ALEXA INSIDER SKILL

INTRODUCTION

The goal of this project is to develop an ALEXA skill using AWS Lambda and DynamoDB to help and assist students in seeking information through Alexa about the courses, registrations, deadlines, university policies and professors. Additionally, this skill offers several engaging features such as scheduling appointments, getting information about courses, inspiring quotes from Avengers, playing games like rock, paper scissors, fortune cookie and others with Alexa. This skill also provides features to get music-related information such as lyrics of a specific song, popular tracks by artists along with weather forecasts for any city. By utilizing this skill, students can access important information easily and accurately through Alexa rather than spending time searching for it online.

In recent years, the world has seen a significant rise in the use of voice assistants such as Amazon's Alexa. With the advancements in Artificial Intelligence and Natural Language Processing, these assistants have become more intelligent and user-friendly. The Alexa skill that I have developed is aimed at making the life of students easier by providing them with quick and easy access to information related to their studies, appointments, and other interesting activities.

DATA COLLECTION

The data for this project was sourced from the Utah State University website and the People's University web page. Information regarding computer science courses, including course descriptions, course offerings, and prerequisites, was collected from the Utah State University website. While data pertaining to professors' research interests, publications, projects, and teaching were sourced from the People's University webpage. The data was organized into key-value pairs and stored in a DynamoDB table, each assigned a unique ID, to support Alexa's responses to user queries. The collected data is presented in the form of answers to various questions, with each intent having a specific table containing unique IDs for both course and professor details.

In order to generate responses to user queries, custom skill intents were developed using the Alexa Skill Kit. This involved creating separate intents for each question, defining relevant utterances to anticipate how users would ask the question, and identifying the necessary slots required to accept user inputs and extract specific values from the DynamoDB database. This process allowed for the creation of a voice interaction model that could effectively respond to user inquiries. By combining the efficiency and scalability of AWS Lambda with the functionality of DynamoDB, the project can provide accurate and timely responses to user inquiries. With the ability to store and retrieve data in the form of key-value pairs, DynamoDB provides a highly efficient and cost-effective solution for handling large volumes of data. This ALEXA skill provides students with an efficient and convenient way to access important information and engage with fun features, all in one place.

DESIGNING THE CUSTOM VOICE INTERACTION MODEL

Using the Alexa Skill kit, a custom skill can be created in the Alexa developer console by following the steps below.

- Login to Developer Console
- Click “create skill” to start creating a skill (app)
- Provide skill name and preferred language
- Choose the model type “custom” to create a custom interaction model.
- Choose a specific programming language, such as “python” to host the skill.
- Choose a “start from scratch” template to start building the skill.
- Define a unique invocation name.
- Create skill intents - intents are the actions a skill can perform.
- Define sample utterances for each intent and associated slot.
- For example, for course prerequisites related questions, can create an intent named “RequirementIntent” and define sample utterances user might say to trigger the intent like "What are the prerequisites for {course}" and a slot called "{course}" to capture the course name.
- Configure the skill's endpoint: The endpoint is the web service that the skill uses to process user requests. In this case, the endpoint is the AWS Lambda function. A Lambda function needs to be set up in the AWS Management Console and then enter its ARN (Amazon Resource Name) in the Alexa Developer Console.

WORKING OF THE MODEL

Alexa Skill Kit was utilized to design and develop intents for each possible question that a user may ask. To ensure that the skill can understand the various ways that users may phrase their questions, a series of utterances were entered into the skill kit. And slots were created to enable the skill to retrieve specific values from DynamoDB based on the user's inputs. The model is designed to be intuitive and user-friendly, allowing students to easily access information about courses, professors, registration, deadlines, and university policies with ease.

General Flow of Custom Skill:

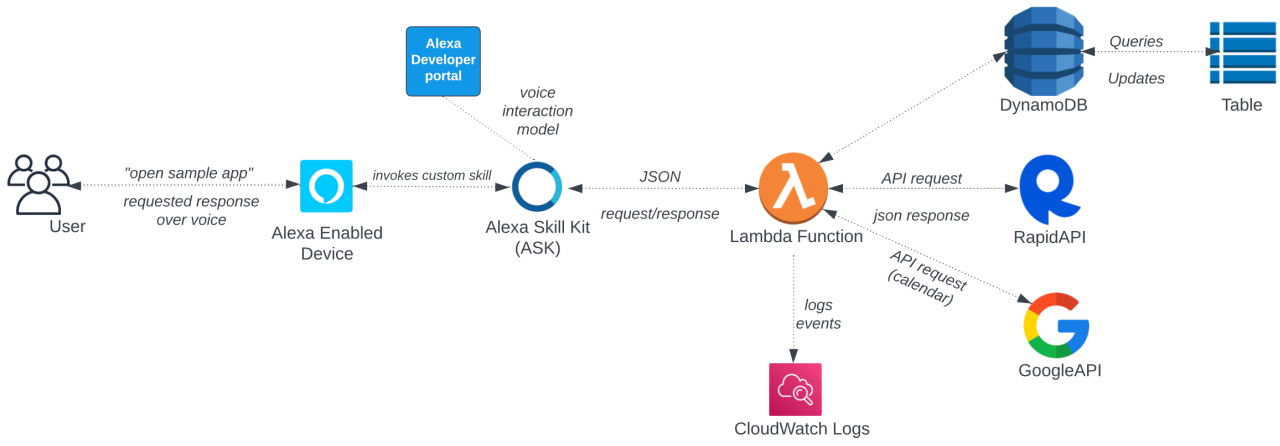
- A user invokes the skill by saying the invocation name, "Open sample app"
- The user speaks a question or command to the skill, which is then processed by the Alexa service. For example, user may ask, what are the prerequisites for {course} (slot here can be a course name such as advanced databases)
- Alexa uses a request-response mechanism via HTTPs interface.
- The skill sends the request to the service for processing.
- The skill recognizes the user's request and maps it to an appropriate intent and routes it to the custom skill's endpoint - an AWS Lambda function.

