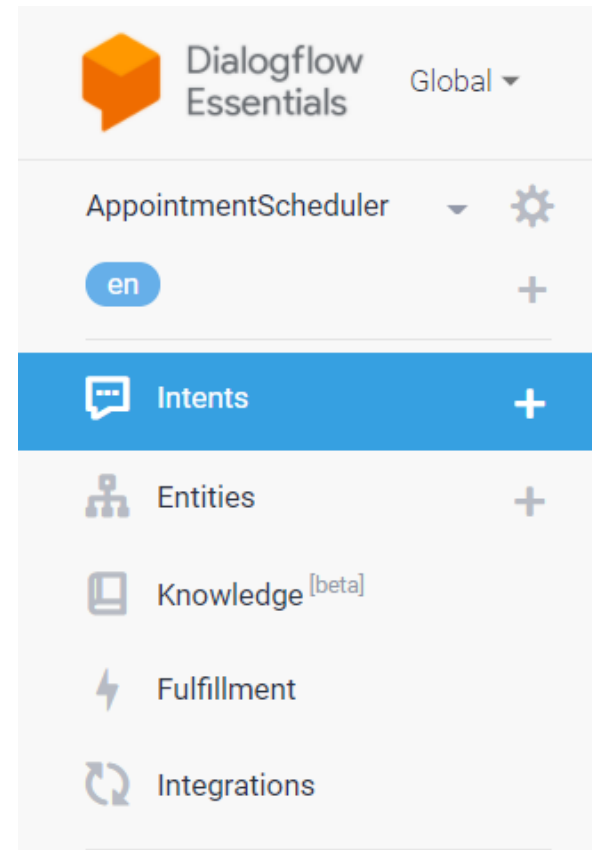
Workspace

Collection of artifacts –
Intents, Entities and Dialogs

Build

Deploy

Improve / Metrics



Intents

Goal or purpose of the user's input. Adding examples helps our virtual assistant understand different ways in which people would say them.

Provide at least 5 examples for each intent.

Schedule Appointment

SAVE

Training phrases ?

Search training phrases 🔍 ^

⚠️ Template phrases are deprecated and will be ignored in training time. More details [here](#).

When a user says something similar to a training phrase, Dialogflow matches it to the intent. You don't have to create an exhaustive list. Dialogflow will fill out the list with similar expressions. To extract parameter values, use [annotations](#) with available [system](#) or [custom](#) entity types.

” Add user expression

” Vehicle registration appointment for Monday

” Set an appointment at 2 PM on Monday for license

” License renewal appointment on Monday

” Set an appointment for driver license for 3 PM tomorrow

” Can you book an appointment at 6pm on Dec 21

Entities

Specifics in the user's input that the chatbot can provide different response to a particular intent.

Adding values and synonyms to entities helps the chatbot learn and understand important details that the users mention.

AppointmentType

SAVE



☒ Define synonyms  ☐ Regexp entity  ☐ Allow automated expansion ☐ Fuzzy matching 

Vehicle registration	Vehicle registration, registration, car registration, register
Driver license	Driver license, DL, license, test, driving test, test ID
Others	Others, something else, none
Click here to edit entry	

Contexts

Contexts are used to control the flow of conversation. Using contexts, you can handle end-user expression like “what is **its** price?” or “where are **they**?”. It needs to be provided with context in order to correctly match an intent.,

Contexts ?



car-attribute-set (X) Add input context

1 car-attribute-set (X) Add output context



Some Tips before we design, build and deploy chatbots

- Start on paper and have a conversation with a friend (discover intents, entities and create dialog).
- Keep it natural – let the users digress and come back to finish the current conversation (digression).
- Remember details about the users (context).
- Enable Fuzzy logic (think more like a human).

