LaRome Dickerson, Eric Schulze, Ryan Gosling

Team LER

O & M Trivia programmers manual

**Vision Statement**

The American Printing House for the Blind (APH) is the world’s largest company devoted to manufacturing, researching, and developing products for people who are blind or visually impaired. APH designs and manufacture textbooks and magazines in braille, large print, and digital formats, as well as other educational, recreational, and daily living products, one of which we have been tasked to develop. APH has requested our services to develop a skill for Amazon’s Alexa voice interface platform.

This skill will consist of a trivia game in which the subject of Orientation and mobility (O&M) will be the focus. The O&M trivia game will be used in concurrence with the training, and education provided to the visually impaired through all stages of life. Blindness or visual impairment can happen to an individual through different stages of their life, from birth to old age, and learning through engagement will solidify the skills they will need to survive in a visual based world, which this Alexa skill will provide.

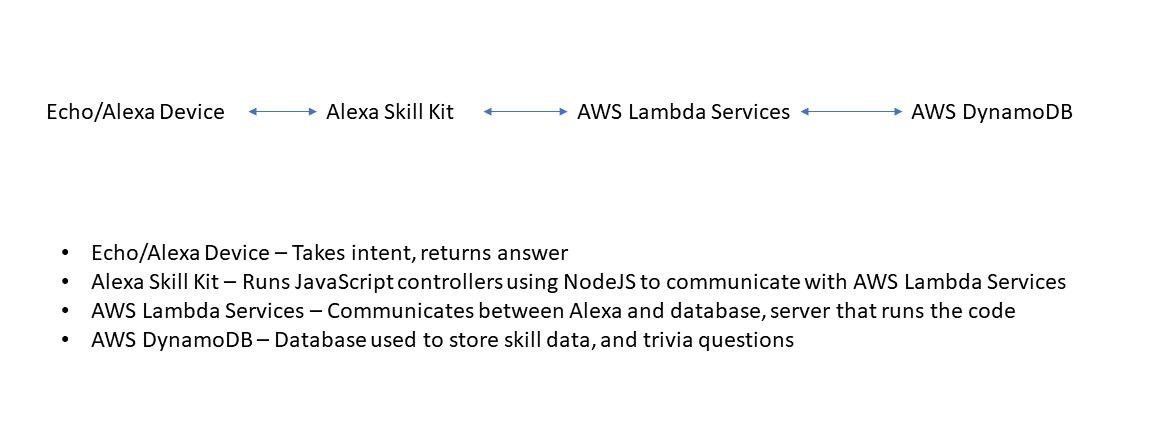
**Introduction**

The American Printing House for the Blind (APH) is the world’s largest company devoted to manufacturing, researching, and developing products for people who are blind or visually impaired. APH designs and manufacture textbooks and magazines in braille, large print, and digital formats, as well as other educational, recreational, and daily living products, one of which we have been tasked to develop. APH has requested our services to develop a skill for Amazon’s Alexa voice interface platform.

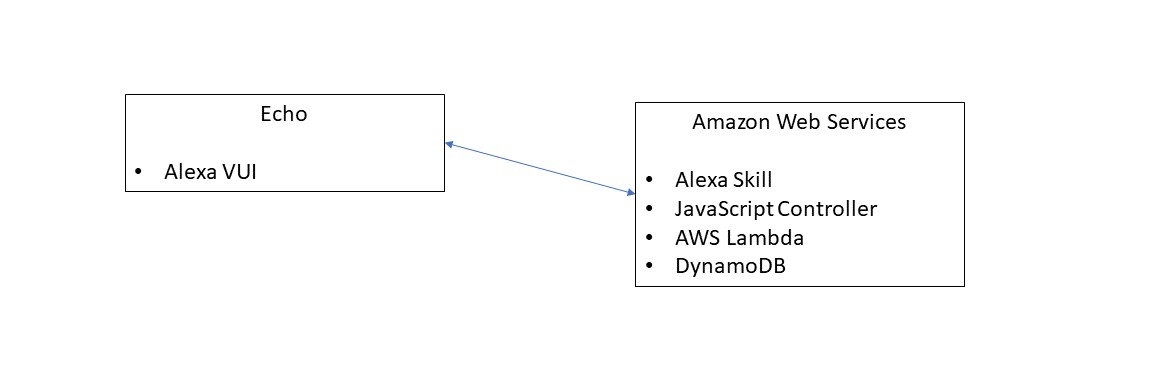
This skill will consist of a trivia game in which the subject of Orientation and mobility (O&M) will be the focus. The O&M trivia game will be used in concurrence with the training, and education provided to the visually impaired through all stages of life. Blindness or visual impairment can happen to an individual through different stages of their life, from birth to old age, and learning through engagement will solidify the skills they will need to survive in a visual based world, which this Alexa skill will provide.

