Introduction to Voice Assistants and Speechly

Mohammad Vali

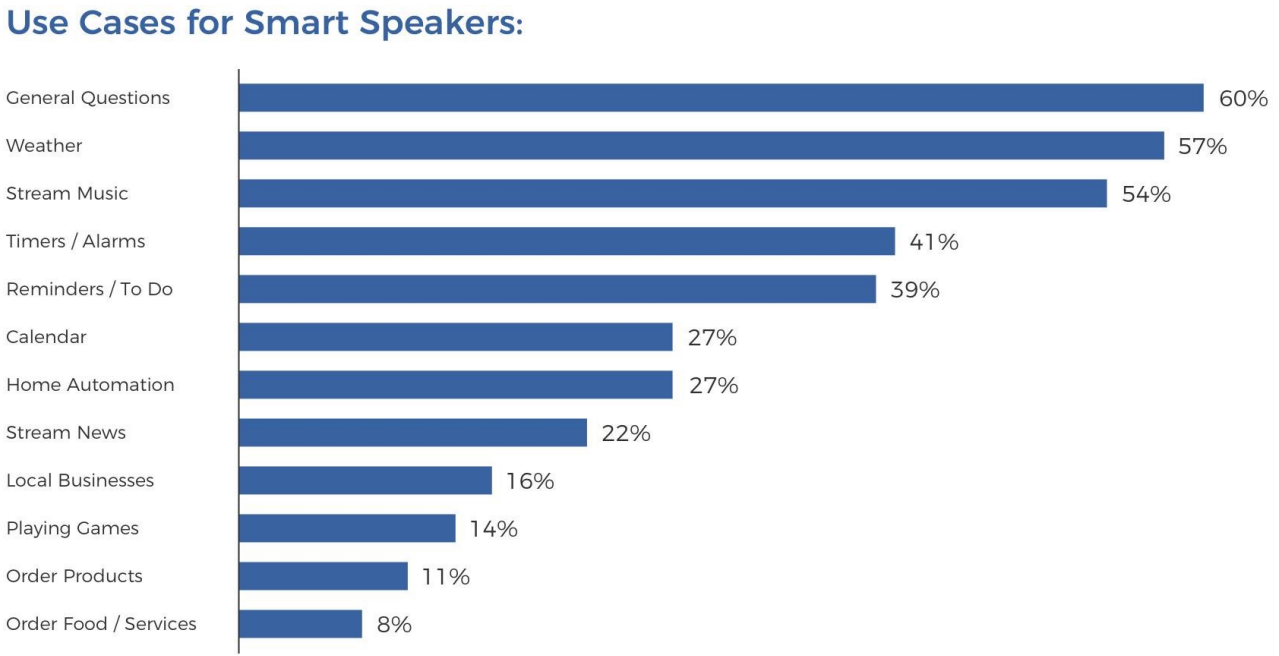
27 January 2022

❖ Why and When?

$$\begin{cases} cooking \\ driving \\ excercising \\ multitasking\ on\ work \end{cases} \longrightarrow$$

Offers an opportunity for voice interaction

❖ What?



