

# **Building User Interfaces**

# **Dialogflow 1**

## **An Introduction**

### **Professor Bilge Mutlu**

# What we will learn today?

- Introduction to Conversational Interface Technologies
- Introduction to Dialogflow
- Dialogflow Building Blocks, Part 1
- Let's Make an Agent
- Assignment Preview

# Introduction to Conversational Interface Technologies

# What is a conversational interface?

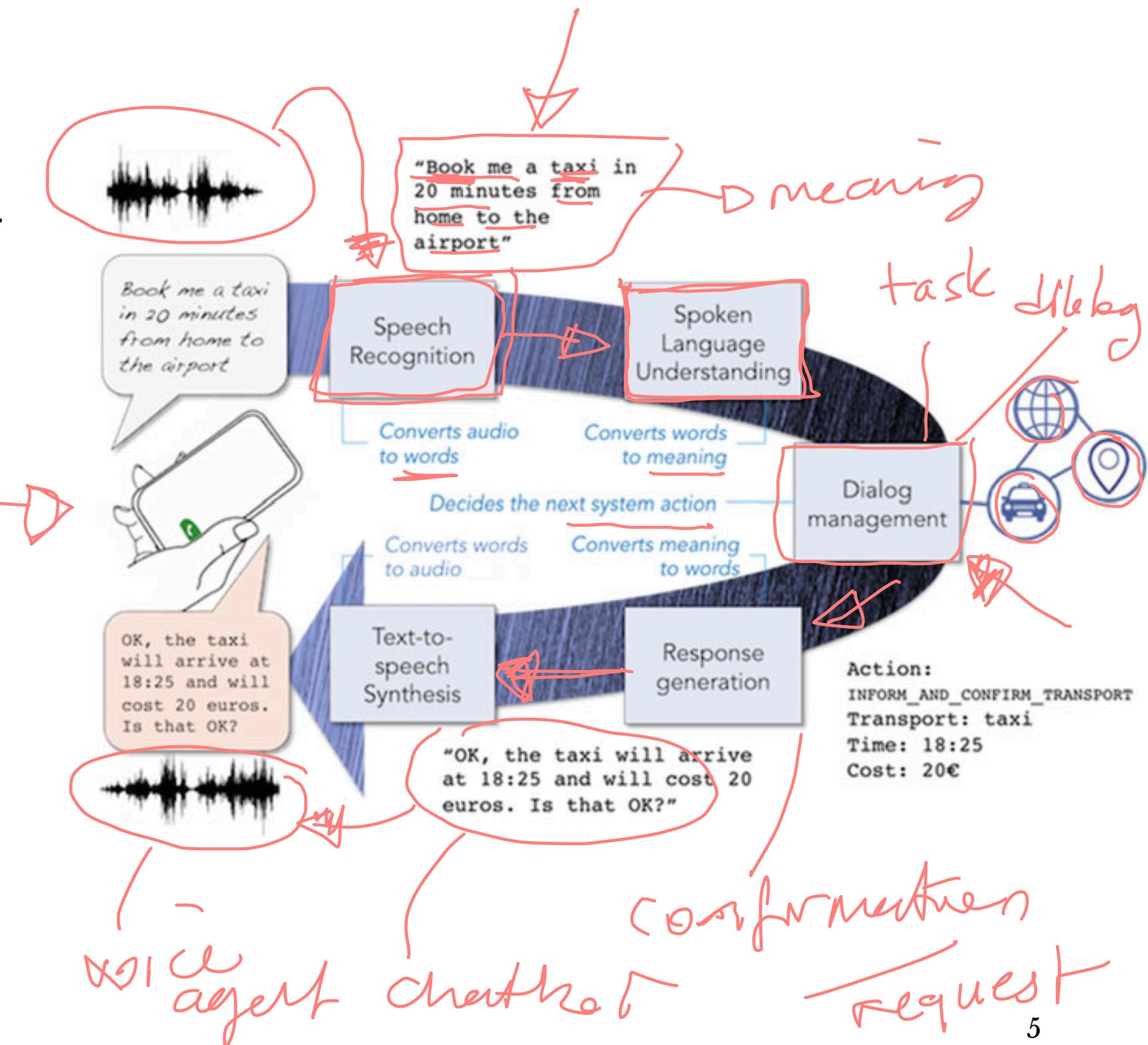
**Definition:** "Technology that supports conversational interaction with these VPAs [virtual personal assistants] by means of speech and other modalities."<sup>1</sup>

<sup>1</sup> McTear et al., 2016, *The Conversational Interface*

# Components of Conversational Interfaces<sup>2</sup>

Conversational interfaces integrate a number of technologies:

- Speech recognition
- Spoken language understanding
- Dialog management
- Response generation
- Text-to-speech synthesis



<sup>2</sup>Image source: McTear et al., 2016, *The Conversational Interface*

# The First Conversational Interface<sup>3</sup>

ELIZA was the first conversational interface developed by Joseph Weizenbaum at the MIT Artificial Intelligence Laboratory in 1964-1966.

```
Welcome to

      EEEEEEE  LL   IIII  ZZZZZZZZ  AAAAAA
      EE       LL   II    ZZ   AA   AA
      EEEEEEE  LL   II    ZZZ  AAAAAAAA
      EE       LL   II    ZZ  AA   AA
      EEEEEEE  LLLLLL  IIII  ZZZZZZZZ  AA   AA

Eliza is a mock Rogerian psychotherapist.
The original program was described by Joseph Weizenbaum in 1966.
This implementation by Norbert Landsteiner 2005.

ELIZA: Is something troubling you ?
YOU: Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU: They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU: Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU: He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU: It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU: |
```

<sup>3</sup> Image source: [Wikipedia: ELIZA](#)

# Types of Conversational Interfaces

- Spoken dialog systems
- Voice user interfaces
- Embodied conversational agents
- Social robots
- Chatbots



Customer service  
Alexa, Google Home "always on"  
video game  
characters.

# Platforms for Building Conversational Interfaces

In this course, we will be using Dialogflow by Google.

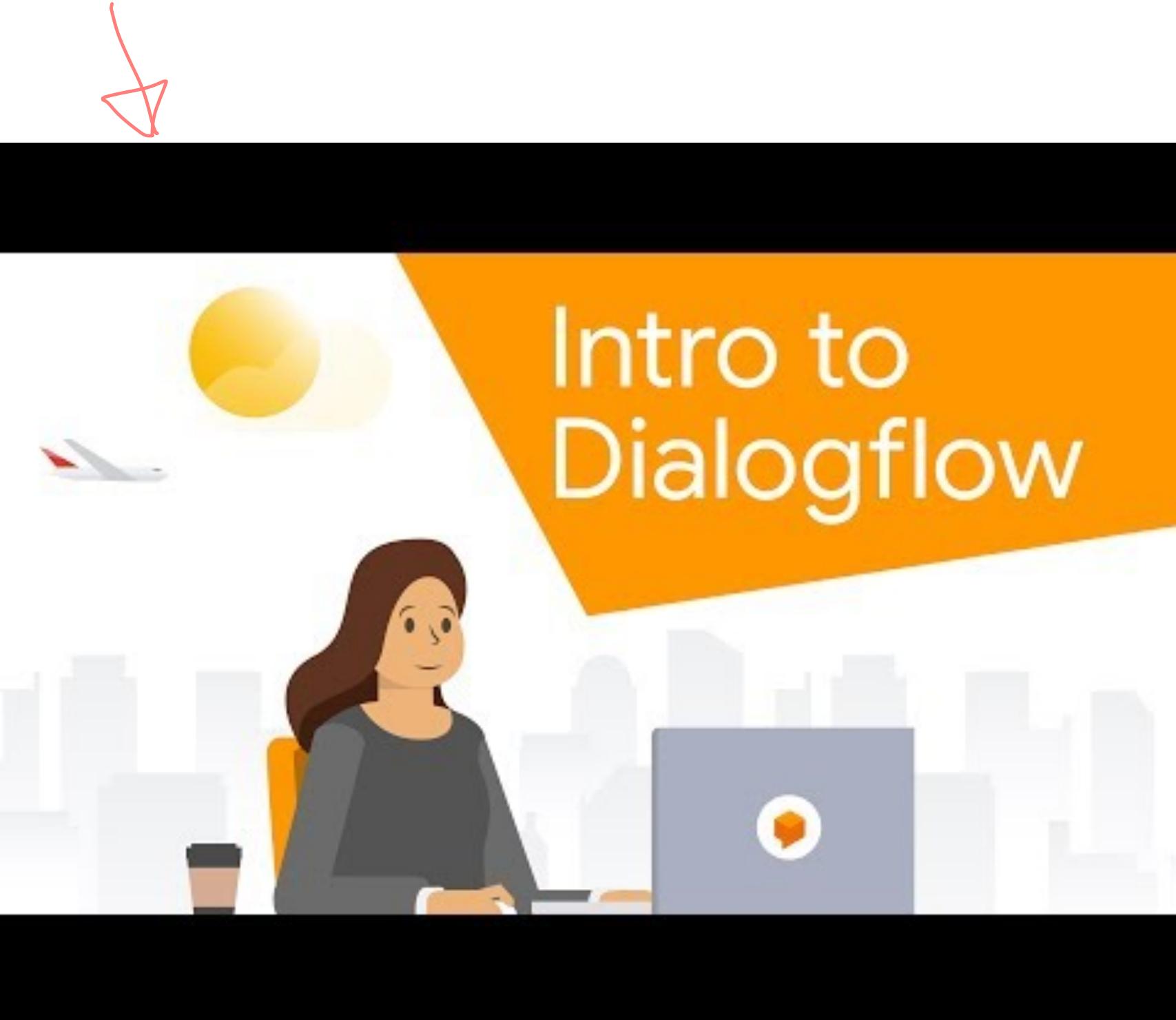
However, there are alternatives:

- Amazon Lex (Commercial) ←
- Mycroft (Open Source) ←
- CoreNLP (Academic) ← } free )

# Introduction to Dialogflow

# What is Dialogflow?

*Dialogflow* is an end-to-end, build-once deploy-everywhere development suite for conversational interfaces for websites, mobile applications, and IoT devices (e.g., smart speakers).<sup>4</sup>

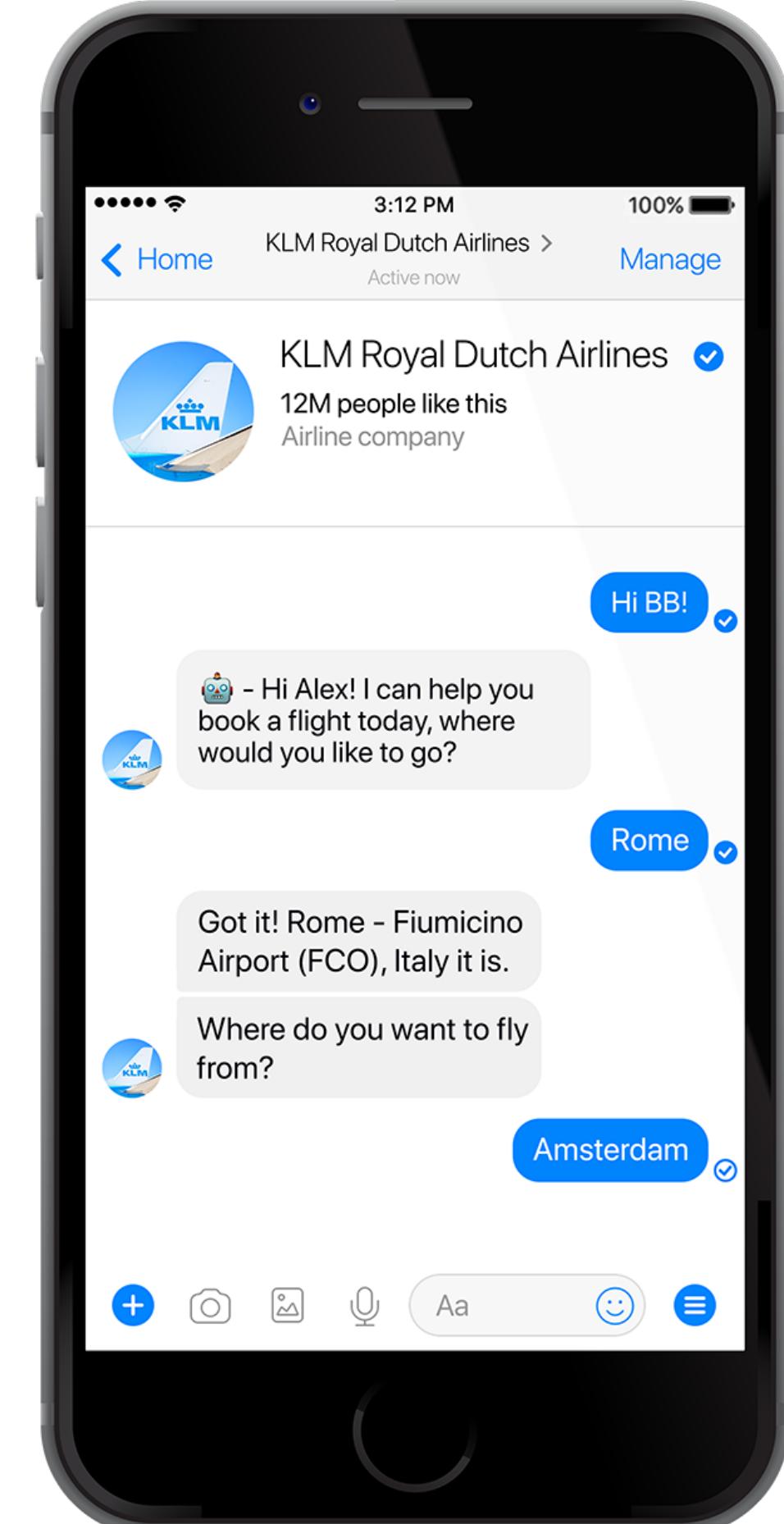
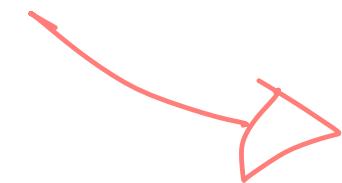


<sup>4</sup> Video source [Youtube](#)

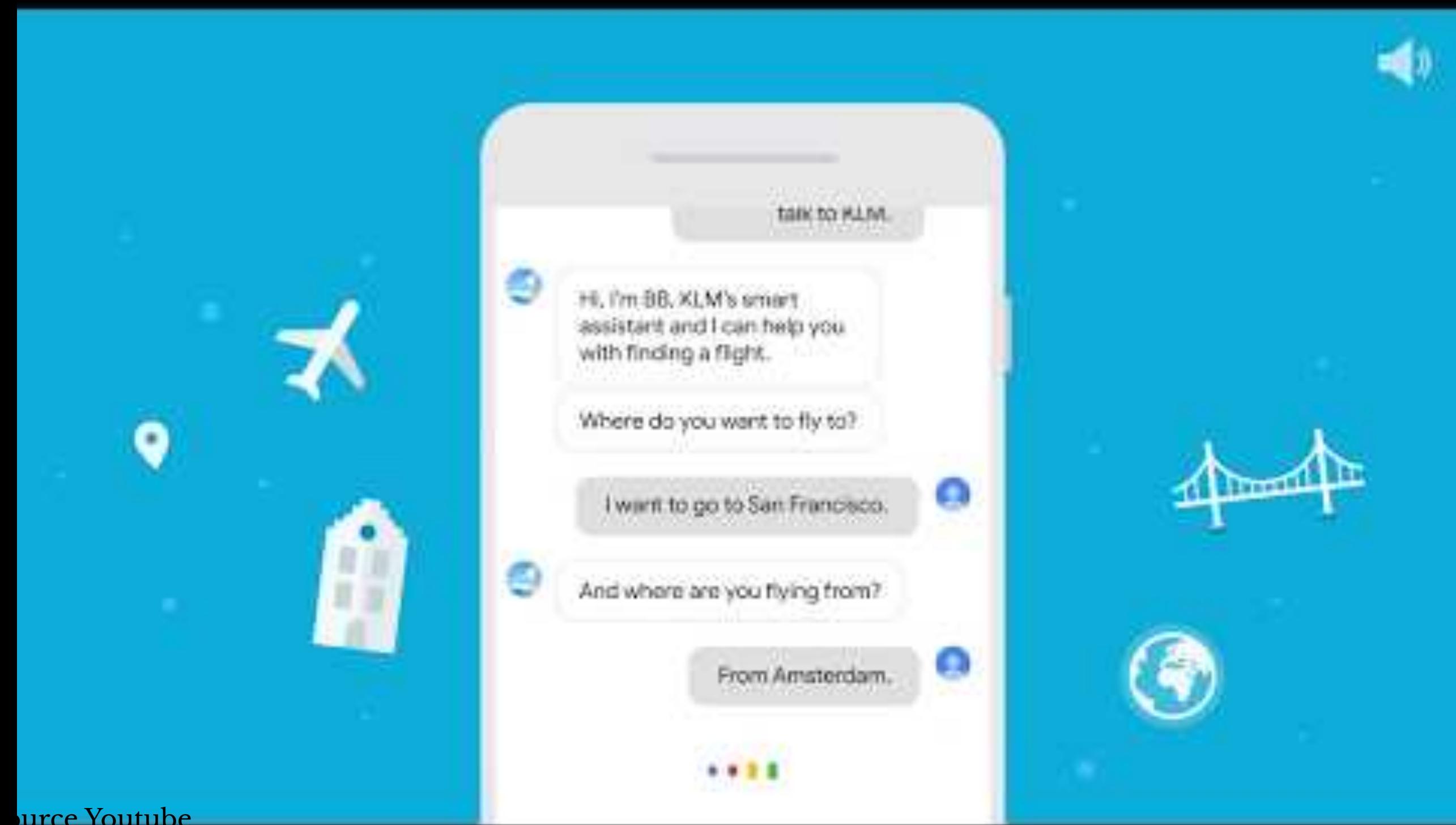
# Case Study: KLM BB<sup>5</sup>

KLM used Dialogflow to create an agent to purchase travel as well as travel preparation.

