

Building Chatbots using Amazon Lex: A Primer

This blog post will introduce a newbie to building a chatbot using the [Lex](#) methodology/toolkit from Amazon. We will demonstrate the basic building blocks and then move on to creating a live application. Our approach will be a step-by-step method. Here is the main landing page for Lex:

Amazon Lex

Conversational interfaces for your applications

Powered by the same deep learning technologies as Alexa

GET STARTED WITH AMAZON LEX

Product Details Pricing Customers FAQs Developers Amazon Connect Integration

Amazon Lex is a service for building conversational interfaces into any application using voice and text. Amazon Lex provides the advanced deep learning functionalities of automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU) to recognize the intent of the text, to enable you to build applications with highly engaging user experiences and lifelike conversational interactions. With Amazon Lex, the same deep learning technologies that power Amazon Alexa are now available to any developer, enabling you to quickly and easily build sophisticated, natural language, conversational bots ("chatbots").

See Amazon Lex at Work and Play

```
graph LR; Lex[Amazon Lex] --> Lambda[AWS Lambda]; Lex --> Polly[Amazon Polly]; Lex --> CloudWatch[CloudWatch]; Lambda --> DMS[DynamoDB]; Lambda --> SES[Amazon SES]; Lambda --> CloudWatch; Polly --> CloudWatch; Other[Other AWS Services]
```

You can find a working model of a sample chatbot at the URL below:

<http://vardangupta.s3-website-us-east-1.amazonaws.com/>

Some readings:

<https://work.qz.com/1147692/harness-the-power-of-bots-to-automate-the-busy-work/>

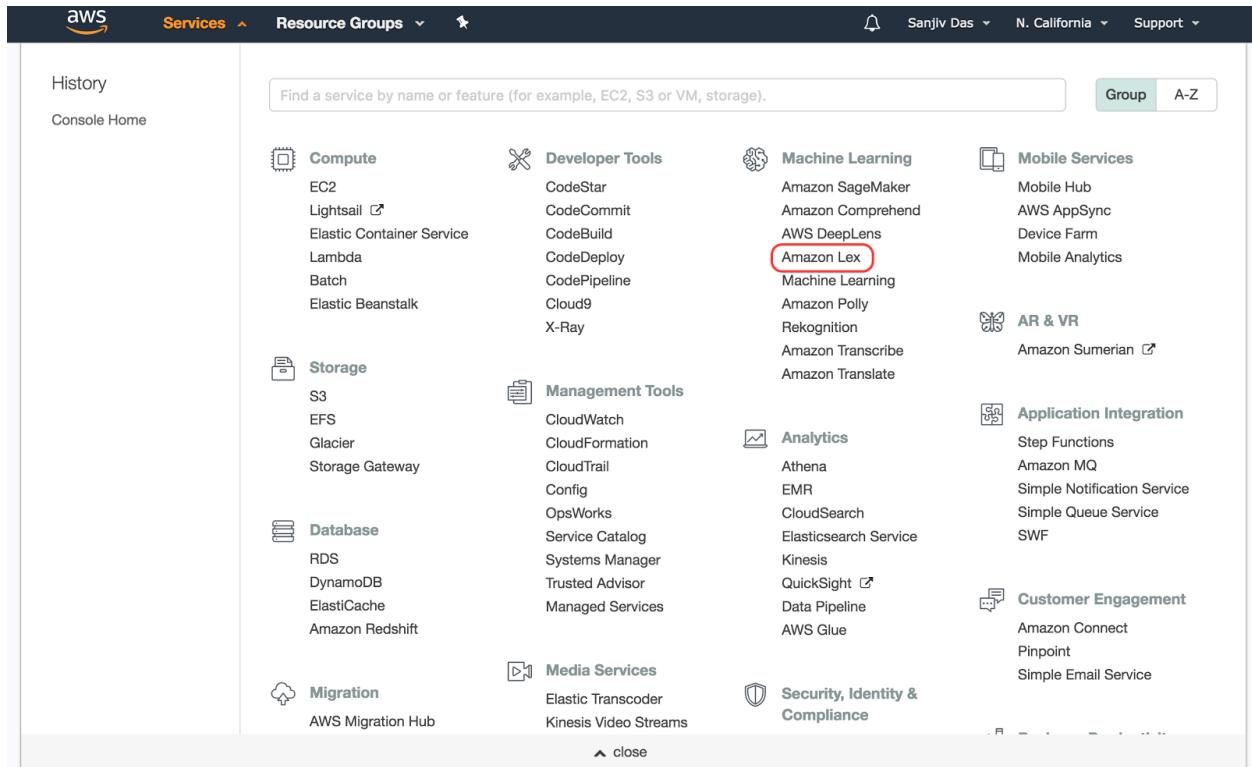
<https://www.datasciencecentral.com/profiles/blogs/beginners-guide-to-chatbots>