- The AWS Lambda function receives the request and extracts the user's intent and any slot values that were provided, which are placeholders for specific values (ex: {course}, {term}, {date} etc)
- The AWS Lambda function uses the intent and slot values to query DynamoDB for the requested information.
- DynamoDB returns the requested data to the AWS Lambda function.
- The AWS Lambda function formulates a response based on the requested data and returns it to the Alexa service in json format.
- The service receives a POST request (JSON) and generates a response.
- The skill delivers the response to the user through Alexa's voice interface.

The voice interaction model contains several intents, for example, an intent to schedule an appointment takes two slot values from the user's inputs, if the user won't specify the slot values in their command/question, Alexa asks for the input by prompting the user to provide inputs otherwise lambda function will not be able to formulate a valid response.

To generate Alexa prompts, the slot filling restriction should be turned on for the required slots for an intent. Only then mapping works in the lambda function to retrieve values from DynamoDB for a specific intent. This can be assured by the ElicitSlotDirective will prompt the user to provide the missing slot value by asking a follow-up question.

Moreover, the pre-defined slot type Amazon.SearchQuery can be used to capture free-form text input from the user, which can be useful for skills that involve searching for information or products. And the order of defining slots is extremely important to capture the inputs from the user correctly. For multiple slots, enabling auto delegation can prompt the user for all missing slots at once, rather than asking for each missing slot value separately.



ABOUT THE INTENTS IN THE INTERACTION MODEL

APIs used:

Google calendar API has been used to integrate calendars to add meeting reminders to the user's calendar when scheduled an appointment at a specific time and the user will also get email notifications about the appointment details including time and date. This can help users stay organized and manage their schedules more effectively.

Rapid API has been used to integrate several interesting automations using apis. By incorporating these features, the aim is to provide users with a more comprehensive and entertaining experience that goes beyond just retrieving information about courses and professors. By leveraging the functionality of the different APIs, the skill provides a more diversified experience that can better meet the users' needs.

Course and Professor Information Intents:

On launching the app, Alexa prompts the user with a message to schedule an appointment with a professor, which triggers a specific intent if the user wants to schedule one. Intents related to courses retrieve information from a DynamoDB table and provide specific values based on user input when triggered by an utterance from the user. The handler takes in a slot value for all the intents that are designed especially for course related questions, and matches it with the keys in the Dynamodb table. Once the matching keys are found, the corresponding values are retrieved and concatenated to form the response. The handlers of all the intents also take care of handling scenarios if requested slot value is not found in the table or when the required slot has not been filled. The handlers for almost all the intents for which questions are defined, works this way.

One example of how Alexa's intent works is when a user asks about the courses taught by the professor. Alexa prompts the user to provide a term, such as fall or spring, to retrieve the courses being offered during that semester. This is achieved by matching the user input with the keys defined in the database. If there is no match, Alexa will prompt the user for a valid input. The handlers for all intents are designed to handle such scenarios.

In addition to retrieving information about courses, the intents also provide information about professors. These intents can answer questions related to a professor's current and previous projects, publications, contact details, education, expertise, and other related information. On the other hand, course intents are focused on providing details about course materials, homework, discussions, prerequisites, grading policy, and other relevant information. Another example of a professor's intent, users can get information on the professor's current work by saying, "professor's ongoing projects". This utterance is then recognized by alexa and the corresponding intent will be triggered to generate a response. In addition, there are also other intents that require input from the user to provide specific information about the professor when requested.

Featured Intents:

Weather Forecast - Provides weather forecast for any city
This intent handler extracts the city and day slots from the user's request and sends a GET request to a weather API using the RapidAPI platform. The API returns a JSON response that includes the current weather condition and temperature for the specified location, as well as the forecast for the next few days. The response is then sent back to the voice assistant to provide users with accurate weather information.

Playlist Intent - returns top 10 songs for an artist
This intent handler extracts artist name from the user's request to make a GET request to the Deezer API and retrieve a list of track titles associated with a specified artist. This intent enables users to discover and enjoy popular tracks of artists over voice commands.

Lyrics Intent - provides lyrics for a song
This intent handler extracts the user's requested song to get the lyrics of a song. It uses a rapid API to retrieve the lyrics and returns the first ten lines to the user. The output is in the form of text that can be spoken out by Alexa.

