

Building User Interfaces

Dialogflow 1

An Introduction

Professor Yuhang Zhao

Final

- Final exam will be remote
- Dec 23, 12:25-2:25 pm
- Alternative only for extreme cases - we will send out a poll
- 60% (after midterm) vs. 40% (before midterm)
- More bonus quizzes

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?

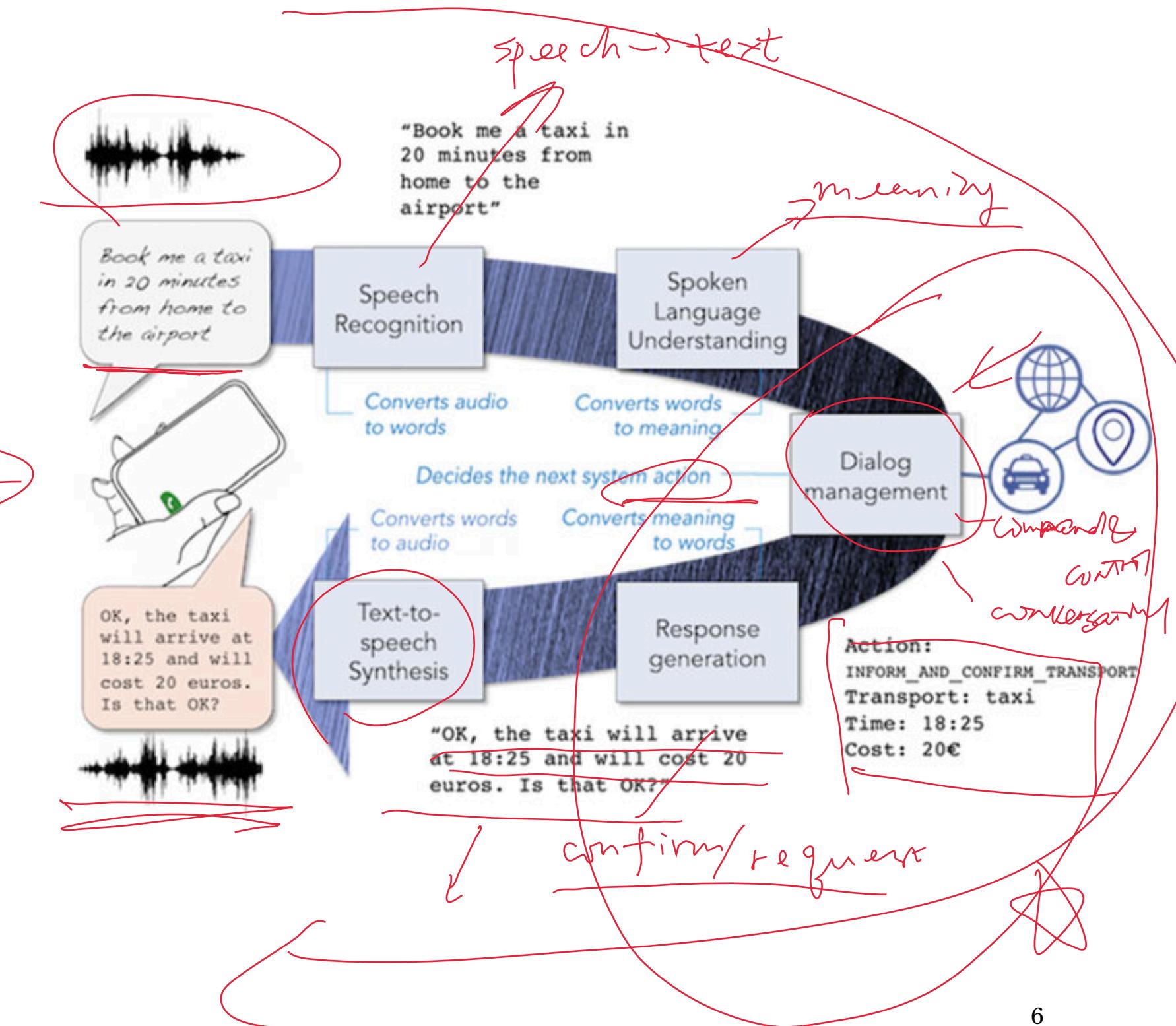
Definition: "Technology that supports conversational interaction with these VPAs [virtual personal assistants] by means of speech and other modalities."¹

¹ McTear et al., 2016, *The Conversational Interface*

Components of Conversational Interfaces²

Conversational interfaces integrate a number of technologies:

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



² Image source: McTear et al., 2016, *The Conversational Interface*

The First Conversational Interface³

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.