Let's see how the KLM BB works...



<sup>5</sup> Image source [Dialogflow](#)



Source Youtube

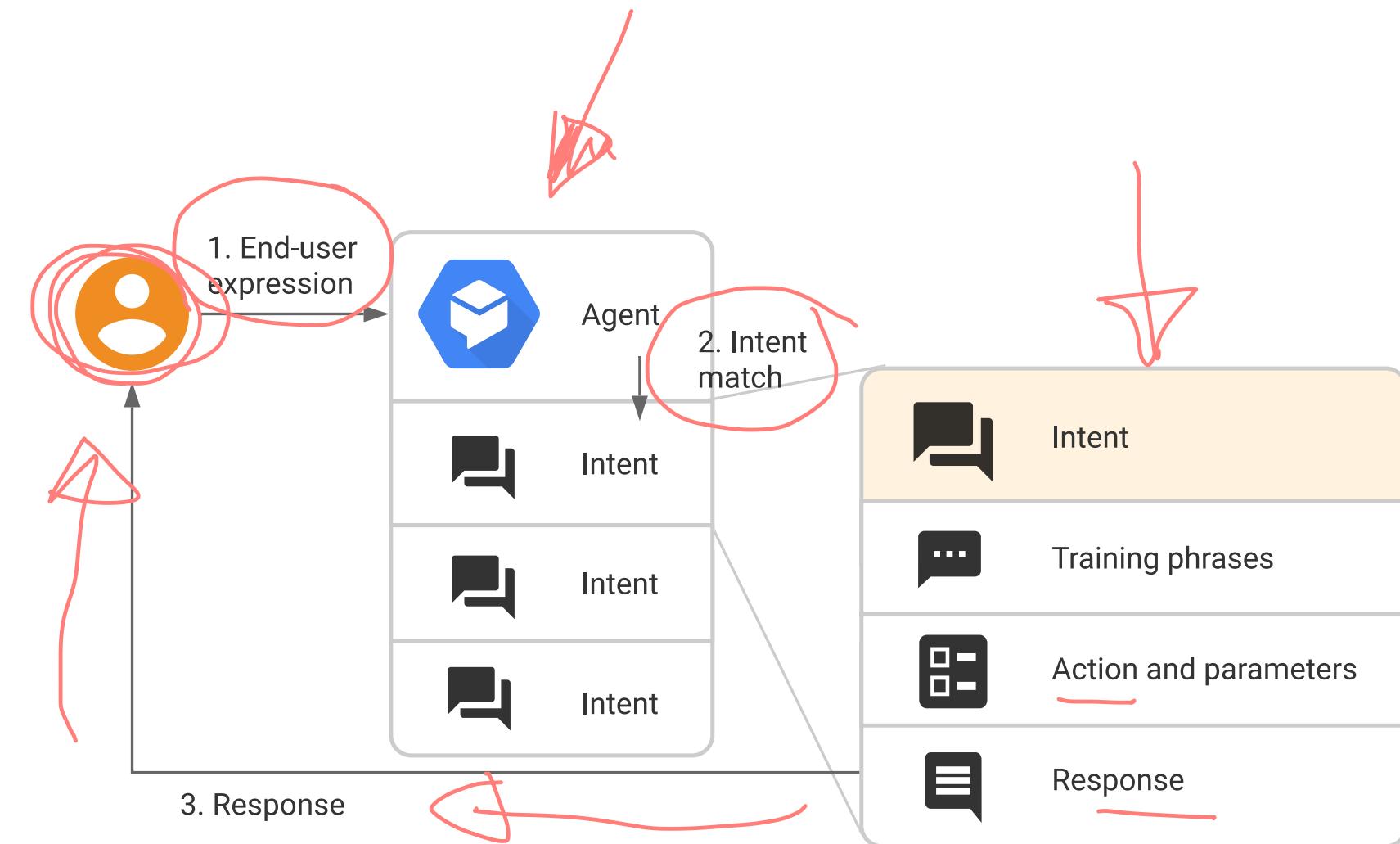


urce

# How does Dialogflow work?<sup>8</sup>

The process within Dialogflow involves:

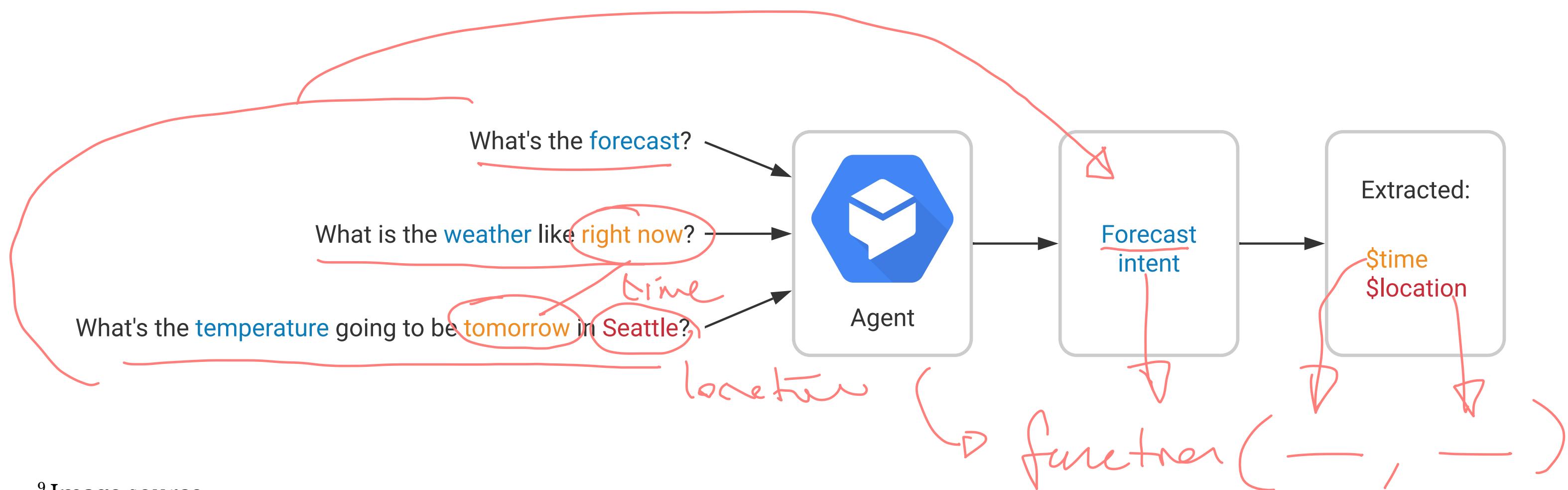
1. User expression
2. Intent matching
3. System response



<sup>8</sup>[Image source](#)

# What is an agent?

**Definition:** A Dialogflow agent is a virtual agent that handles conversations with users (similar to a human call agent).<sup>9</sup>



