Development of Voice Commands in Digital Signage for Improved Indoor Navigation Using Google Assistant SDK

DAVID SHEPPARD, NICK FELKER, and DR. JOHN SCHMALZEL

RowanUniversity

Outline

- Motivating application
- Constraints
- Details of device development
- Results
- Future work

Image source: *Kriesten objekt design GmbH* on Wikimedia

Indoor Navigation

- Indoor navigation is not easy, especially in a new place
- Traditionally maps are not interactive
- Current digital signs are more dynamic, but often fail to provide custom information or do so in a nonintuitive way
- Many digital signs are large and expensive

Image source: *Kriesten objekt design GmbH* on Wikimedia

Application Problem

- Develop a Smart Sign as a small, voice-forward digital sign
- Use a microphone and Wi-Fi connection to accept and process verbal requests
- Use voice input and natural language processing allowing users to find information faster than a sign with only a touchscreen
- Potential for incorporating Google Assistant to provide natural user-to-device communication

Constraints

- Ease of use: The Smart Sign must be intuitive to interact with and encourage user interaction
- Ease of development: The Smart Sign software must be easy to modify such that its maps and other information can be easily updated
- Portability: The Smart Sign must be easy to install in an indoor location
- Affordability: The Smart Sign must not be cost-prohibitive

Design Approach

- Based on the Android Things Pico Pro Maker kit

Development board designed by Google (180EUR)

- Primary hardware of the device:
  - Pico i.MX7 development board with:
    - Two ARM® Cortex®-A7 cores (up to 1.2 GHz)
    - An ARM® Cortex®-M4 core
  - 5-inch touch display
  - Wi-Fi antenna
  - USB-C power cable

Design Approach

- Software:
  - Android 8.1 operating system
  - UI development familiar to Android users
  - Able to receive Android version updates
  - App written in Java
- 3D printed case to house the Android Thing

The Smart Sign

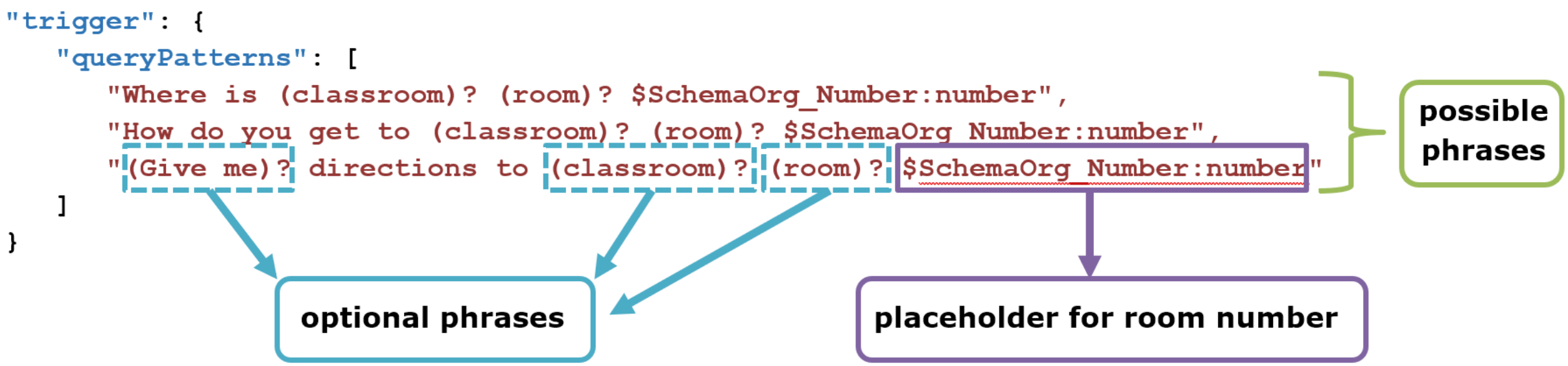
User Interface

- Simple and intuitive
- An IDLE state provides the user with general information about the time, location, and weather
- From IDLE, the user initiates voice commands by a button press—due to high ambient noise
- Users ask:
  - Questions developed specifically for the Smart Sign
    - “Where is room 337?”
  - Questions already understood by the Google Assistant
    - “What is the current temperature?”