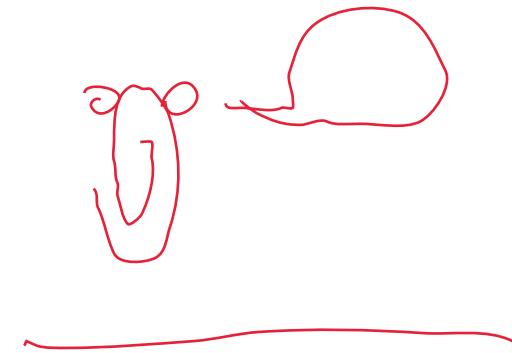
ELIZA: Is something troubling you ?
YOU: Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU: They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU: Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend 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: |
```



³ Image source: [Wikipedia: ELIZA](#)

Types of Conversational Interfaces

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



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)

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).⁴

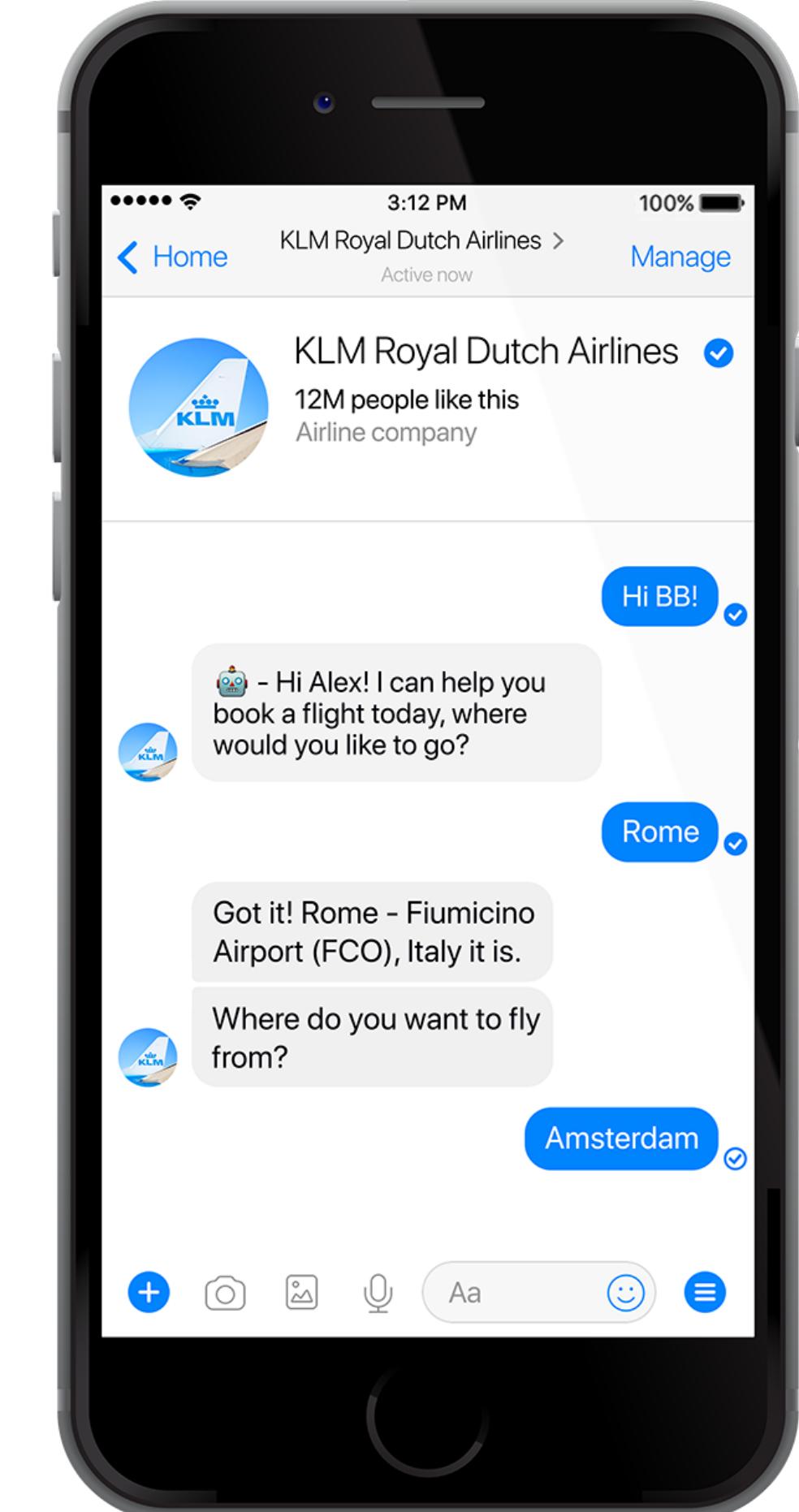


⁴ Video source [Youtube](#)

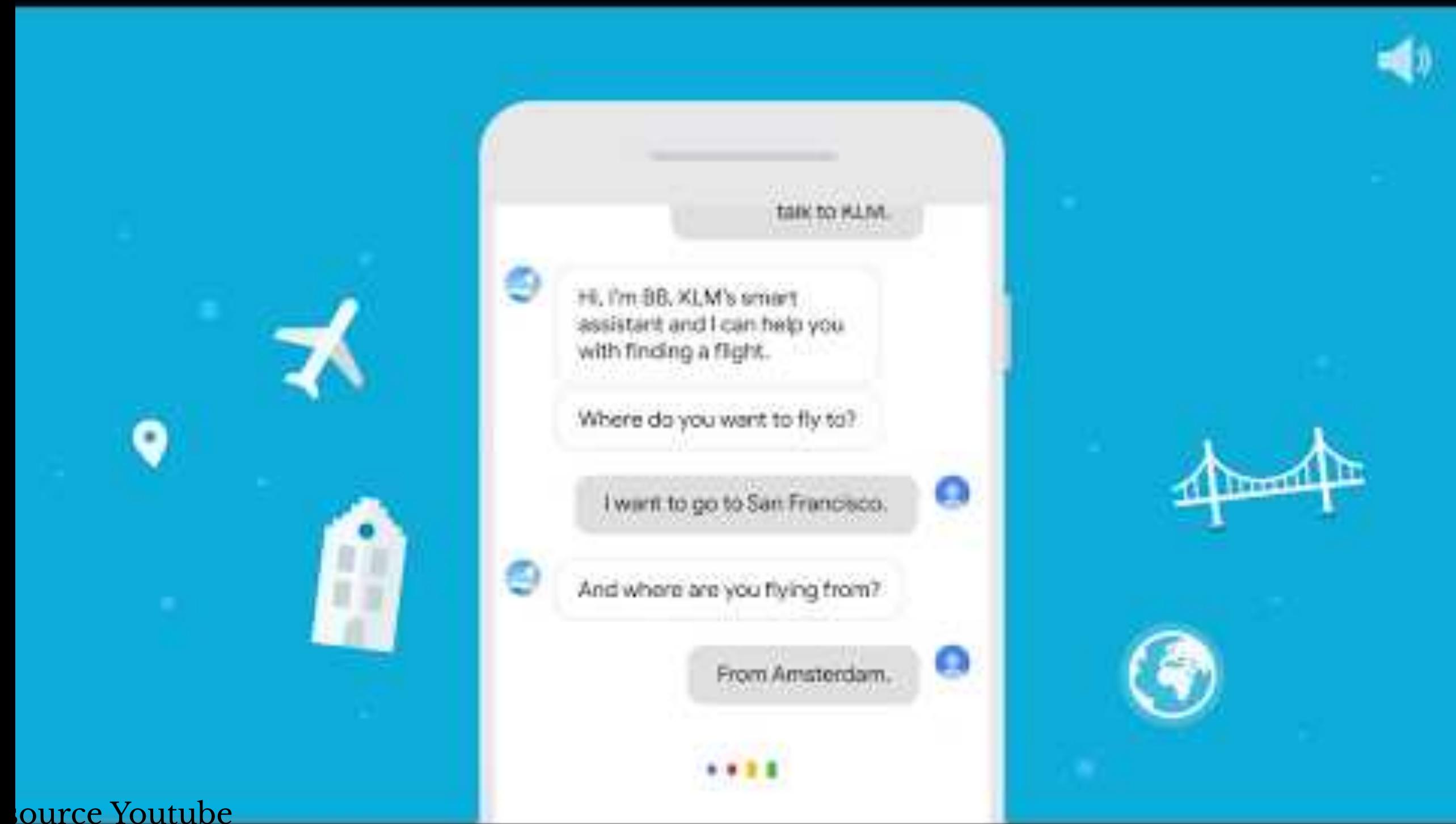
Case Study: KLM BB⁵

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

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



⁵ Image source [Dialogflow](#)



source Youtube



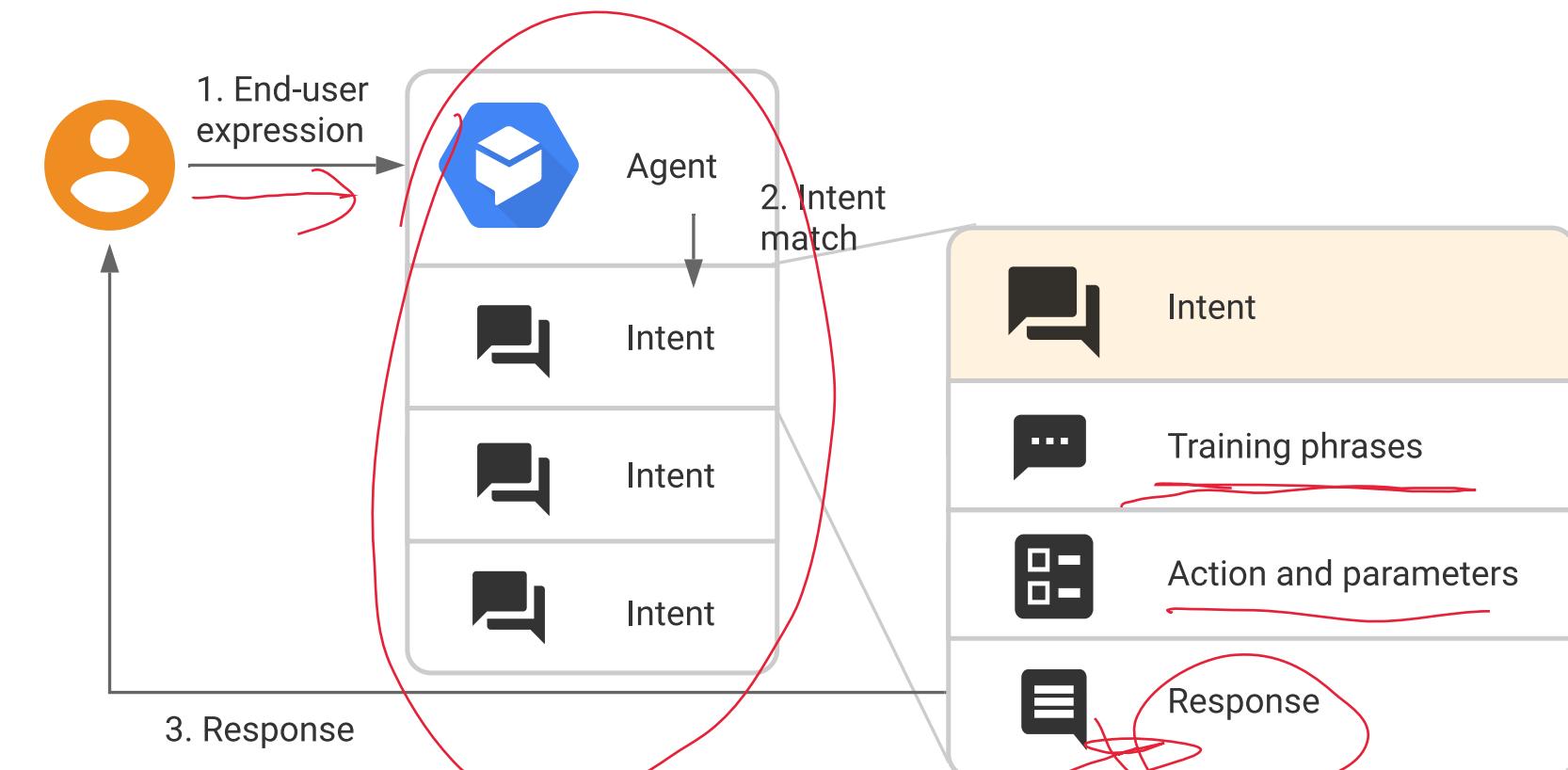
ource



How does Dialogflow work?⁸

The process within Dialogflow involves:

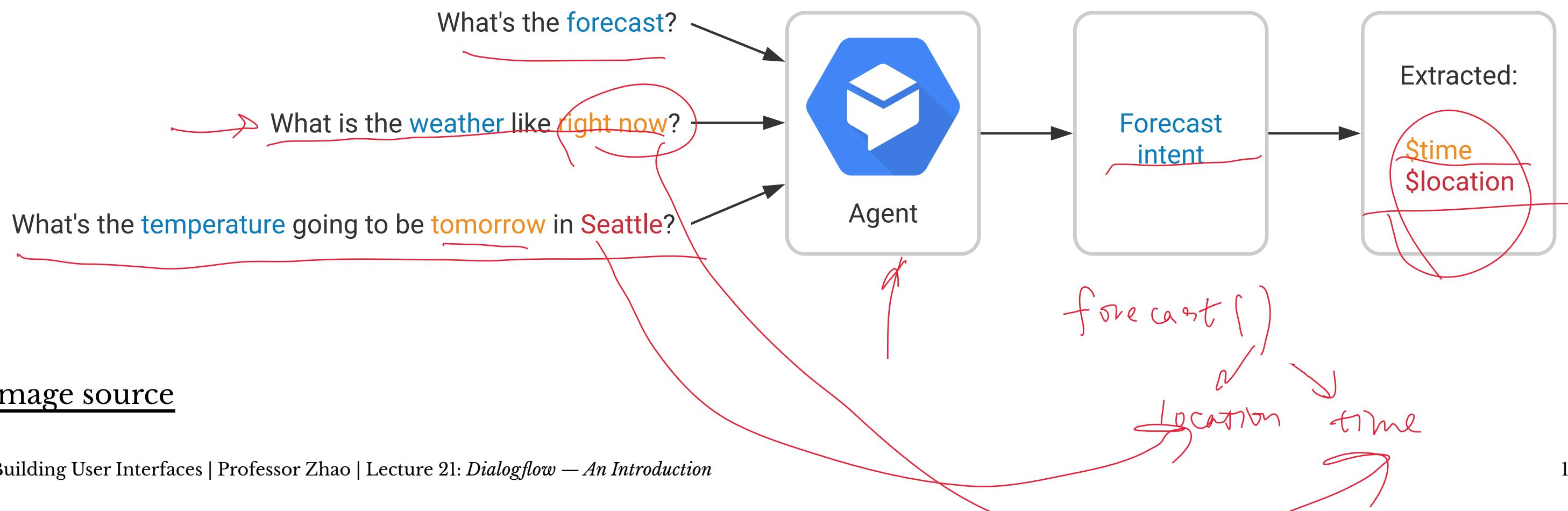
1. User expression
2. Intent matching
3. System response



⁸[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).⁹

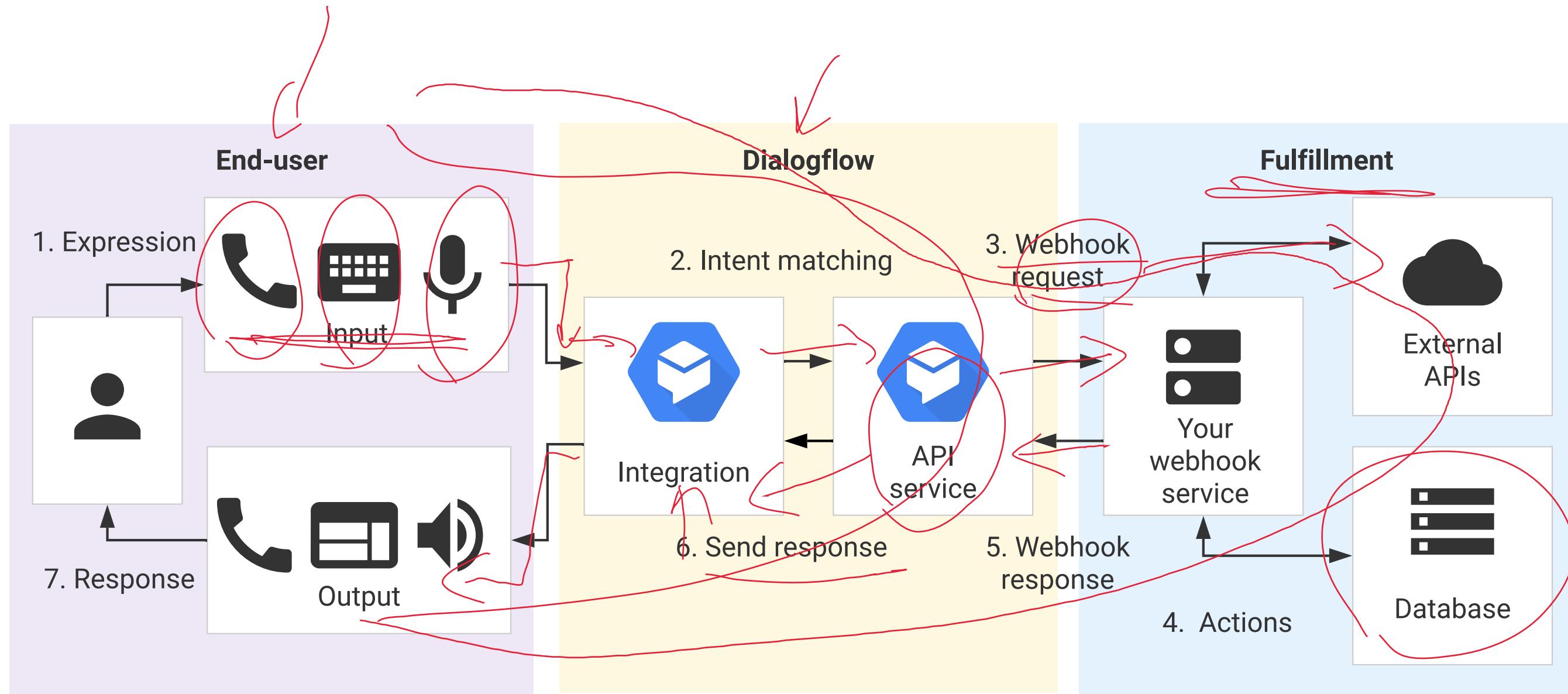


⁹ Image source

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

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

The End-to-end Dialogflow Workflow¹⁰



¹⁰ [Image source](#)

Dialogflow Building Blocks, Part 1

We will cover Part 2 in the next class.