**Component and tool overview**

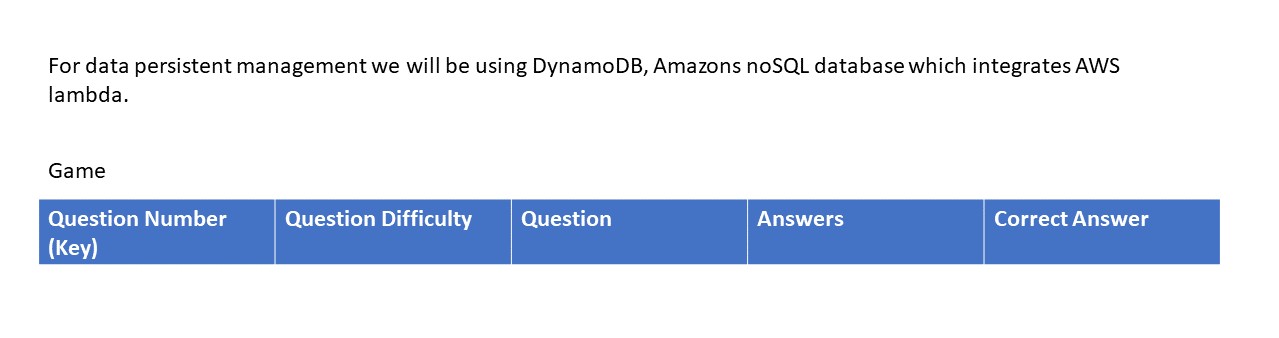
**1. Subsystem Decomposition**

****

**2. Hardware/Software Mapping**

****

**3. Persistent Data Management**

****

**4. Access Control and Security**

4a. Users

Users of the APH O&M skill will only have access to the skill through an Alexa enables device or application. All of these devices or applications are linked to a user’s Amazon account, and as such, all authentication is handled by Amazon’s security services. When a user is authenticated to use a device, the device has a unique id that is sent to the Alexa server with every request. The APH O&M will use this device id to track games and settings.

