Finding Your Voice: Building Screenless Interfaces with Node.js
What is NPR?

A quick explainer for the Canadians in the audience:
Who am I?

▷ Sr. full-stack web developer
▷ At NPR since March 2014
▷ Part of a skunkworks team focused 100% on voice UI development
  ○ Formed in September 2017
Why voice UI development?

Then:

Now:
Hey Siri, tune to NPR

“Alexa, what's in the news?”

Ok Google, play the latest news from NPR
“smart speakers”
“smart speakers”
“voice assistants”
How Amazon views Alexa

“Alexa, set the thermostat to 25 degrees.”

“Okay.”

“I'd like to reorder paper towels please.”

“Alexa, thank you!”

“No problem.”
“What would you want to know about voice UI development?”
What can I actually make?
Is it possible to build one app for Amazon Echo, Google Home, and Apple HomePod?
1. What can you actually make?

To understand the present, we must understand the past.
A brief timeline of voice assistants

2014
Amazon Echo launches November 6

2015
Amazon Alexa Skills Kit launches (June)

2016
Google Assistant (May) + Google Home (November)

2017
Samsung Bixby (August) + Microsoft Cortana (October)

2018
Apple HomePod (February)
A natural evolution

1. add voice activation to existing custom app ecosystem
2. add content via RSS feeds
3. add support for custom "skills"
A natural evolution

1. add voice activation to existing custom app ecosystem

2. add content via RSS feeds

3. add support for custom “skills”
Conclusions

▷ Amazon has a 2-year lead
▷ Only Amazon and Google have fully developed ecosystems
▷ A big focus is adding access to news and podcasts via RSS
▷ Home automation is secondary
2. Can you build one “skill” to rule them all?

tl;dr yes ... and no
Alexa + Google ecosystems

▷ Heavily leverage their existing cloud infrastructure
  ○ AWS Lambda + Google Cloud Functions

▷ Can also build a traditional REST API accessed by their services
The request/response flow

request

Your code

response
The request/response flow

P.S. all the NLP and ML happens here
The future is “serverless”

▷ Others can speak more eloquently on this subject than me
  ○ Several other talks on serverless

▷ Let's just assume we want to use Lambda or Cloud Functions...

▷ ... node.js wins!
The official SDKs are not bad

Alexa node.js SDK:

github.com/alexa/alexa-skills-kit-sdk-for-nodejs

Actions on Google node.js SDK:

github.com/actions-on-google/actions-on-google-nodejs
Examples from Alexa SDK

```java
responseBuilder.speak("Hello!");
responseBuilder.reprompt("Hello?");
responseBuilder.withSimpleCard("Card Title", "Content!");
responseBuilder.addAudioPlayerPlayDirective(...url);
```
WFUV is your station. There is a three second pause here then I continue.

When I wake up, I speak slowly.

I can speak with my normal pitch, and also with a low pitch.
Backends-for-Frontends (BFFs)
Two “skills”, one codebase

BFF on AWS Lambda

BFF on AWS Lambda

The "real" API

>60% shared code with separate view layers

different builds using Gulp
Generic Response Model

class GenericResponseModel {
    public audioUrl = ''; 
    public outputSpeech = ''; 
    public repromptSpeech = ''; 
    public cardTitle = ''; 
    public cardContent = ''; 
    public cardImage = ''; 

    ...
}

Challenges

▷ Text-to-Speech (TtS) is still king
  ○ Google didn't even add support for their native audio player until February 2018
▷ No access to the user's location
▷ Error handling is interesting!
  ○ User might not even trigger your skill
Conclusions

▷ The code is not hard

▷ Understanding platform limitations and user expectations are
Open source opportunities

▷ Would it be helpful to have a formalized framework?
Open source opportunities

▷ Would it be helpful to have a formalized framework?
  ○ Not really. The code is not hard.

▷ What we struggle with the most: QA
  ○ We need something like Selenium or Nightwatch.js for voice UI
P.S.
Are you excited about QA for voice UI?
... 'cause we're hiring!

n.pr/tech-jobs
Resources from NPR

▷ NPR + Edison Research Smart Audio Report
▷ Talking Back To Your Radio: How We Approached Voice UI (npr.design)
▷ How To Prototype For Audio-Rich Voice Experiences Without Really Trying (npr.design)
▷ My talk on the ethics of voice UI
Resources from others

▷ Alexa Skill Blueprints (Amazon)
▷ Conversation design (Google)
▷ Designing Voice Experiences (Smashing Mag)
▷ Storyline: Create Alexa skills without coding
▷ Intelligent Assistants Have Poor Usability: A User Study of Alexa, Google Assistant, and Siri (Nieman Norman Group)
Thank you!

nara@nara.codes

@xiehan

https://npr.codes