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 Dialogflow Agent Settings page. The top navigation bar includes 'General', 'Languages', 'ML Settings', 'Export and Import', 'Speech', 'Share', and 'Advanced' tabs, with 'General' being the active tab. A red oval highlights the 'CS639DemoAgent' title at the top. A red arrow points from the 'Get familiar with agent settings.' text to the title. Another red arrow points from the 'LOG SETTINGS' section to the 'Delete Agent' button at the bottom. A red circle labeled 'DANGER ZONE' is drawn around the 'Delete Agent' button. The right side of the screen features a sidebar with 'Try it now' and a microphone icon, and a link to 'See how it works in Google Assistant'.

CS639DemoAgent

General Languages ML Settings Export and Import Speech Share Advanced

DESCRIPTION
Describe your agent

DEFAULT TIME ZONE
(GMT-6:00) America/Chicago
Date and time requests are resolved using this timezone.

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	85c321bebf844e1bbe1732b6d1419b8
Developer access token	4aa35bdac21b402fab22caef2f675662

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 the agent "RobotPlanner". The left sidebar lists various sections: RobotPlanner, General, Languages, ML Settings, Export and Import (which is highlighted with a red oval), Speech, Share, Advanced, Intents, Entities, Knowledge [beta], Fulfillment, Integrations, Training, Validation [beta], History, Analytics, and Prebuilt Agents. The main area is titled "RobotPlanner" and contains a "SAVE" button and a three-dot menu. Below the tabs, there are three