Smart speaker usage in the US

❖ Anatomy of a voice command

1. Intent ➡ The broader objective of a user's voice command

$$\begin{cases} Low\ utility \\ High\ utility \end{cases}$$

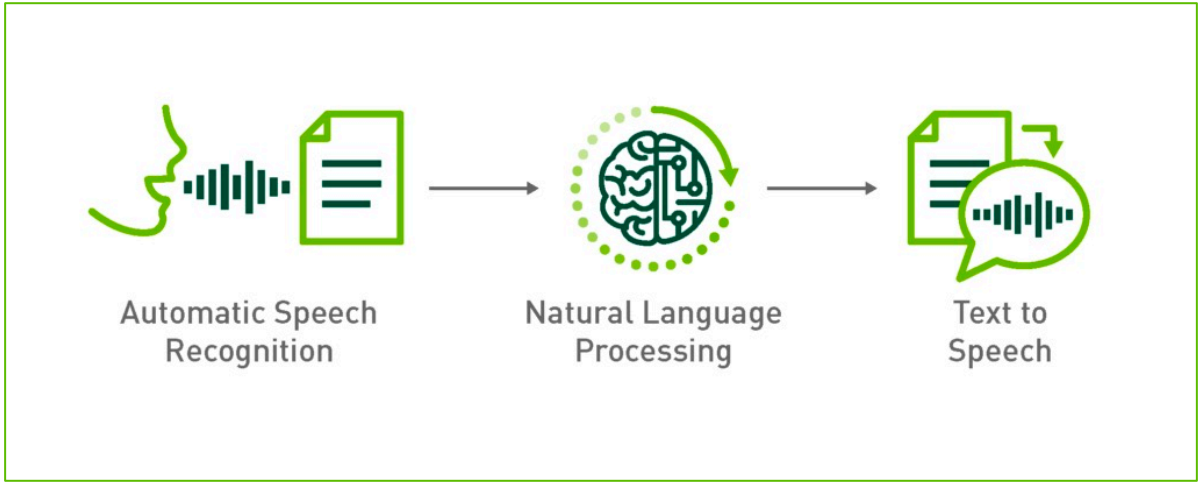
- $Low\ utility$ → Vague tasks; Hear more about Helsinki
- $High\ utility$ → Very specific tasks; Turning off the lights

2. Utterance ➡ How the user phrases a command → Play …; I want to hear …

3. Slot ➡ More information required (or optional)

- Request a taxi (default time = ASAP)
- Request a taxi at 1 PM (default time = 1 PM)

❖ Architecture of voice assistants



❖ Anatomy of a conversational AI interaction

STEP 1 — Wake-Up Word Detection
Also called 'keyword spotting', this component detects a spoken wake-up phrase such as 'OK Google', 'Hey Siri', or 'OK MindMeld'. It runs in low power mode locally on device. When a wake-up word is detected, resource intensive cloud-based speech recognition is activated.

STEP 2 — Speech Recognition
Also called 'speech-to-text' or ASR, this component converts spoken audio into a text transcript. Today, all major OS platforms provide built-in, speaker-independent ASR with near-human accuracy. They rely on hybrid local and cloud-based models and deliver results in real-time.

STEP 3 — Domain Classification
Domain classification is the first natural language processing, or NLP, step. It analyzes the ASR transcripts and categorizes each request into one of a set of pre-defined domains. Each domain typically reflects a unique knowledge area such as 'weather', 'sports', 'traffic', etc.

STEP 4 — Intent Classification
The second NLP step is intent classification. This step categorizes each request into one of a set of pre-defined intents. Each intent reflects what the user is trying to accomplish, and it prescribes a specific action or answer type which defines the desired outcome of each request.

STEP 5 — Entity Recognition
The third NLP step is entity recognition. This component is responsible for identifying important words and phrases in each request. For example, in the request 'play Vogue by Madonna', 'Vogue' is an entity identifying the song title and 'Madonna' is an entity identifying the artist.

STEP 6 — Role Classification
The fourth NLP step is role classification or role labeling. This component assigns pre-defined labels or categories to entities of a particular type. For example, for the request 'book a flight from SFO to JFK', the entity 'SFO' might be labeled 'origin' and 'JFK' might be labeled 'destination'.

STEP 7 — Entity Resolution
The fifth NLP step is entity resolution. This step matches each identified entity with a real-world, disambiguated canonical concept or object. For example, the entity 'Obama' could be resolved to 'President Barack Obama', and 'iPhone 6' could be resolved to a specific product SKU.

STEP 8 — Language Parsing
The last NLP step is language parsing. This step can involve both dependency parsing and semantic parsing. It is responsible for determining the relationships between the individual entities identified in each request.

STEP 9 — Question Answering
Question answering identifies the best answer candidates to satisfy each request. It relies on a knowledge base, containing comprehensive catalog or product data for example, to check the validity of each candidate response and provide recommendations and suggestions.

STEP 10 — Language Generation
Language generation interprets the output of the NLP components and the question answerer in order to generate a human-language response for each request. This response is intended to resemble a real-world natural language interaction.