Deploy Chatbots to Various Channels



WhatsApp with Twilio

Make the assistant available to customers by adding it to WhatsApp



Slack

Make the assistant available to customers by adding it to a Slack app as a bot user



Facebook Messenger

Make the assistant available to customers through Facebook Messenger on the web or on native mobile clients



SMS with Twilio

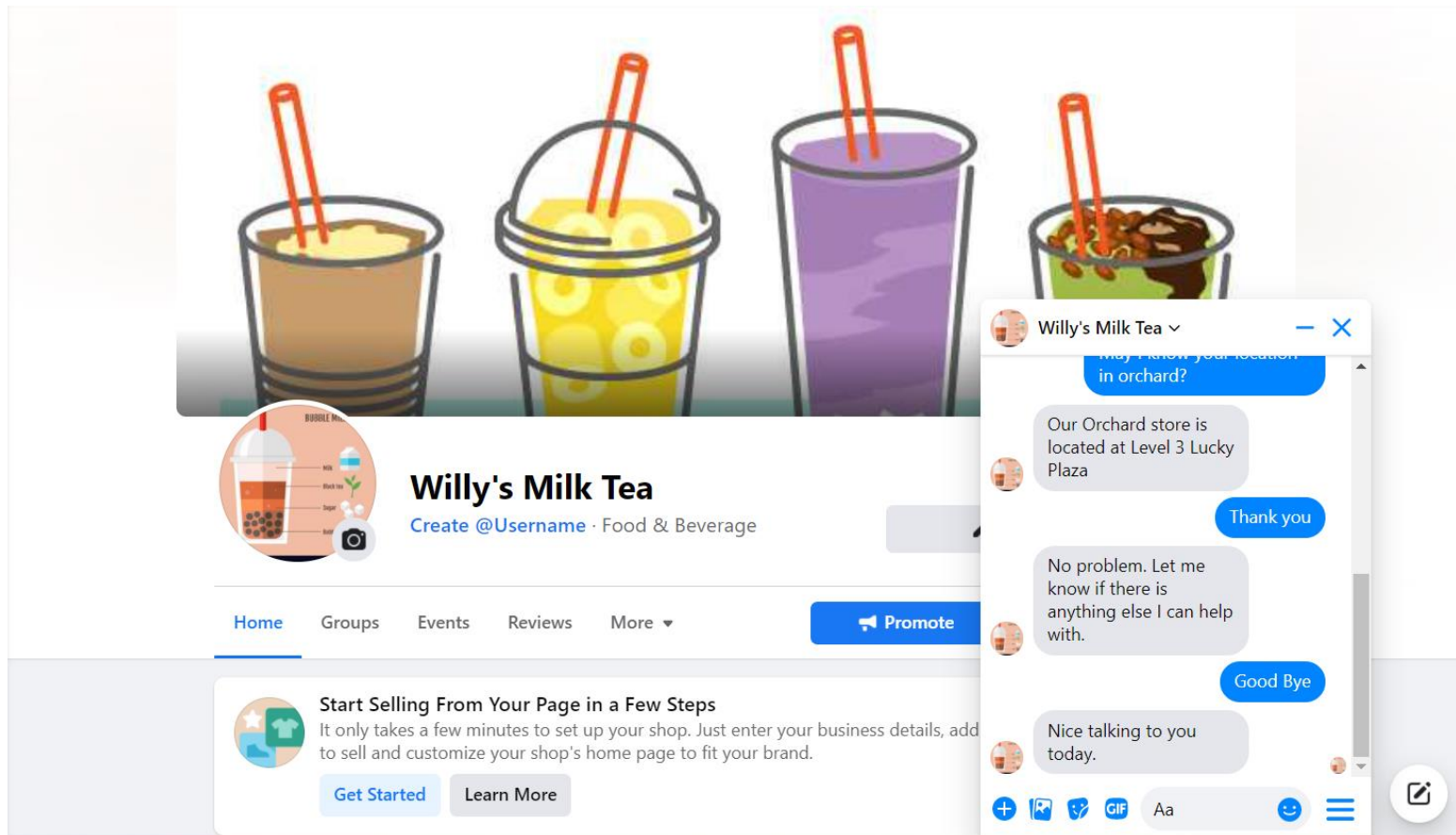
Send and receive SMS text messages



Intercom

Connect the assistant to an Intercom app so it can help human agents address customer queries