blue buttons: "EXPORT AS ZIP", "RESTORE FROM ZIP", and "IMPORT FROM ZIP". The "EXPORT AS ZIP" button is also highlighted with a red oval. To the right, there is a "Try it now" button and a microphone icon. A note says "Please use test console above to try a sentence." Below that, a link says "See how it works in Google Assistant" with a blue arrow icon.

RobotPlanner

General Languages ML Settings Export and Import Speech Share Advanced

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

EXPORT AS ZIP RESTORE FROM ZIP IMPORT FROM ZIP

Create a backup of the agent

Replace the current agent version with a new one. All the intents and entities in the older version will be deleted.

Upload new intents and entities without deleting the current ones. Intents and entities with the same name will be replaced with the newer version.

Try it now

Please use test console above to try a sentence.

See how it works in Google Assistant ↗

Agent Speech

The screenshot shows the Dialogflow interface for managing agent speech settings. On the left is a sidebar with navigation links: 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". Under "TEXT TO SPEECH", there is a checked checkbox for "Enable Automatic Text to Speech" with a note about output audio encoding set to "16 bit linear PCM (signed, little-end...)" and a dropdown menu. Below that is a "VOICE CONFIGURATION" section with a note about configuring the synthesized voice. A red oval highlights the "Agent Language" dropdown, which is set to "en (English)". On the right side, 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?