<https://www.datasciencecentral.com/profiles/blogs/under-the-hood-with-chatbots>

Basic Setup

- (1) Sign up for an AWS account and sign in: <https://aws.amazon.com/console/>

- (2) Go to the dashboard and search for Lex under AWS services. Open the lex services and create the new chatbot. Select the purpose of your chatbot from existing examples or create your own. To make this demonstration even more explanatory let's move ahead with the custom bot.



When you click on "Amazon Lex" you reach this screen.

AWS Services Resource Groups Sanjiv Das N. Virginia Support

Amazon Lex

Amazon Lex is a service for building conversational interfaces using voice and text. With Lex, the same deep learning engine that powers Alexa is now available to any developer, enabling you to bring sophisticated, natural language chatbots to your new and existing applications.

[Get Started](#)

Getting Started Guide



High Quality Deep Learning Technologies

Powered by the same technology as Alexa, Lex provides both automatic speech recognition (ASR) and natural language understanding (NLU) technologies to create a



Seamlessly Deploy and Scale

You can build, test, and deploy your chatbots directly from the AWS Management Console. Lex allows you to easily publish your voice or text chatbots, so you can access them from mobile apps, web apps, and multiple chat



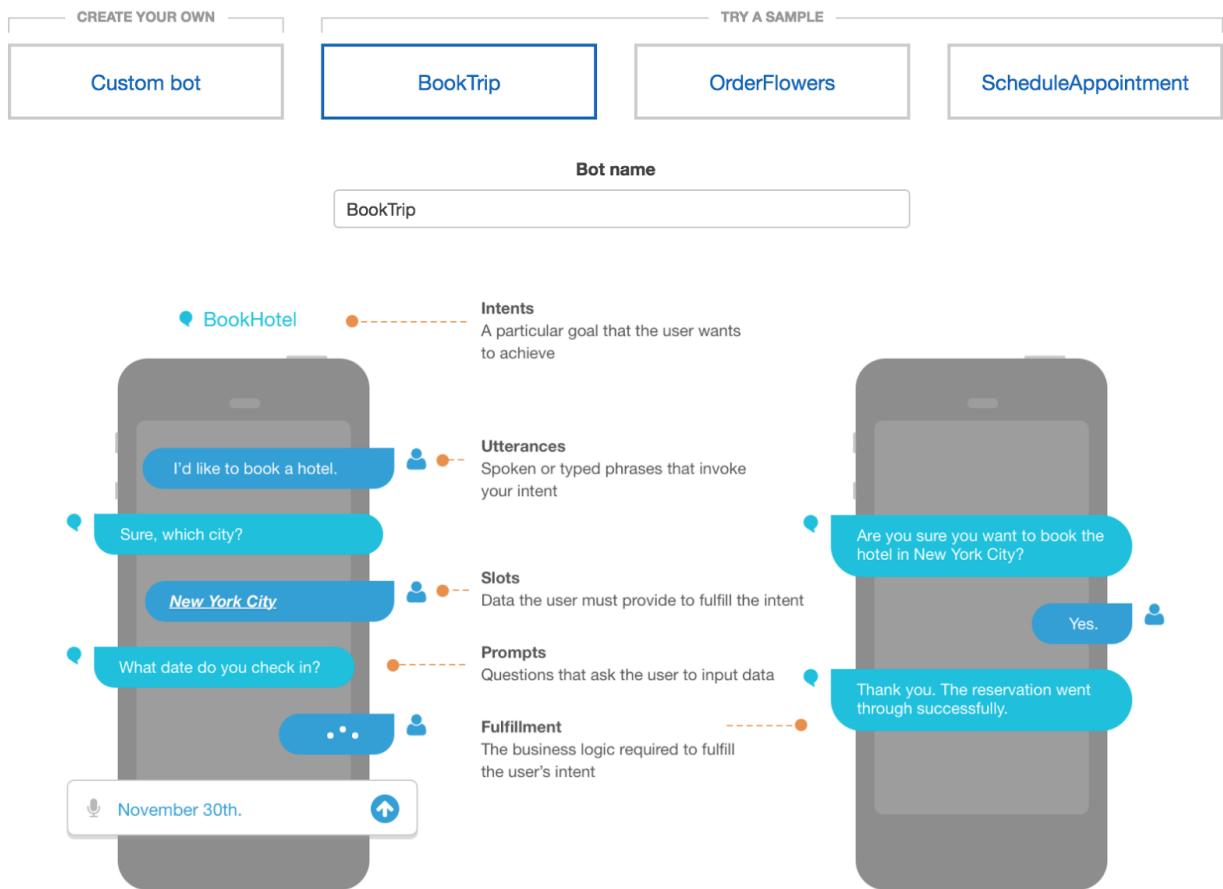
Built-in Integration with the AWS Platform

