



VOICED BASED STUDY APPLICATION

Capstone Project Spring 2023
Final Presentation
Angelica Golphin
Joy Wilson

A person wearing a striped shirt is sitting at a desk, writing in a notebook with a pen in their right hand and typing on a laptop with their left hand. The scene is brightly lit, and the background is slightly blurred.

INTRODUCTION

- Student Doctors Preparing for Board Testing
- Multitasking Study Aid
- Easy Access to Alexa Enabled Devices
- Create a VUI Alexa Skill

Alexa

- Voice User Interface
- Flow of Program
- Build within Developer Console

The screenshot displays the Alexa Developer Console interface. At the top, a navigation bar includes links for 'Your Skills', 'Doctor Prompter', 'Build', 'Code', 'Test', 'Distribution', 'Certification', and 'Analytics'. The 'Build' tab is active. On the left, a sidebar menu shows a language dropdown set to 'English (US)' and a 'CUSTOM' section with expandable items: 'Invocations', 'Interaction Model', 'Assets', 'Slot Types (2)', 'Multimodal Responses', and 'Interfaces'. The main content area is titled 'Design and skill building resources' and is divided into two columns. The left column contains 'Design' resources (Alexa design guide) and 'Build' resources (Custom skill building guide, APL documentation, Make money). The right column, titled 'Building your skill', provides instructions to test the skill and features a checklist with two items: '1. Invocation Name' and '2. Intents, Samples, and Slots', both marked as 'REQUIRED' and completed with green checkmarks.

English (US)

CUSTOM

- > Invocations
- > Interaction Model
- ∨ Assets
- > Slot Types (2)
- Multimodal Responses
- Interfaces

Design and skill building resources

Design

[Alexa design guide](#)

Learn how to design engaging and creative visual responses for your skills using APL.

Build

[Custom skill building, a step-by-step-guide](#)

Follow this basic guide on how to build a custom skill.



[Alexa Presentation Language \(APL\) documentation](#)

Use our APL technical documentation to build skills with full-featured, responsive, and interactive screen displays.

[Make money](#)

Building your skill

Complete these steps to be able to test your skill using the simulator in the test tab, or with your echo device.

- REQUIRED** 1. Invocation Name > 
Enter an invocation name for your skill
- REQUIRED** 2. Intents, Samples, and Slots > 
Add at least one intent and one sample utterance

Database

- User Friendly
- Easy to Add Questions
- Web Based

The screenshot shows the MD2B application interface. The top navigation bar includes 'MD2B', 'Data', 'Automations', and 'Interfaces'. Below this, there are tabs for 'Data', 'User Persistence', and '+ Add or import'. A secondary navigation bar contains 'Views', 'Grid view', 'Hide fields', 'Filter', 'Group', 'Sorted by 1 field', 'Color', and 'Share view'. On the left, there is a search bar 'Find a view' and a 'Create...' dropdown menu with a 'Grid' option. The main content area displays a table with the following data:

	A Topic	Question	A Answer
1	anatomy	The common carotid artery bifurcates at what level?	C4
2	anatomy	Lumbar punctures are typically performed between what spinal levels?	L3-L5
3	anatomy	The trachea bifurcates at what spinal level?	T4
4	anatomy	Avascular necrosis of the head of the femur is most commonly due to insufficiency in what vascular structure?	medial cir
5	cardiology	Which artery supplies the majority of the anterior of the heart?	Lateral an
6	cardiology	What is the most common cardiac tumor?	metastasi