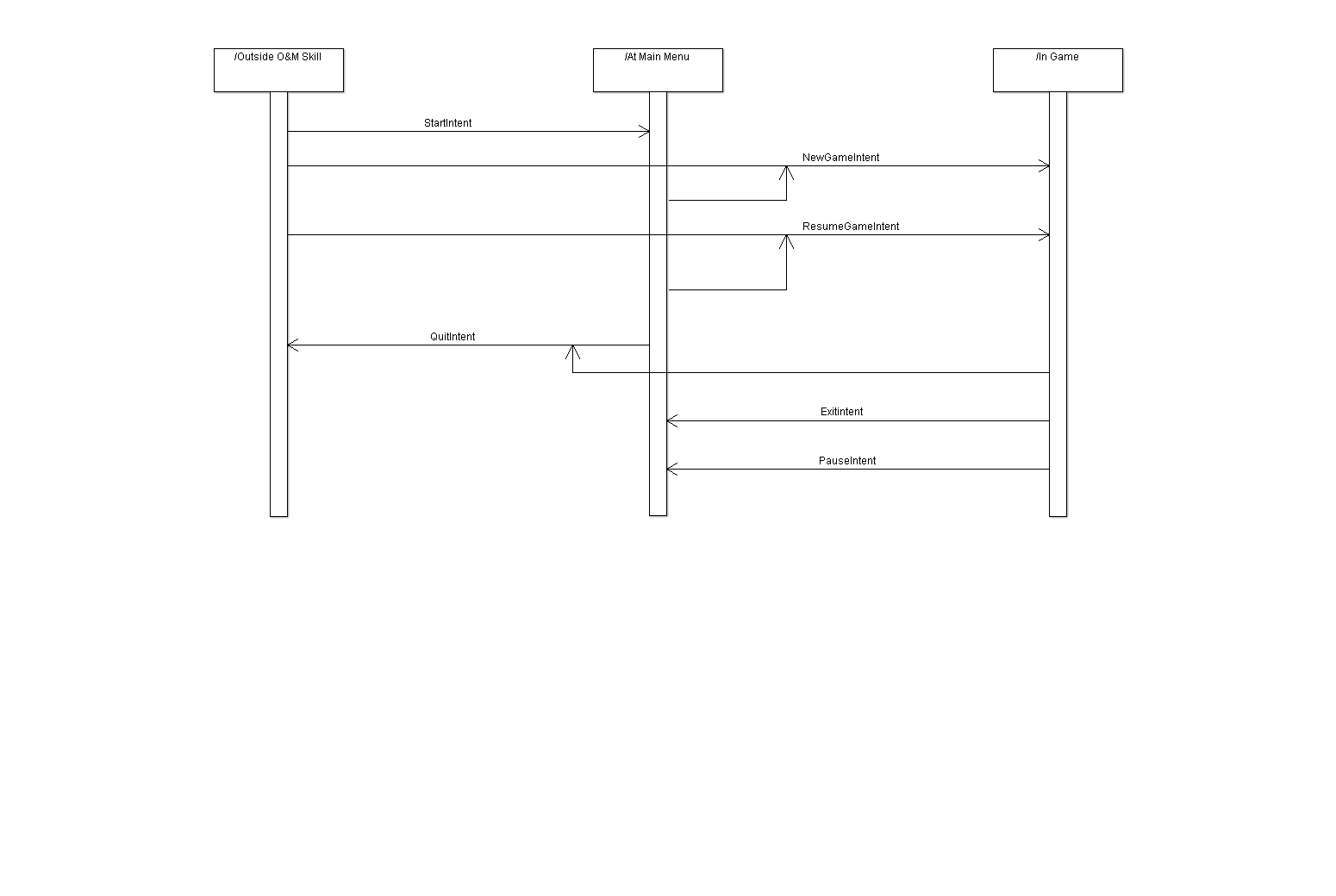
Users will be able to use all VUI features of the Interaction Model. There will not be any distinction between different kinds of users.

4b. APH System Admin

A System Admin from APH will have access to the Interaction Model, the Hosted Service, and the DynamoDB through the Amazon AWS portal. This will allow for updates and revisions of any of the system components. These services will be passed on to APH at the conclusion of the project.

**5. Global Software Control**

5a. Stage 1: Outside APH O&M Skill

For Stage 1 and Stage 2, the only control that will need to be handled will be the utterance-intent-response loop. These actions will navigate between all of the stages. (See diagram for intent actions)

5b. Stage 2: At Main Menu

For Stage 1 and Stage 2, the only control that will need to be handled will be the utterance-intent-response loop. These actions will navigate between all of the stages. (See diagram for intent actions)

6c. Stage 3: In Game

A screenshot of a cell phone

Description generated with high confidenceFor Stage 3, there will be a repeated loop through each question. This loop will terminate either if there is no response after three attempts to garner a response, or if the list of questions has been completed (i.e. the game is finished). See the following UML diagram for the loop actions.

**6. Boundary Conditions**

6a. Start Up

The APH O&M skill will be initiated when a user gives the key words “APH Trivia Game” to an Alexa device. This can be used in any of the different utterances that are provided for start up. The Amazon Alexa service will handle the initial VUI, and when the service finds a request for the O&M skill, it will start the skill

6b. Shut Down

The APH O&M skill will be shut down when a user uses the key word “Quit” to an Alexa device. Using the utterance “Alexa Quit” will automatically shut down the skill, saving the state of the current game. The Amazon Alexa service will handle all of the exit processes to take the user out of the O&M skill.

6c. Errors and Exceptions

The APH O&M skill will handle errors in two ways. The first will assume an error in the VUI. This will cause a repeat of the last phrase that was given to the Alexa device. If this error is caused twice in a row, one more repeat phrase will be given, with examples of the utterances that can be said. At this point, it will be assumed that there is no one playing the game and the game will save state and return to the main menu.