Amazon Lex has native interoperability with several AWS services such as Amazon Cognito, AWS Lambda, Amazon DynamoDB, Amazon CloudWatch, and AWS Mobile Hub,

The main components of the bot are shown below: Intents, Utterances, Slots, Prompts, and Fulfillment.

Create your Lex bot

Amazon Lex enables any developer to build conversational chatbots quickly and easily. With Amazon Lex, no deep learning expertise is necessary—you just specify the basic conversational flow directly from the console, and then Amazon Lex manages the dialogue and dynamically adjusts the response. To get started, you can choose one of the sample bots provided below or build a new custom bot from scratch.



Building a Custom Bot

We will pick a custom bot and fill out the basic information in the form depending on the type of use.

Create your Lex bot

Amazon Lex enables any developer to build conversational chatbots quickly and easily. With Amazon Lex, no deep learning expertise is necessary—you just specify the basic conversational flow directly from the console, and then Amazon Lex manages the dialogue and dynamically adjusts the response. To get started, you can choose one of the sample bots provided below or build a new custom bot from scratch.

CREATE YOUR OWN

TRY A SAMPLE

Custom bot

BookTrip

OrderFlowers

ScheduleAppointment

Bot name: LaptopEnquiry

Language: English (US)

Output voice: None. This is only a text based application.

Session timeout: 5 min

IAM role: AWSServiceRoleForLexBots ⓘ
Automatically created on your behalf

COPPA: Please indicate if your use of this bot is subject to ⓘ
the Children's Online Privacy Protection Act
(COPPA). [Learn more](#)

Yes No

[Cancel](#) [Create](#)

You will land on your chatbot console where you can edit and customize your chatbot. Before moving forward, let's understand some basic terminologies that we will be using throughout the tutorial.

- **Intent:** Represents an action that the user wants to perform.
- **Utterances:** Sentences by a user to represent an intent. Example: I want to enquire about a laptop.
- **Slot:** Sets of information supplied by the user and required by the intent for it to get completed. The information may or may not be mandatory and is asked from the user while chatting with the user i.e., during the runtime.
- **Slot Types:** Each slot belongs to a particular type and has different properties related to the type of input from the user. Lex has hundreds of inbuilt slot types but users can always define their own.

We take the example of a "Laptop Enquiry" for this tutorial.

This is the dashboard of the chatbot where all the functionality can be modified. Firstly we have to define a new **intent** and its operation.

Welcome to your bot editor. You can start right away by adding an intent using the **Create Intent** button in the Intents section of the left navigation.

+ Create Intent

Components of your bot.

BookHotel

- Intents**: A particular goal that the user wants to achieve.
- Utterances**: Spoken or typed phrases that invoke your intent.
- Slots**: Data the user must provide to fulfill the intent.
- Prompts**: Questions that ask the user to input data.
- Fulfillment**: The business logic required to fulfill the user's intent.

The diagram illustrates a conversation flow between a user and a bot. The user says "I'd like to book a hotel." The bot asks "Sure, which city?" The user replies "New York City". The bot asks "What date do you check in?" The user replies "November 30th". The bot then asks "Are you sure you want to book the hotel in New York City?" The user replies "Yes". Finally, the bot says "Thank you. The reservation went through successfully."