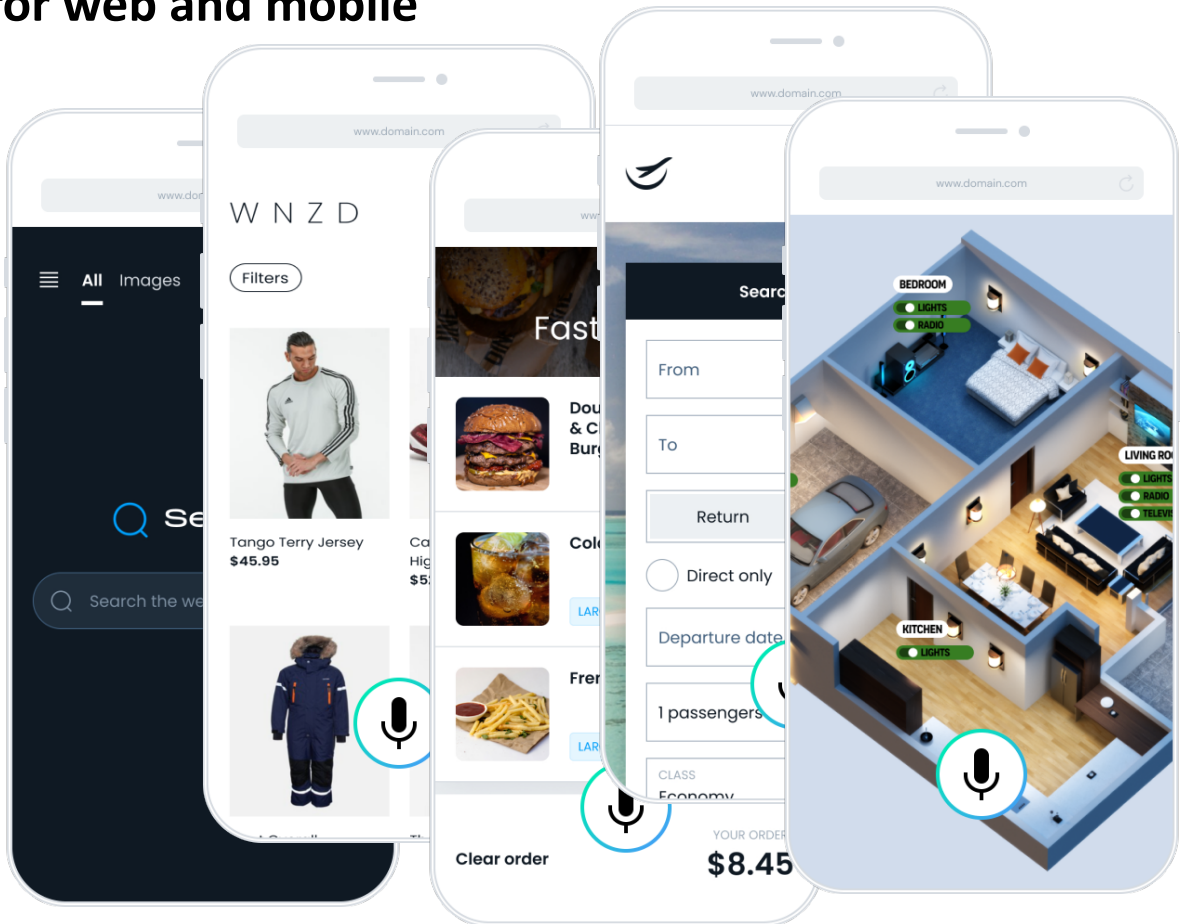
STEP 11 — Dialogue Management
Dialogue management tracks the context of a conversation across multiple interactions and formulates the appropriate response to return to the user at each stage of the conversation. The response includes the natural language reply as well as other interactive elements.

STEP 12 — Speech Synthesis
Also called 'text-to-speech', speech synthesis converts text-based natural language responses into spoken audio which can be read aloud to the user. Most major OS platforms and devices provide built-in text-to-speech capabilities.

