Lab 3: Expand your Game Engine: Hello Buttons

Overview

Using the skill you built in Lab 1 and the Buttons you added in Lab 2, in this lab you will learn how to employ click events you receive from buttons using the Alexa Skills Kit. When completed, the user will be able handle button events in the Hello Buttons skill. This extended skill will handle a button click and expect a voice response from the user.

The skill you create in this lab should match the following dialog model:

Alexa, open Hello Buttons.

Welcome to Hello Buttons Skill. Tell me your favorite color.

Purple.

Click on the button you wish to change to purple.

User clicks a button and it animates and changes to specified favorite color.

Objectives

After completing this lab, you will be able to:

- Have a deeper understanding of the Game Engine.
- Understand how to handle button clicks.
- Emphasize voice first in their Button skill design.

Prerequisites

This lab requires:

- Access to a notebook computer with Wi-Fi, running Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat).
- An Internet browser such as Chrome, Firefox, or IE9 (previous versions of Internet Explorer are not supported).

- Completion of Lab 1: Build your First Skill: Hello Buttons
- Completion of Lab 2: Animate your Buttons: Hello Buttons

Configuration Source Files

https://s3.amazonaws.com/alx306/ALX306-lab3.zip

- HelloButtons/
  - lambda/custom/
    - node_modules/
    - hello-buttons.js
    - package-lock.json
  - models/en-US.json
  - archive.zip
  - skill.json
- ALX306-Lab3.pdf (this document)

Task 1: Using Buttons with Voice

Overview

Alexa is the voice service that powers Amazon Echo. Alexa provides capabilities, called skills, which enable customers to interact with devices using voice (answer questions, play music, and more).

It is important to remember, when designing an Alexa skill, you need to make sure voice is first. This can easily be lost as you incorporate a screen or other gadgets. The skill needs to be useful with just a voice enabled device, and the other components add emphasis.

The skill you are about to make asks the user for their favorite color. It is a fun, short dialog with Alexa, but can surprise and delight users when she prompts them to choo`se a button and changes the color to what they specified.

Task 3.1: Create a new intent

3.1.1 Navigate to the Amazon Developer Portal at https://developer.amazon.com/alexa.

3.1.2 When signed in, click **Your Alexa Dashboards** in the upper right.

3.1.3 Choose **Get Started** under Alexa Skills Kit.

3.1.4 Click on your Hello Buttons skill

3.1.5 On the left, click to Interaction Model. Click the black **Launch Skill Builder BETA** button if it does not already navigate you to the skill builder beta.

3.1.6 In the navigation menu on the left, Add+ next to Intents

3.1.7 Under “Create a new custom intent” name your intent FavoriteColorIntent

3.1.8 Create a slot called favoriteColor

3.1.9 Set the slot type to AMAZON.Color

3.1.10 Add the sample utterance “my favorite color is {color}” and hit enter

3.1.11 Provide more examples of sample utterances. Think about what users might say when prompted with, “What is your favorite color?”

3.1.12 Save Model

3.1.13 Build Model

Task 3.2: Update the language model and add your intent

Use the files from Lab 2 to update your skill. If you get stuck, you can download the Lab 3 files.

3.2.1 Click on Code Editor and copy the JSON

3.2.2 Using your favorite text editor (ex: atom.io), paste the new interaction model into *HelloButtons/models/en-US.json*

3.2.3 Still in your text editor, open *HelloButtons/lambda/custom/hello-buttons.js*

3.2.4 In `LaunchRequest` from Lab 2, the `.speak` prompt is not a question. The dialog needs to change so Alexa prompts the user for their favorite color. It should look something like: `this.response.speak("your prompt").listen("your reprompt after");`

3.2.5 Since we are listening for the users response to our prompt, we do not want to end the session. Set `this.handler.response.response.shouldEndSession` to `false` instead of deleting it in `LaunchRequest`.

3.2.6 Remove any occurrence of `button_up_event` and `buttonup`, this will no longer be necessary for our skill. Remove associated directives.