weather - general - query

weather - rain - query

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

Intents In Dialogflow

The screenshot shows the Dialogflow interface for managing intents. A red oval highlights the 'RobotPlanner' project name at the top left. A red box encloses the 'Intents' section in the sidebar. Red arrows point from the 'Intents' label in the sidebar to the 'Default Fallback Intent' and 'Default Welcome Intent' cards in the main list. A red line runs vertically down the left side of the interface.

Intents

CREATE INTENT

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.

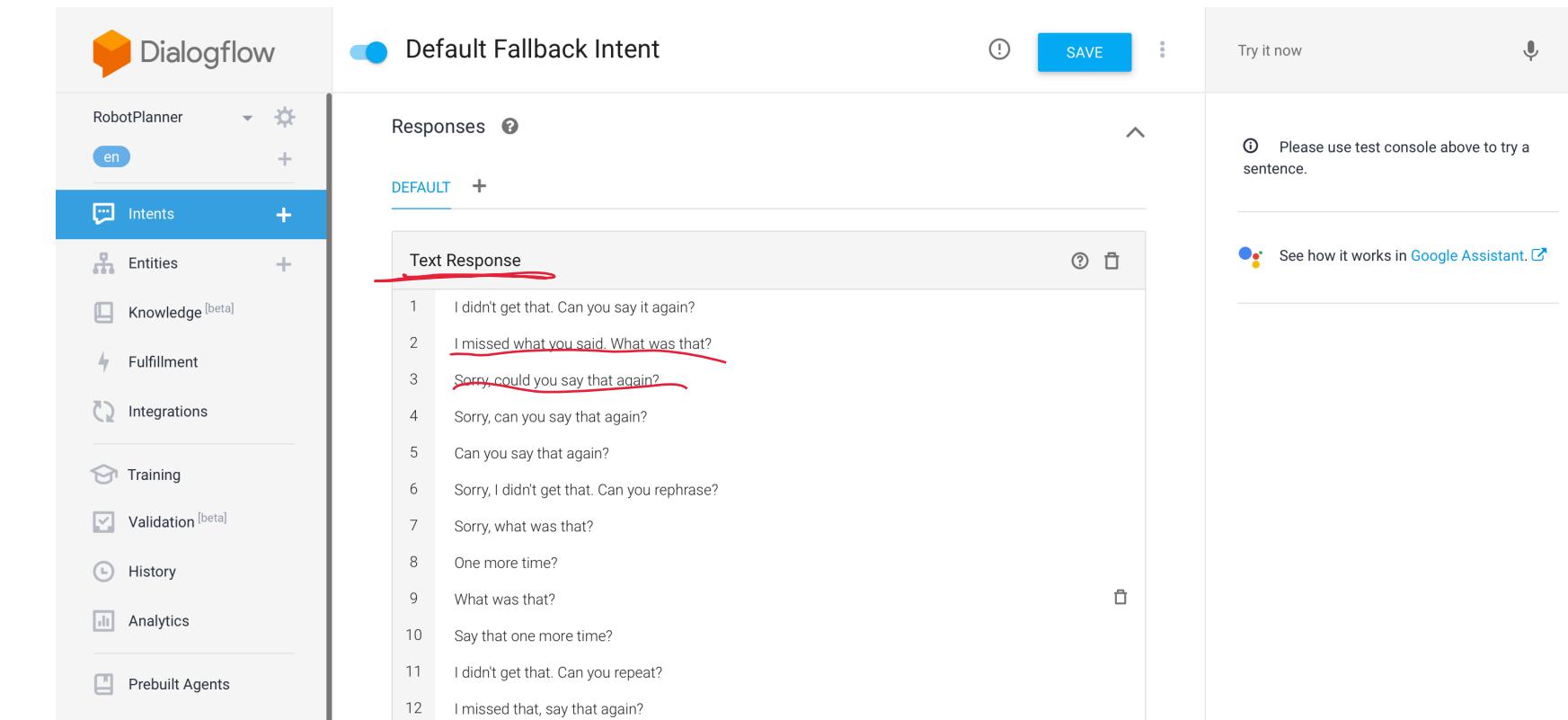
Set-up [Google Assistant integration](#).

RobotPlanner

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

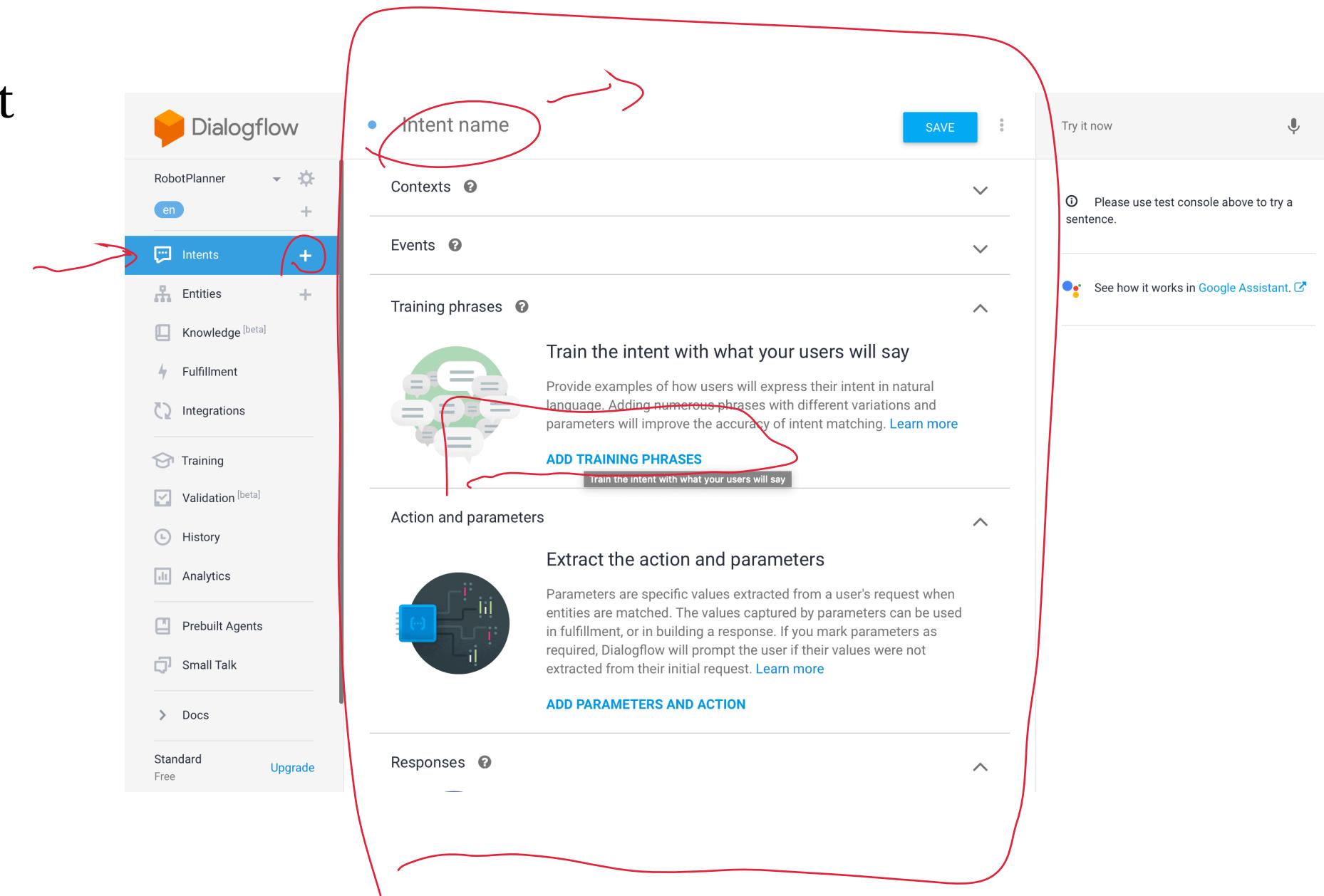
Default Fallback Intents

- 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.



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.

• Pickup Command

Training phrases ?

” 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

SAVE

⋮

Test

Test your agent using the Default Welcome Intent

The screenshot shows the Dialogflow web interface. On the left, a sidebar menu includes options like CS639DemoAgent, en, Intents (selected), Entities, Knowledge [beta], Fulfillment, Integrations, Training, Validation [beta], History, Analytics, Prebuilt Agents, Small Talk, Docs (with Standard Free and Upgrade), Support, Account, and