Custom Commands

- Written for the Google Assistant to support the Smart Sign
  - “Where is room ____?”
  - Enable the Smart Sign to respond to users in a much more familiar way, conversational way
  - Custom commands are defined and written in a JSON file that is included in the app
- Approach allows the Smart Sign to be adapted to other similar applications

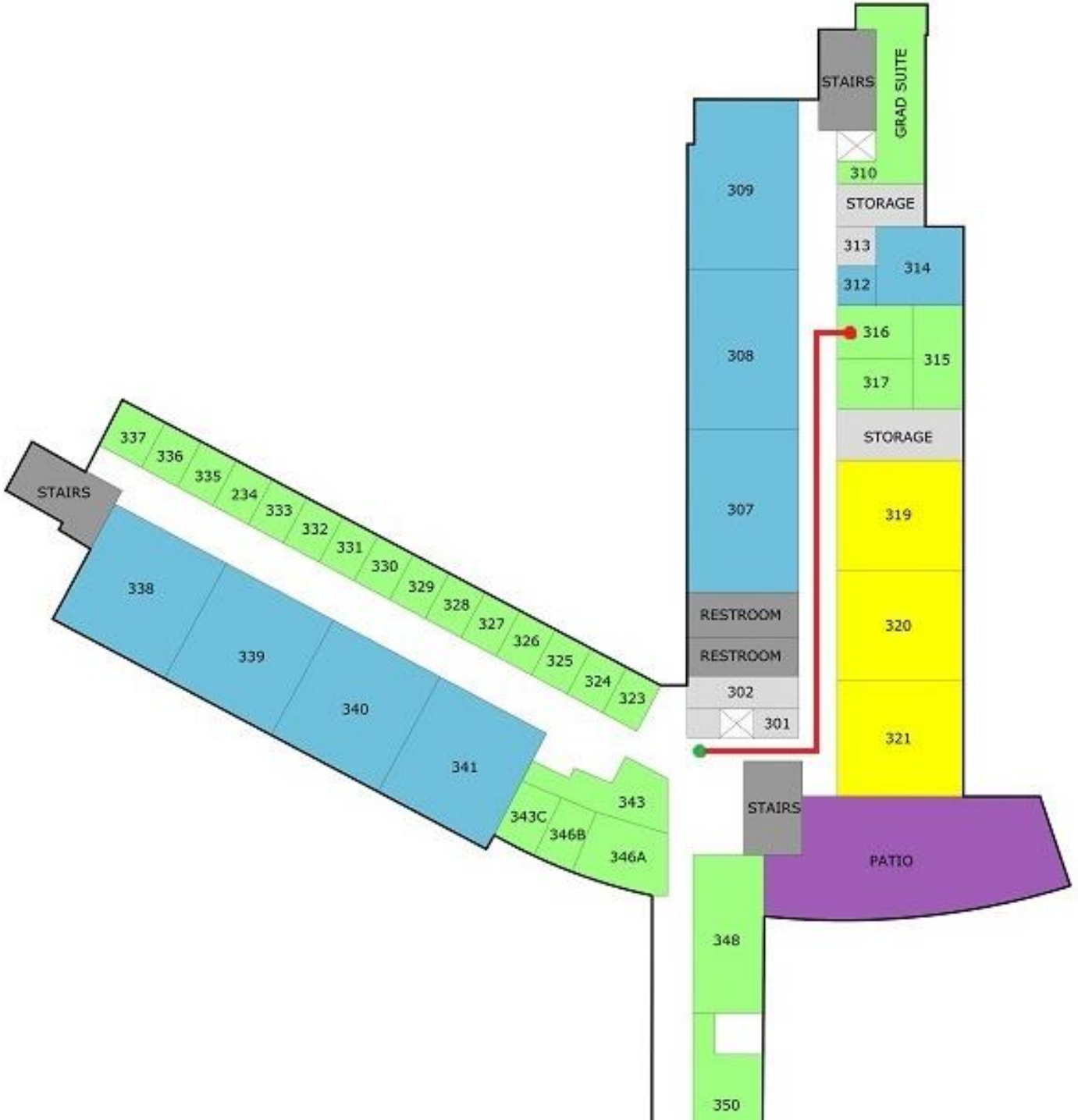
Writing Custom Commands: A sample from actions.json