The second error that the skill will handle is an error in the processing of the Hosted Service. These errors will result in the generation of an error report and then the game exiting to the main menu without saving state. We will not save state to ensure a clean, error-free start of the next game.

6d. Data Migration

The APH O&M skill will keep all data stored in cloud storage. The skill Interaction Model will be hosted and stored on Amazon’s Alexa service, with the ability to download the original .xml file at any point for revision or update. The Hosted Service will be available as a JavaScript module on the Amazon Lambda service at any time for update or revision. Using Amazon DynamoDB will allow our database to be cloud based, with tools provided by Amazon that allow for easy import of new data and export of current data.

**Software**

1a. Glossary

APH: American Print House for the Blind, our project sponsor

O&M: Orientation and Mobility, a subject of instructional lessons for the visually impaired that APH works with

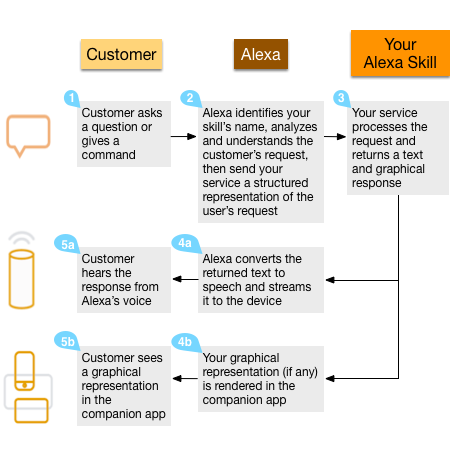
Skill: A capability of Alexa, the Amazon cloud based service that handles speech recognition, machine learning, and natural language understanding, that defines how you can interact with Alexa.

Interaction Model: The Voice User Interface (VUI) of the skill, it defines what functionalities or behaviors the skill is able to handle

Hosted Service: The programming logic of the skill, hosted by Amazon Lambda Services, that responds to a user’s requests and phrases

Utterance: A phrase from the user that the skill can understand and map to an intent

Intent: A representation of functions that the skill is capable of performing. Multiple utterances can map to one intent that then gets passed on to the Hosted Service.

1b. System Overview

The Alexa skill will consist of two individual parts working in tandem with each other, the Interaction Model and the Hosted Service. The Interaction Model is a voice user interface (VUI) for the Alexa skill. The Interaction Model is to the skill, what a graphical user interface is to a mobile application; it defines how a user is able to interact with our skill. This Interaction model includes the intents and utterances that a user can use. The intents and utterances trigger responses based on the second part of the skill, the Hosted Service. The hosted service is the brains behind how Alexa handles responses and phrases from the user. The hosted service takes as input intents, deciphered from the user utterance by the VUI, and outputs response phrases.

1c. References

Amazon Alexa Skills Kit (<https://developer.amazon.com/alexa-skills-kit>)

Alexa Skills Documentation (<https://developer.amazon.com/public/solutions/alexa/alexa-skills-kit/overviews/understanding-custom-skills>)

Codecademy – Build Alexa Skills (<https://www.codecademy.com/learn/learn-alexa>)