This is the screen which comes up when we create a new intent or we are currently on any of the intent. Here, inside the sample utterances we define various sentences or responses from the user, that will tell the chatbot about the intent of user.

Add intent

Create new intent

Search within existing intents

Custom intents: BuyLaptop

Built-in intents: AMAZON.CancelIntent, AMAZON.HelpIntent, AMAZON.LoopOffIntent, AMAZON.SearchIntent

AWS Lambda function Return parameters to client

Cancel **Add**

Save Intent **Detach intent**

The dialog box is titled "Add intent". It has two main sections: "Create new intent" and "Search within existing intents". The search bar shows "14 intents". Below the search bar, there are two sections: "Custom intents" (listing "BuyLaptop") and "Built-in intents" (listing "AMAZON.CancelIntent", "AMAZON.HelpIntent", "AMAZON.LoopOffIntent", and "AMAZON.SearchIntent"). At the bottom of the dialog, there are buttons for "Cancel" and "Add", and links for "Save Intent" and "Detach intent".

Next, we define multiple utterances so that the user may ask for any information in many different ways from the system.

LaptopEnquiry Latest

Editor Settings Channels Monitoring

Intents Laptop

Slot types No slots created

Error Handling

Laptop Latest

Sample utterances

- I am looking for a laptop
- I am looking for laptops
- I want to buy a laptop
- purchase a laptop
- laptop computer
- laptop
- computer

Lambda initialization and validation

Slots

Priority	Required	Name	Slot type	Prompt
		e.g. Location	e.g. AMAZON.U...	e.g. What city?

Confirmation prompt

Fulfillment

AWS Lambda function Return parameters to client

We add in new slots.

LaptopEnquiry Latest

Editor Settings Channels Monitoring

Intents Laptop

Slot types NewOrOld Spend

Error Handling

Laptop Latest

Sample utterance

Purchasing a new or old laptop

Slot Resolution

Expand Values

Restrict to Slot values and Synonyms

Value

e.g. Small

New

Cancel Save slot type Add slot to intent

Lambda initialization and validation

Slots

Priority	Required	Name	Slot type	Prompt
1.	<input checked="" type="checkbox"/>	NewOrOld	NewOrOld	e.g. What city?
2.	<input checked="" type="checkbox"/>	Spend	Spend	Do you want a new or old laptop?

Confirmation prompt

Laptop

Sample utterances

I would like to book a flight.

I am looking for a laptop

I am looking for laptops

I want to buy a laptop

purchase a laptop

laptop computer

laptop

computer

Slots appear here

Lambda initialization and validation

Slots

Priority	Required	Name	Slot type	Prompt
1.	✓	NewOrOld	e.g. AMAZON.U...	e.g. What city?
2.	✓	Spend	Spend	Do you want a new or old laptop?

Now, add in all possible ways in which you may want the bot to respond.

Spend Prompts

e.g. What is your destination?

What is your budget?

Up to how much are you willing to spend?