Speechly

Voice interface API for web and mobile

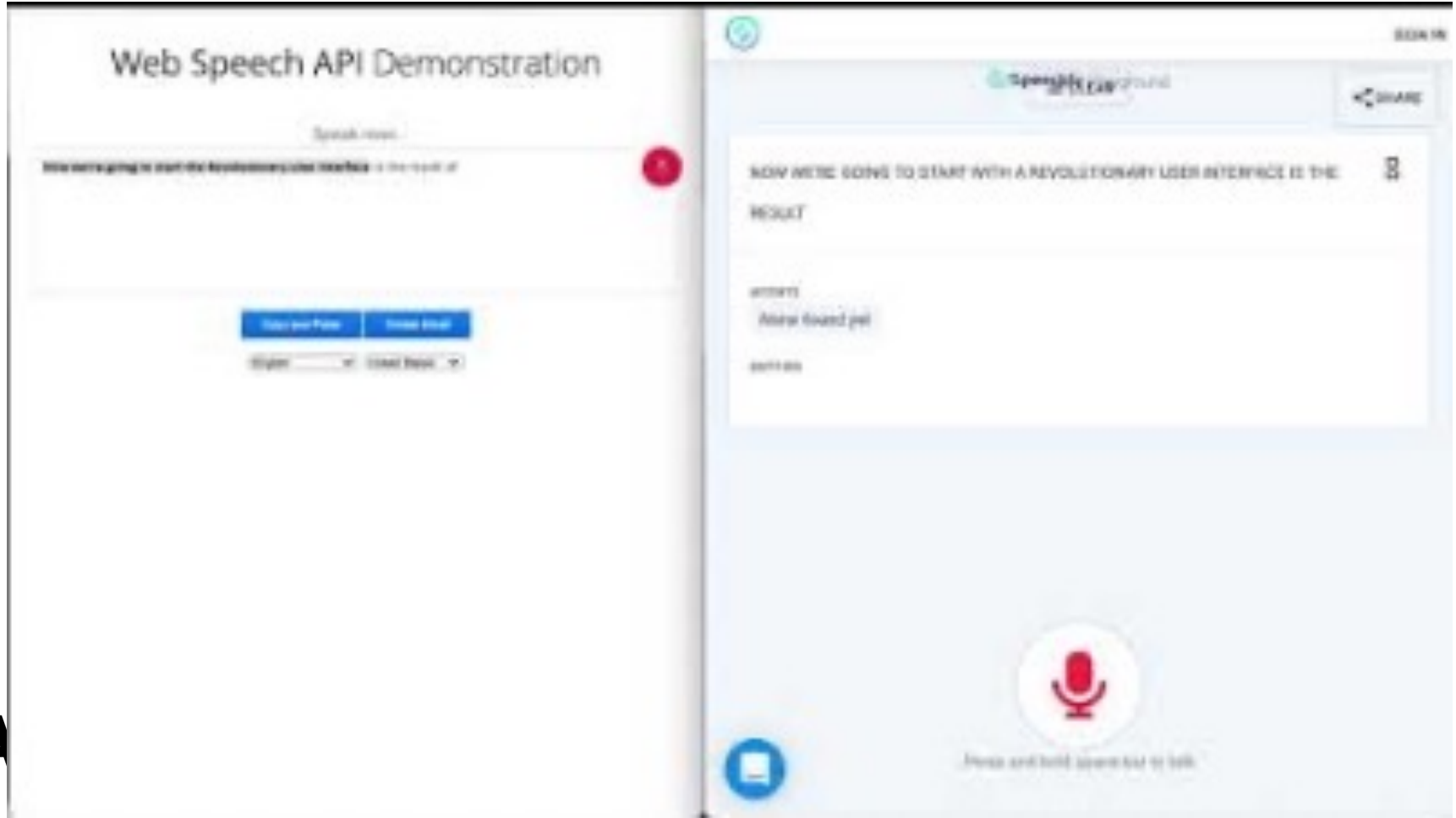
- Create voice-enabled applications
- Incorporate speech recognition technology **faster** and **more accurate** than Big Tech companies (Apple, Microsoft, Google, Amazon, Facebook)



