

# **Software Specification Report**

**Team MGN**

**October 6, 2017**

# **Breakdown of Individual Contributions**

## **Joseph Olin (Team Face)**

- Responsible for communication with American Printing House for the Blind
- Responsible for report proofing and submissions
- Will work on the server side of the app
- Will work on the client side of the app
- Will work on integration testing the app

## **Ben Pister (Wizard of the Web)**

- Responsible for maintenance of the project website
- Will work on unit testing the server side of the app

## **Michael Roark (Chart Master)**

- Responsible for compiling the reports prior to proofing
- Responsible for general documentation of the project
- Will work on the server side of the app
- Will work on the client side of the app

# **Section 1 Introduction**

## **Section 1.1 Purpose**

The purpose of our software project is to provide a more convenient and efficient way for visually impaired people to learn mobility skills. Efforts are being waged to help visually impaired to live a life as independent as possible, as close as possible to those who have no visual impairment.

## **Section 1.2 Definitions**

Google Assistant: The Google Assistant is a command interface that is a ramped up version of the Google Now, Google's voice activated assistant[ CITATION Sar16 \l 1033 ].

Google Home: The Google Home device is a hands-free assistant made to provide a personalized experience based on each user's voice[ CITATION Ovent \l 1033 ].

Node.js: Node.js is an event-driven JavaScript runtime that is built to be scalable[ CITATION Abo17 \l 1033 ].

Visually Impaired: Refers to the reduction in eyesight that restricts normal function of visual performance. This may allow for an interface in some cases, since some have very low, but some, visibility.

Dialog: In our case, a dialog is an interface that will be used to verbally direct the user to interact with the trivia game. This will provide them direction in the same manner that a visual interface directs non-visually impaired people or more importantly, as non-visually impaired can have conversation over the visible content[ CITATION Sar17 \l 1033 ].

User: A User in our instance is any individual that will benefit from a verbal interface to facilitate learning in a manner that they may not be able to experience in any other manner.

## **Section 1.3 System overview**

Our system will be built on the extensible Google Actions SDK. We will be creating our own dialog dependent on the game type. Our own system will also be extensible in so much that it will allow for new questions to be added as necessary to facilitate a most enjoyable user experience.

# **Section 2 Product perspective**

## **Section 2.1 System Interfaces**

Our system interfaces directly to Google Assistant, a virtual personal assistant software created by Google. Since this app uses the Google Actions SDK, Google Assistant provides the one and only portal for using the Google Home Trivia App System. Once Google Assistant initiates the client side of our app, the client side handles the data it receives to a processing layer that Google provides which formats the request for the server. Once the server receives this data from this layer, it does the necessary processing and gives the response back to a layer provided by Google. Google formats the response and then gives it to Google Assistant, and the cycle continues.

Our system composes the Google Home portion of the Trivia Framework. An Amazon Alexa Trivia Game app is also in the works which will consume the same data as the Google Home Trivia Game app. However, the only common piece between the two apps will be the part of the framework that provides data to them.

## **Section 2.2 User Interfaces**

Google Assistant itself has relatively little in terms of UI elements. Furthermore, since this is an app designed for visually impaired and blind students, it's extremely probable that what UI elements exist for Google Assistant won't be utilized. Rather, the focus will be on refining the audio conversation carried on between our system and the user.

## **Section 2.3 Hardware interfaces**

Google Assistant runs on select Android devices, as well as on Google Home devices. Thus, it through the usage of these physical devices that users will actually use our system.

## **Section 2.4 Software interfaces**

The client side of the system is relatively simple. It contains but one software interface, which is provided by the Google Assistant app. This app is installed on Google Home devices, and many Android devices. This interface processes the client side code we submit to Google. The server-side of the system has a number of smaller interfaces. It is composed of a series of Node.js modules, one of which acts as the command interpreter of the system, passing commands that use different modules for different actions.

## **Section 2.5 Communication Interfaces**

There are two primary communication interfaces in our system. The first communication interface exists to feed audio input from the user using Google Assistant into the client side of our app. This same interface gives the actual output from the client side once it receives a response to Google Assistant so output can be transmitted to the user as audio.

The second communication interface exists between the client side and the server side of the application. This is represented via an HTTPS endpoint specified on the client side that, when fired, executes a request to the server over the network. The server side, when finished, returns a response via the same network communication interface.

## **Section 2.6 Memory Constraints**

Although many Android devices running Google Assistant boast considerable disk and memory space, Google Home devices themselves allow for very little onboard storage (it relies heavily on server-side storage and processing). Thus, as much processing and data storage that can be handled by the server-side of our system, the better.

## **Section 2.7 Operations**

The app will have multiple modes of operation. The app will support multiplayer, which will dictate to some extent how the users' scores will be maintained during the game session. Also, the app will support multiple difficulty levels, which will dictate what questions that are asked to the user.

## **Section 2.8 Site Adaptation Requirements**

Little to no site adaptation is expected for this project. There might be some expense to APH, depending on if we host the server-side of the app on a server provided by an external organization or not. However, as simple as it would be to host our server-side implementation on Firebase or an APH server, it's not likely that this expense will be realized.