Different ways in which the bot may respond

Maximum number of retries

2

Corresponding utterances

e.g. I would like to go to {toCity}

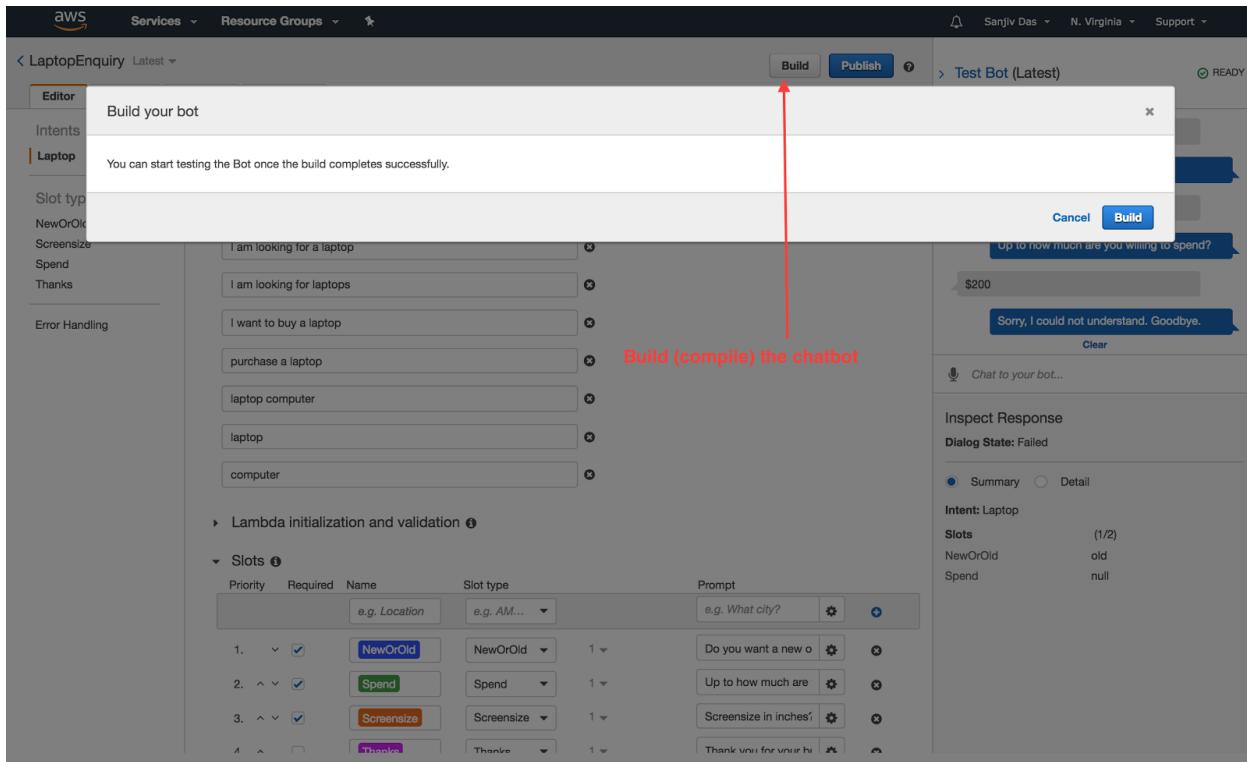
What is your {Spend}

Prompt response cards

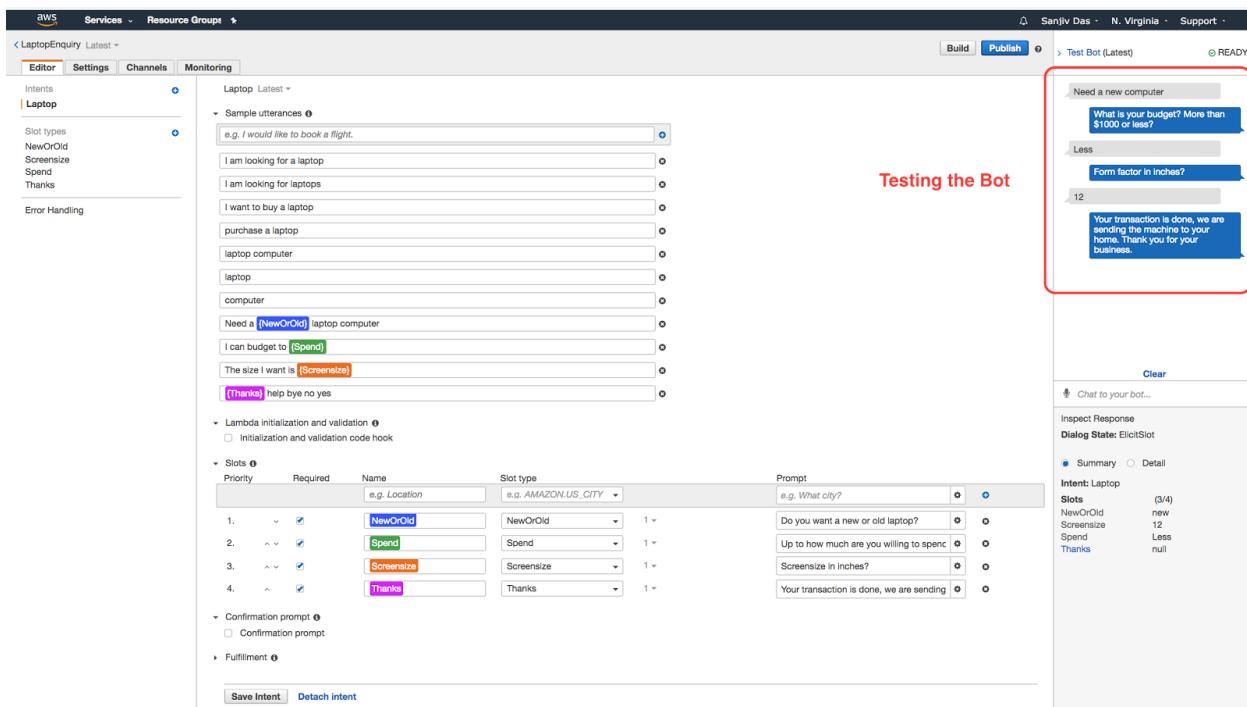
0

Card image	Card title	Card subtitle	Preview
e.g. url	e.g. Card title text	e.g. Card subtitle text	Facebook
None	e.g. Button title		
None	e.g. Button title		
None	e.g. Button title		

Then, build the chatbot.



Finally, we test the bot.



We may then publish the bot.

Here is what you get after publishing the bot.

You can set up a callback URL in the channels tab.

AWS Services - Resource Groups -

LaptopEnquiry Latest - Sanjiv Das - N. Virginia - Support -

Editor Settings Channels Monitoring

Build Publish > Test Bot (Latest) READY

Facebook

Fill in the form below and click activate to get a callback URL to use with Facebook. You can generate multiple callback URLs. [Learn more](#) on steps to integrate with Facebook.

Channel Name*

Channel Description

IAM Role AWSServiceRoleForLexChannels
Automatically created on your behalf

KMS Key aws/lex

Alias*

Verify Token* Verify Token

Page Access Token* Page Access Token

App Secret Key* App Secret Key

* Required Field

Activate

Callback URLs

Fill in the form above and click activate to get a callback URL. You can generate multiple callback URLs.

Type an utterance below to begin a conversation with your chatbot

Chat to your bot... Clear

Inspect Response

Done!

Edited out material