Aalto University
School of Electrical
Engineering

Google webspeech API

Speechly speech recognition



❖ More videos and demos for Speechly

- ❑ **Comparing Alexa and Speechly in commercial apps**

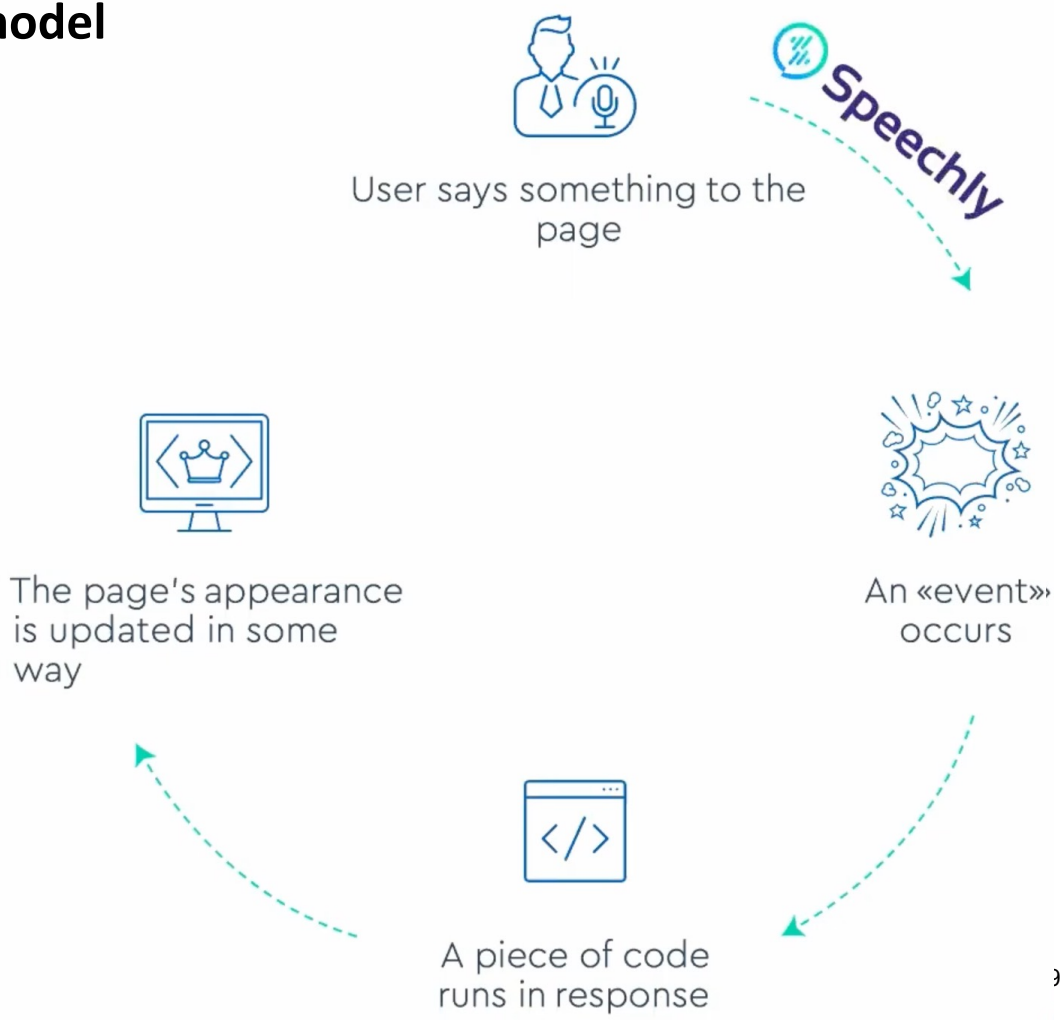
  https://www.youtube.com/watch?v=XJ4BnEIiAjo

- ❑ **Fashion Ecommerce, Flight Booking, Search Engine, Instant Messaging, and Smart Home (videos and demos)**

  https://www.speechly.com/demos

❖ Speechly UI programming model

- In speechly, the UI reacts to voice events and everything else remains the same as before
- Speechly adds voice functionality to your webpage or mobile applications



❖ Speechly Natural Language Understanding

- Provide example utterances
- Mark spans that should trigger events
- Tag the spans to identify between different events

