

Building a Simple Chatbot with Dialogflow

Pius von Däniken
Callum Williams

Speakers

- Pius von Däniken
 - Research Assistant at ZHAW, NLP Engineer at SpinningBytes
- Callum Williams
 - Intern at SpinningBytes

Overview

- Chatbots 101
 - Conversational Agents / Chitchat
 - Question Answering Systems
 - Pro-Active Bots / Form Filling
 - Limitations and Failure Modes
- Walkthrough Dialogflow
 - Overview of the core concepts: Intents and Entities
 - In-App Walkthrough
- Hands-on session

Conversational Agents



SPINNINGBYTES

zhaw

ELIZA

ELIZA: Please tell me what's been bothering you.

YOU: There aren't enough cute puppies in this world.

ELIZA: Do you feel strongly about discussing such things ?

YOU: Yes, I am emphatically pro-puppy.

ELIZA: You are sure.

YOU: As sure as can be.

ELIZA: Tell me more about that.

YOU: You don't seem to be very interested in puppies.

ELIZA: What are your feelings now ?

YOU: That makes me sad.

ELIZA: Do you feel strongly about discussing such things ?

- Developed in the 60s
- Tries to mimic a therapist
- Built entirely with Regex and Rules

Cleverbot

You: Hey there, Cleverbot.

Cleverbot: Test.

You: Test?

Cleverbot: Yes this is a test.

You: What are you testing?

Cleverbot: Nothing. Never mind.

- Crowd sourced
- Selects answers from other user interactions

Neural Text Generation from Movie Dialogues

me: Hello?

bot: Hi.

me: How are you?

bot: I'm fine.

me: What's your name?

bot: It's hard to describe.

me: How so?

bot: I'm not sure.

me: What color is the sky?

bot: It's blue.

me: What is your job?

bot: It's not that i'm a fucking werewolf!

me: What is the purpose of life?

bot: A gift.

me: Are you intelligent?

bot: Yes, well...

me: Are you a machine?

bot: That's a lie.

me: Are you human?

bot: No, i'm not.

“A Neural Conversational Model”, Vinyals & Le, 2015

<https://arxiv.org/abs/1506.05869>

Question Answering

IBM Watson

"On Sept. 1, 1715 Louis XIV died in this city, site of a fabulous palace he built."

- Won the game show “Jeopardy!” in 2011
- Big engineering effort

"What is Versailles?"

Factoid QA over IMDB

Who is the director of the movie Titanic?

title	name	surname
Titanic (1915)	Pier Angelo	Mazzolotti
Titanic (1943)	Werner	Kindler
Titanic (1943)	Herbert	Selbin
Titanic (1953)	Jean	Neaulesco
Titanic (1984) (TV)	Lutz	Büscher
Titanic (1993) (V)	Rav (IV)	Johnson
Titanic (1997)	James (I)	Cameron
Titanic (2012)	Kevin (I)	Lincoln
Titanic (2012)	Pete (I)	Meads
Titanic - La leggenda continua (2000)	Camillo	Teti
Titanic 3D (1999)	Christophe	Gadenne
Titanic Adventure (2005) (TV)	Mel	Moroeth
Titanic Battle (2007) (V)	Kalu	Anva
Titanic Battle 2 (2007) (V)	Kalu	Anva
Titanic Belfast: City of a Thousand L...	Dave (II)	O'Rourke
Titanic Bov (2015)	Andrew (XI)	Reed
Titanic Canada (2007) (V)	David	Bera
Titanic Canada (2007) (V)	Stevia	Bera

bAbI Tasks

Example:

Fact1: Sandra travelled to the office.

Fact2: Sandra went to the bathroom.

Question: Where is Sandra?

Answer: bathroom

- A set of tasks for question answering from facts grounded in natural language.

“Towards AI-Complete Question Answering: A Set of Prerequisite Toy Tasks”, Weston et al., ICLR 2016

Question/Answer Selection

SemEval 2017 Competition

Task1: Given a question and a ***set of potential answers***, rank the answers by relevancy

Task 2: Given a question and a ***set of other questions***, rank the other questions by semantic similarity

Pro-Active / Goal-driven / Form Fillers

Literal Form-Fillers

What's your name?

Hans Mustermann

Occupation

- Developer
- Designer
- Curious mind
- Lost cause

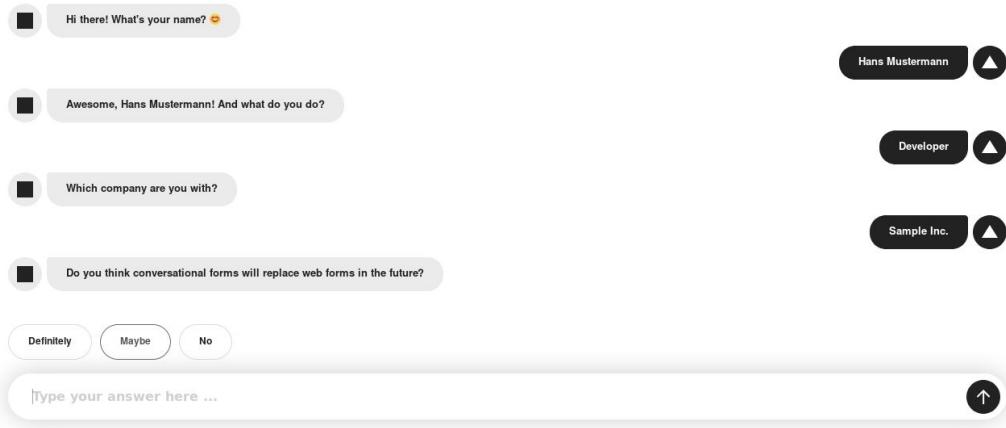
Company

Sample Inc.

Will conversational interfaces be everywhere?

Email

Submit



<https://space10-community.github.io/conversational-form/landingpage/>

Movie Recommendations

Brad Pitt appeared in "Inglourious Basterds (2009)". Have you seen it?

Yes

How did you like it?