Logout. The main area is titled "Intents" and shows a list with "Default Fallback Intent" and "Default Welcome Intent". A red box highlights the "Default Welcome Intent" row. To the right, a detailed view of the intent is shown. It starts with "USER SAYS" (Hello!) and "DEFAULT RESPONSE" (Hi! How are you doing?). Below this, it shows the "INTENT" (Default Welcome Intent) and "ACTION" (input.welcome). A red box highlights the "ACTION" field. At the bottom, there's a "DIAGNOSTIC INFO" section. A large red bracket on the right side groups the "USER SAYS", "DEFAULT RESPONSE", "INTENT", and "ACTION" sections. A "COPY CURL" button is located at the top right of this group.

Entities

What are entities?

Let's consider those requests again.

— How do I get to *Middleton*?

destination (location)

— What is the price of a basketball?

price - query (object)

— Buy one box of tissues from *Amazon*.

purchase (item , service , amount)

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 interface for creating a new entity. The left sidebar lists 'RobotPlanner' (en), 'Intents', 'Entities' (highlighted with a red circle), 'Knowledge [beta]', 'Fulfillment', 'Integrations', 'Training', 'Validation [beta]' (checked), 'History', and 'Analytics'. The main area shows a form for defining an entity. The 'Entity name' field has 'object' handwritten above it with a red oval. Below it are 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.' A table below has five rows, each with a 'Click here to edit