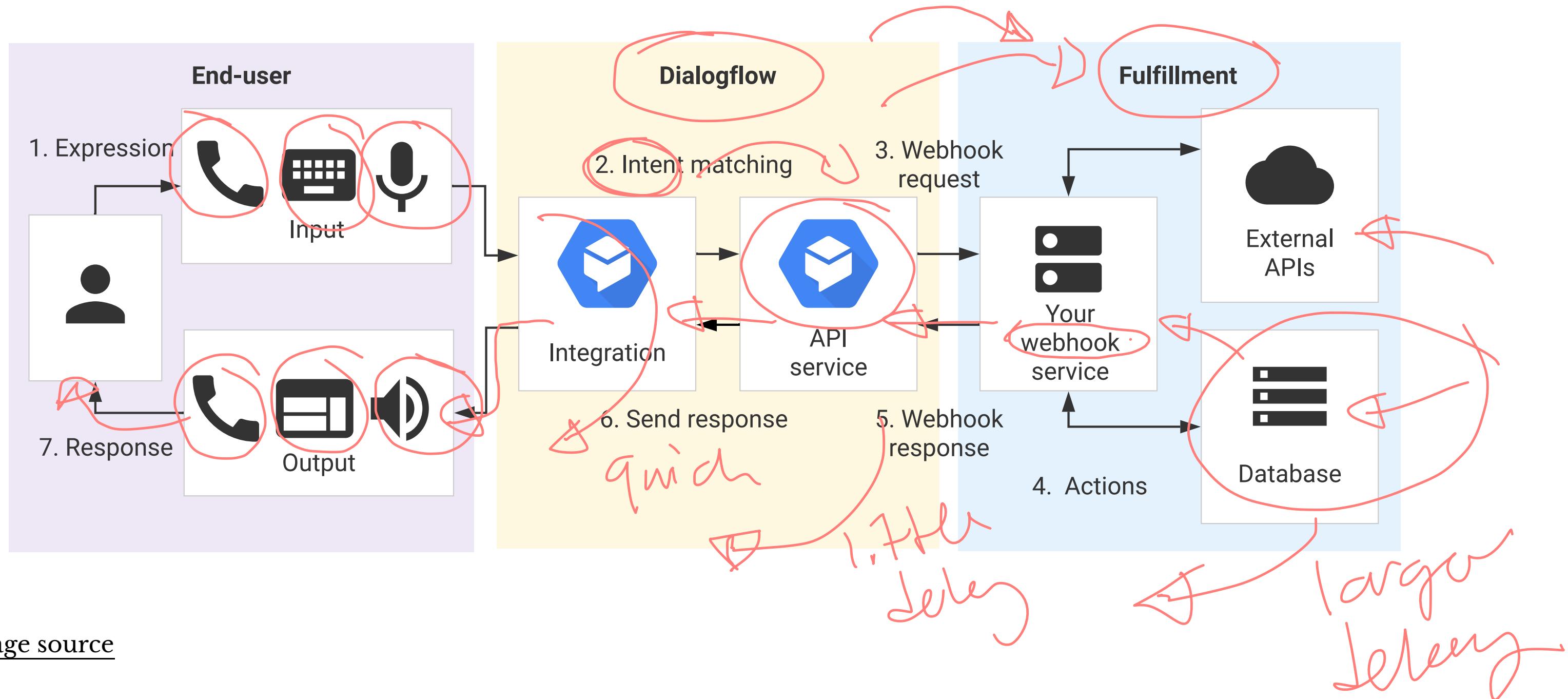
<sup>9</sup> Image source

Agents are high-level containers for a number of building blocks:

- Agent settings
- Intents
- Entities
- Knowledge
- Integrations
- Fulfillment

today  
Wed

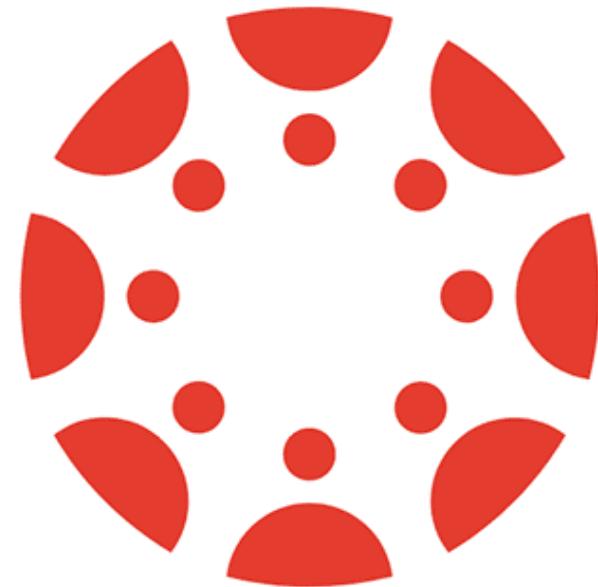
# The End-to-end Dialogflow Workflow<sup>10</sup>



<sup>10</sup> [Image source](#)

# Quiz 1

Complete the Canvas quiz.



canvas

# Dialogflow Building Blocks, Part 1

We will cover Part 2 in the next class.

wed.

# Getting Started with Dialogflow

1. Login to the Dialogflow <
2. Go to the Dialogflow console <
3. Create a new *agent* <

# Agent Settings

Get familiar with agent settings.

The screenshot shows the 'General' tab of the CS639DemoAgent settings page. Several UI elements are annotated with red arrows:

- A red arrow points to the orange icon in the top-left corner of the main content area.
- A red arrow points to the 'Default Time Zone' dropdown menu, which is set to '(GMT-6:00) America/Chicago'. A checkmark is placed next to this entry.
- A red arrow points to the 'Project ID' field, which contains 'cs639demoagent-jmmvnw'.
- A red arrow points to the 'Service Account' field, which contains 'dialogflow-cpgcaj@cs639demoagent-jmmvnw.iam.gserviceaccount.com'.
- A red arrow points to the 'Client access token' field, which contains '85c321bebfb844e1bbe1732b6d1419b8'.
- A red arrow points to the 'Developer access token' field, which contains '4aa35bdac21b402fab22caeef2f675662'.
- A red circle highlights the 'Log interactions to Dialogflow' toggle switch, which is turned on. Below it, a note says 'Collect and store user queries. Logging must be enabled in order to use Training, History and Analytics.'
- A red arrow points to the 'Delete Agent' button at the bottom right of the page.

At the top right, there is a 'Try it now' button and a microphone icon. On the far right, there is a note: 'Please use test console above to try a sentence.' and a link to 'See how it works in Google Assistant'.

General Languages ML Settings Export and Import Speech Share Advanced

CS639DemoAgent

SAVE

DESCRIPTION  
Describe your agent

DEFAULT TIME ZONE  
(GMT-6:00) America/Chicago

GOOGLE PROJECT

Project ID	cs639demoagent-jmmvnw
Service Account	dialogflow-cpgcaj@cs639demoagent-jmmvnw.iam.gserviceaccount.com

API VERSION

V2 API  
Use Cloud API as default for the agent. Your webhook will receive and return V2 format messages.

BETA FEATURES

Enable beta features and APIs  
Be the first to get access to the newest features and latest APIs. ([Full V2-beta API reference](#))

API KEYS (V1)

Client access token	85c321bebfb844e1bbe1732b6d1419b8
Developer access token	4aa35bdac21b402fab22caeef2f675662

LOG SETTINGS

Log interactions to Dialogflow  
Collect and store user queries. Logging must be enabled in order to use Training, History and Analytics.

Log interactions to Google Cloud  
Write user queries and debugging information to Google Stackdriver.

DANGER ZONE

Delete Agent  
Are you sure you want to delete agent CS639DemoAgent? This will destroy the agent with all corresponding data and cannot be undone!

DELETE THIS AGENT

# Agent Exporting

The screenshot shows the Dialogflow interface for managing an agent named "RobotPlanner". The left sidebar lists various sections: RobotPlanner (selected), Intents, Entities, Knowledge [beta], Fulfillment, Integrations, Training, Validation [beta], History, Analytics, and Prebuilt Agents. The main area is titled "RobotPlanner" and shows tabs for General, Languages, ML Settings, Export and Import (which is underlined and highlighted with a red arrow), Speech, Share, and Advanced. Under the "Export and Import" tab, there are three buttons: "EXPORT AS ZIP" (Create a backup of the agent), "RESTORE FROM ZIP" (Replace the current agent version with a new one. All the intents and entities in the older version will be deleted.), and "IMPORT FROM ZIP" (Upload new intents and entities without deleting the current ones. Intents and entities with the same name will be replaced with the newer version.). A blue "SAVE" button is located at the top right. To the right of the main area, there is a "Try it now" section with a microphone icon and a note: "Please use test console above to try a sentence." Below that is a link: "See how it works in Google Assistant" with a blue arrow icon.

# Agent Speech

The screenshot shows the Dialogflow interface for a project named "RobotPlanner". The left sidebar lists various components: RobotPlanner (selected), en, Intents, Entities, Knowledge [beta], Fulfillment, Integrations, Training, Validation [beta] (checked), History, Analytics, Prebuilt Agents, and Small Talk. The main area is titled "RobotPlanner" and has tabs for General, Languages, ML Settings, Export and Import, **Speech**, Share, and Advanced. The "Speech" tab is active. It contains sections for "IMPROVE SPEECH RECOGNITION QUALITY" and "TEXT TO SPEECH". In the "TEXT TO SPEECH" section, "Enable Automatic Text to Speech" is checked. Below it, "Output Audio Encoding" is set to "16 bit linear PCM (signed, little-end...)" with a dropdown arrow. A red arrow points from the bottom of the "Output Audio Encoding" dropdown towards the "Agent Language" section. The "VOICE CONFIGURATION" section includes "Configure your agent's synthesized voice in the V2 API and Telephony integration.", "Agent Language" (set to "en (English)"), and a "Voice" dropdown. On the right, there is a "Try it now" button and a microphone icon, along with a note: "Please use test console above to try a sentence." and a link to "See how it works in Google Assistant".

# Intents

# What are intents?

Consider the following user requests:

- What is the weather like today?
- Will it rain sometime today?

*locater*

weather - general (today)

weather - rain - likelihood (today)

What is the intent of these requests?

# What are intents?

**Definition:** Intents are the goals of the user that are expressed to the agent.

In the previous examples, despite their different framing, the user was expressing a desire to know what the weather will be.

That is their *intent*.

## More Intent Examples

Let's look at some more requests and identify their intents:

- How are you? ↗ small talk, greeting
- How do I get to Middleton? ↗ directions
- What is the price of a basketball? ↗ price inquiry
- Buy one box of tissues from Amazon. ↗ purchase

# Intents In Dialogflow

The screenshot shows the Dialogflow interface for creating intents. A red arrow points from the left margin to the 'Intents' button in the sidebar. Another red arrow points to the 'CREATE INTENT' button at the top right of the main area. A red circle highlights the 'CREATE INTENT' button. Red underlines are placed under 'Default Fallback Intent' and 'Default Welcome Intent'. A red flag-like mark points to the 'No regular intents yet.' message.