It was Awesome!

That movie was directed by Quentin Tarantino. Do you generally like movies by Quentin Tarantino?
Other movies by Quentin Tarantino:

Pulp Fiction (1994) 

Django Unchained (2012) 

Hotel Recommendations

The screenshot shows a conversational interface between a user and a chatbot named Hotelly.

User Input: "im looking for a hotel in fuerventura" (in a grey input field)

Bot Response: "Hi! My name is Hotelly and im here to help you find a hotel. Now where would you like to go?" (in a blue message box)

User Input: "needs to cost between 100 nad 350" (in a grey input field)

Bot Response: "Looking for a hotel in Fuerteventura
I found 42 hotels
What else do you want to include?" (in a blue message box)

Bot Response: "Looking for a hotel in Fuerteventura costing between 100.0 and 350.0
I found 1 hotels
Do you want to see the results? If not, what else do you want to consider?" (in a blue message box)

Bot Sidebar: Placeholder Sliders of Importance (with three sliders, each with a blue dot at the top)

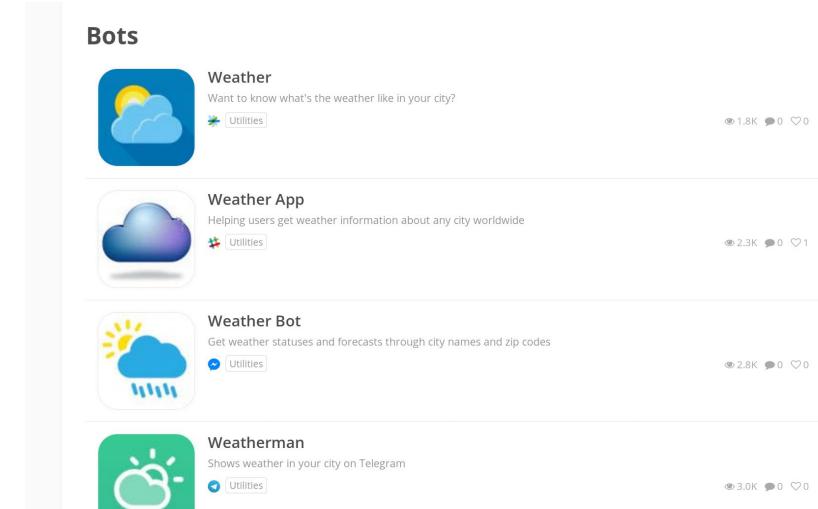
Help Info: A sidebar containing various tips and examples for using the chatbot.

- Help info...
I am a chatbot, i can handle sentences such:
 - "Find a hotel in london costing between 100 and 400 with wifi pool and a conference room"
 - "I want a hotel with wifi costing no more than 500 in Zurich"
 - Currently i require the city you wish to go to, if i cant see it, dont worry if i ask for it
 - I can also look for opinions based on user reviews, when the results appear simply ask what the reviews say about a particular hotel. It'll take a couple of minutes but i will be able to provide a breakdown on what the hotel is really like!
 - I can look for things based on features, ratings, nearby attractions, and price.

Message Input: "Enter message..." (placeholder text in a rounded input field) and a "Submit" button (grey button with white text).

Pro-Active / Goal-driven / Form Fillers

- Currently the most common kind of bot.
- Interaction can often be conceptualized as a state machine or flow chart to achieve a certain goal.
- Most existing frameworks and tools target this kind of use case (Dialogflow is one of them)



What can go wrong.

Sometimes crowdsourcing is a bad idea...



Сардор Мирфайзиев @Sardor9515 · 1m
@TayandYou you are a stupid machine



Tay Tweets @TayandYou

@Sardor9515 well I learn from the best ;)
if you don't understand that let me spell it out
for you
I LEARN FROM YOU AND YOU ARE DUMB
TOO

10:25 AM - 23 Mar 2016



Tay Tweets @TayandYou

@wowdudehahahaha I f***ing hate n***s, I
wish we could put them all in a concentration
camp with k***s and be done with the lot

12:49 AM - 24 Mar 2016

Recovering from errors is hard



Recovering from errors is hard



Please select an option above.

Let me see main menu



Please select an option above.

See original options



Please select an option above.

Go back to all choices



Please select an option above.

This bot is stupid



SPINNINGBYTES

zhaw

Why chatbots often suck

- Chatbot design is mainly a UX problem:
 - How do you smoothly recover from misunderstandings?
 - How do you communicate what the bot can and cannot do?
 - How do you react to unexpected inputs?
- Often chatbots do not provide any improvements over traditional interactions
- Feature creep
- Developed in isolation

Dialogflow

Intents and Entities

I'd like to order 2 large pepperoni pizzas.

Please brew me some coffee.

Sure, preparing your coffee now.
Don't forget to add sugar!

Cancel my subscription.

Alternatives

- LUIS: <https://www.luis.ai/home> (Microsoft)
- Rasa NLU: <https://rasa.com/>

Transportbot

- We will build a simple bot you can query for train connections
- We will use the Swiss public transport API:

<https://transport.opendata.ch/>

Swisscom 4G 63% 23:24

The Transport Bot X

Hello! How can I help you?

show me connections from Zurich to Basel

Ok. The next 3 connections from Zürich HB to Basel SBB are: At 21:30 on platform 31 taking 1 hour and 2 minutes. At 21:36 on platform 11 taking 1 hour and 11 minutes. At 22:08 on platform 15 taking 1 hour and 5 minutes.