The screenshot shows the Amazon Lex console interface. On the left, the navigation bar includes 'Editor', 'Settings', 'Channels', and 'Monitoring'. The main area is titled 'BuyLaptop Latest'. The 'Intents' section shows 'BuyLaptop' selected. The 'Slot types' section lists 'Binary', 'LaptopUses', and 'Screensize'. The 'Error Handling' section is collapsed. The 'BuyLaptop' intent details are shown, including sample utterances like 'I would like to buy a laptop' and slots for 'OldPackageOrNew', 'LaptopUses', 'ScreenSize', and 'InvestmentAmout'. To the right, a 'Test Bot' window shows a conversation starting with 'I would like to buy a laptop', followed by a prompt 'Is there any package you are interested in or you want help to select one?'. The user responds 'new', and the bot asks 'What will be the 3 main uses of this'. Below this is a text input field 'Chat to your bot...' and an 'Inspect Response' panel showing the dialog state as 'ReadyForFulfillment'.

BuyLaptop Latest ▾

▼ Sample utterances ⓘ

e.g. I would like to book a flight. +

I would like to buy a laptop ×

{OldPackageOrNew} ×

{LaptopUses} ×

I want a {ScreenSize} ×

I have a budget of {InvestmentAmout} ×

▼ Slots i

Priority	Required	Name	Slot type	Prompt	
		e.g. Location	e.g. A... ▾	e.g. What city?  	
1.	✓	OldPackageC	Binary ▾	1 ▾	Is there any packa  
2.	✓	LaptopUses	Laptop... ▾	1 ▾	What will be the 3  
3.	✓	ScreenSize	Scree... ▾	1 ▾	What is the prefer  
4.	✓	InvestmentAr	AMAZ... ▾	Built-in ▾	How much are yo  

I would like to buy a laptop

Is there any package you are interested in or
you want help to select one?

new

What will be the 3 main uses of this
laptop?

programming

What is the preferred screen size?

13 inch

How much are you planning to invest?

1000\$

Intent BuyLaptop is ReadyForFulfillment:
InvestmentAmout:1000
LaptopUses:programming
OldPackageOrNew:new ScreenSize:13
inch