

Cloud Assignment 1

Team:

Abhishek Kandukuri (ak9691)

Sidharth Prakash (sx2236)

UI endpoint: <http://cloud-as1-chatbot.s3-website-us-east-1.amazonaws.com/>

Front-End:

The front end is hosted on a publicly accessible S3 bucket. The sdk folders in the front-end code have been modified according to the configurations in API Gateway.

API Gateway:

It has been configured to route incoming requests to the lambda function cloud-as1-route-to-lex.

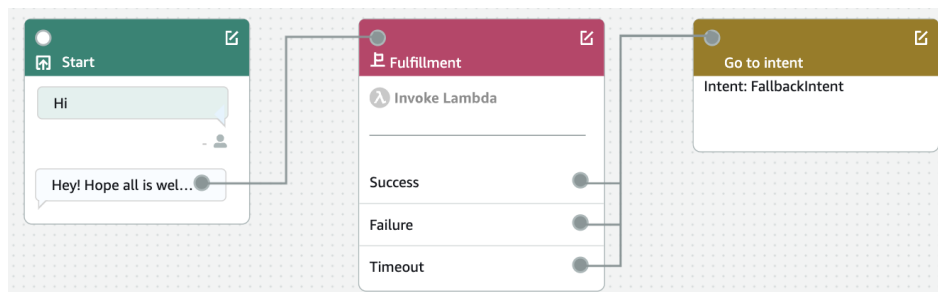
Lambda functions:

- **cloud-as1-route-to-lex (If0)** : This function routes the messages between API gateway and lex. It also checks the dynamoDB for any previous suggestions if the user requires.
- **cloud-as1-lex-to-sqs (If1)**: This function is triggered from a lex fulfillment hook and parses the information from lex to a suitable format and pushes it to the queue.
- **test1 (If2)** which does the following:
 - ❖ Get the attributes of the request of the customer (location, cuisine, number of people etc)
 - ❖ Get the top 5 results from Elastic Search for the cuisine requested by the customer . These results contain only the Business id and cuisine.
 - ❖ The Business id of the top 5 results is then used to query additional data such as address, restaurant name etc. from DynamoDB
 - ❖ The response with detailed information about the top 5 restaurants is then inserted into another DynamoDB table called “user_suggestions”. This table is used to return previous search results of a user.
 - ❖ Send an email to the user with the curated list of suggestions using SES service.

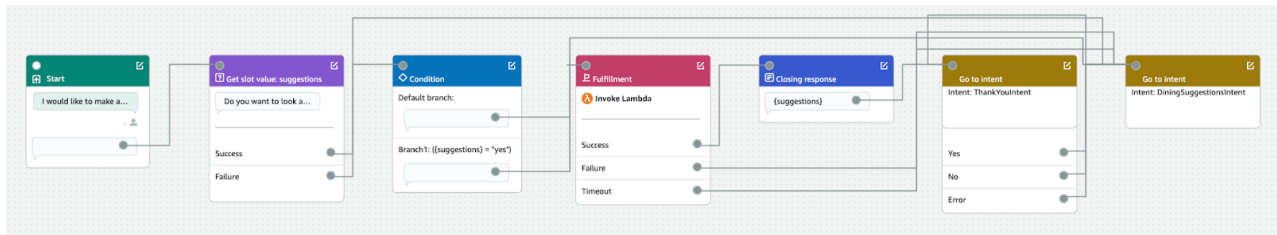
Lex Chatbot:

We have set up the following Intents for our chatbot.