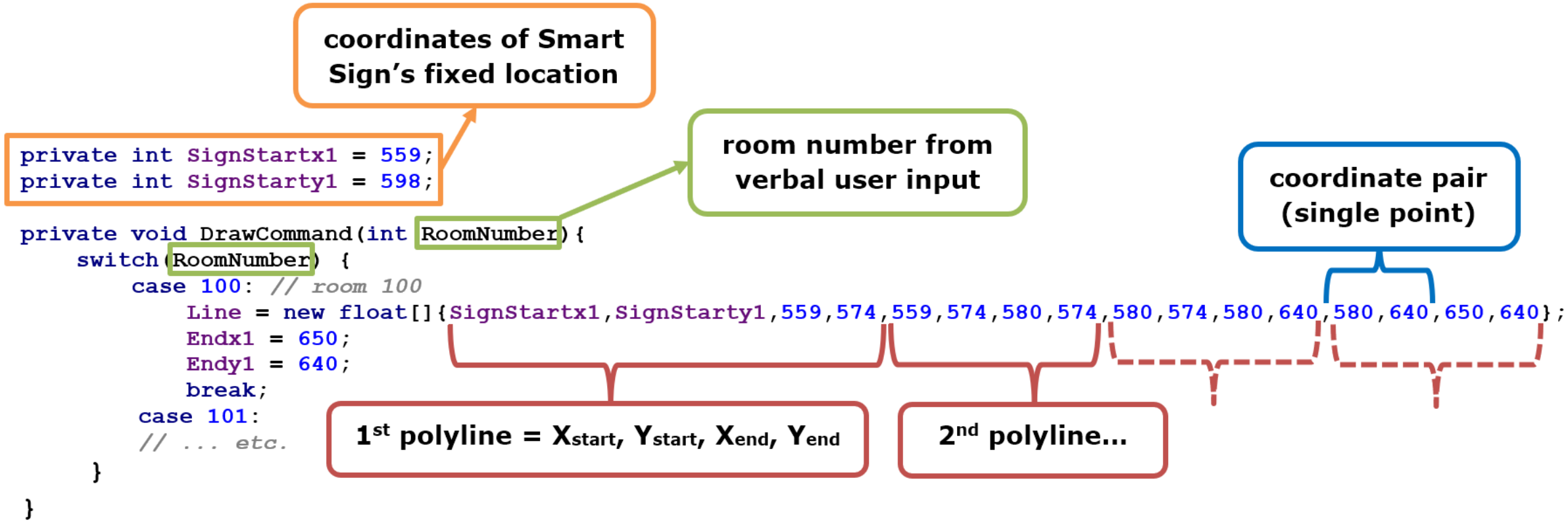
Giving Directions

- Directions are drawn over a map of the building
- Directions consist of:
  - Green circle at current location
  - Red circle at destination
  - Polylines representing the optimal route
- The endpoints of each polyline and locations of each circle are represented as coordinate pairs corresponding to the pixels in the image of the floor map.
- Directions are currently preprogrammed for each room
  - This could be eventually implemented into a mapping algorithm so that polyline parameters do not have to be written manually