Dialogflow

RobotPlanner en +

Intents + Entities + Knowledge [beta] Fulfillment Integrations Training Validation [beta] History Analytics

Intents

Search intents

Default Fallback Intent

Default Welcome Intent

No regular intents yet. [Create the first one.](#)

Intents are mappings between a user's queries and actions fulfilled by your software. [Read more here.](#)

Before you start, check out [Prebuilt Agents](#), a collection of agents developed by the Dialogflow team.

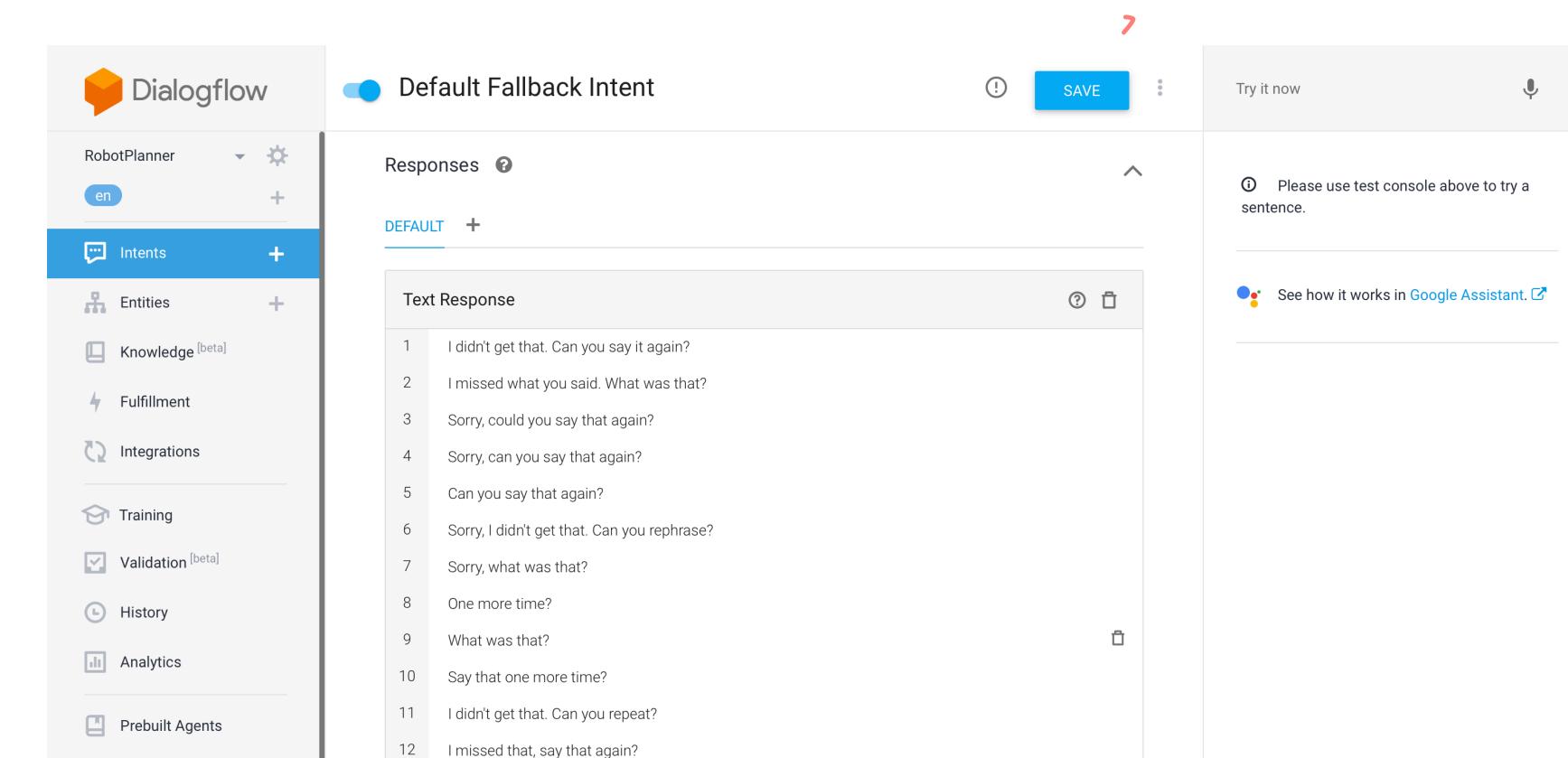
Try it now

Please use test console above to try a sentence.

Set-up Google Assistant integration.

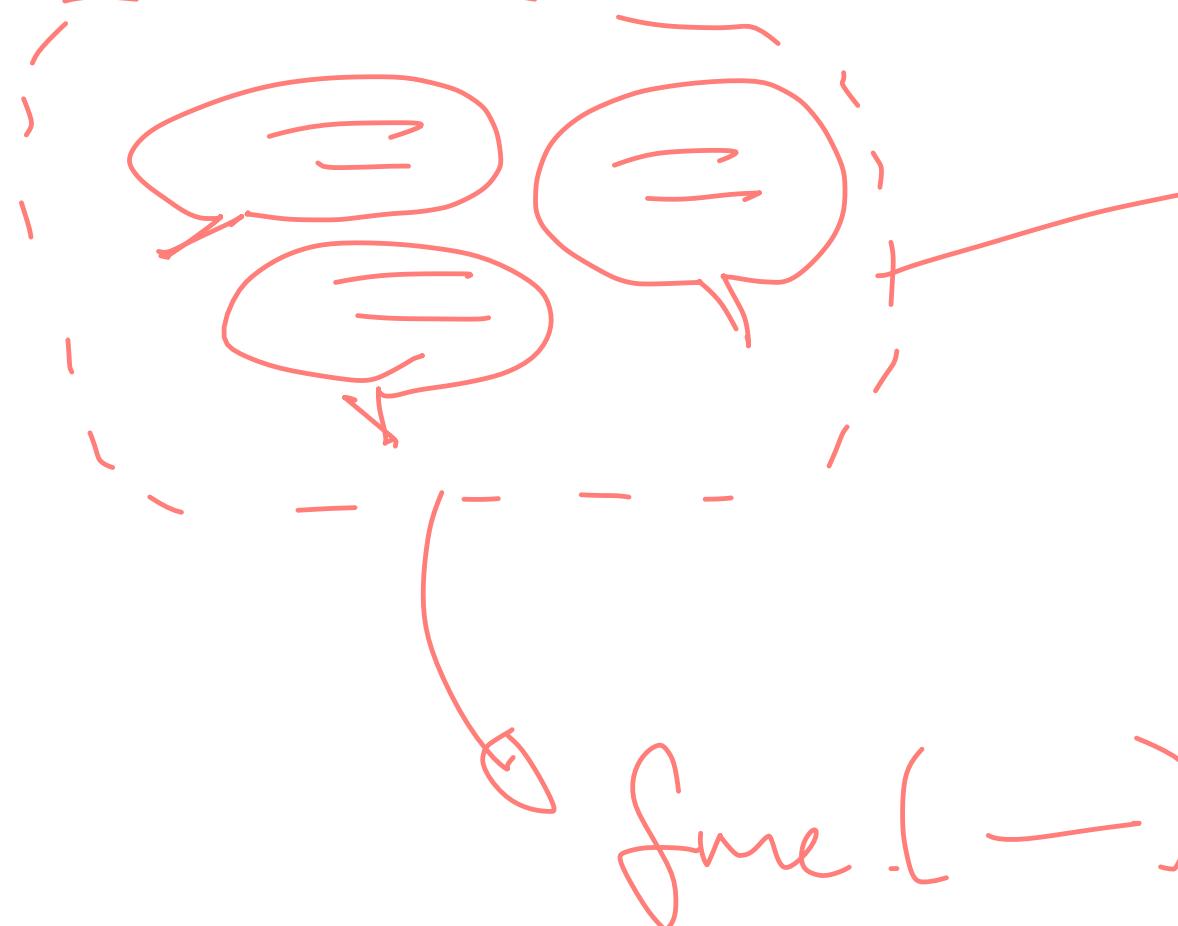
# Default Fallback Intents

- Engaged if no other intents are recognized.
- Could be a result of not being able to "hear" the user, or unable to parse what they said.
- Can provide training examples of things that will serve as negative examples for your desired intents.



# Creating Intents

- Allow the user to say that they want a robot to pick something up.
- We will start with training phrases.
- Should try to create at least 10.



Intent name SAVE

Contexts

Events

Training phrases

Train the intent with what your users will say

Provide examples of how users will express their intent in natural language. Adding numerous phrases with different variations and parameters will improve the accuracy of intent matching. [Learn more](#)

**ADD TRAINING PHRASES**

Action and parameters

Extract the action and parameters

Parameters are specific values extracted from a user's request when entities are matched. The values captured by parameters can be used in fulfillment, or in building a response. If you mark parameters as required, Dialogflow will prompt the user if their values were not extracted from their initial request. [Learn more](#)

**ADD PARAMETERS AND ACTION**

Responses

# Creating Intents

- Can you get the screwdriver for me?
- Please get the green ball.
- Pick up that red cube.
- etc.