The client-side of our application will be provided to Google Home devices by Google via Google's internal process for publishing apps for testing and/or release. Thus, we won't have to install the app on any physical devices.

## **Section 2.9 Product functions**

The product will administer a trivia game by asking the user questions via audio vectors regarding Orientation and Mobility. The user's response will prompt the game to reply that the answer was either correct or incorrect. Then, the game will continue for as many questions as desired, until the end of the game session has been reached.

## **Section 2.10 User characteristics**

The majority of users of this product will be visually impaired and blind children receiving Orientation and Mobility training. Orientation and Mobility instructors will also likely use the product to some extent so they can instruct students in the product's usage.

## **Section 2.11 Constraints, assumptions, and dependencies**

Since this app will run using Google Assistant on Google Home devices, it is heavily dependent upon Google's Actions SDK. If anything were to happen to Google or the Google Actions SDK, our project would have to be discontinued.

The release of the app is also dependent upon acceptance by Google's Action admissions.

Failure of the app to be accepted by Google's admissions process would require changes to the app so it would pass admissions.

It is assumed that this app will be used primarily via Google Home devices, and not for Android devices running Google Assistant.

# **Section 3 Specific Requirements**

## **Section 3.1 External interface**

The Google Home will be our platform, and is the external interface users will interact with.

## **Section 3.2 Functional Requirement**

Users must be able to start a Quiz.

Users must be able to choose a difficulty for the quiz.

Users must be able to choose the number of players, one through four.

Users must be able to answer questions given to them and be given a response, verifying or correcting them.

Optionally, players may choose their own difficulty setting.

Validity checks will be handled by the Google Home device, and all strings will be parsed by the server.

## **Section 3.3 Sequence of operations**

1. A user will begin the main program.
2. If a difficulty was not defined alongside the startup, prompt the user for difficulty.
3. The server will prompt the user for number of players.
4. The server will load a quiz and all of its questions and answers.
5. The server will question the users.
6. Users will respond with their answer, either 1,2,3,4/a,b,c,d, or True/False depending on the question type.
7. The server will congratulate or console users in events of success or failure respectively.
8. The system will repeat the previous 3 steps [Ask APH] times.

## **Section 3.4 Performance Requirements**

The Server must be able to manage [ask APH] sessions at any given time.

The server will handle one string per query per session. For each query the server will return a response.

All queries must be met within 5 seconds of a request sent from the Google Home device.

Otherwise the device will terminate the session.

## **Section 3.5 Design Constraints**

### ***Section 3.5.1 Standards Compliance***

The greatest risk of this project is information being incorrect, or the quiz incorrectly evaluating answers to give users incorrect information. As the quizzes are intended to help children learn how to orient themselves, and give them mobility tips incorrect information if followed could lead to injury, or death in extreme cases.

### ***Section 3.5.2 Logical Database Requirement***

The database will consist of questions, answers, correct answers, question type, and identifiers for each question, and quiz. The server will ask questions from each quiz and score students as they answer.

The database will be accessed once per quiz session. If a session times out persistent data will be cleared.

## **Section 3.6 Software System Attributes**

### ***Section 3.6.1 Reliability***

The quizzes will be as reliable as the platform they run on, and the information provided to us from The American Printing House for the blind.

### ***Section 3.6.2 Availability***

The Google Home platform keeps all its apps readily available to all users making this app highly available to all who want to use it, this is true since it is an on-demand system. A session will expire after a period of time determined by the Google Home time out. If the quiz would time out on the device the server can be safe in assuming that the session on the Home has also ended. Once this occurs a full restart must occur. No data on the session will remain once it has expired.

### ***Section 3.6.3 Security***

The Google Home sends string responses to a server where it parses the string and returns sound values, strings or other data, thus making a secure system that is difficult to attack. Protecting the network that the server is hosted on is where security concerns should be focused.

### ***Section 3.6.4 Maintainability***

Maintainability will be easy. The framework is designed with the main principle being easy to expand and therefore maintain. The largest obstacle for someone in the future will be changes to the Google Home platform. Either The SDK depreciation, or total overhaul of the system architecture.

### ***Section 3.6.5 Portability***

This framework will be used to create trivia games specifically for the Google Home. This means portability is extremely low, as it will only work on one system.

## **Section 3.7 Other Requirements**

We are required to use the Actions SDK for better control of the code, and operation of the quiz.

We are required to generate quizzes from a json file.

We are required to use Node.js for server side.

## REFERENCES

- About Node.js*. (2017). Retrieved from Node.js: <https://nodejs.org/en/about/Overview>. (Current). Retrieved from Madeby-Google: [https://madeby.google.com/home/?utm\\_source=en-offnet-na-sem](https://madeby.google.com/home/?utm_source=en-offnet-na-sem)
- Perez, S. (2017, March 2). *Google Assistant Hits More Android Devices Today*. Retrieved from Tech Crunch: <https://techcrunch.com/2017/03/02/google-assistant-hits-more-android-devices-today/>
- Purewal, S. J. (2016, October 4). *The Difference Between Google Now and Google Assistant*. Retrieved from cnet.com: <https://www.cnet.com/how-to/the-difference-between-google-now-and-google-assistant/>