Navigation View



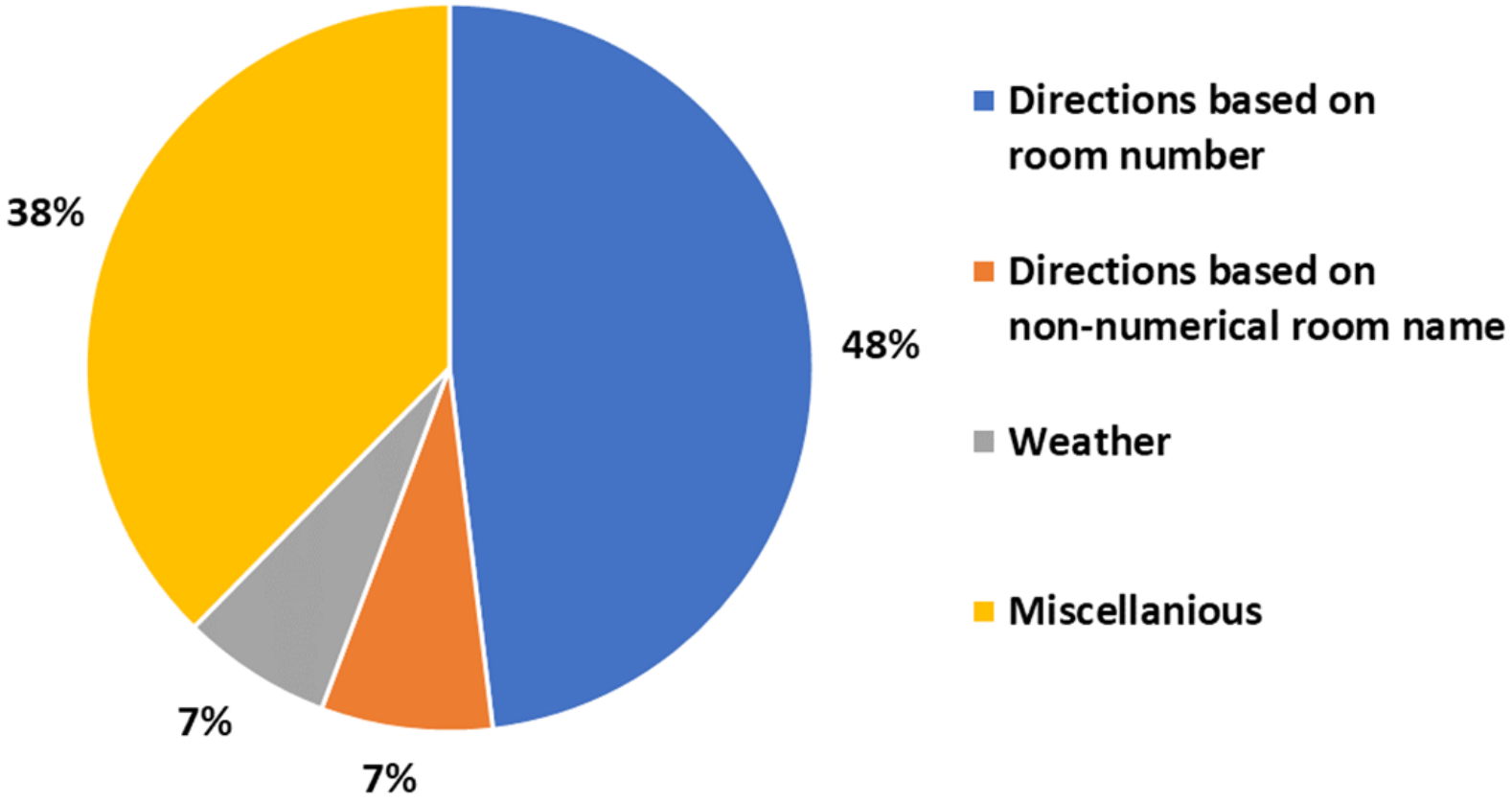
Mapping Function: Sample from NavigationPath.java



α Testing

- The Smart Sign α test
  - One week duration
  - Engineering building (Engineering Hall, Rowan Univ.)
  - >200 queries processed
- Testing results
  - Analyzed using associated “My Activity” webpage
  - Provided insights into features that should be added to the Smart Sign

Query Categories



Internal Testing Findings

- The Smart Sign effective for path guidance
- Users liked the Smart Sign's ability to receive quick and succinct directions
- Integration with Google Assistant allowed users to ask questions they are accustomed to using with other digital assistants
- The Smart Sign had trouble understanding some requests
  - Attributed to poor quality microphone
- Support for non-numerical room names needed to be added
  - (This was completed after testing)

Future Work

1. Expand the library of custom commands
   - Increase repertoire of requests the Smart Sign understands
   - Added recognition of non-numerical room names (e.g. “office”)
2. Implement server-side integration of the code
   - Allows the application to be modified remotely, further expanding the ease of development
3. Add features to the IDLE state
   - Announcements/Calendar events
   - Use suggestions for the Smart Sign.
   - Incorporation of advertisements

Future Work

4. Extend the concept to instrument/process control
   - Natural voice command set
     - Audible SCPI—e.g., MEAS:VOLT:DC

References

- R. Want and B. N. Schilit, ”Interactive Digital Signage,” in Computer, vol. 45, no. 5, pp. 21-24, May 2012.
- “NXP i.MX7D,” developer.android.com, Sept. 25, 2018. [Online]. Available: https://developer.android.com/things/hardware/imx7d. [Accessed Nov. 18, 2018].
- “Google Assistant SDK for Devices," developers.google.com, Aug. 8, 2018. [Online]. Available: https://developers.google.com/assistant/sdk/overview. [Accessed Nov. 11, 2018].
- “Internet Protocol Version 6: IPv6 for Consumers,” fcc.gov, Sept. 8, 2017. [Online]. Available: https://www.fcc.gov/consumers/guides/internet-protocol-version-6-ipv6-consumers. [Accessed Dec. 6, 2018].
- G. R. Hiertz, D. Denteneer, L. Stibor, Y. Zang, X. P. Costa and B. Walke, "The IEEE 802.11 universe," in IEEE Communications Magazine, vol. 48, no. 1, pp. 62-70, January 2010.

Acknowledgements

The support of Google through the donation of Android Things is gratefully acknowledged.

Thank You!

The code for the Smart Sign is open-source and can be found at https://github.com/RowanPWLab/smart-signs