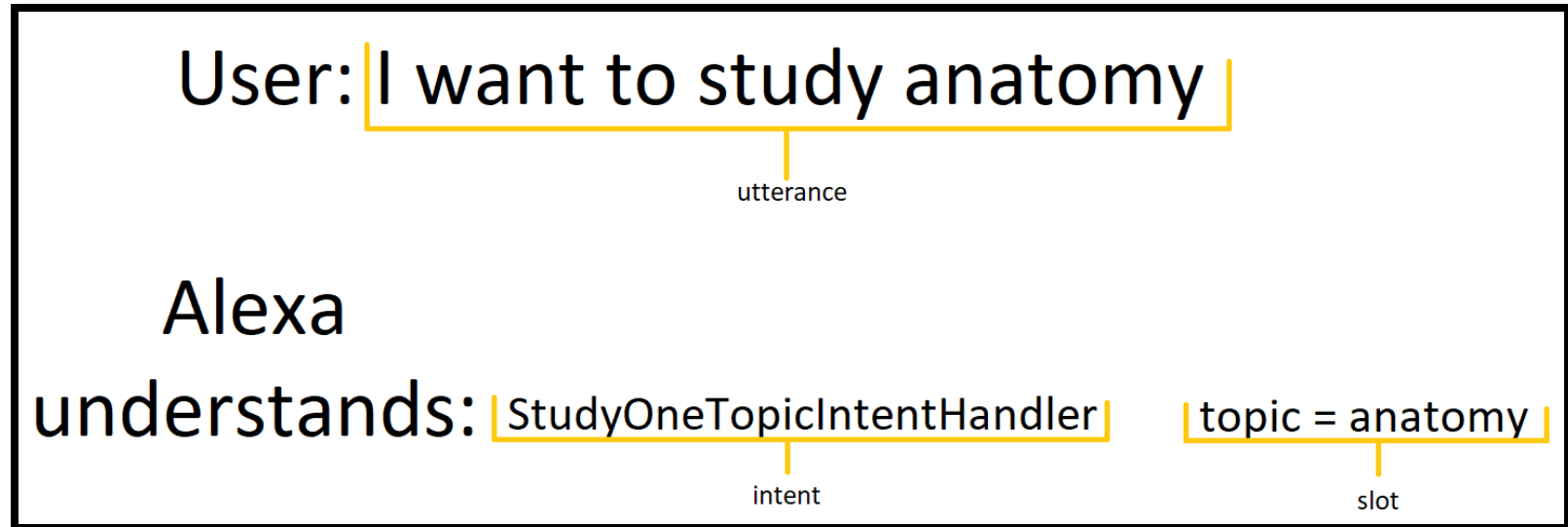
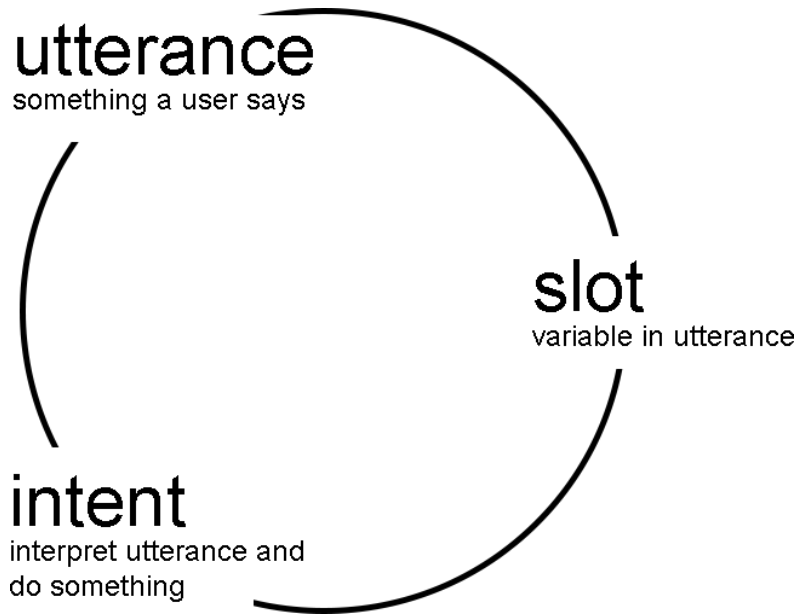
DOCTOR PROMPTER

- How Does It Work?
 - Sequence of Events
 - Persistence
- How Scalable Is It?
 - Adding Questions and Topics
 - Complete Change of Quiz