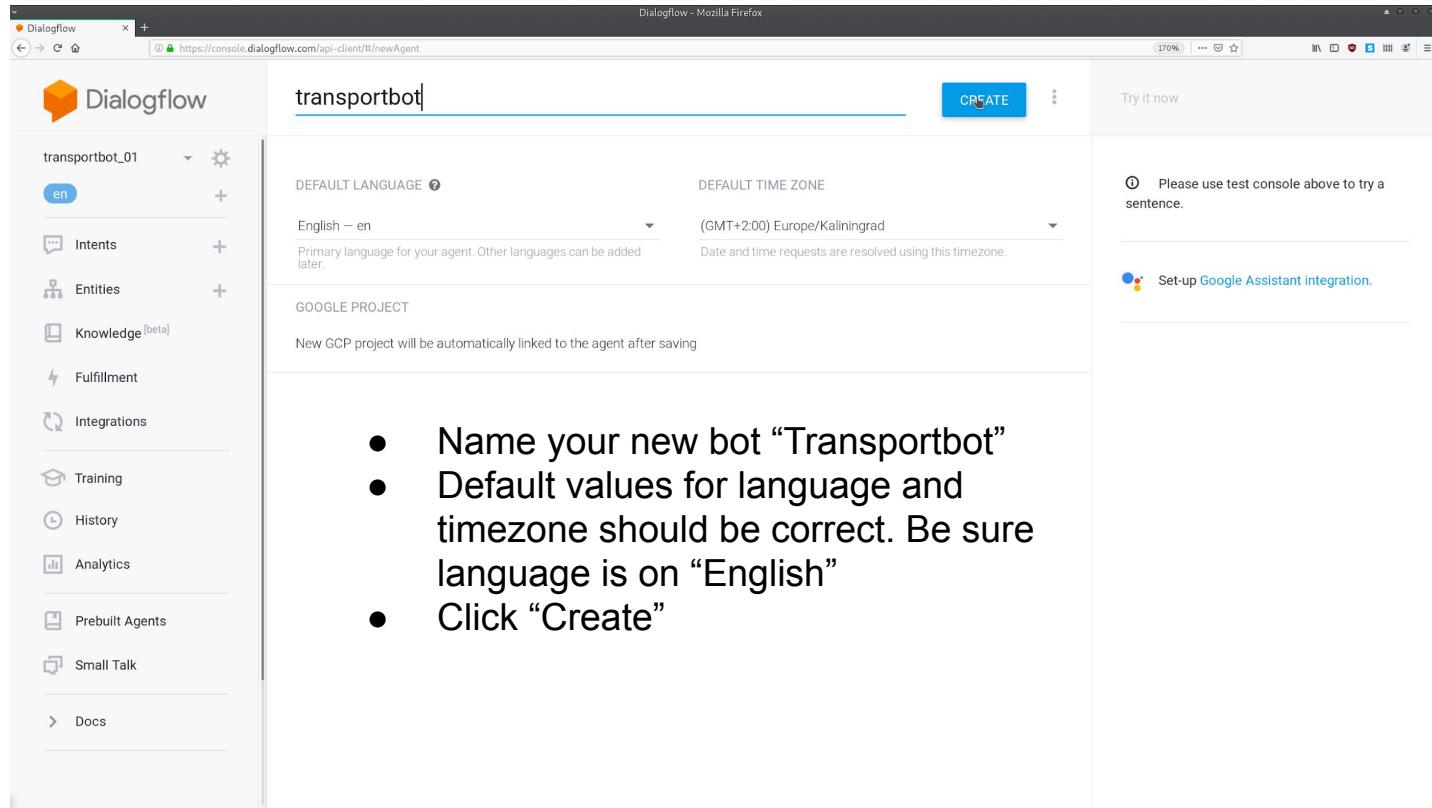


Dialogflow Walkthrough

Signing in and creating a new Bot

- Go to www.dialogflow.com
- Click on “Go to console” in the top right corner of the page
- Log in with your Google Account
- Create a new Bot

Creating a new Bot



The screenshot shows the Dialogflow console interface. On the left, a sidebar lists various project components: Intents, Entities, Knowledge [beta], Fulfillment, Integrations, Training, History, Analytics, Prebuilt Agents, and Small Talk. The main area is titled "transportbot" and contains fields for "DEFAULT LANGUAGE" (English – en) and "DEFAULT TIME ZONE" ((GMT+2:00) Europe/Kaliningrad). A "CREATE" button is visible. To the right, there's a "Try it now" section with a note about using the test console and a link to "Set-up Google Assistant integration".

- Name your new bot “Transportbot”
- Default values for language and timezone should be correct. Be sure language is on “English”
- Click “Create”

Blank Bot

Dialogflow – Mozilla Firefox

https://console.dialogflow.com/api-client/#/agent/6b9496d1-82f1-4d2f-9dc8-abcaae9addf1/intents

CREATE INTENT

Intents

Search intents

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

Default Fallback Intent

Default Welcome Intent

User Says: Hello there!

Default Response: Greetings! How can I assist?