Games - Rock paper scissors, Fortune cookie
This intent handler is designed to handle requests made by users to play rock-paper-scissors with ALexa. It takes the user's move as input and generates a random move for Alexa. Based on the moves, it randomly chooses an action and compares it to the user's action to determine the winner. The response is then returned to the user.
Another intent has been designed to handle requests for a fortune cookie message. It uses the RapidAPI to generate a random fortune message and returns it to the user.

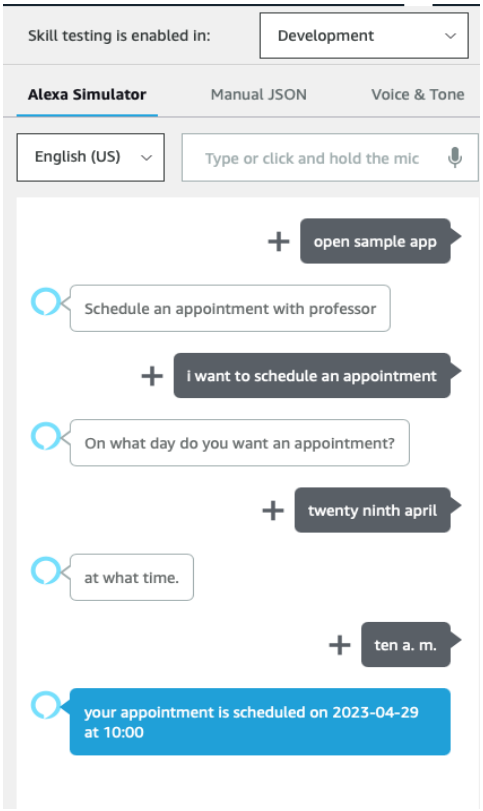
AvengersQuote - Gives a random quote from Avengers
This intent retrieves a random quote spoken by a character from the Marvel Avengers movies. The handler then constructs a response message containing the quote and character name, which is then spoken by Alexa to the user.

Words Intent - Translates a word in another language
This feature is designed to handle user's requests to translate a word from english to another language. It utilizes NLP translation API from rapid API to perform the translation. It takes the word and the target language as inputs from the Alexa request, and then makes an API request to get the translated word. The response is then returned as output and spoken by alexa.

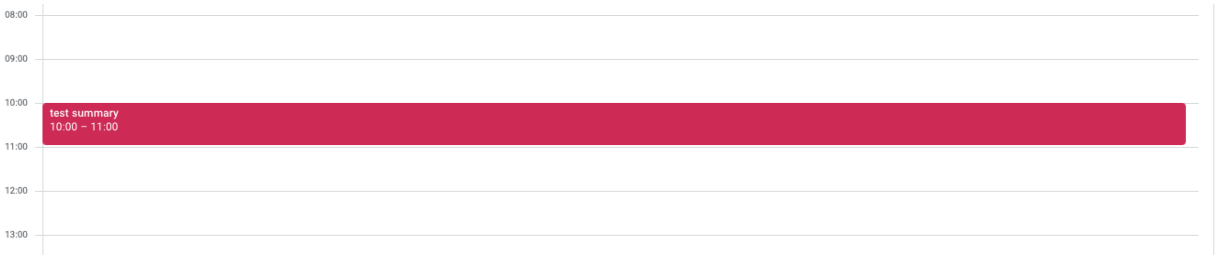
Space news Intent - provides space news headlines
This intent is responsible for retrieving the latest news related to space. The handler accesses the spaces news API and extracts the top five titles from the articles and reads them out loud to the user as a response. User has to say something like "recent space news".

Testing The Skill:

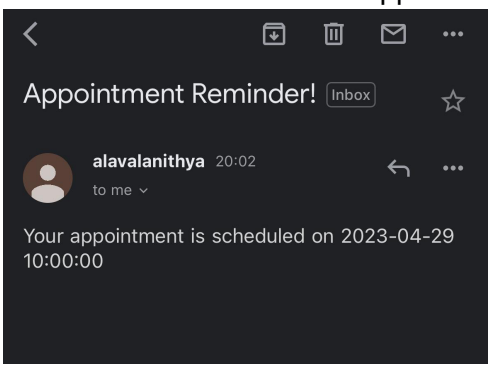
Alexa Simulator schedules an appointment based on the user's input. “Schedule an appointment with professor” message indicates the app launch.



Can check the reminder on the associated Google calendar!

