Amazon Lex

What is Amazon Lex?

Amazon Lex is a tool that enables the developer to make conversational applications. The purpose of the application is to provide a interface where user can interact with the custom made models via texting or speech.

For example, let’s say that the user is interested in knowing the program of event ‘X’.

Incase the developer has provided a what we call ‘**Intent**’, the application is able to handle the users input and return data accordingly to the request.

Intents are usually just a bunch of functions connected to a certain name, and that name usually has some custom made **slots** (keywords) connected to it. When these slots(keywords) are used in input, Amazon Lex finds out what intent they are related to, and has this information included in the **JSON-formatted** request.

This JSON-data is given to **Main handler-function** as a parameter.This main handler function is the main function of **Lambda function**, and lambda function is basically the whole program. All the functionalities of intents, data parsing, etc is done in the lambda function. How it is done is just a matter of the developer(s), how they, he or she wants to implement everything.

Following the previous example, user’s question could be something like this:

“Libby, what is the program of the evening?”

Here we first a have a **wakeword**, which is **Libby** in this case. This will tell the application to start listening the users input. If the input is in form of text, wakeword is not needed. Amazon Lex itself does not support wakeword, so that has to be implemented with different technology.

The next part of the sentence usually has slots(keywords) init. In this case the words program and evening would be those. Slots will help Amazon Lex to know that the question needs to be handled with certain intent. In our case the intent could be named something like **program-intent**. After that we could call for a function like returnProgram(). Notice that all of the intent, slot and function namings are arbitrary.

This function would return a JSON-file, with information of the output-text/speech included in it. After that it is Amazon Lexas job to get the output-text/speech out of the JSON-file.