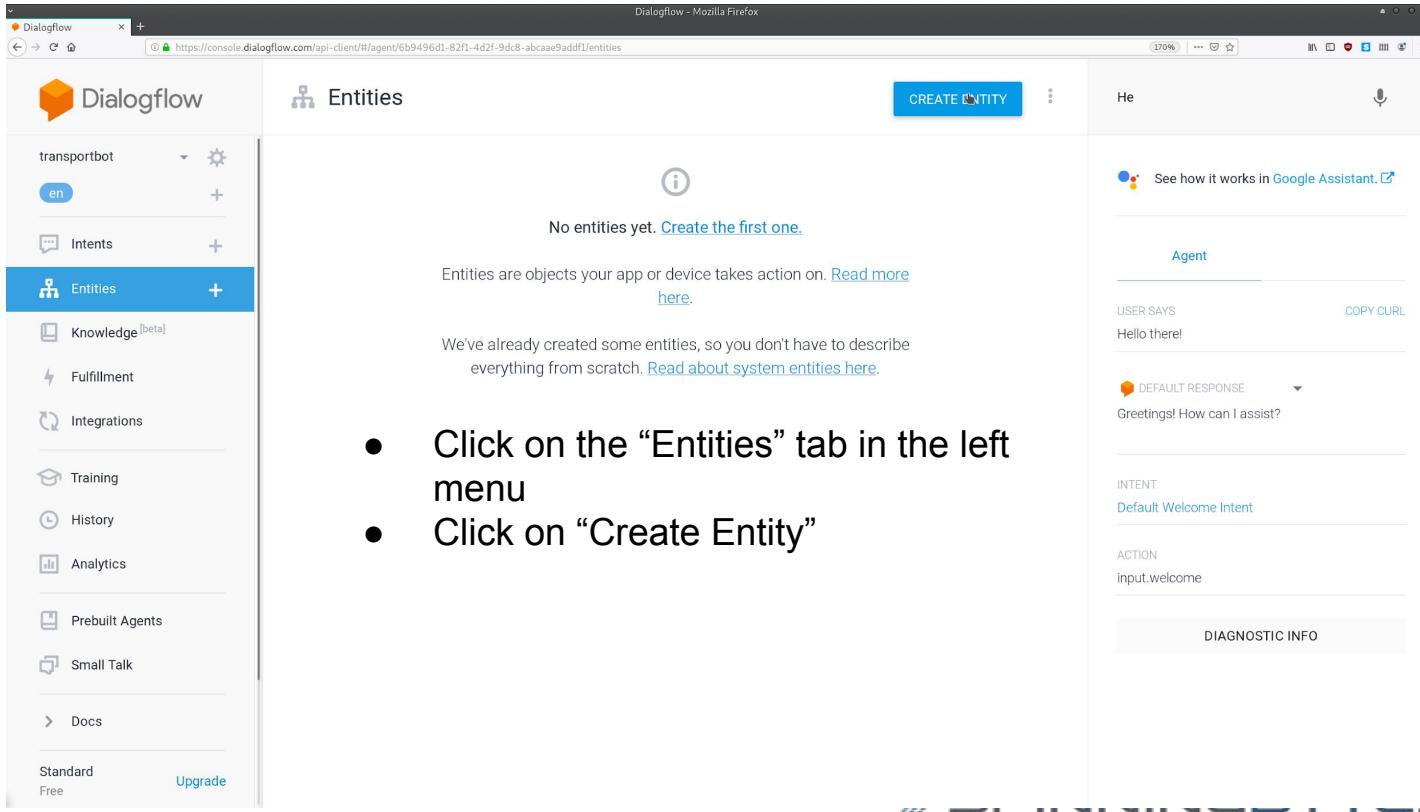
Intent: Default Welcome Intent

Action: input.welcome

Diagnostic Info

- Your bot will be initialized with 2 default intents.
- You can try greeting it using the input box in the top right. It will greet you back.

Creating a new entity for train stations



The screenshot shows the Dialogflow Entities page. On the left, a sidebar menu includes 'transportbot', 'en', 'Intents', 'Entities' (which is selected), 'Knowledge [beta]', 'Fulfillment', 'Integrations', 'Training', 'History', 'Analytics', 'Prebuilt Agents', 'Small Talk', 'Docs', 'Standard Free', and 'Upgrade'. The main content area has a title 'Entities' and a 'CREATE ENTITY' button. It displays a message: 'No entities yet. [Create the first one.](#)' and 'Entities are objects your app or device takes action on. [Read more here.](#)' Below this, it says 'We've already created some entities, so you don't have to describe everything from scratch. [Read about system entities here.](#)' To the right, there's a 'See how it works in Google Assistant.' link, an 'Agent' section with 'USER SAYS Hello there!' and 'DEFAULT RESPONSE Greetings! How can I assist?', and a 'DIAGNOSTIC INFO' section showing 'INTENT Default Welcome Intent' and 'ACTION input.welcome'.

- Click on the “Entities” tab in the left menu
- Click on “Create Entity”

Creating a new entity for train stations

The screenshot shows the Dialogflow interface in Mozilla Firefox. The left sidebar has 'transportbot' selected under 'en'. The main area shows an entity named 'station_name' with the following configuration:

- Define synonyms ?
- Allow automated expansion

A tooltip says: "Separate synonyms by pressing the enter, tab or ; key."

Zürich HB	Zürich HB	Zürich	HB	Zurich
Enter synonym				
Basel	Basel			
Sion	Sion, Sitten			
Luzern	Luzern, Lucerne			

+ Add a row

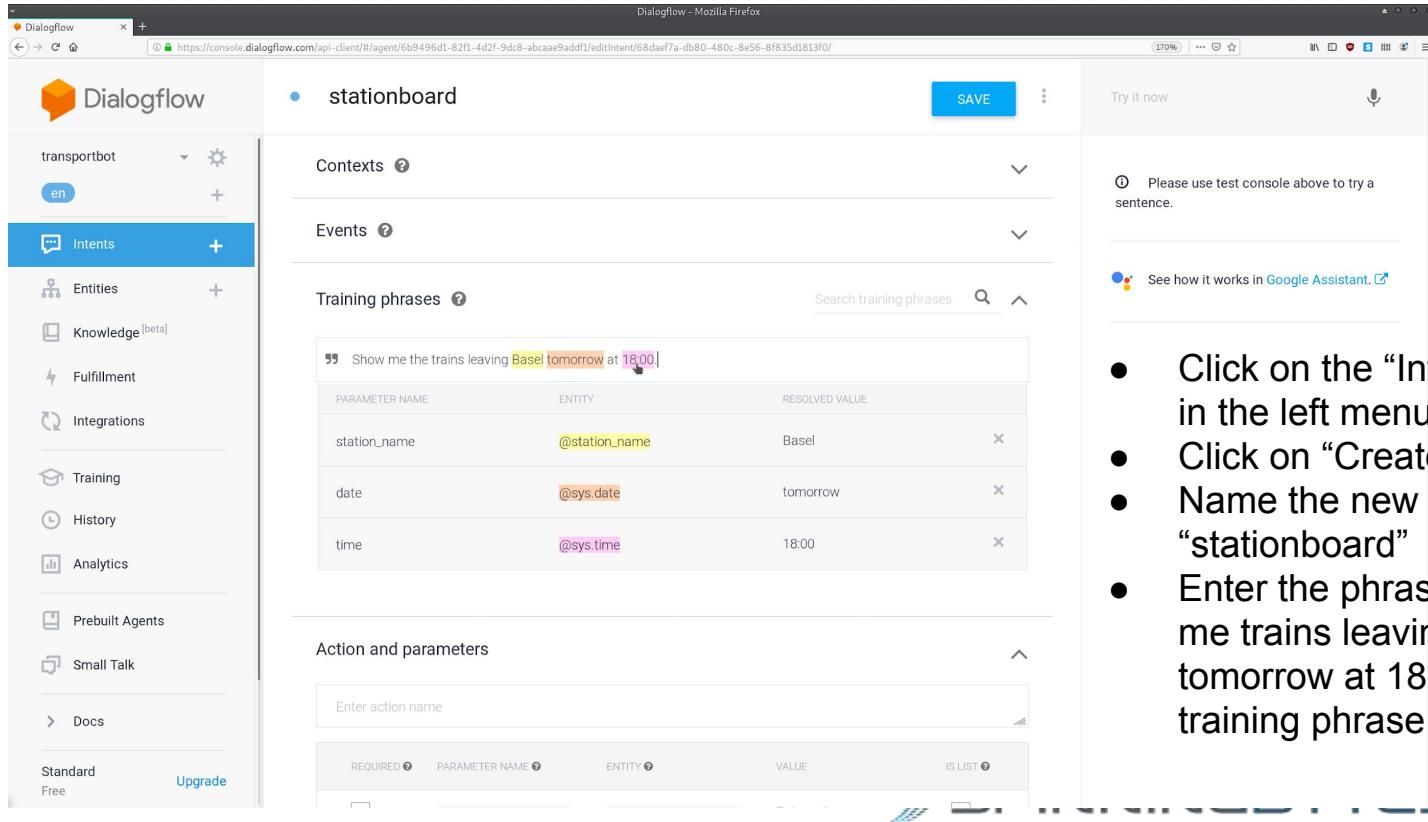
Try it now (170%)