Notice how the color is highlighted? More on that next.

The screenshot shows the Dialogflow interface for creating an intent named "Pickup Command". The intent is circled in red. Below it, a list of "Training phrases" is shown, each preceded by a double quotes icon. Some words in the phrases are highlighted in yellow and circled in red, corresponding to the underlined words in the list above. A search bar at the top right allows for searching training phrases. A blue "SAVE" button is in the top right corner, and three vertical dots indicate more options.

Training phrase
” Add user expression
” Take that.
” Grab this from me.
” Take this wrench
” Pick up the hammer over there.
” How about getting that box of screws for me?
” Can you get the screwdriver for me?
” Please get the green ball
” Get the green ball
” Grab the toy
” Pick up that red cube

# Test

Test your agent using the *Default Welcome Intent*

The screenshot shows the Dialogflow web interface. On the left, a sidebar lists various sections: CS639DemoAgent (selected), en, Intents (highlighted in blue), Entities, Knowledge [beta], Fulfillment, Integrations, Training, Validation [beta] (checked), History, Analytics, Prebuilt Agents, Small Talk, Docs (expanded), Standard (Free), Upgrade, Support, Account, and Logout.

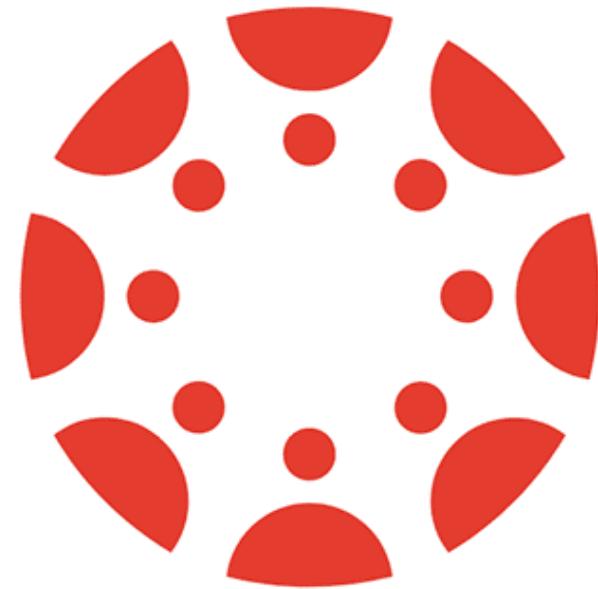
The main area is titled "Intents". It features a "CREATE INTENT" button and a search bar labeled "Search intents". Below the search bar, two intents are listed: "Default Fallback Intent" and "Default Welcome Intent". A red arrow points from the text "Test your agent using the Default Welcome Intent" to the "Default Welcome Intent" link.

A large red arrow points from the "Default Welcome Intent" link to the "Default Welcome Intent" section in the main content area. This section contains a message from the user "Hello!" and a response from the agent "Hi! How are you doing?". The "Hi! How are you doing?" message is circled in red.

The interface also includes sections for "USER SAYS" (Hello!), "INTENT" (Default Welcome Intent), "ACTION" (input.welcome), and "DIAGNOSTIC INFO". A "COPY CURL" button is located at the top right of the main content area.

# Quiz 2

Complete the Canvas quiz.



canvas

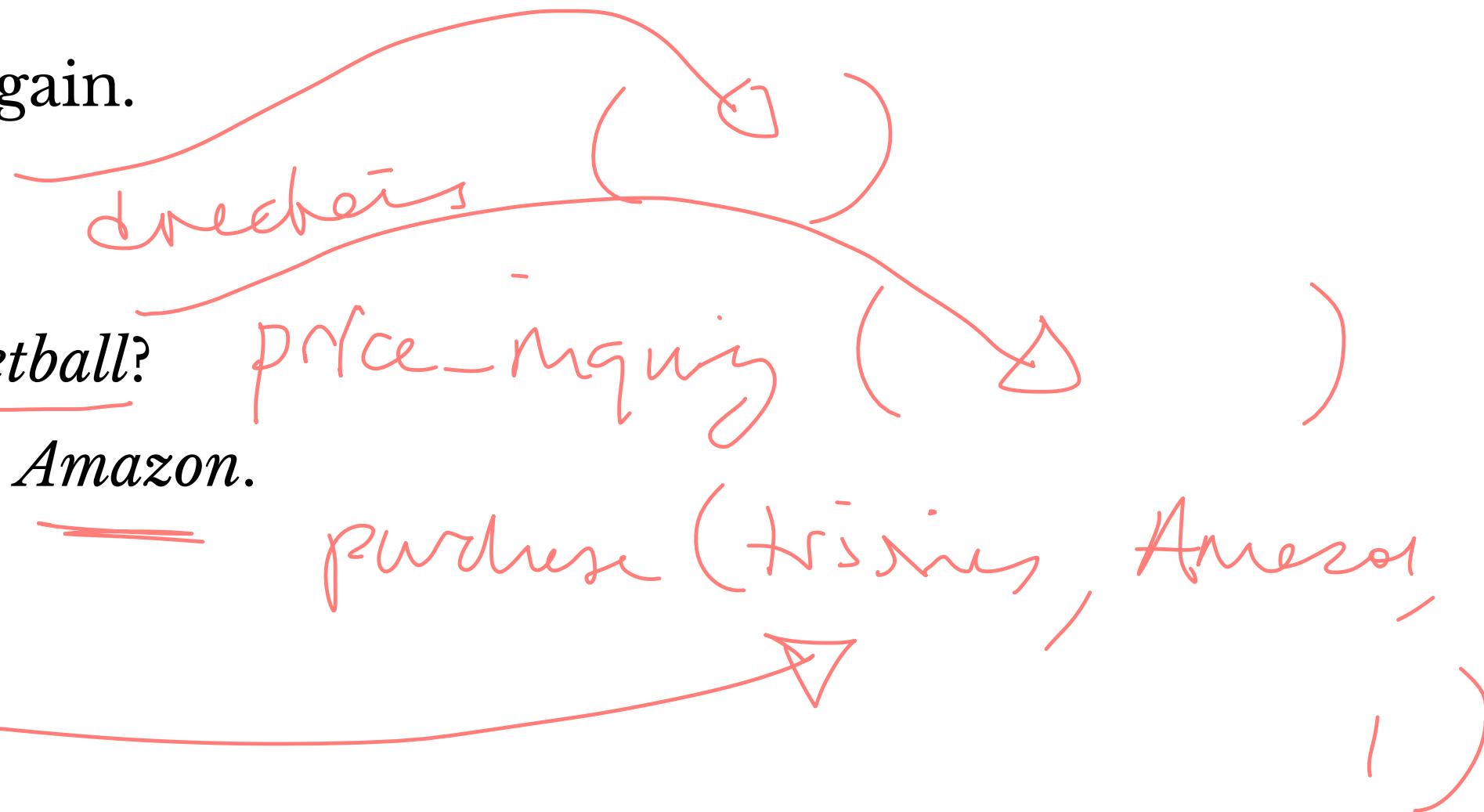
# Entities

---

# What are entities?

Let's consider those requests again.

- How do I get to Middleton?
- What is the price of a basketball?
- Buy *one* box of *tissues* from *Amazon*.



# What are entities?

Sometimes, users' intents are more specific, and have an intent based around a certain item or *entity*.

**Definition:** Entities allow for more specificity of requests, without exploding the intent space.

Thus, if the request was:

**What is the weather like today in Seattle?**

The *intent*: weather inquiry; *entity*: Seattle, *today*

# Entities in Dialogflow

Let's define some things that the robot can pick up.

The screenshot shows the Dialogflow Entities interface. At the top, there is a red annotation with the word "object" written in red cursive. Below it, the "Entity name" field contains "object". To the right of the entity name is a blue "SAVE" button, which is circled in red. The interface includes a sidebar with "RobotPlanner" (en), "Intents", "Entities" (selected), "Knowledge [beta]", "Fulfillment", "Integrations", "Training", "Validation [beta]" (checked), "History", and "Analytics". There are also checkboxes for "Define synonyms" (checked), "Regexp entity", "Allow automated expansion", and "Fuzzy matching". A note says "Separate synonyms by pressing the enter, tab or ; key." At the bottom, there are five rows of placeholder text "Click here to edit entry" and a "+ Add a row" button. On the right side, there are links for "Try it now", "See how it works in Google Assistant", and a note "Please use test console above to try a sentence."

# Entities in Dialogflow

Remember I mentioned some objects when creating my intents.  
Let's add those here.

- Cube
  - Sphere
  - Screwdriver
  - etc.
- 

# Entity Entries and Synonyms

- Cube (Box)
- Container (Box, Bin)
- Sphere (Ball)
- Screwdriver

The screenshot shows a user interface for managing entity entries and synonyms. At the top, the word "object" is circled in red. Below it, there are several checkboxes: "Define synonyms" (which is checked), "Regexp entity", "Allow automated expansion", and "Fuzzy matching". The main area displays a table of entities and their synonyms:

Entity	Synonyms
cube	cube
container	container, box, bin
sphere	sphere, ball
screwdriver	screwdriver
hammer	hammer, mallet
wrench	wrench

An input field labeled "Enter synonym" is shown next to the wrench row. A "Click here to edit entry" link is at the bottom. A blue "SAVE" button is in the top right corner. A "More options" menu icon is also present.