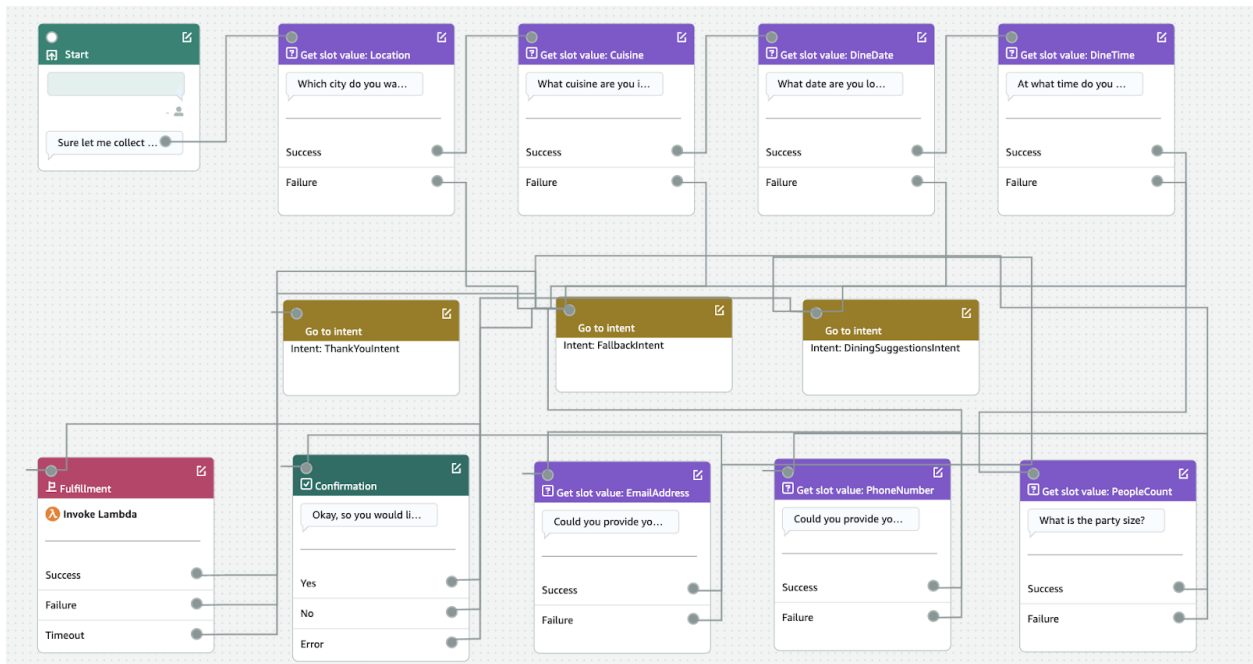
GreetingIntent



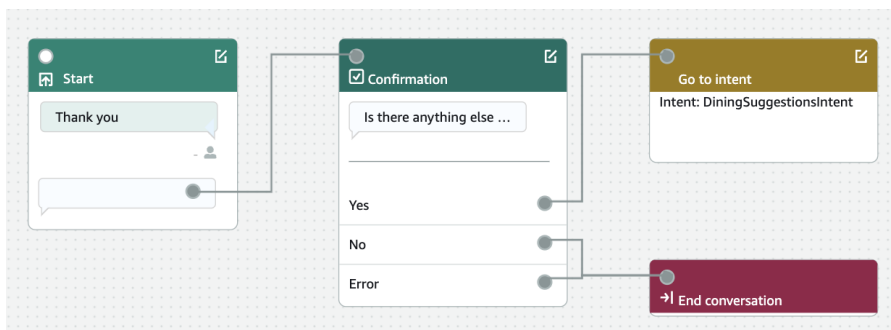
ExistingSuggestionIntent



DiningSuggestionIntent



ThankYouIntent



YELP API Data Collection

Data was collected from yelp API using parameters mentioned in the documentation here https://docs.developer.yelp.com/reference/v3_business_search

Restaurant data was collected for the following cuisines

```
cuisines = ['italian', 'chinese', 'mexican', 'greek', 'french', 'korean',  
'jewish', 'japanese', 'malaysian',  
            'mediterranean', 'persian']
```

from different restaurants located in and around Manhattan. This data was then formatted into the required format and inserted into DynamoDB table called “yelp-restaurants”

The code for this is available in dynamo_db_insertion_final.py file

Screenshot of data in DynamoDB

This table has more items to retrieve. To retrieve the next page of items, choose **Retrieve next page**.

Items returned (50)

Business_ID	Address	Cuisine	insertedAtTimestamp	Latitude	Longitude	Name
XjeGryxde-tQZF_Ewu...	408 Main St	korean	26/02/2023 21:40:31	40.853626...	-73.974899...	Gamja
b_OBvpng5a8Ufg8gJ...	217-21 Ja...	italian	26/02/2023 21:45:44	40.71815	-73.73724	CD Res
uaFHq-a5XqF-bsOK...	811 8th Ave	chinese	26/02/2023 21:27:28	40.761628...	-73.98688	Real K
lHSVn0TaX8xXb3wcQ...	118A Eldrid...	chinese	26/02/2023 21:42:41	40.71839	-73.9917399	Vaness
aSprnLY-6GkMk1cuv0...	248 5th Ave	italian	26/02/2023 21:29:53	40.675332...	-73.981340...	Al Di L
IYVeY3q78hefg6o3IH...	17-43 E 16...	chinese	26/02/2023 21:34:25	40.8290427	-73.9280077	Noodle
YwpP-mgXV5N35xhLi...	453 Rogers ...	japanese	26/02/2023 21:...	40.66127	-73.95342	Names
pBgnAbZQ\$svv_HJGR...	789 Allerto...	mexican	26/02/2023 21:38:09	40.86566	-73.86396	Latin F
jYia13i0CBwNXadzAA...	8814 4th Ave	japanese	26/02/2023 21:25:32	40.62106	-74.02927	Ariranc
QHUGMN9uizw6kAUA...	46 Christie Pl	mediterranean	26/02/2023 21:38:35	40.9902295	-73.8070063	PopoJi

Elastic Search (Amazon OpenSearch Service)

First, an Elastic Search index called “restaurants” was created . Partial information from the data collected in DynamoDB was inserted into an index called “restaurants” with type “restaurant”

Screenshot of data in Elastic search for the restaurant data with index “restaurants”

The screenshot shows the Elastic Search web interface. On the left, there's a sidebar with a search bar and filter options. The main area displays search results for the 'restaurants' index. The results are shown as a list of documents, each containing fields like Business_ID, Cuisine, _id, _type, _index, and _score. The interface is clean and modern, with a dark header and a light main content area.