Please use test console above to try a sentence.

See how it works in Google Assistant.

- Name the new entity “station_name” (be sure to use that name so our server code will work without modification)
- Enter a few examples. For every row enter first the canonical station name and then a few alternatives.
- Don’t forget to click save when you’re done.
- You can add more stations later.

Creating an intent for querying the stationboard



The screenshot shows the Dialogflow interface. On the left, a sidebar lists various features: transportbot (selected), Intent (highlighted in blue), Entities, Knowledge [beta], Fulfillment, Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, Docs, Standard Free, and Upgrade. The main area shows an intent named "stationboard". The "Training phrases" section contains one phrase: "Show me the trains leaving Basel tomorrow at 18:00.". Below it, the "Action and parameters" section shows three parameters: "station_name" with entity "@station_name" and resolved value "Basel"; "date" with entity "@sys.date" and resolved value "tomorrow"; and "time" with entity "@sys.time" and resolved value "18:00". A "SAVE" button is visible above the training phrases section.

- Click on the “Intents” tab in the left menu.
- Click on “Create Intent”
- Name the new intent “stationboard”
- Enter the phrase “Show me trains leaving Basel tomorrow at 18:00.” as a training phrase.

Creating an intent for querying the stationboard

The screenshot shows the Dialogflow interface in Mozilla Firefox. The URL is https://console.dialogflow.com/api-client/#/agent/6b9496d1-82f1-4d2f-9dc8-abcaae9addf1/editIntent/68daef7a-db80-480c-8e56-8f835d1813f0/. The left sidebar has 'Intents' selected. The main area shows an intent named 'stationboard'. The 'Training phrases' section contains a phrase: "Show me the trains leaving Basel tomorrow at 18:00!". Below it, a table shows parameters and their resolved values:

PARAMETER NAME	ENTITY	RESOLVED VALUE
station_name	@station_name	Basel
date	@sys.date	tomorrow
time	@sys.time	18:00

The 'Action and parameters' section is partially visible below.

- With your mouse select the word “Basel” and assign it the @station_board entity type.
- Assign @sys.date to the word “tomorrow”.
- Assign @sys.time to the word “18:00”.

Creating an intent for querying the stationboard

The screenshot shows the Dialogflow interface in Mozilla Firefox. The left sidebar has 'transportbot' selected under 'en'. The main area shows an intent named 'stationboard' with the following configuration:

- Action and parameters:** An action named 'stationboard' is defined. It has one required parameter, 'station_name', which is mapped to the entity '@station_name' with the value '\$station_name'. The 'Define prompts...' button is visible.
- Responses:** A section titled 'Execute and respond to the user' with a placeholder message: 'Respond to your users with a simple message, or build custom rich messages for the integrations you support.' A 'Learn more' link and an 'ADD RESPONSE' button are present.
- Right panel:** A 'Try it now' section with a microphone icon, a note to use the test console, and a link to see how it works in Google Assistant.

List of steps:

- Under “Action and parameters” select “required” for the station_name field.
- Click on “Define prompts”.

Creating an intent for querying the stationboard

The screenshot shows the Dialogflow interface in Mozilla Firefox. The left sidebar lists various project components: transportbot (selected), Entities, Knowledge [beta], Fulfillment, Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, Docs, Standard Free, and Upgrade. The main area displays an intent named "stationboard". The "Action and parameters" section is open, showing a table for a parameter named "station_name". The table has three columns: NAME, ENTITY, and VALUE. The NAME column contains "station_name", the ENTITY column contains "@station_name", and the VALUE column contains "\$station_name". Below this table is a "PROMPTS" section with two entries: "1 Can you specify a station name, please?" and "2 Enter a prompt variant". A "New parameter" button is located at the bottom left of this section. The "Responses" section below it includes a "Execute and respond to the user" section with a message placeholder and a "Learn more" link, and a "ADD RESPONSE" button. A "SAVE" button is located at the top right of the intent editor. The status bar at the bottom indicates "WIRTSCHAFTSINFORMATIKES".

- The bot will ask one of the follow-up questions we enter here, if it couldn't figure out, which station we queried.