+ Add a row

# Tagging Entities in Intents

Entities can be explicitly tagged in intents, if they are not automatically detected.

PARAMETER NAME	ENTITY	RESOLVED VALUE
color	@sys.color	red
object	@object	cube

# Automated expansion

Allows dialogflow to extrapolate to new objects

object

object

SAVE ⋮

Define synonyms ?  Regexp entity ?  Allow automated expansion ?  Fuzzy matching ?

cube	cube
container	container, box, bin
sphere	sphere, ball
screwdriver	screwdriver
hammer	hammer, mallet
wrench	wrench

Click here to edit entry

+ Add a row

The screenshot shows a user interface for managing automated expansion rules. At the top right are 'SAVE' and '⋮' buttons. Below them are four checkboxes: 'Define synonyms' (checked), 'Regexp entity' (unchecked), 'Allow automated expansion' (checked and highlighted with a red oval), and 'Fuzzy matching' (unchecked). A table lists objects and their expanded forms. Red arrows point from the expanded forms ('box, bin', 'sphere, ball') back to their respective original objects ('container', 'sphere'). A red oval also surrounds the 'Allow automated expansion' checkbox. At the bottom is a link to 'Edit entry'.

# Required Entities

Suppose you want to require the user provide some entity. You can make it required, and specify how you want the agent to respond if it isn't provided.

Action and parameters ^

Enter action name

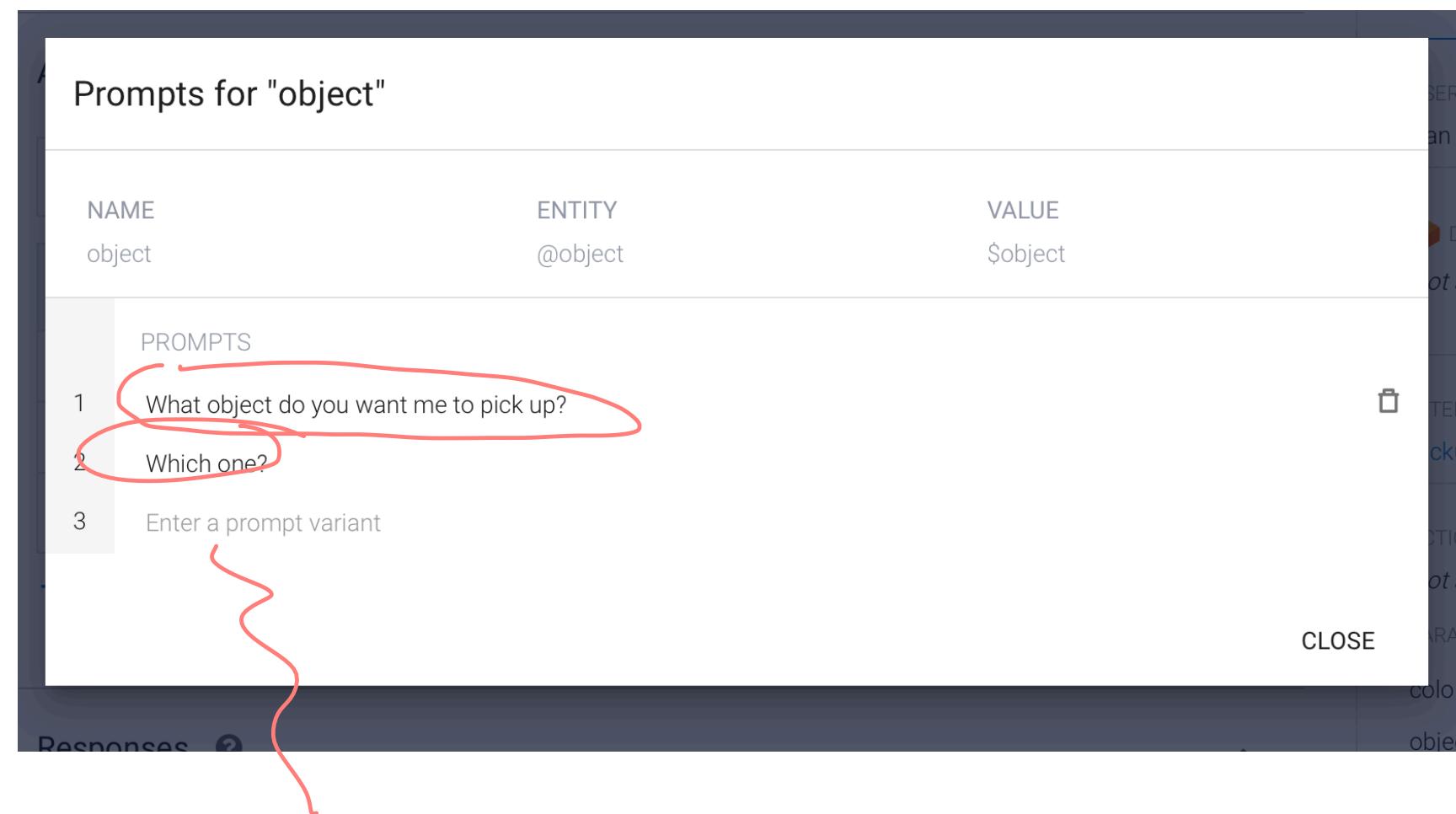
REQUIRED	PARAMETER NAME	ENTITY	VALUE	IS LIST	PROMPTS
<input type="checkbox"/>	color	@sys.color	\$color	<input type="checkbox"/>	-
<input checked="" type="checkbox"/>	object	@object	\$object	<input type="checkbox"/>	<span style="border: 1px solid red; padding: 2px;">Define prompt s...</span>
<input type="checkbox"/>	Enter name	Enter entity	Enter value	<input type="checkbox"/>	-

[+ New parameter](#)



# Specifying prompts

Specify in the modal what prompts to use to query the required entity.



# Tweaking your intents for required entities

Remove the entity from the prompt if it was too general. Then the agent will inquire if it gets that prompt.

Note: This could also be handled by context, which will be discussed next build class.

The screenshot shows a list of training phrases in Dialogflow. Red arrows point to several entities in the phrases, indicating they are being reviewed or modified. The entities are highlighted in pink, yellow, and purple boxes. The list includes:

- "Take that."
- "Grab this from me."
- "Take this wrench"
- "Pick up the hammer over there."
- "How about getting that box of screws for me?"
- "Can you get the screwdriver for me?"
- "Please get the green ball"
- "Get the green ball"
- "Grab the toy"
- "Pick up that red cube"

At the top right, there is a search bar labeled "Search training phrases" with a magnifying glass icon, and a back arrow icon.

# Dialogflow Documentation

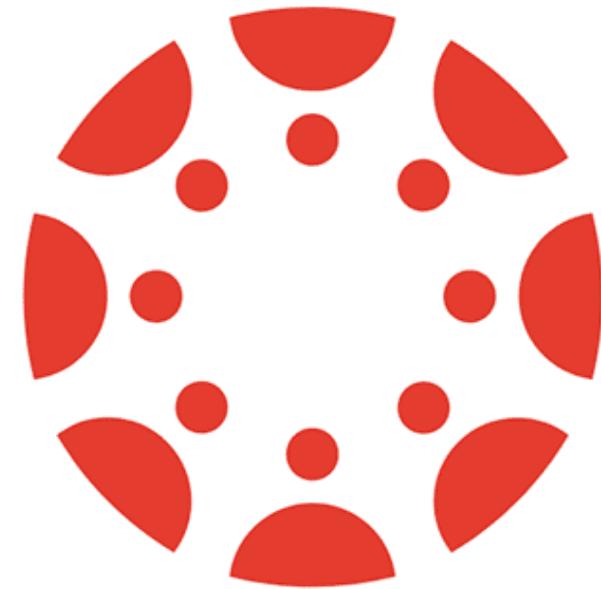
Full Documentation



# Let's Build An Agent

# Quiz 3

Complete the Canvas quiz.