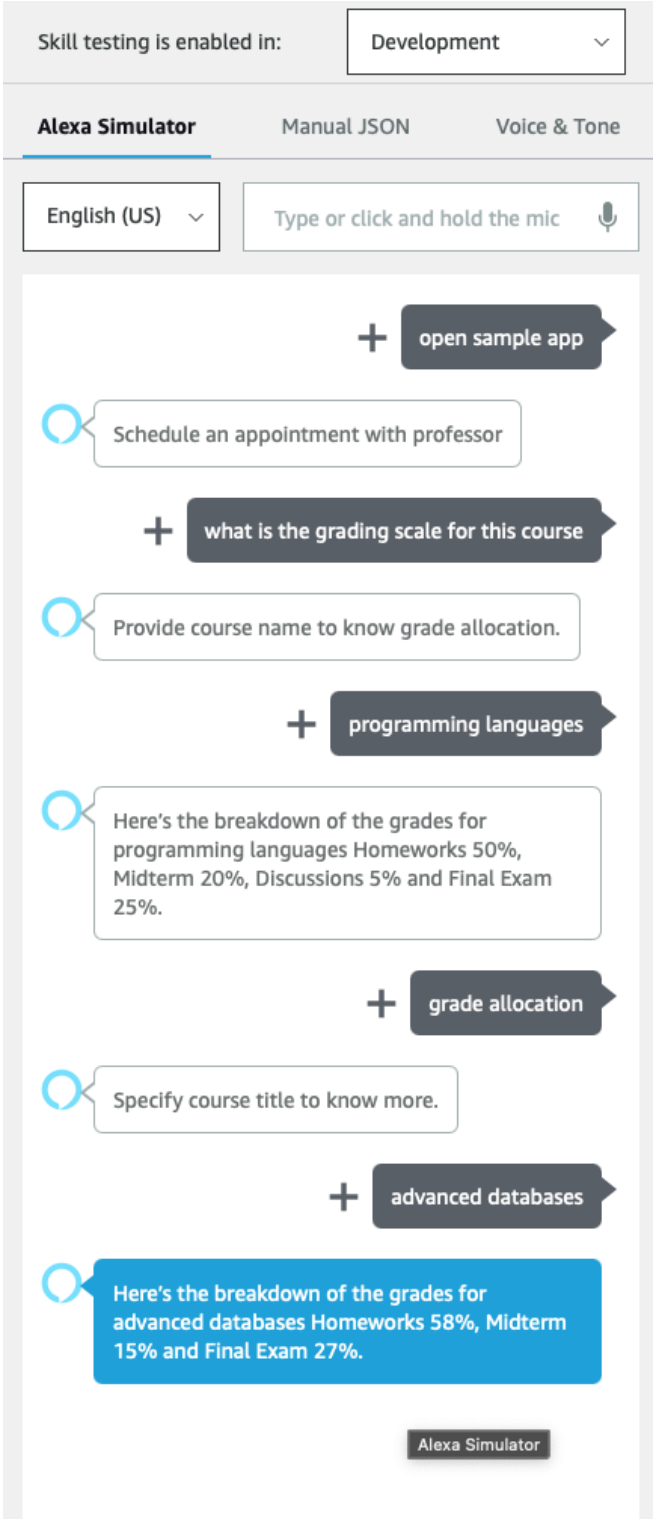


Email notification about the appointment.



Testing one of the course intents:

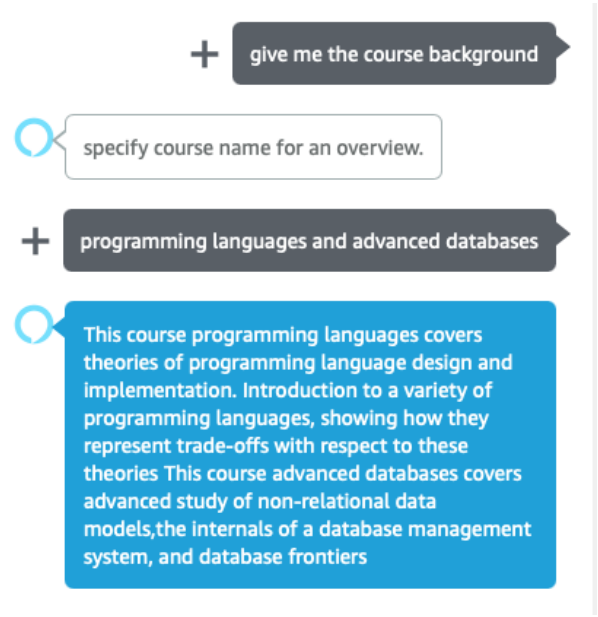
To know the grading scale of a specific course:



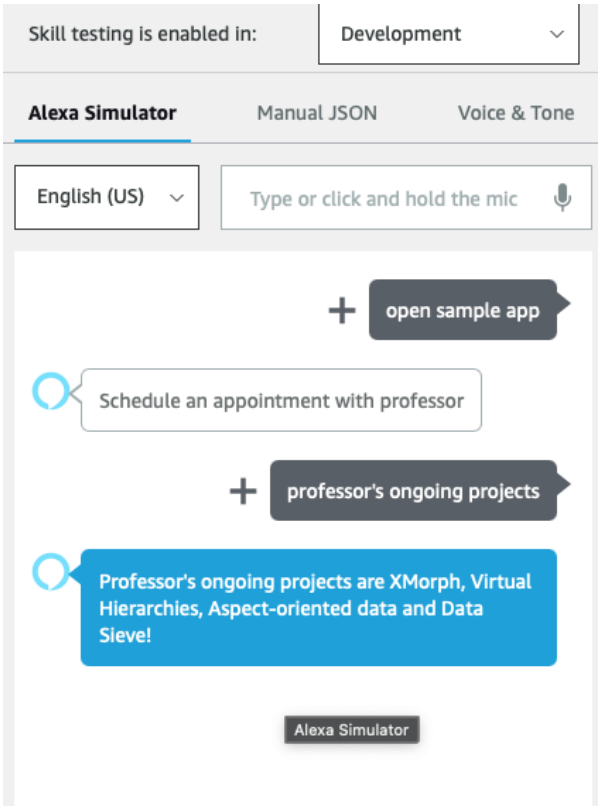
User Utterances: *what is the grading scale for this course/ grade allocation*
Alexa prompts: *Provide course name to know grade allocation (asking to fill slot value)*
User input: *programming languages/advanced databases*
Alexa Response: Here's the breakdown of the grades for {slot}