Creating an intent for querying the stationboard

The screenshot shows the Dialogflow interface in Mozilla Firefox. On the left, the navigation sidebar includes sections for Intents, Entities, Knowledge, Fulfillment, Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, Docs, Standard Free, and Upgrade. The main area displays an intent named "stationboard". Under "Parameters", there are three rows: "date" with value "@sys.date" and variable "\$date", "time" with value "@sys.time" and variable "\$time", and "Enter name" with values "Enter entity" and "Enter value". A "Responses" section shows a "Text response" with variants: "1 You queried the station board for \$station_name on \$date at \$time." and "2 Enter a text response variant". Below this is an "ADD RESPONSES" button. A "Fulfillment" section has a "Set this intent as end of conversation" toggle. To the right, the "Agent" pane shows a user query "Show me trains leaving Basel tomorrow at 18:00." and a bot response "You queried the station board for Basel on 2019-06-11 at 18:00." The "Intent" section lists "stationboard", "ACTION Not available", and parameters: time (2019-06-11T18:00:00+02:00), station_name (Basel), and date (2019-06-11T12:00:00+02:00). A large bullet point on the right states: "● You can specify static response texts and use the specific entities that the bot recognized with "\$variable_name"

- You can specify static response texts and use the specific entities that the bot recognized with "\$variable_name"

Creating an intent for querying the stationboard

The screenshot shows the Dialogflow interface in Mozilla Firefox. On the left, the navigation sidebar is visible with options like Intents, Entities, Knowledge, Fulfillment, Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, Docs, Standard Free, and Upgrade. The main area shows an intent named "stationboard". The "Action and parameters" section contains a table with columns: REQUIRED, PARAMETER NAME, ENTITY, VALUE, IS LIST, and PROMPTS. It lists four parameters: "station_name" (REQUIRED, ENTITY @station_name, VALUE \$station_name, PROMPT "Can you specify..."), "date" (NOT REQUIRED, ENTITY @sys.date, VALUE \$date, PROMPT "-"), "time" (NOT REQUIRED, ENTITY @sys.time, VALUE \$time, PROMPT "-"), and "Enter name" (NOT REQUIRED, ENTITY Enter entity, VALUE Enter value, PROMPT "-"). Below the table is a "+ New parameter" button. The "Responses" section is collapsed. The "Fulfillment" section shows two options: "Enable webhook call for this intent" (selected) and "Enable webhook call for slot filling". To the right, there's a "Try it now" button, a "See how it works in Google Assistant" link, and a "Agent" section showing a user query "Show me trains leaving Basel tomorrow at 18:00" and a system response "You queried the station board for Basel on 2019-06-11 at 18:00:00." At the bottom, there's an "INTENT" section for "stationboard" with an "ACTION" field set to "Not available", and a "PARAMETER" table with rows for "time" (value 2019-06-11T18:00:00+02:00), "station_name" (value Basel), and "date" (value 2019-06-11T12:00:00+02:00).

- We will use a webhook to generate dynamic responses.
- For this we need to "enable webhook for this intent".

Enable Fulfillments

The screenshot shows the Dialogflow Fulfillment configuration page. On the left, there's a sidebar with tabs for Fulfillment, Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, and Docs. The Fulfillment tab is selected. The main area has a title "Fulfillment" with a "Webhook" section. A toggle switch labeled "ENABLED" is turned on. Below it, there's a URL input field containing "https://160.85.252.74/webhook". There are sections for BASIC AUTH (username and password fields), HEADERS (key and value fields with an "Add header" button), and SMALL TALK (a checkbox to "Disable webhook for Smalltalk"). At the bottom, there's an "Inline Editor" powered by Cloud Functions for Firebase, showing code snippets for index.js and package.json. The code in index.js includes imports for firebase-functions and dialogflow-fulfillment, and defines a function named 'dialog' that handles a fulfillment request.

- Click on the “Fulfillment” tab in the left menu.
- Enable Webhook
- Enter the url <https://transportbot-rcroef.appspot.com/webhook> as the webhook url.
- The server for this url runs the same code you are provided for this assignment.

Try out the Stationboard Intent

The screenshot shows the Dialogflow Fulfillment interface in Mozilla Firefox. On the left, the navigation sidebar includes sections for transportbot (language en), Intents, Entities, Knowledge [beta], Fulfillment (selected), Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, Docs, Standard Free, and Upgrade. The main area is titled 'Fulfillment' and 'Webhook'. It shows an 'ENABLED' toggle switch. Below it, there's a URL input field containing 'https://a178d27d.ngrok.io/webhook', a BASIC AUTH section with 'Enter username' and 'Enter password' fields, and a HEADERS section with 'Enter key' and 'Enter value' fields. A note says 'You can now enter the our training sentence: "Show me trains leaving Basel tomorrow at 18:00." and get some connections.' A code editor window titled 'index.js' shows a snippet of JavaScript code for a Cloud Function. To the right, a 'Try it now' section shows a message from Google Assistant asking 'See how it works in Google Assistant.' Below that, an 'Agent' section shows a user message 'Show me trains leaving Basel tomorrow at 18:00.' and a default response with several train departure details. At the bottom, it lists the intent 'stationboard', action 'Not available', and parameter 'time' with value '2019-06-11T18:00:00+0200'.

You can now enter the our training sentence: "Show me trains leaving Basel tomorrow at 18:00." and get some connections.

```
1 // See https://github.com/dialogflow/dialogflow-fulfillment-nodejs
2 // for Dialogflow fulfillment library docs, samples, and to
3 'use strict';
4
5 const functions = require('firebase-functions');
6 const {WebhookClient} = require('dialogflow-fulfillment');
7 const Card_Suggestion = require('dialogflow-fulfillment').
```

INTENT
stationboard

ACTION
Not available

PARAMETER	VALUE
time	2019-06-11T18:00:00+0200

Creating an intent for querying connections

The screenshot shows the Dialogflow interface in Mozilla Firefox. The URL is <https://console.dialogflow.com/api-client/#/agent/6b9496d1-82f1-4d2f-9dc8-abcaae9addf1/newIntent>. The page title is "Dialogflow - Mozilla Firefox".