canvas

# Assignment Preview

# Assignment Overview

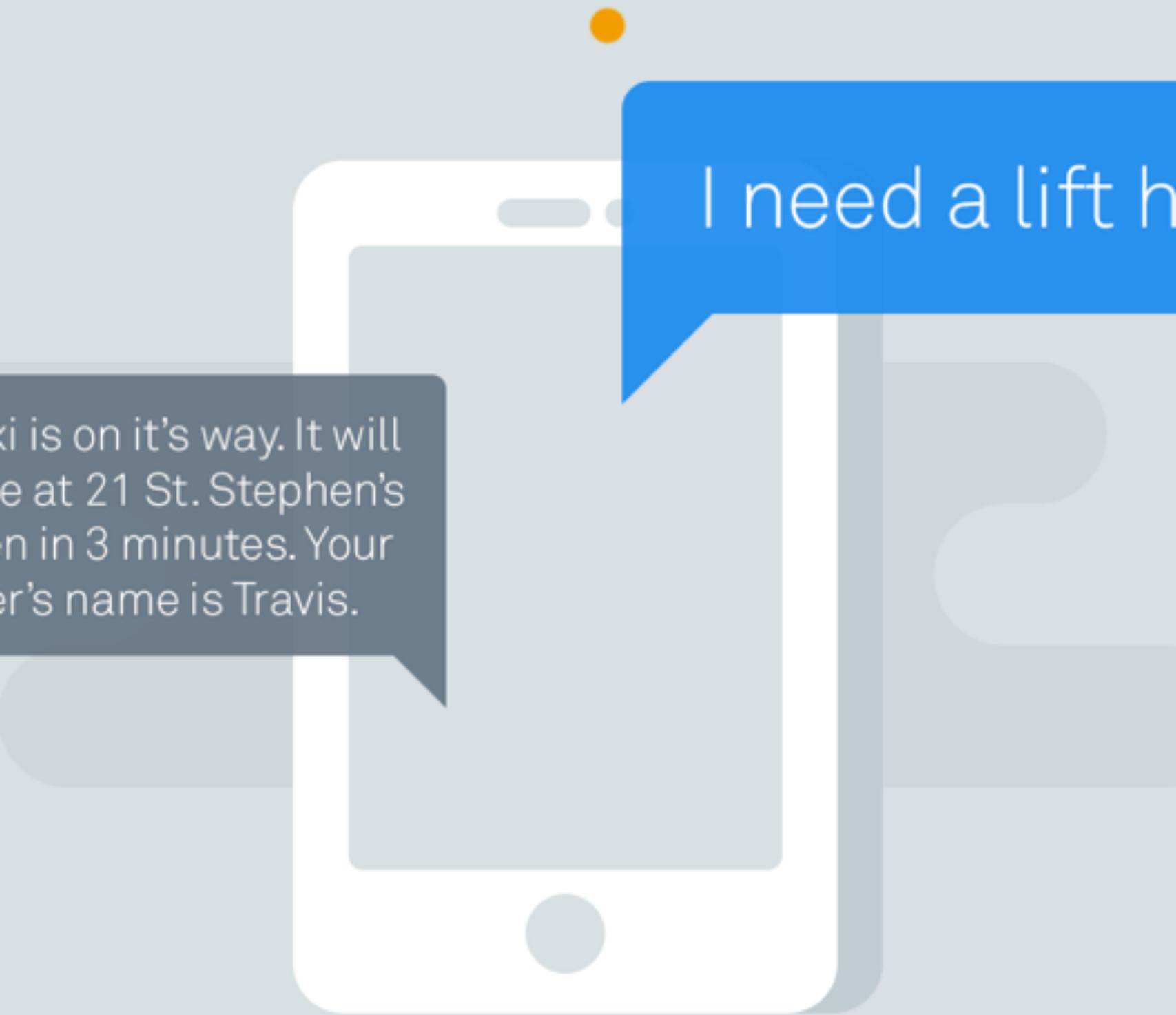
We will create a voice assistant for a fictional online clothing retail store, called *WiscShop*.

- Dialogflow  $\alpha$  — Develop specifications
- Dialogflow  $\beta$  — Implementation
- Dialogflow  $\gamma$  — User evaluation

# Dialogflow Alpha – Develop specifications<sup>11</sup>

Experience prototyping to develop specifications for the voice assistant.

- Study the WiscShop store system
- Bodystorm how the voice assistant can support a set of capabilities
- Extract Dialogflow intents, entities, etc.



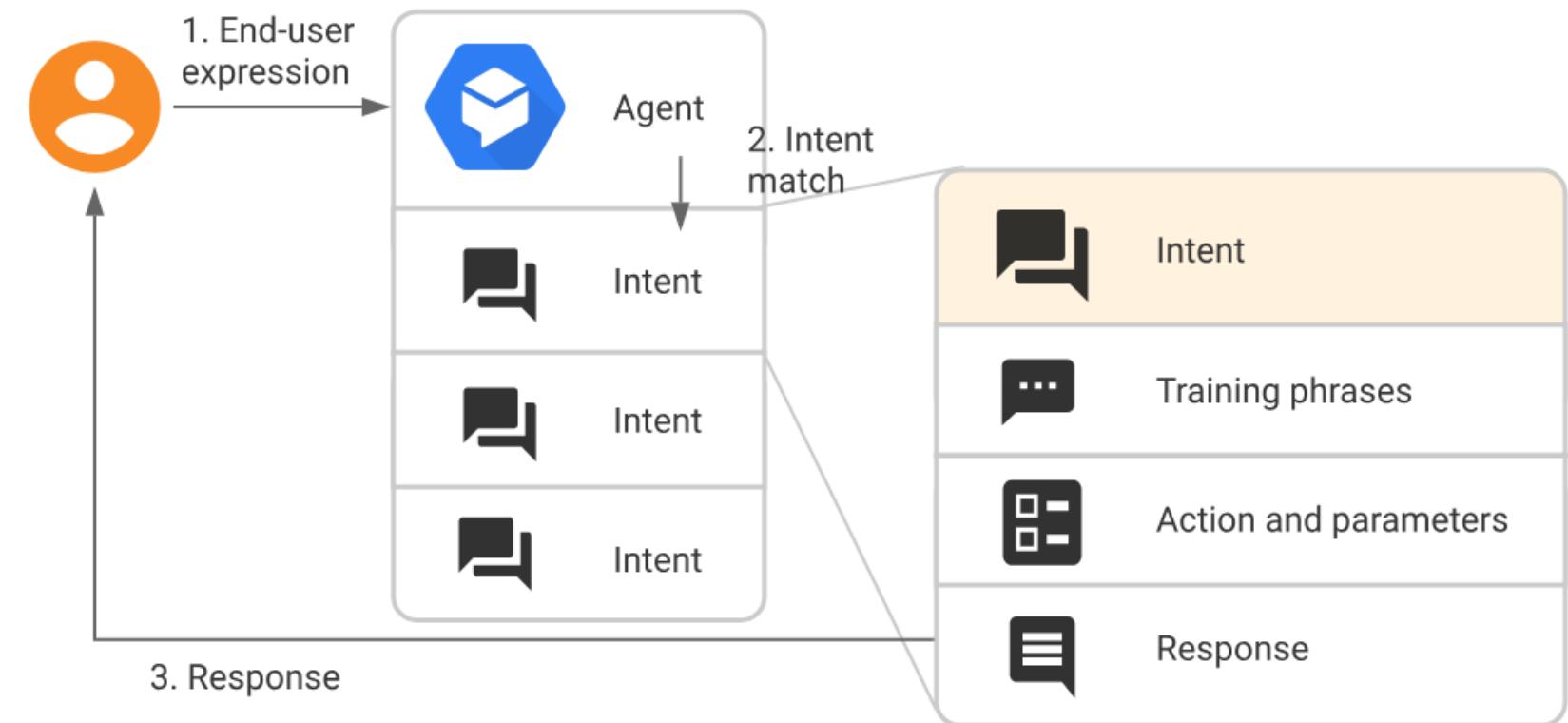
<sup>11</sup> [Image source](#)

# Capabilities

- Create a voice agent that allows a user to navigate a visual shopping interface.
- Allow the user to navigate from anywhere back to home, or to their cart.
- Allow the user to return to the previous page.
- Allow the user to specify the category of items they want to look at.
- Allow the user to specify attributes/tags to search in.
- Allow the user to navigate to product pages.
- Allow the user to add items (when on that product page) to their cart.
- Allow the user to go to checkout and complete the process.

# Dialogflow Beta – Implementation<sup>8</sup>

Implement the voice assistant agent using the specifications extracted in Dialogflow  $\alpha$ .



<sup>8</sup>[Image source](#)

# Server API

Route	Auth Required	Token Required	Get	Post	Put	Delete
/login	✓		✓			
/users				✓		
/users/ <username>		✓	✓	✓	✓	✓
/tags			✓			
/categories			✓			
/products			✓			
/products/ <product_id>			✓			
/products/ <product_id> /tags			✓			
/products/ <product_id> /reviews			✓			
/products/ <product_id> /reviews/ <review_id>			✓			
/application		✓	✓		✓	
/application/tags		✓	✓			✓
/application/tags/ <tag_value>		✓		✓		✓
/application/messages		✓	✓	✓		✓
/application/messages/ <message_id>		✓	✓		✓	✓
/application/products		✓	✓			✓
/application/products/ <product_id>		✓		✓		✓

# Pages

home  
category:<category>  
product:<product>  
cart-current  
cart-review  
cart-confirmed

# Dialogflow Gamma — User evaluation<sup>12</sup>

Design and perform a mini usability test over Zoom.

- Develop user study protocol.
- Recruit two volunteers.
- Administer the protocol.
- Analyze and report your findings.



<sup>12</sup> [Image source](#)

# What have we learned today?

- Introduction to Conversational Interface Technologies
- Introduction to Dialogflow
- Dialogflow Building Blocks, Part 1
- Let's Make an Agent <
- Assignment Preview —