Deployment



Time To Exercise



Lab Time

Plan for Today

- Finish Topic 5
- Finish Lab 5: exercise 4-1 to 4-3
- Quiz

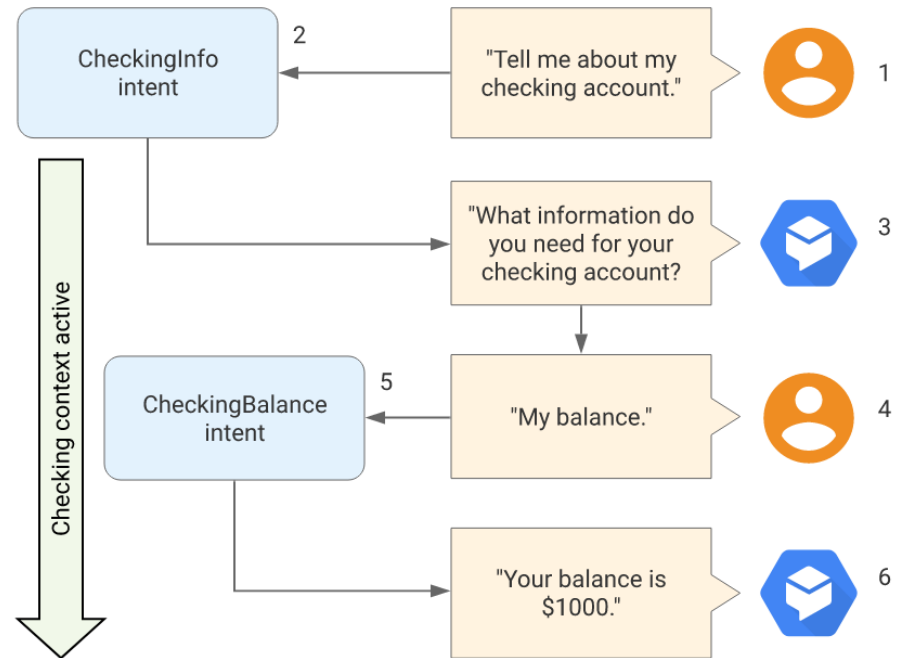
Define Contexts in Chatbot

AI HUMAN INTERFACE

Contexts

- **Contexts:** Dialogflow contexts are similar to natural language context. If a person says to you "they are orange", you need context in order to understand what "they" is referring to. Similarly, for Dialogflow to handle an end-user expression like that, it needs to be provided with context in order to correctly match an intent.
- Using contexts, you can control the flow of a conversation. When an intent is matched, any configured output contexts for that intent become active. While any contexts are active, Dialogflow is more likely to match intents that are configured with input contexts that correspond to the currently active contexts.

Contexts



1. The end-user asks for information about their checking account.
2. Dialogflow matches this end-user expression to the **CheckingInfo** intent. This intent has a checking output context, so that context becomes active.
3. The agent asks the end-user for the type of information they want about their checking account.
4. The end-user responds with "my balance".
5. Dialogflow matches this end-user expression to the **CheckingBalance** intent. This intent has a **checking** input context, which needs to be active to match this intent. A similar **SavingsBalance** intent may also exist for matching the same end-user expression when a savings context is active.
6. After your system performs the necessary database queries, the agent responds with the checking account balance.

Advantages of using Context

Don't repeat yourself

While conversing about a particular topic with our chatbot, you don't need to repeatedly mention the topic. Just use 'it' and the chatbot will understand what 'it' is referring to. The bot understands the context of your conversation.