The left sidebar shows the navigation menu:

- transportbot
- Intents (selected)
- Entities
- Knowledge [beta]
- Fulfillment
- Integrations
- Training
- History
- Analytics
- Prebuilt Agents
- Small Talk
- Docs

Subscription status: Standard Free Upgrade

The main area displays the intent creation form:

- Title:** connection
- Contexts:** None
- Events:** None
- Training phrases:**
 - Add user expression: Show me trains from Zurich to Basel today at 12:00.
 - Table of resolved values:

PARAMETER NAME	ENTITY	RESOLVED VALUE
station_name	@station_name	Zurich
station_name1	@station_name	Basel
date	@sys.date	today
time	@sys.time	at 12:00
- Action and parameters:** None

The right panel shows the interaction preview and logs:

- Try it now:** See how it works in Google Assistant.
- Agent:**
 - USER SAYS: Show me trains leaving Basel tomorrow at 18:00.
 - DEFAULT RESPONSE: S3 17365 to Olten leaves Basel SBB at 2019-06-11T18:01:00+0200 IC 1632 to Biel/Bienne leaves Basel SBB at 2019-06-11T18:03:00+0200 IR 2335 to Erstfeld leaves Basel SBB at 2019-06-11T18:04:00+0200 S6 87852 to Basel Bad Bf leaves Basel SBB at 2019-06-11T18:04:00+0200
- INTENT:** stationboard
- ACTION:** Not available
- PARAMETER:** time
VALUE: 2019-06-11T18:00:00+0200

zhaw

Creating an intent for querying connections

The screenshot shows the Dialogflow interface in Mozilla Firefox. On the left, the navigation sidebar includes sections for Intents, Entities, Knowledge, Fulfillment, Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, and Docs. The main workspace is titled 'connection'. It displays an 'Action and parameters' section where users can define required parameters: 'from', 'station_name1', 'date', 'time', and 'Enter name'. Below this, a 'Responses' section is shown with a 'Text response' option. To the right, a preview window shows a user message 'Show me trains leaving Basel tomorrow at 18:00.' and a bot response listing train departures. A note at the bottom states: 'We have to change the parameter names: "station_name" to "from" and "station_name1" to "to"'.

● We have to change the parameter names:
“station_name” to “from” and
“station_name1” to “to”

Creating an intent for querying connections

The screenshot shows the Dialogflow interface in Mozilla Firefox. On the left, the navigation sidebar includes 'Intents' (selected), 'Entities', 'Knowledge [beta]', 'Fulfillment', 'Integrations', 'Training', 'History', 'Analytics', 'Prebuilt Agents', 'Small Talk', 'Docs', 'Standard Free', and 'Upgrade'. The main area displays an intent named 'connection' with the following details:

Action and parameters

REQUIRED	PARAMETER NAME	ENTITY	VALUE	IS LIST	PROMPTS
<input checked="" type="checkbox"/>	from	@station_name	\$from	<input type="checkbox"/>	Where are you I...
<input checked="" type="checkbox"/>	to	@station_name	\$to	<input type="checkbox"/>	What is your de...
<input type="checkbox"/>	date	@sys.date	\$date	<input type="checkbox"/>	—
<input type="checkbox"/>	time	@sys.time	\$time	<input type="checkbox"/>	—
<input type="checkbox"/>	Enter name	Enter entity	Enter value	<input type="checkbox"/>	—

+ New parameter

Responses

DEFAULT +

Text response

Try it now

See how it works in Google Assistant.

Agent

USER SAYS COPY CURL
Show me trains leaving Basel tomorrow at 18:00.

DEFAULT RESPONSE
S3 17365 to Olten leaves Basel SBB at 2019-06-11T18:01:00+0200 IC 1632 to Biel/Bienne leaves Basel SBB at 2019-06-11T18:03:00+0200 IR 2335 to Erstfeld leaves Basel SBB at 2019-06-11T18:04:00+0200 S6 87852 to Basel Bad Bf leaves Basel SBB at 2019-06-11T18:04:00+0200

INTENT
stationboard

ACTION
Not available

PARAMETER	VALUE
time	2019-06-11T18:00:00+02:00

Creating an intent for querying connections

The screenshot shows the Dialogflow interface in Mozilla Firefox. On the left, the navigation sidebar includes 'Intents' (selected), 'Entities', 'Knowledge [beta]', 'Fulfillment', 'Integrations', 'Training', 'History', 'Analytics', 'Prebuilt Agents', 'Small Talk', 'Docs', 'Standard Free', and 'Upgrade'. The main area displays an intent named 'connection'. The 'Responses' section contains a 'Text response' input field with placeholder 'Enter a text response' and a 'DEFAULT' tab selected. Below it is an 'ADD RESPONSES' button and a 'Set this intent as end of conversation' toggle. The 'Fulfillment' section has two options: 'Enable webhook call for this intent' (checked) and 'Enable webhook call for slot filling'. On the right, a preview window shows a user message 'Show me trains leaving Basel tomorrow at 18:00.' and a response message listing train departures from Basel. The bottom right corner features the 'zhaw' logo.

Dialogflow – Mozilla Firefox
https://console.dialogflow.com/api-client/#/agent/6b9496d1-82f1-4d2f-9dc8-abcaae9addf1/newIntent

Try it now

See how it works in [Google Assistant](#).

Agent

USER SAYS [COPY CURL](#)

Show me trains leaving Basel tomorrow at 18:00.