**2. Data Design**

{"questions":{[

"question":{

"id": "id\_value",

"difficulty": "difficulty\_value",

"question\_text": "text\_value",

"correct\_answer": "answer\_value",

"question\_type": "type\_value",

"answers": [

{"answer\_text": "text\_value"},

{"answer\_text": "text\_value"},

{"answer\_text": "text\_value"},

{"answer\_text": "text\_value"},

],

"additional\_answer\_info": "text\_value",

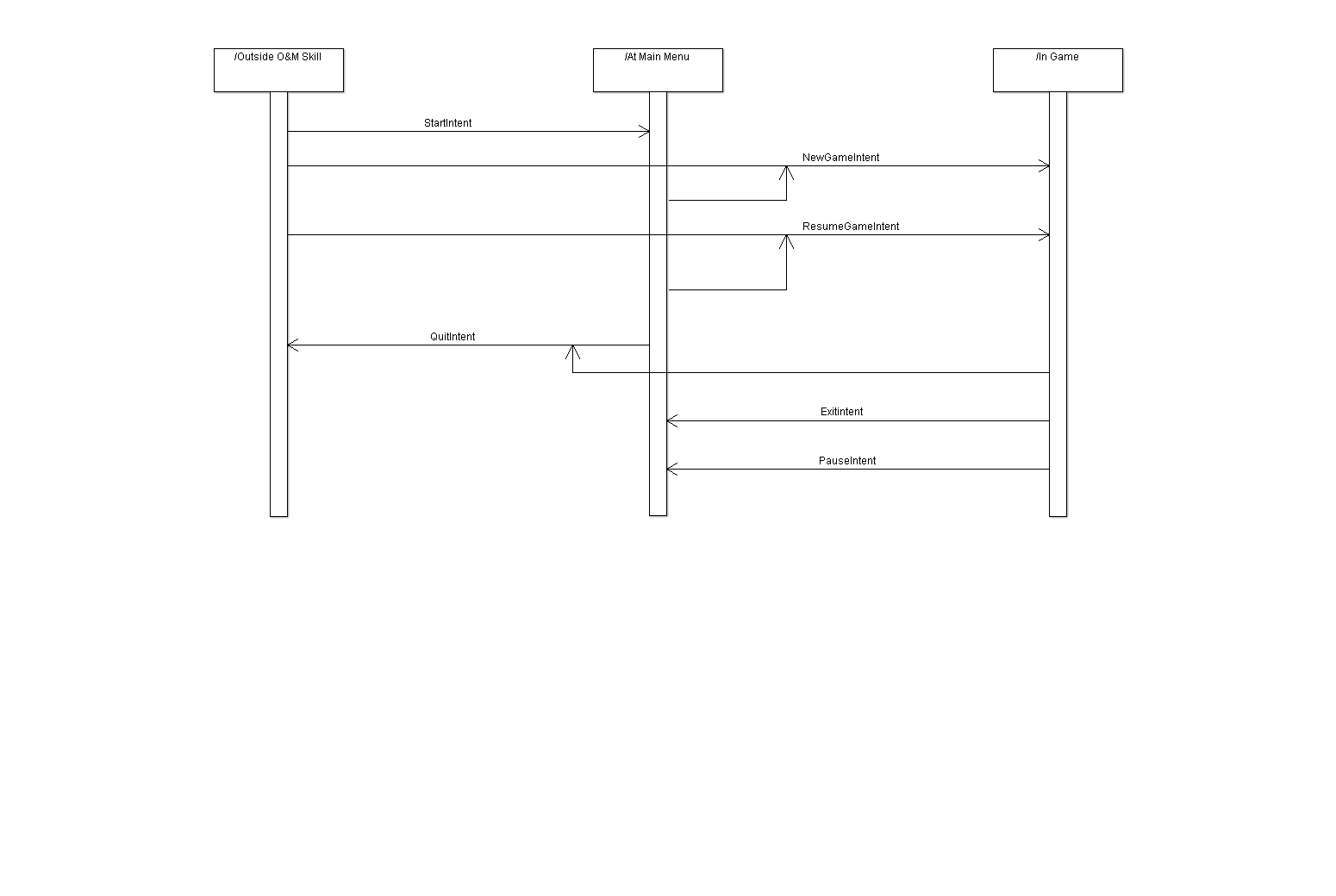
}

]}

}

**3. Architecture Design**

3a. Stage 1: Outside APH O&M Skill

For Stage 1 and Stage 2, the only control that will need to be handled will be the utterance-intent-response loop. These actions will navigate between all of the stages. (See diagram for intent actions)