Front End Structure

Interaction Model



Front End Code

Interaction Model

```
134 ▾ const StudyOneTopicIntentHandler = {
135 ▾   canHandle(handlerInput) {
136     return Alexa.getRequestType(handlerInput.requestEnvelope) === 'IntentRequest'
137     && Alexa.getIntentName(handlerInput.requestEnvelope) === 'StudyOneTopicIntent';
138   },
139 ▾   async handle(handlerInput) {
140     const topic = Alexa.getSlotValue(handlerInput.requestEnvelope, 'topic');
141     const questions = await getQuestions(topic);
142 ▾   if (questions.length === 0) {
143     const speakOutput = `Sorry, I don't have any questions on ${topic}. Please try another topic.`;
144     return handlerInput.responseBuilder
145     .speak(speakOutput)
146     .reprompt(speakOutput)
147     .getResponse();
148 ▾ }
```

Back End Code

Session Attributes

```
handlerInput.attributesManager.setSessionAttributes({
  topic: topic, //store topic
  question: question.question, //store question
  answer: question.answer, //store current question's answer
  isTopicMode: true, //In OneTopic mode
  questions: questions, //Store all questions within the topic
  currentQuestionIndex: 0, //current question #
  hasNextQuestion: questions.length > 1, //checks for next question
  nextQuestion: questions.length > 1 ? questions[1] : null, //how many questions
  score: { correct: 0, total: 0 }, // score property
  correctAnswers: 0, // initialize correctAnswers to 0
  // totalQuestions: questions.length // set totalQuestions to the number of questions
  totalQuestions: 5 // set to 5 for only 5 questions to be asked
});
```

```
*****
```


Back End Code

Check Answer

```
248 // If storedAnswer is an array, check if userAnswer matches any of the elements in the array
249 if (Array.isArray(storedAnswer)) {
250     const isCorrect = storedAnswer.some(answer => answer.toLowerCase() === userAnswer.toLowerCase());
251
252     if (isCorrect) {
253
254         // Increment the number of correct answers and total questions asked
255         sessionAttributes.correctAnswers = sessionAttributes.correctAnswers + 1;
256
257         // This is where Alexa picks a random string for answer notification.
258         speakOutput = messages.correctAnswer();
259
260         // If there is a next question, ask it
261         if (sessionAttributes.hasNextQuestion) {
262             const nextQuestion = sessionAttributes.nextQuestion;
263
264             // This gives alternative sayings for next question and then gives the next question.
265             speakOutput += `${messages.anotherQuestion()}: ${nextQuestion.question}.`;
266
267             attributesManager.setSessionAttributes(sessionAttributes); //I don't think I need this.
268
269             if (sessionAttributes.currentQuestionIndex < sessionAttributes.totalQuestions) {
270
271                 sessionAttributes.currentQuestionIndex = sessionAttributes.currentQuestionIndex + 1; //current question #
272                 sessionAttributes.question = sessionAttributes.questions[sessionAttributes.currentQuestionIndex].question;
273                 sessionAttributes.answer = sessionAttributes.questions[sessionAttributes.currentQuestionIndex].answer;
274
275                 // If at the last question
276                 if (sessionAttributes.currentQuestionIndex >= sessionAttributes.totalQuestions - 1){
277                     sessionAttributes.hasNextQuestion = false; //checks for next question
278                     sessionAttributes.nextQuestion = null; //how many questions
279                 }
280                 // Else loop to the next question
281                 else
282                 {
283                     sessionAttributes.hasNextQuestion = true; //checks for next question
284                     sessionAttributes.nextQuestion = sessionAttributes.currentQuestionIndex + 1 < sessionAttributes.totalQuestions ? sessionAttributes.questions[sessionAttributes.currentQuestionIndex + 1] : null; //how many questions
285                 }
286
```

CHALLENGES

- Skill Name
- Database API Implementation
- Program Flow
 - CheckAnswer Handler(s)
 - Setting Up Session Attributes
- Public Skill
 - User Testing
 - Privacy Policy





SOFTWARE SIMULATION



Our Amazing Clients

Neesha Jahani
Trey Walker



Conclusion

- Thank You
- Questions??