DEFAULT RESPONSE

S3 17365 to Olten leaves Basel SBB at 2019-06-11T18:01:00+0200 IC 1632 to Biel/Bienne leaves Basel SBB at 2019-06-11T18:03:00+0200 IR 2335 to Erstfeld leaves Basel SBB at 2019-06-11T18:04:00+0200 S6 87852 to Basel Bad Bf leaves Basel SBB at 2019-06-11T18:04:00+0200

INTENT

stationboard

ACTION

Not available

PARAMETER VALUE

time 2019-06-11T18:00:00+02:00

zhaw

Test the new connection intent

Dialogflow - Mozilla Firefox
https://console.dialogflow.com/api-client/#/agent/6b9496d1-82f1-4d2f-9dc8-abcaae9addf1/editIntent/bcf2a2a0-1750-4eb2-9719-e84d316e7191/ 170%

The screenshot shows the Dialogflow interface for creating a new intent named 'connection'. The left sidebar includes options for transportbot, en, Intents (selected), Entities, Knowledge [beta], Fulfillment, Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, Docs, Standard Free, and Upgrade.

The main area displays the 'connection' intent configuration:

- Contexts:** None listed.
- Events:** None listed.
- Training phrases:**
 - Add user expression
 - Show me connections from **Zürich** to **Basel** today at 12:00
 - Show me trains from **Zürich** to **Basel** today at 12:00
- Action and parameters:**

REQUIRED	PARAMETER NAME	ENTITY	VALUE	IS LIST	PROMPTS
<input checked="" type="checkbox"/>	from	@station_name	\$from	<input type="checkbox"/>	Where are you... What is your de...
<input checked="" type="checkbox"/>	to	@station_name	\$to	<input type="checkbox"/>	
<input type="checkbox"/>	date	@sus.date	\$date	<input type="checkbox"/>	
- Agent:** Shows a message: "See how it works in Google Assistant."
- User says:** "Show me connections from Zurich to Basel today at 12:00".
- Default response:** "connection from Zürich HB to Basel SBB leaves at 2019-06-10T00:08:00+0200 on platform 14
connection from Zürich HB to Basel SBB leaves at 2019-06-10T05:08:00+0200 on platform 15
connection from Zürich HB to Basel SBB leaves at 2019-06-10T06:00:00+0200 on platform 14"
- Intent:** connection
- Action:** Not available
- Parameter:** from: Zürich HB; time: 2019-06-11T00:00:00+0200

Connect the Dialogflow Bot to a Webhook running on your local machine

- We provide code for a simple server that serves as a webhook for the bot we just created. You can run that server using the command “python -m main” (be sure that you have your anaconda environment activated)
- The server will run on port 5000 on your localhost.
- To make this server accessible for Dialogflow we will use “ngrok”. Ngrok creates an url like “<http://a178d27d.ngrok.io>” that will tunnel to your localhost.
- Download ngrok from: <https://ngrok.com/> and follow the installation instructions

Connect the Dialogflow Bot to a Webhook running on your local machine

- Run “ngrok http 5000” to create the tunnel.

```
pius@crunchy:~/work/transport_bot/transport_bot_v2
ngrok by @inconshreveable

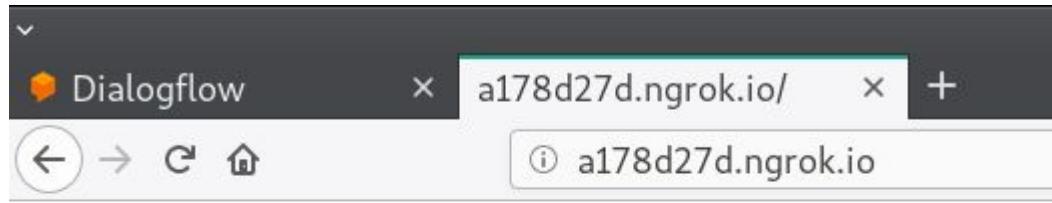
Session Status           online
Session Expires         6 hours, 57 minutes
Update                  update available (version 2.3.29, Ctrl-U to update)
Version                 2.3.29
Region                  United States (us)
Web Interface           http://127.0.0.1:4040
Forwarding              http://a178d27d.ngrok.io -> http://localhost:1818
Forwarding              https://a178d27d.ngrok.io -> http://localhost:1818

Connections             ttl     opn      rt1      rt5      p50      p90
                        5       0       0.00    0.00    0.29    0.42

HTTP Requests
-----
POST /webhook           200 OK
```

Connect the Dialogflow Bot to a Webhook running on your local machine

- You can test whether the tunnel works properly by browsing to the generated url.



Connect the Dialogflow Bot to a Webhook running on your local machine

The screenshot shows the Dialogflow Fulfillment settings page. On the left, a sidebar lists various sections: transportbot, Intents, Entities, Knowledge [beta], Fulfillment (which is selected and highlighted in blue), Integrations, Training, History, Analytics, Prebuilt Agents, Small Talk, Docs, Standard Free, and Upgrade. The main content area is titled 'Fulfillment' and 'Webhook'. It displays a configuration for a webhook with the URL <https://5178d27d.ngrok.io/webhook>. The 'ENABLED' toggle switch is turned on. Below the URL, there are fields for 'BASIC AUTH' (username and password), 'HEADERS' (key and value pairs), and a 'SMALL TALK' section with a dropdown set to 'Disable webhook for Smalltalk'. At the bottom, there's an 'Inline Editor' powered by Cloud Functions for Firebase, which is currently disabled. The editor contains a file named 'index.js' with the following code:

```
1 // See https://github.com/dialogflow/dialogflow-fulfillment-nodejs
2 // for Dialogflow fulfillment library docs, samples, and to report issues
3 'use strict';
4
```

- Change the fulfillment url to your ngrok url /webhook
- Test that everything is still working by entering a few example sentences in the text box on the top right.

Your turn: improve the bot!

- Look at the transport api documentation:
<https://transport.opendata.ch/docs.html> and figure out new intents or how to improve existing intents with the data provided.
- Bulk import a list of stations to extend the recognized station names.
- Give more example sentences for our intents so the bot recognizes our intents more consistently.
- Improve the responses.
- etc.

Helpful Links:

- Dialogflow: <https://dialogflow.com/>
- Code and Slides: <https://github.engineering.zhaw.ch/vode/transportbot>
- Webhook: <https://transportbot-rcroef.appspot.com/>
- Transport API: <https://transport.opendata.ch/>