entry' link and a red arrow pointing to it. A '+ Add a row' button is at the bottom. On the right, there are 'SAVE' and 'Try it now' buttons, and a note: 'Please use test console above to try a sentence.' At the bottom right is a 'See how it works in Google Assistant' button.

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

object

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
Enter synonym	

Click here to edit entry

+ Add a row

The screenshot shows a dialog flow entity editor for an entity named 'object'. The entity name is underlined in red. There are four checkboxes at the top: 'Define synonyms' (checked), 'Regexp entity' (unchecked), 'Allow automated expansion' (checked), and 'Fuzzy matching' (unchecked). Below the checkboxes is a table with two columns. The first column contains the original entity names, and the second column contains their corresponding synonyms. The rows are: cube (synonym: cube), container (synonym: container, box, bin), sphere (synonym: sphere, ball), screwdriver (synonym: screwdriver), hammer (synonym: hammer, mallet), and wrench (synonym: wrench). A red box highlights the 'hammer' row. At the bottom right of the table is a button labeled 'Enter synonym'. A red box highlights the entire table area. At the bottom of the editor is a link 'Click here to edit entry'.

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	x
object	@object	cube	x

Automated expansion

Allows dialogflow to extrapolate to new objects

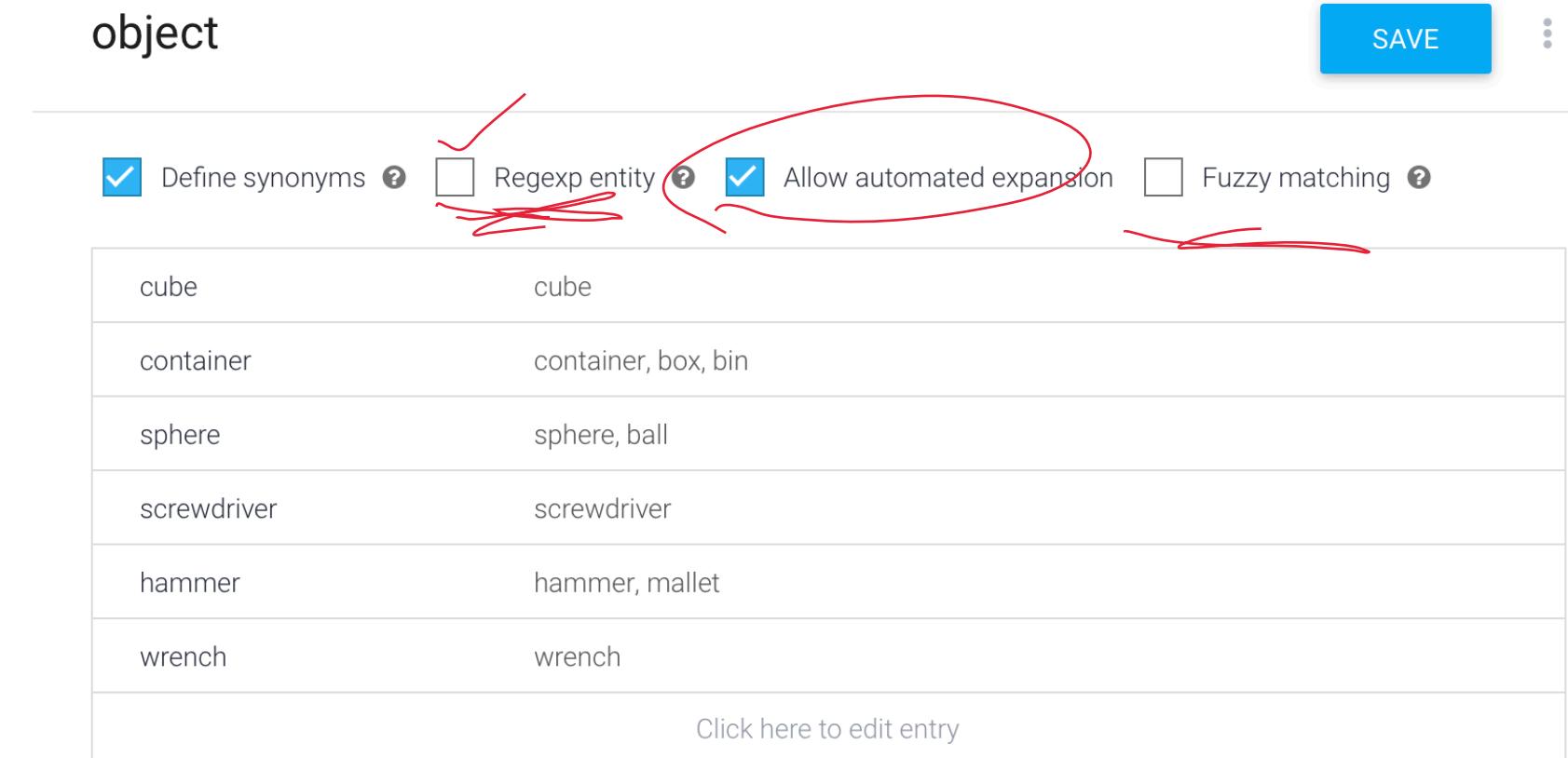