[ ] : Span >>> contains relevant info to your app

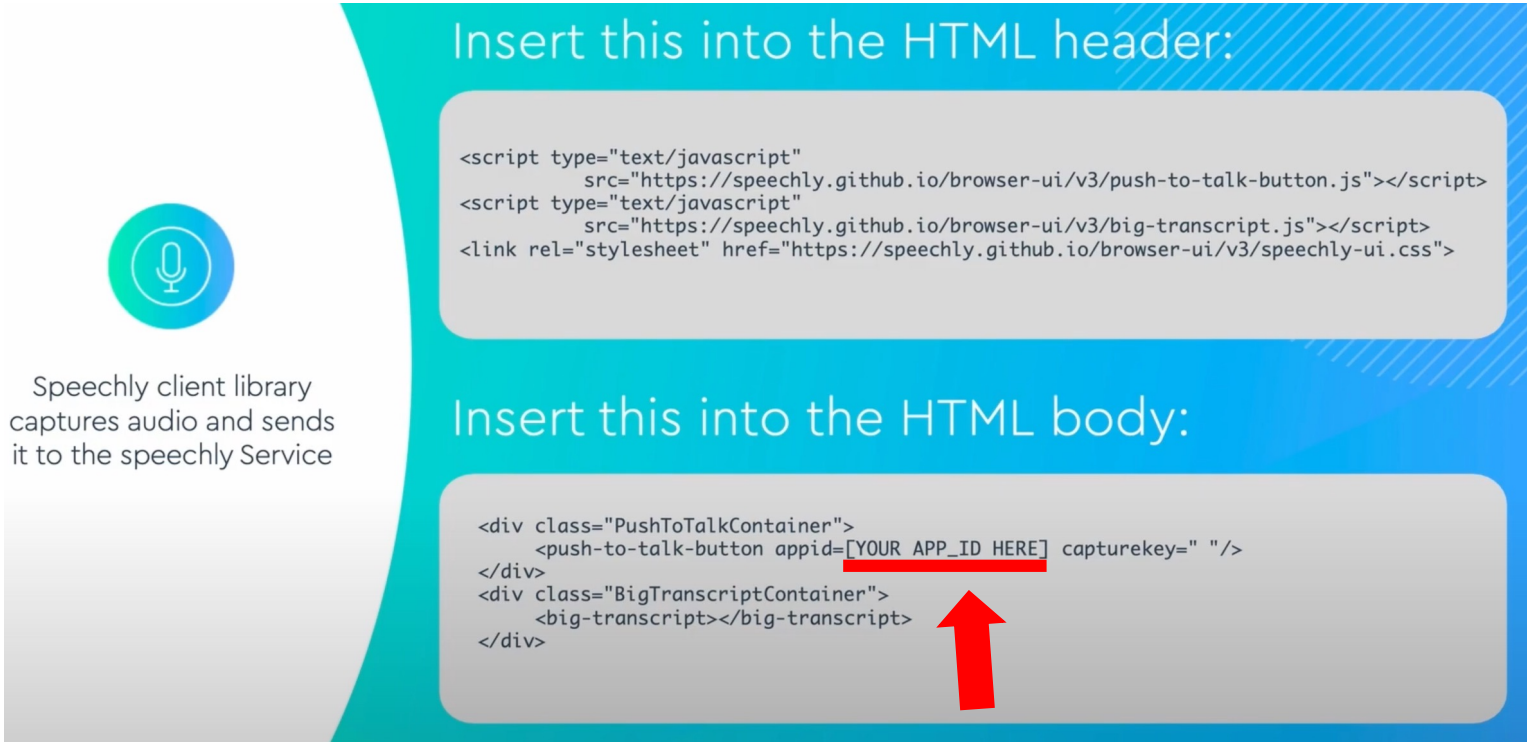
( ) : Tag >>> maps each span to its specific event

```
*filter show me [blue](color) [jeans](category)
*filter do you have that in [red](color)
*filter i'm looking for [mens](deparment) [t-shirts](category)
*filter can you sort by [price](sort)
```

- show me blue jeans
- do you have that in red
- i'm looking for mens t-shirts
- can you sort by price