Business_ID	Cuisine	_id	_type	_index	_score
XjeGryxde-tQZF_Ewu7NCw	korean	FR7kmIYBK81j1dtclFpc	restaurant	restaurants	0
IYVeY3q78hefg6o3IHbFSw	chinese	Oh7kmIYBK81j1dtclloq	restaurant	restaurants	0
pBgnAbZQsSvv_HJGR1ydRQ	mexican	HB7kmIYBK81j1dtcllrz	restaurant	restaurants	0
AqQkBsacSskJN2FdPeBkmg	chinese	Hx7kmIYBK81j1dtcmfPS	restaurant	restaurants	0
VuR31hMA88aH42jrBM_CBA	french	IB7kmIYBK81j1dtcmFqv	restaurant	restaurants	0
FXhSEZF1qjMZh4hR0_8JMQ	chinese	JR7kmIYBK81j1dtcmIqm	restaurant	restaurants	0
IaVv4YRvo3Z1pxQX6ub1FA	mediterranean	Jx7kmIYBK81j1dtcmIpv	restaurant	restaurants	0
Ry9w-jFALWYhaz16jyIKw	italian	LR7kmIYBK81j1dtcmVrp	restaurant	restaurants	0
PqgnNrUtU7XTKQvUM4cyqQ	mexican	OB7kmIYBK81j1dtcmIpd	restaurant	restaurants	0
bR2IOg3D113G4TzKYbEwFA	mediterranean	SR7kmIYBK81j1dtcmVp4	restaurant	restaurants	0
vFmBN7U25QAipTjK_EFmfA	chinese	Sh7kmIYBK81j1dtcmVre	restaurant	restaurants	0

Code for querying DynamoDB and inserting partial data into Elastic Search is available in “elastic_search_final.py”.


Note: This is considered a single user service . Hence, we store the results of a user in DynamoDB and return them to the user when they request a search next time

Note: The sender and receiver emails are predefined in the lambda. This was done because, SES service required emails to be verified before they can be used to send/receive emails. Hence, random emails will not work even if they are asked in the chatbot.


Extra Credit :

“User_suggestions” table in DynamoDB is used to store previous search results of the user . These are shown to the user the next time they use the chatbot, upon request.


Chatbot Execution




Hey! Hope all is well. How can I help you today?




Do you want to look at previous suggestions?




Some of the suggestions are:
Gamja Tang Tang, 408 Main St
Buddakan, 75 9th Ave
Best Dumplings, 16 Humphrey St
Aura Cocina, 315 Meserole St



Do you want to search for more reservations?



Sure let me collect the details



Which city do you want to search in?

Hi

15:42

I want to make some reservation

15:43

yes

Type message...

Some of the suggestions are:
Gamja Tang Tang, 408 Main St
Buddakan, 75 9th Ave
Best Dumplings, 16 Humphrey St
Aura Cocina, 315 Meserole St
Fette Sau, 354 Metropolitan Ave

yes

15:44

Type message...



Manhattan



What cuisine are you interested in?

Chinese



What date are you looking for?

tomorrow



At what time do you want to dine?

7pm

Type message...



What is the party size?

4



Could you provide your phone number?

9113671234

15:45



Could you provide your email address ?

test1@test.com



Okay, so you would like to find a restaurant for a party size of 4 belonging to chinese cuisine in Manhattan on 2023-03-04 at 19:00. And you want to send the suggestions to the number 9113671234 and email test1@test.com. Is this correct?

Type message...



the number 9113671234 and email test1@test.com. Is this correct?

yes



Great! Let me check



Yes, have initiated the search. You will be getting the restaurant suggestions on your registered email test1@test.com.



Is there anything else I can help you with?

no



Great! Have a good day.

Type message...

Screenshot of email suggestions



abhishekkandukuri@gmail.com via amazonses.com

to me ▾



8:10 PM (0 minutes ago)



Hello and welcome to our Dining Concierge Bot! We're thrilled to assist you with your dining needs.

Thank you for choosing our service. We have curated an exciting list of restaurants of Chinese cuisine in Manhattan for 4 people. Here are your suggestions:

1. Vanessa's Dumpling House, 118A Eldridge St
2. Jing Fong Restaurant, 202 Centre St
3. Mr Q's Noodle, 5302A 8th Ave
4. Jade Spoon Asian Cuisine, 1473 Weaver St
5. Great Wall, 527 Lenox Ave

Thanks,

Dining Bot Team

Abhishek (ak9691@nyu.edu) Sidharth (sx2236@nyu.edu)