Advantages of using Context

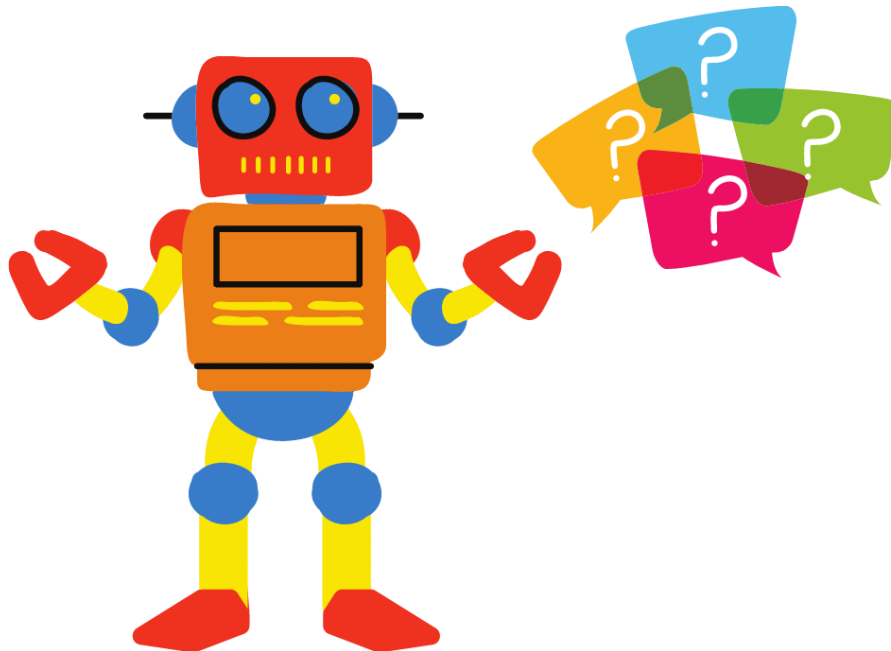
More natural conversation

Similar to how humans converse with each other, they don't mention the topic of the conversations repeatedly. They understand the topic and the context of their conversation. Our chatbot's ability to recognize the context of conversation makes talking to the bot feel natural.



How to make your chatbot more advanced

Good Fallback is the key!



Fallback Node

We will have a fallback node at the bottom, it ensures that we always have a reply for the user.

The fallback interaction controls the conversation flow by showing a general message when the bot cannot match the user query with any of the interaction. The interaction prompts the user suggesting rewording the phrase, showing possible options or performing specific actions.



Do you have Hojicha Milk Tea?



Says “I am sorry, but I do not understand what you are talking about”

Fallback Node = Not Smart?

Fallback node does not mean our chatbot is not smart enough to understand human language, but it is due to the lack of confidence in the current state of artificial intelligence and natural language processing.



How to Design Fallback



Do you have Hojicha Milk Tea?



Says “Opps! I didn’t get that”

- Imagine, we’ve likely received generic responses like this when the chatbot does not understand us.
- If our bot fails, try to **divert the user’s focus** from the failure event and **restore some delight** to the user’s experience.

Creative Fallback Messages

- Consider multiple fallback messages and rotate them randomly, so that any misunderstanding doesn't seem like an error 404.
- If our fallback messages are creative enough, we might be able to delight our users unexpectedly.

404
Not Found



Do you have Hojicha Milk Tea?



“Hmmm... I’ve never heard that before. Not sure how to respond to that 😊”

“ Would you like to inquire about our store locations?”

Gracefully Refocus Users

- If the communication breakdown with users occurs in the middle of the conversation, we can serve up two messages in succession.
- The first is the fallback message, the second can be a reminder. In this sense, we can move the focus back to the goals and not let the misunderstanding becomes a distraction in user experience.



Do you have Hojicha Milk Tea?



“Hmmm... I’ve never heard that before. Not sure how to respond to that 😊”

“ Would you like to inquire about our store locations?”

Tastefully Redirect Users

- A good strategy to tell the users right at the beginning of a conversation if they get stuck, they can always just ask for help.
- When our bot receives a request it doesn't understand, we can serve up the creative fallback message, followed by a second reminder that help is one input away.



Do you have Hojicha Milk Tea?



“Hmmm... I’ve never heard that before. Not sure how to respond to that 😊”

“ If you want to talk to a customer service, say ‘yes’.”

Summary

- Learn to deploy a simple chatbot to Facebook Messenger.
- Learn to use context variable to store information and maintain the state of a conversation.