❖ Deployment of Speechly in a webpage

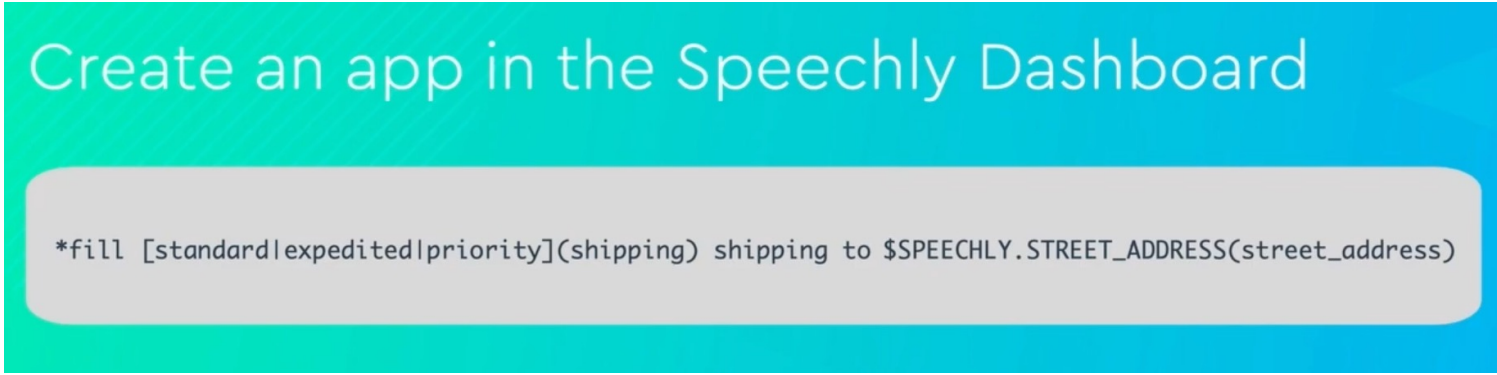
➢ **First step** ➡ Add Speechly client to the webpage ➡ A microphone appears on the webpage, but it doesn't work yet!



❖ Deployment of Speechly in a webpage

➢ **Second step** ➡ Create your application based on the expected utterance examples

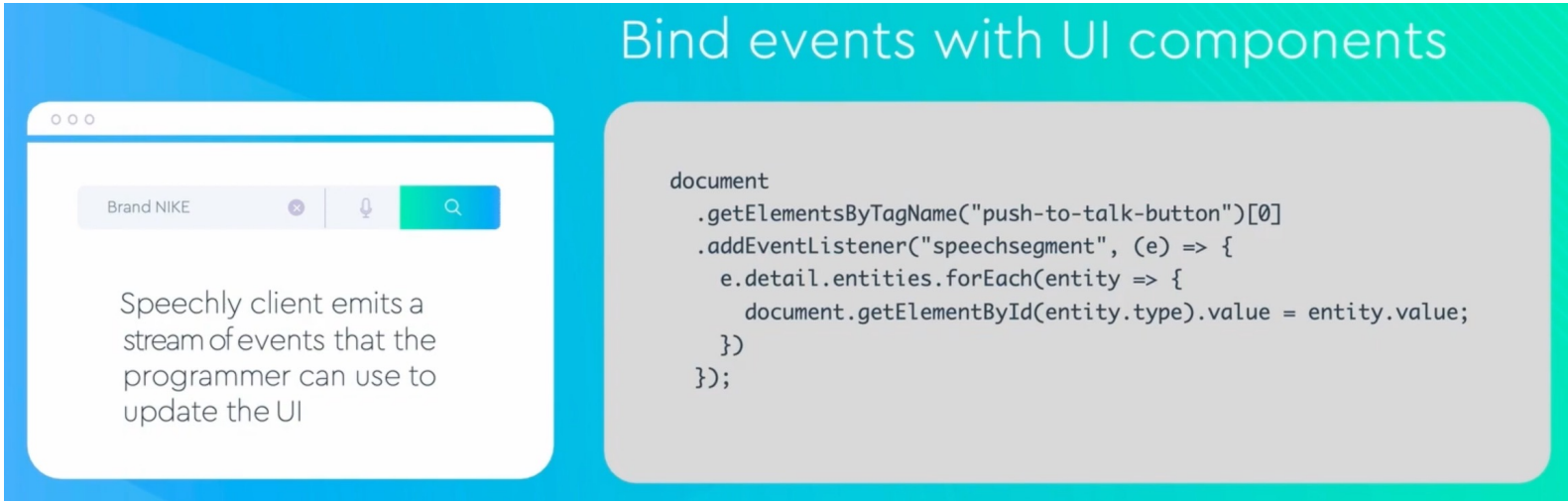
- ❑ A shipping application filling form



The most important step!

❖ Deployment of Speechly in a webpage

➢ **Third step** ➡ Bind the stream of data events provided by Speechly client to the UI components



❖ Useful references for Speechly

- ❑ **Speechly documentation (to learn more)**

  https://docs.speechly.com

- ❑ **Video series on how Speechly works**

  https://docs.speechly.com/quick-start/how-speechly-works/

- ❑ **Book a demo meeting**

  https://www.speechly.com/contact