object

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



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

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"/>	Define prompt s...
<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.



Prompts for "object"

NAME	ENTITY	VALUE
object	@object	\$object

PROMPTS

- 1 What object do you want me to pick up?
- 2 Which one?
- 3 Enter a prompt variant

CLOSE

Responses ?

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.

Training phrases [?](#)

Search training phrases Q ^

” 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

Dialogflow Documentation

Full Documentation

Let's Build An Agent

Assignment Preview

Assignment Overview

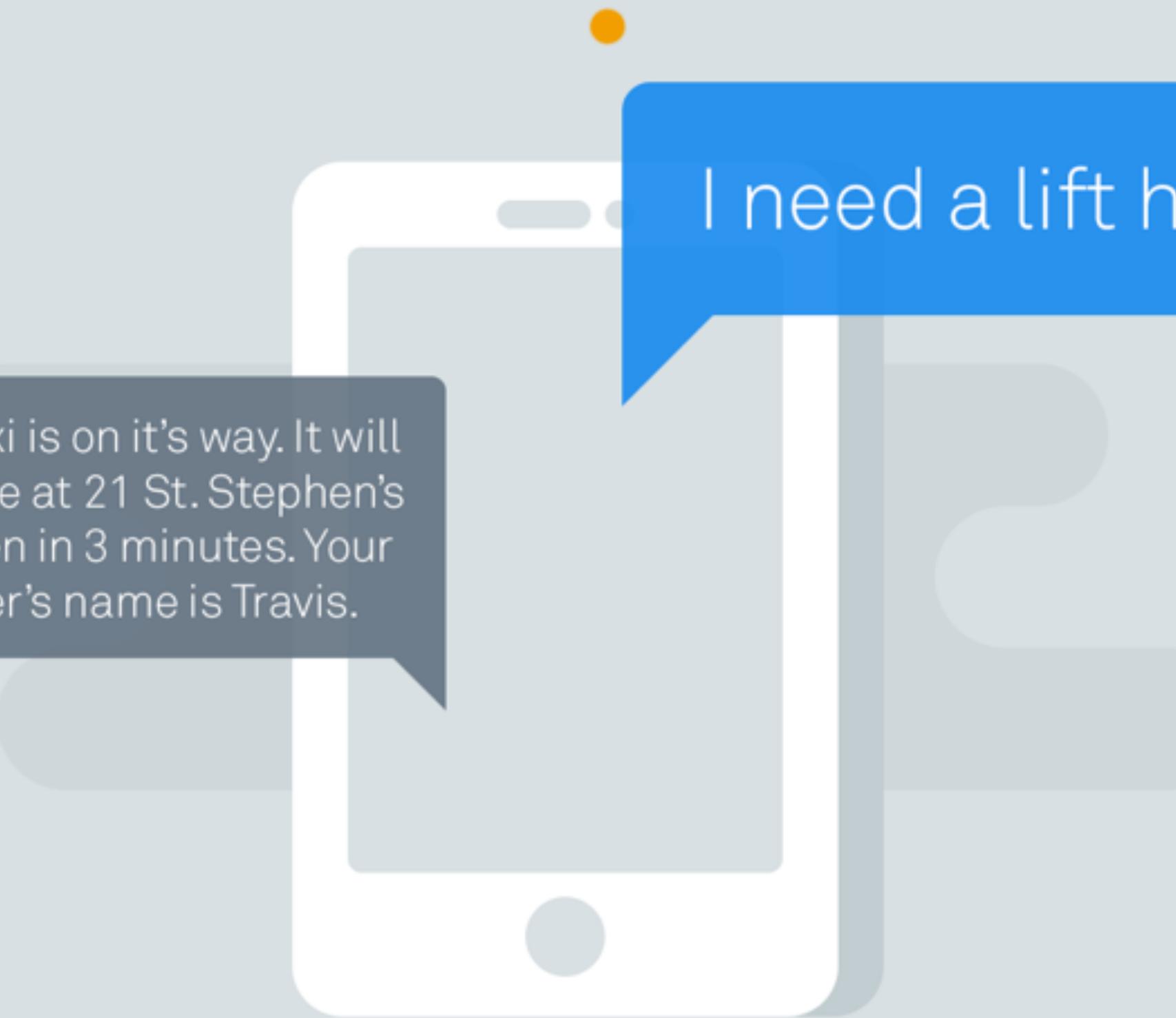
We will create a voice assistant for a fictional online clothing retail store, called *WiscShop*.

- Dialogflow α — Develop specifications
- Dialogflow β — Implementation
- Dialogflow γ — User evaluation

Dialogflow Alpha – Develop specifications¹¹

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.



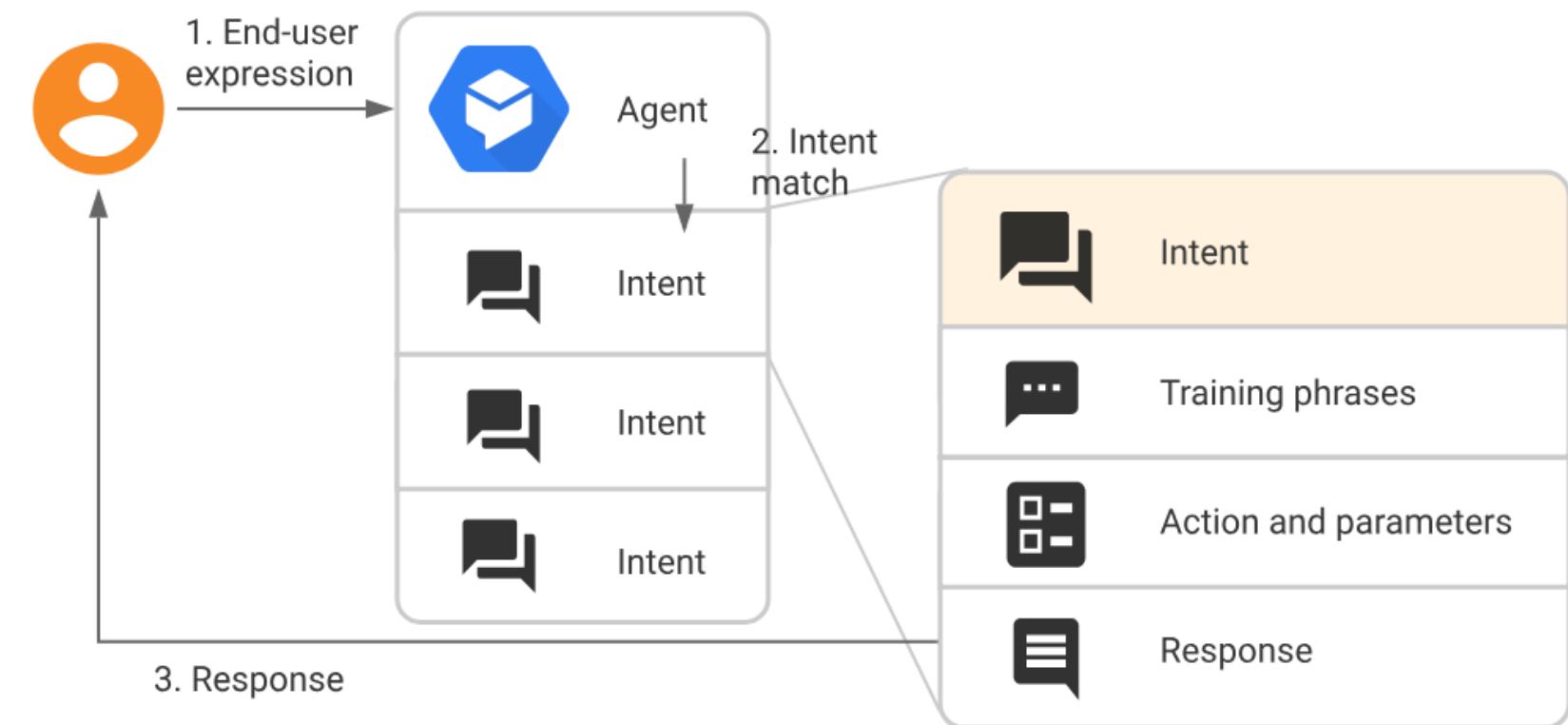
¹¹[Image source](#)

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 – Implementation⁸

Implement the voice assistant agent using the specifications extracted in Dialogflow α .



⁸[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 evaluation¹²

Design and perform a mini usability test over Zoom.

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



¹² [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