User can also provide multiple inputs

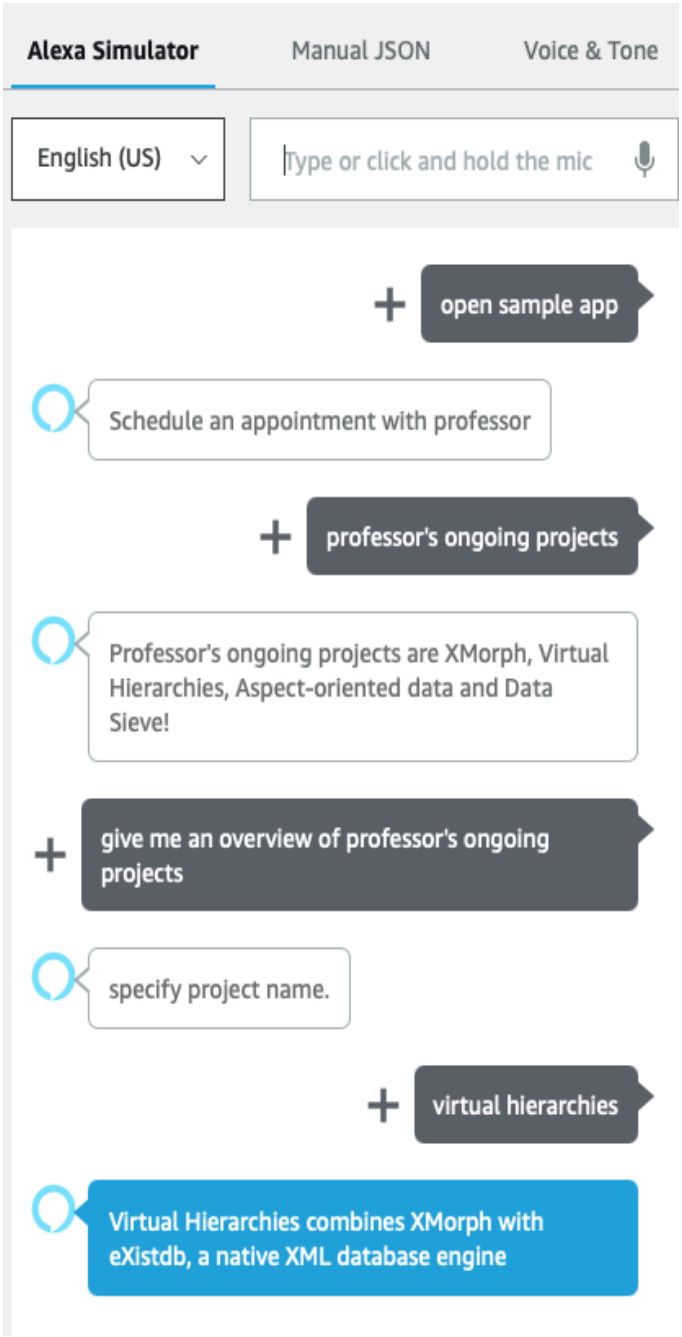
To know more about the background of a course: Here Alexa provides response for both the inputs: programming languages and advanced databases



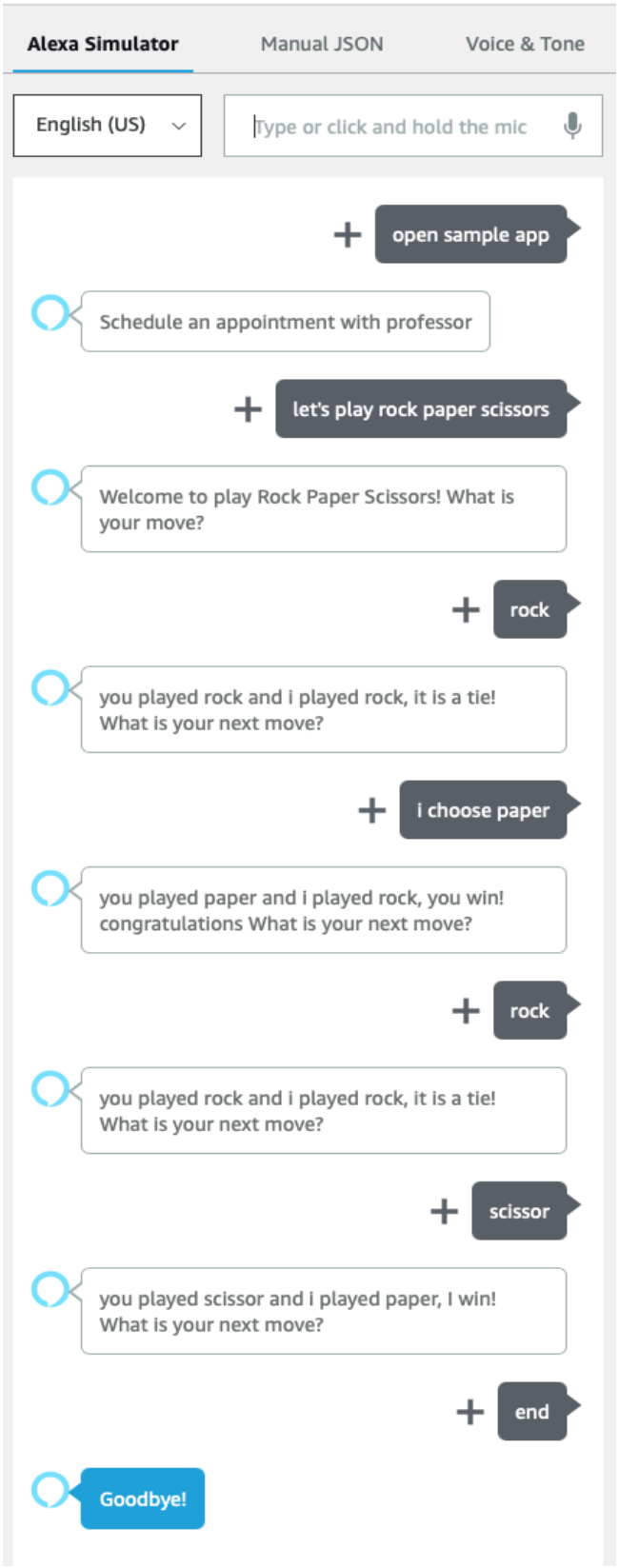
Testing one of the professor's intents:



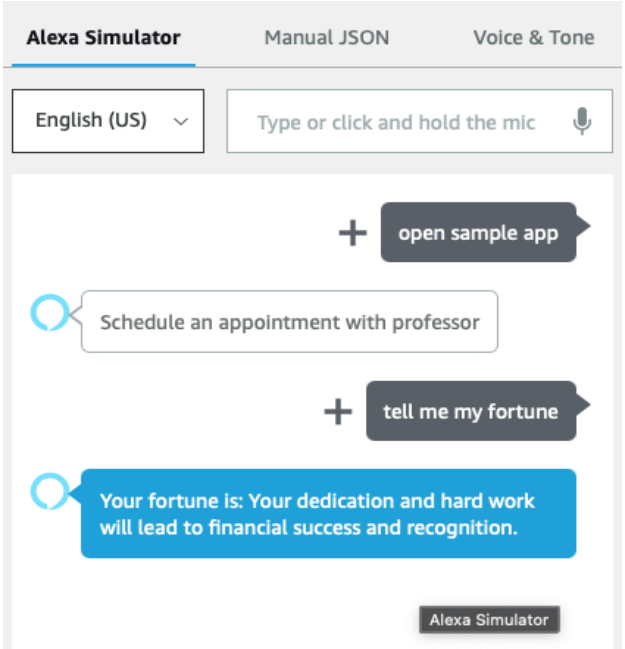
If user wants to know about a specific project, Alexa asks for user input if not provided.



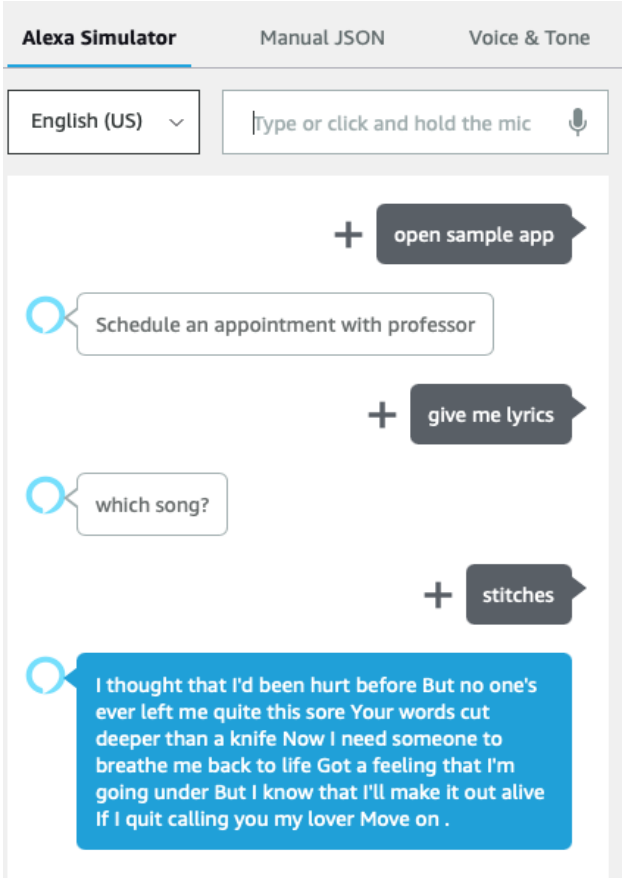
Rock Paper Scissors Intent:



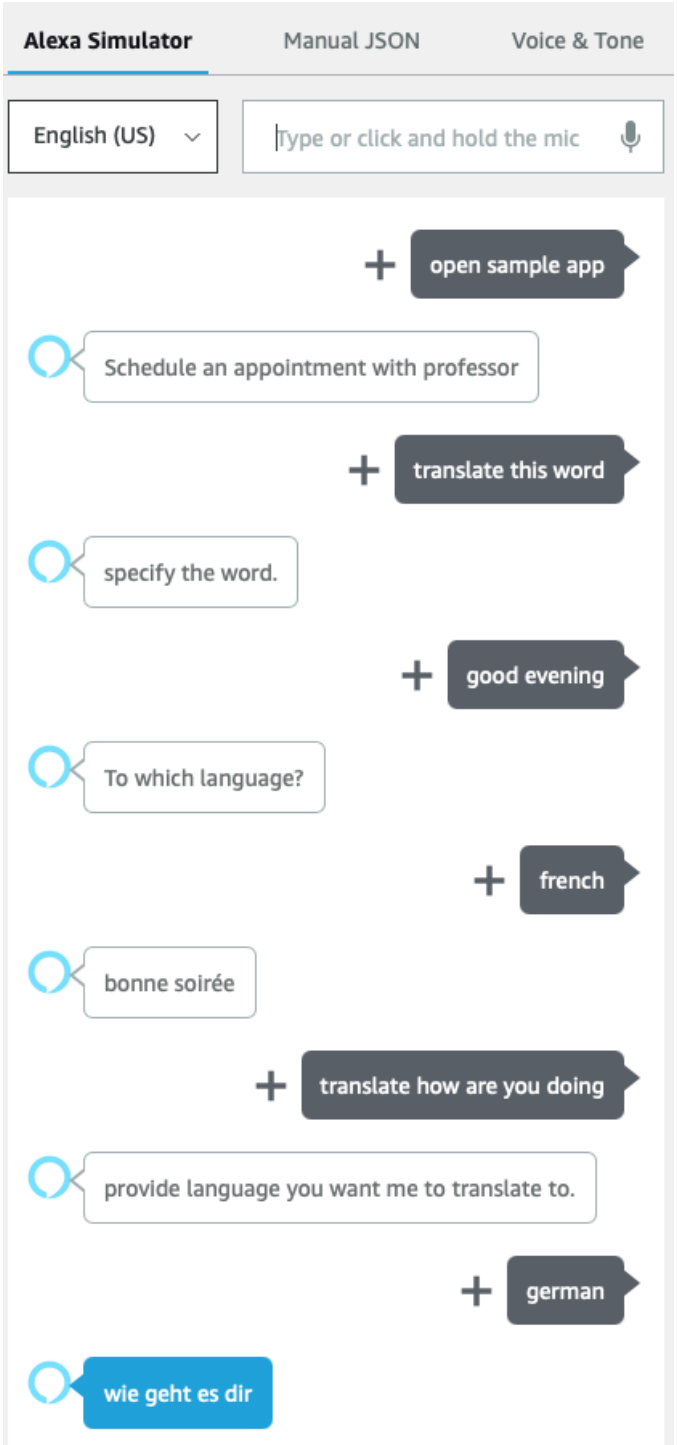
Fortune Cookie Intent:



Song Lyrics Intent:



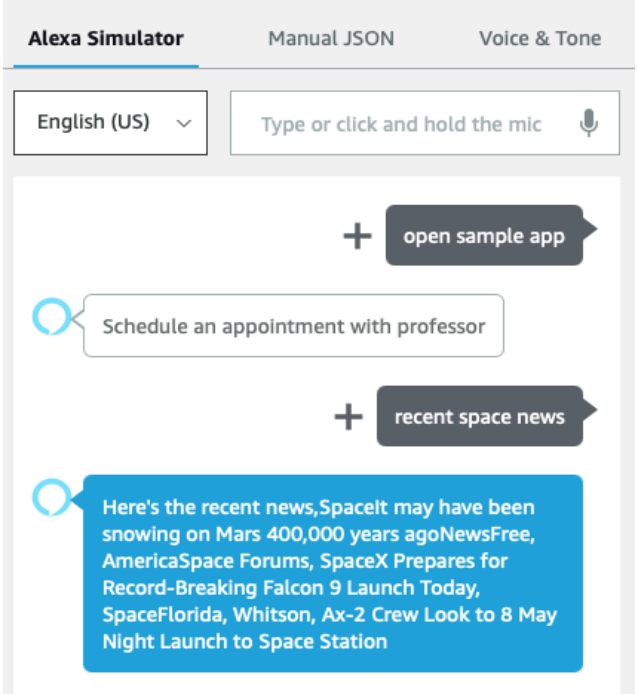
Word Translator Intent:
(This skill can prompt for multiple required slot inputs)