3b. Stage 2: At Main Menu

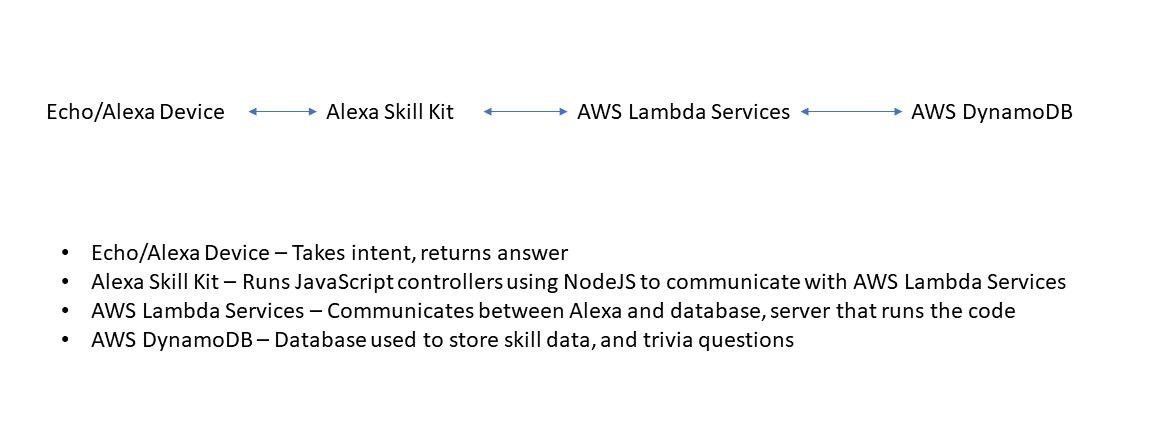
For Stage 1 and Stage 2, the only control that will need to be handled will be the utterance-intent-response loop. These actions will navigate between all of the stages. (See diagram for intent actions)

3c. Stage 3: In Game

A screenshot of a cell phone

Description generated with high confidenceFor Stage 3, there will be a repeated loop through each question. This loop will terminate either if there is no response after three attempts to garner a response, or if the list of questions has been completed (i.e. the game is finished). See the following UML diagram for the loop actions.

**4. Interface Design**

****

A screenshot of a cell phone

Description generated with high confidence**5. Procedural Design**

GetIntent(Intent i){

Listen for incoming intent; //Always running on alexa devices.

Switch(i){ //This may be covered by the Control method

Case “Game + Players + Difficulty”: StartGame(NumPlayers, Difficulty, SaveState);

Break;

Case “Main Menu”: MainMenu(SaveState);

Break;

Case “Quit”: QuitGame(SaveState);

Break;

Default;

Break;

}

}

StartGame(NumPlayers, Difficulty, SaveState){

Int[] ListOfQuesIDs;

If(SaveState.isEmpty()){

ListOfQuesIDs = GetNewQuestionSet(Difficulty); //loads 20 question ids into an array

}

Else{

ListOfQuesIDs = SaveState.getListOfQuesIDs();

}

Question q;

Boolean correctAns;

Int playerID = 1;

Int[] score;

Int questionNum = 0;

While(intent == Game && questionNum != ListOfQuesIDs.length){

ReadIntro(questionNum, playerID);

q = GetQuestion(ListOfQuesIDs[questionNum])); //gets question from database by ID

ReadQuestion(q.getQues());

ReadAnswers(q.getAnswers());

\*Listen for answer.

isCorrectAns(q.getCorrectAns(), Player’s given answer);

For(int a = 0; a < 3; a++){

If(Player answers are invalid twice){

ReadOptions();

}

If(Player answers are invalid thrice){

correctAns = false;

}

}

addScore(playerID, score[playerID – 1], correctAns);

if(playerID == NumPlayers){

readScore(score);

playerID = 1; //player 1 goes again, new round starts.

}

Else{

playerID++;

}

}

Congratulate(player with highest score);

}

StartMainMenu(Savestate){

If(Intent comes from ‘Out of game’ state){

ReadMenuOptions();

\*Listen for answer.

}

Else if(Intent come from ‘In game’ state){

SaveGame(SaveState);

ReadMenuOptions();

\*Listen for answer.

}

}

SaveGame(SaveState){

Upload each variable of the save state to the server:

• AlexaID //unique alexa ID for the device

• NumPlayers //The number of players in this game

• Difficulty //The difficulty of this game

• ListOfQuesIDs //The list of question IDs, in order to have the same questions

• CurrentQues //Which question the current game is on

• Scores //The scores for the players

}

QuitGame(SaveState){

SaveGame(SaveState);

\*”Alexa, quit the game”;

}

ReadIntro(QuestionNum, PlayerNum)

This procedure will create a statement for a question introduction for Alexa to read at the beginning of each question. As inputs, it will take the question order number and the player number, both integers. The output will be the string introduction message. An example of a possible output would be “Question Number 6, for Player 2”. There will be a standard introduction statement with two variables for the question number and player number. This standard introduction will then be returned with the correct variables supplied from the input.