3.2.7 Add an intent called `FavoriteColorIntent` to your code. The pattern is `'FavoriteColorIntent': function() {}`

3.2.8 Your new intent has a slot value associated to it. Navigate to the GitHub Cookbook and add helper function `getSlotValues` to your code: http://bit.ly/2iQjcRa

Within `FavoriteColorIntent`, call `getSlotValues` to get the user’s favorite color: `let slotValues = getSlotValues(this.event.request.intent.slots);`

3.2.9 The `slotValues` variable is a JSON object that contains the filled slot values required by the utterance according to the user’s response. In our case, the only slot value was `color`. Access the resolved color based on what the user said from `slotValues`:

```
let favoriteColor = slotValues.color.resolved;
```

Task 3.3: Adjust button according to voice input

3.3.1 Add a `colorNameToHex` helper function to your code: http://bit.ly/2BaIi3M (change the name from colourNameToHex to colorNameToHex)

3.3.2 Within `FavoriteColorIntent`, create a variable that is the hex number of your user's favorite color using this helper function:

```
let favoriteColorHex = colorNameToHex(favoriteColor);
```

3.3.3 Throughout your code, delete the lines with `buildButtonUpAnimationDirective`.

3.3.4 Rename and update occurrences of the `breathAnimationRed` variable to `breathAnimation`.

3.3.5 Update the initialization of `breathAnimation` to start at white instead of red:

```
let breathAnimation = buildBreathAnimation('000000', 'ffffff',
30, 1200);
```

3.3.6 Back in `FavoriteColorIntent`, set your `breathAnimation` variable to update to the user's favorite color:

```
breathAnimation = buildBreathAnimation('000000',
favoriteColorHex, 30, 1200);
```

3.3.7 Now, when the user presses the button, it will use the redefined `breathAnimation` color according to what the user stated versus white. In keeping voice first, you did not have to change any aspects of the GameEngine. Add dialog to `FavoriteColorIntent` so Alexa prompts the user to choose a button:
`this.response.speak("Choose a button to change to " + favoriteColor);`

3.3.8 Set your `FavoriteColorIntent` response to ready:

```
this.emit(':responseReady');
```

Task 3.4: Upload code to Lambda and test

3.4.1 Create a zip file containing hello-buttons.js, package.json, and node_modules

3.4.2 Navigate to your Hello Buttons AWS Lambda project.

3.4.3 Under Function Code – Function Package, click Upload.

3.4.4 Upload your zip file, and click save.

3.4.5 Test your skill in AWS Lambda and with your device.

If your code is broken, use both the Service Simulator in the developer portal and AWS Lambda to look at the details and logs for information about your errors.

3.4.6 Go to developer.amazon.com/alexa and navigate to your skill.

3.4.7 Click on the test tab and be sure the skill is enabled for testing on your account.

3.4.8 Scroll down to the Service Simulator and enter an utterance you'd like to test on (for example, if you wanted to test the `LaunchRequest`, you could simply enter "start" or "launch").

3.4.9 Click the "Ask Hello Buttons" button. If you get the anticipated Service Response on the right, your code is working. If you get an invalid response, find out more

information about the error in AWS Lambda. Copy the Service Request JSON on the left. Notice that this generated a JSON request object for whatever intent your utterance pointed to.

3.4.10 Go to aws.amazon.com and navigate to your Lambda function for Hello Buttons.

3.4.11 In the upper right corner, click on the dropdown next to the Test button.

3.4.12 Click on Configure Test Events.

3.4.13 Select the Create New Test Event radio button and copy your JSON request object into the code editor. Name the test event something corresponding to the intent you are testing for.

3.4.14 Click Create.

3.4.15 Click the Test button. Now you will see more details on any errors you might have. To look at the logs for this intent, click the Logs link next to the execution result.

Task 3.5: Get creative!

Play with your new Favorite Color game and think about how you could expand on it using what you've learned in this workshop. Try completing the following challenges, or think of something new on your own:

3.5.1 Change the animation: have the buttons blink faster or slower.

3.5.2 Update the colors: As the user clicks the buttons, change the color to shades or complements of their favorite color.

3.5.3 Continue the dialog: once the user has stated their favorite color, tell them they could either choose a button or say a new color.

3.5.4 Expand the dialog: have Alexa ask the user what the user's favorite number is between 1 and 5, to pick X number of colors according to their favorite number, to choose a button, and then animate between those colors on that button.

Lab 3 Complete

Congratulations! You have now completed the final lab of the Echo Buttons workshop at AWS RE:Invent!

In Lab 3, you have learned:

- How to add a slot to an intent in your dialog model.
- How to technically access slots within your intent and gain meaning from them for your response.
- How to use voice to control Echo Buttons and keep voice first.
- How to carry over information from your Voice Intents to the Echo Button Game Engine.
- How to update animations and colors for your Echo Buttons.
- How to test your skill's code using the Developer Portal Service Simulator and AWS Lambda.