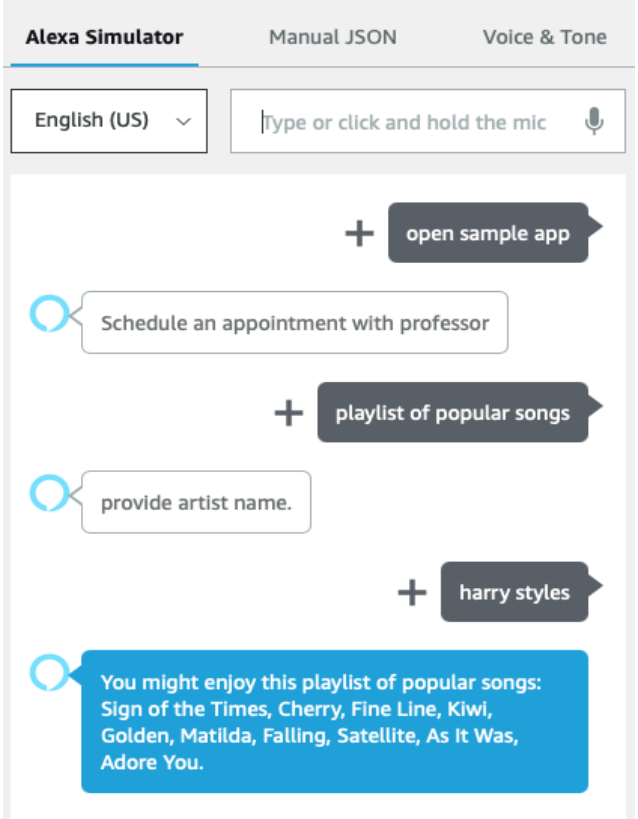
User Utterances: *translate this word*
Alexa prompt 1: *specify the word (asking to fill slot value {word})*
User input 1: *good evening*
Alexa prompt 2: *provide language you want me to translate to (asking to fill slot value {lang})*
User input 2: *French*
Alexa Response: Translated sentence in user requested language.

Space News Intent:

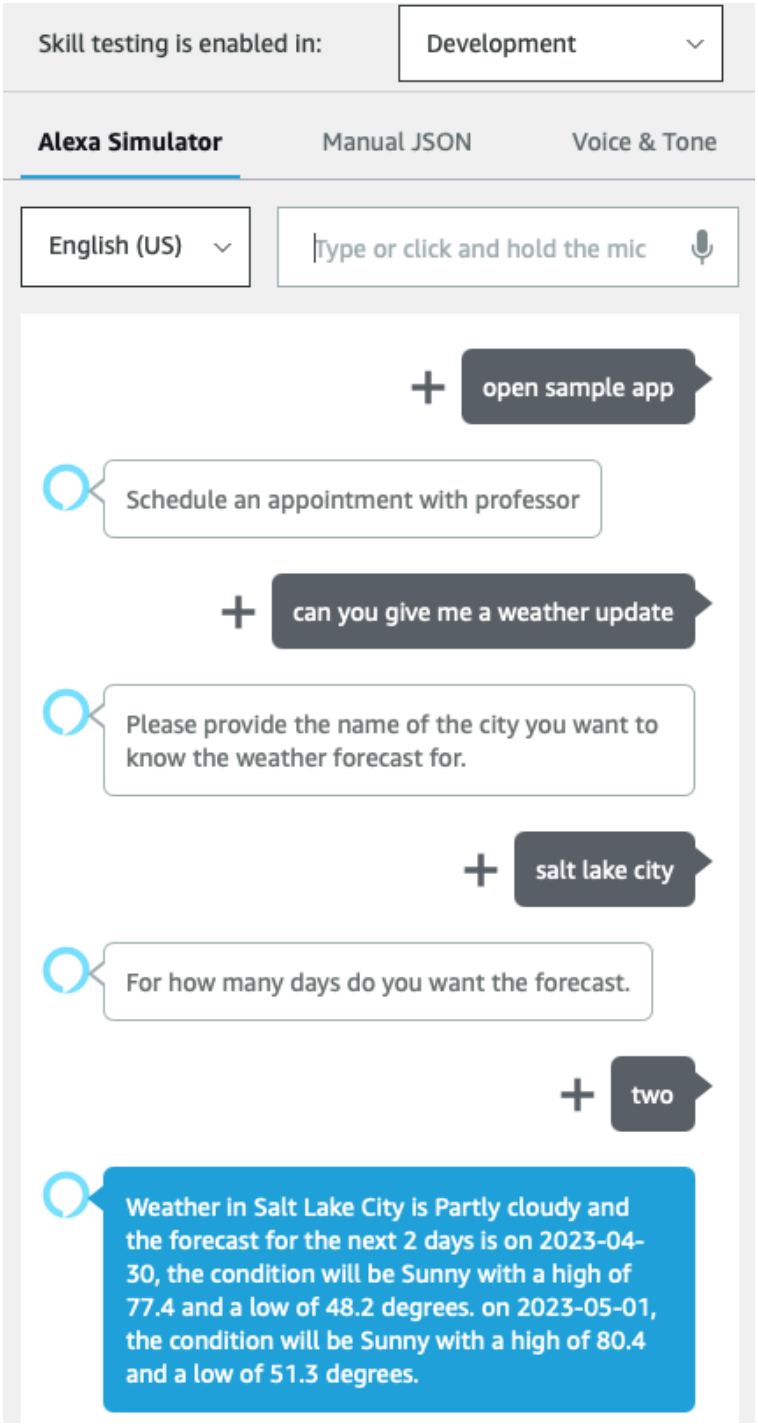
Alexa provides the most recent space news



Playlist Intent:



Weather Forecast Intent:



User Utterances: *can you give me a weather update*
Alexa prompt 1: *provide name of the city (asking to fill slot value {city})*
User input 1: *Salt Lake City*
Alexa prompt 2: *for how many days (asking to fill slot value {day})*
User input 2: *two*
Alexa Response: Weather forecast for user specified city and no.of days.