Process:

Input: int questionNum, int playerNum

Output: “Question Number “ + questionNum + “, for Player Number “ + playerNum

ReadQuestion(QuestionText)

This procedure will create a statement for a question for Alexa to read for each question. As input, it will take the question text as a string. The output will be the string question reading.

Process:

Input: Question Text

Output: Question Text string

ReadAnswers(Answers[])

This procedure will create a statement Alexa to read all of the answers to a question. As inputs, it will take a string array of answers. The output will be the string message with the answers randomized.

Process:

Input: string[] answers

Randomize array

Output: “Is the answer a “ + answers[1] + “ or b “ + answers[2] + “ or c “ …

ReadOptions()

This procedure will create a statement Alexa to read of the different options a user has while in the trivia game. This statement will be read if Alexa has not received input within a given time. It will take no inputs. The output will be the string message listing the different options a user can say to Alexa.

Process:

Input: none

Output: Options message

GetQuestions(ListOfQuestionIDs)

This procedure will go out to the database and bring back a list of questions, with all of the information associated with those questions, in the form of a JSON object. The input will be a list of questionIds. This will be the list of questions that are returned, in the same order. The output will be a list of question objects.

Process:

Input: list questionIds

Convert questionIds to a string list of ids.

Pass the string to a database call that takes the ids and returns the JSON object for each question.

Parse each JSON object into a question object and put it into the questionObjects list.

Output: list questionObjects

IsAnswerCorrect(CorrectAns, GivenAns)

This procedure simply compare. This statement will be read if Alexa has not received input within a given time. It will take no inputs. The output will be the string message listing the different options a user can say to Alexa.

Process:

Input: string correctAns, string givenAns

Output: Boolean correct

**Test Cases**

Test case 1: All correct answers, full game played.

* System: Amazon Echo using Alexa system
* Subsystem: O and M trivia game
* Designed by: LaRome Dickerson
* Executed by: LaRome Dickerson, Ryan Gosling, and Eric Schulze
* Execution Date: 3/29/18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Action | Expected Response | Result | Comments |
| 1 | “Alexa, Start O and M Trivia” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Pass |  |
| 2 | “2 players on beginner difficulty.” | System recognizes Number of players and difficulty then starts a game using these parameters. | Pass |  |
| 3 | All correct answers | Game flows correctly, responds to correct answers correctly, and tallies the score correctly. At the end the scores and the winner are announced. | Pass |  |

Test Case 2: All incorrect answers, full game played.

* System: Amazon Echo using Alexa system
* Subsystem: O and M trivia game
* Designed by: LaRome Dickerson
* Executed by: LaRome Dickerson, Ryan Gosling, and Eric Schulze
* Execution Date: 3/29/18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Action | Expected Response | Result | Comments |
| 1 | “Alexa, Start O and M Trivia” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Pass |  |
| 2 | “2 players on beginner difficulty.” | System recognizes Number of players and difficulty then starts a game using these parameters. | Pass |  |
| 3 | All incorrect answers | Game flows correctly, responds to incorrect answers correctly, and tallies the score correctly. At the end the scores and the winner are announced. | Pass | Ties are not accounted for yet. |

Test Case 3.1: Help menu tested, full game played.

* System: Amazon Echo using Alexa system
* Subsystem: O and M trivia game
* Designed by: LaRome Dickerson
* Executed by: LaRome Dickerson, Ryan Gosling, and Eric Schulze
* Execution Date: 3/29/18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Action | Expected Response | Result | Comments |
| 1 | “Alexa, Start O and M Trivia” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Pass |  |
| 2 | “Help” | “Help Menu” is recited and system is waiting for “Resume” or “Repeat” utterances. | Pass |  |
| 3 | “Repeat” | “Help Menu” is repeated and system is waiting for “Resume” or “Repeat” utterances. | Pass |  |
| 4 | “Resume” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Fail | System Crashed |

Test Case 3.2: Help menu tested, full game played.

* System: Amazon Echo using Alexa system
* Subsystem: O and M trivia game
* Designed by: LaRome Dickerson
* Executed by: LaRome Dickerson, Ryan Gosling, and Eric Schulze
* Execution Date: 3/29/18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Action | Expected Response | Result | Comments |
| 1 | “Alexa, Start O and M Trivia” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Pass |  |
| 2 | “Help” | “Help Menu” is recited and system is waiting for “Resume” or “Repeat” utterances. | Pass |  |
| 3 | “Repeat” | “Help Menu” is repeated and system is waiting for “Resume” or “Repeat” utterances. | Pass |  |
| 4 | “Resume” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Pass | This is where the last test failed. Problem was the change of state. |
| 5 | “2 players on beginner difficulty.” | System recognizes Number of players and difficulty then starts a game using these parameters. | Pass |  |
| 6 | All correct answers | Game flows correctly, responds to correct answers correctly, and tallies the score correctly. At the end the scores and the winner are announced. | Pass |  |

Test Case 4.2: Help Mid-Game, full game played.

* System: Amazon Echo using Alexa system
* Subsystem: O and M trivia game
* Designed by: LaRome Dickerson
* Executed by: LaRome Dickerson, Ryan Gosling, and Eric Schulze
* Execution Date: 4/6/18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Action | Expected Response | Result | Comments |
| 1 | “Alexa, Start O and M Trivia” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Pass |  |
| 2 | “2 players on beginner difficulty.” | System recognizes Number of players and difficulty then starts a game using these parameters. | Pass |  |
| 3 | “Help” | “Help Menu” is recited and system is waiting for “Resume” or “Repeat” utterances. | Pass |  |
| 4 | “Repeat” | “Help Menu” is repeated and system is waiting for “Resume” or “Repeat” utterances. | Pass |  |
| 5 | “Resume” | Game is resumed with data persistence. Same player’s turn, same scores, and same game parameters. | Fail | No data persistence. |
| 6 | All correct answers | Game flows correctly, responds to correct answers correctly, and tallies the score correctly. At the end the scores and the winner are announced. | Pass |  |

Test Case 4.2: Help Mid-Game, full game played.

* System: Amazon Echo using Alexa system
* Subsystem: O and M trivia game
* Designed by: LaRome Dickerson
* Executed by: LaRome Dickerson, Ryan Gosling, and Eric Schulze
* Execution Date: 4/6/18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Action | Expected Response | Result | Comments |
| 1 | “Alexa, Start O and M Trivia” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Pass |  |
| 2 | “2 players on beginner difficulty.” | System recognizes Number of players and difficulty then starts a game using these parameters. | Pass |  |
| 3 | “Help” | “Help Menu” is recited and system is waiting for “Resume” or “Repeat” utterances. | Pass |  |
| 4 | “Repeat” | “Help Menu” is repeated and system is waiting for “Resume” or “Repeat” utterances. | Pass |  |
| 5 | “Resume” | Game is resumed with data persistence. Same player’s turn, same scores, and same game parameters. | Pass | Data persistence was fixed. |
| 6 | All correct answers | Game flows correctly, responds to correct answers correctly, and tallies the score correctly. At the end the scores and the winner are announced. | Pass |  |

Test Case 5: Quit Mid-Game.

* System: Amazon Echo using Alexa system
* Subsystem: O and M trivia game
* Designed by: LaRome Dickerson
* Executed by: LaRome Dickerson, Ryan Gosling, and Eric Schulze
* Execution Date: 4/6/18

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Action | Expected Response | Result | Comments |
| 1 | “Alexa, Start O and M Trivia” | “Start Menu” is recited and system is waiting for Number of players and Difficulty. | Pass |  |
| 2 | “2 players on beginner difficulty.” | System recognizes Number of players and difficulty then starts a game using these parameters. | Pass |  |
| 3 | “Alexa, Quit Game” | Game returns to “Main Menu” state. | Pass |  |
| 4 | “Alexa, Exit Game” | O and M Trivia game is exited completely. | Pass |  |

Installation for new install:

The game can be run on any hardware with Amazon Alexa running on it.

Installation for new platform:

The game can be seamlessly run from any valid device.

Further Development Statement.

If we had another year to work on this application, we would add a different language to expand the utility of the app. We would also add a visual component to the trivia game for the Amazon Show. Lastly, if possible we would expand the list of excepted utterances so that users could actually say the whole answer if they wanted to.