MusiCloud

Vikram Sunil Bajaj vsb259 **Binal Modi** bjm470 **Shivani Payal** smp765 **Ilyas Habeeb** mih278

Abstract

Most music recommendation systems nowadays simply recommend music to their users. But have you ever wanted more? MusiCloud is a music recommendation system with a twist! It recommends music as well as **upcoming concerts** by artists of the recommended music in the user's vicinity.

Keywords: *AWS, S3, Lambda, Lex, API Gateway, SQS, SNS, SES, Elasticsearch Service, DynamoDB, Amazon Machine Learning*

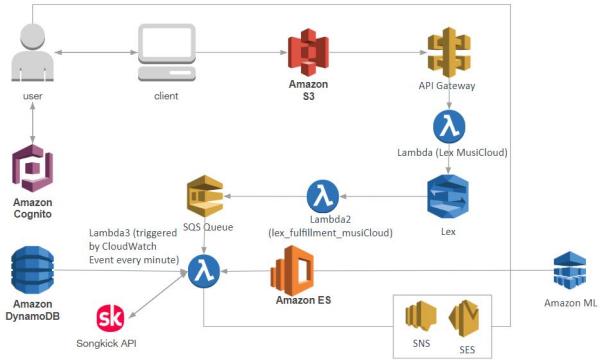
GitHub Link:
https://github.com/vikrambajaj22/aws-musiCloud

1 Introduction

MusiCloud is a system that recommends music to users and suggests places in their vicinity that have upcoming concerts/live music sessions by the artists of the recommended songs.

2 Architecture

The architecture is given in the diagram below:



- The user interacts with MusiCloud via a chatbot on the web client
- The website files are hosted on S3
- The first Lambda function is configured to facilitate the communication between the user and the Lex chatbot. Another Lambda function validates the user's inputs for the MusicSuggestionsIntent of the Lex chatbot
- Once the intent has been fulfilled, another Lambda function pushes the user's responses onto the SQS queue
- The third Lambda function is triggered by a CloudWatch Event trigger every minute. It performs the following:
  - Polls the SQS queue
  - Retrieves a message from the queue
  - Queries the DynamoDB table for artist names based on recommendations stored in Elasticsearch
  - Calls the Songkick API to retrieve upcoming concerts for the retrieved artists
  - Sends an SMS/Email to the user with the results

3 Implementation

- **Cognito**
  - Cognito is used to perform user authentication
  - Cognito provides a **Sign-Up page** to add users to a user pool and a **Login page** to validate users
  - The POST method of the API is secured using Cognito authentication; only successfully logged-in users can call the POST method
- **Lex**
  - Lex is AWS's service to create conversational bots; it serves as the base for the user's interaction with MusiCloud
  - It has 3 intents, MusicSuggestionsIntent being the main one
  - The user's inputs to this intent are validated using a Lambda function and another Lambda function handles the fulfillment of this intent
  - The bot asks the user for his/her preferred genre and location, and also asks for phone and email
- **Lambda**
  As discussed, there are multiple Lambda functions:

- **lex_lambda_musiCloud**: facilitates communication between the user and the Lex bot
- **lambda_lex_validation_musiCloud**: validates the user's inputs when the MusicSuggestionsIntent is fired
- **lex_fulfilment_musiCloud**: handles the fulfillment of the MusicSuggestionsIntent; pushes the user's inputs as a message onto the SQS queue
- **lambda_elasticsearch_dynamo_songkick**: Polls the SQS queue, retrieves a message from the queue, queries the DynamoDB table for artist names based on recommendations stored in Elasticsearch, calls the Songkick API to retrieve upcoming concerts for the retrieved artists, and sends an SMS/Email to the user with the results
- **lambda_csv_reader_musiCloud**: loads records from the test CSV file into DynamoDB

- **Amazon ML**
  - We used the **MetroLyrics dataset** from Kaggle
  - We split it into a training set (80%) and test set (20%)
  - The training set is used to train the recommendation system using Amazon ML. It has attributes including artist, genre, song name and year and has a binary target variable (1: recommended, 0: not recommended)
  - The test set is loaded into DynamoDB
  - Batch predictions are generated on the test set using the trained model, and the recommended songs (with target=1) are stored in Elasticsearch, indexed by genre. We used a score threshold of 0.35 to determine if a song must be recommended (the default threshold is 0.5)

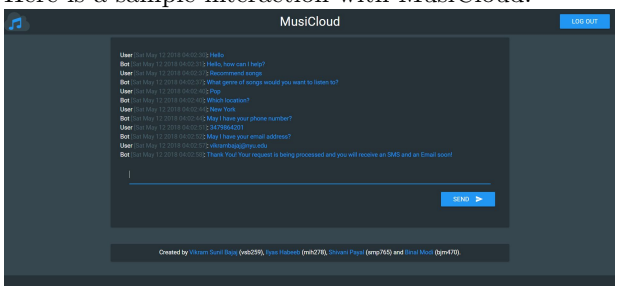
- **Elasticsearch and DynamoDB**
  - In Elasticsearch, we used genre names as indices to facilitate a quicker search
  - Recommended songs were ingested into Elasticsearch using **Logstash**
  - Our DynamoDB table uses genre as the primary partition key and song as the sort key. Records in DynamoDB also contain artist names and year.

- **Songkick API**
  - The **Songkick API** is used to retrieve upcoming concerts for the artists of the recommended songs.
  - It gives multiple concerts for most artists, however the concerts may not be at the location specified by the user during his/her conversation with the Lex bot. In such cases, we return concerts in other locations.

- **SNS/SES**
  - SNS is used to send an SMS to the user with the results
  - SES is used to email the results to the user

4 Results

Here is a sample interaction with MusiCloud:



The user receives an email with the following details:

Hello! Here are our suggestions for pop songs:
1. Song: desire jazzed
Artist: blank jones julian roman wasserfuhr

2. Song: shattered hollow
Artist: first aid kit

3. Song: mind of the wonderful jazzed
Artist: blank jones julian roman wasserfuhr

4. Song: smell the color 9
Artist: chris rice

5. Song: jamais deixarei voce
Artist: bruna karla

6. Song: kann den schwachsinn sduan nde sein
Artist: erste allgemeine verunsicherung

7. Song: si tu me miras
Artist: alejandro sanz

8. Song: domani u00e8 un altro film
Artist: dear jack

9. Song: my silver lining
Artist: first aid kit

10. Song: frozen lake
Artist: first aid kit

Sorry! No events happening around the location you mentioned :(
However, we found the following upcoming events in other locations:
1. Event: Dear Jack and Le Vibrazioni at Phenomenon (June 1, 2018)
Check out the Link: http://www.songkick.com/concerts/33725659-dear-jack-at-phenomenon?utm_source=48677&utm_medium=partner

2. Event: Erste Allgemeine Verunsicherung and EAV at Volksbankmesse (February 8, 2019)
Check out the Link: http://www.songkick.com/concerts/33050524-erste-allgemeine-verunsicherung-at-volksbankmesse?utm_source=48677&utm_medium=partner

3. Event: The Biggest Weekend 2018
Check out the Link: http://www.songkick.com/festivals/2197709-biggest-weekend/id/32659939-the-biggest-weekend-2018?utm_source=48677&utm_medium=partner

Enjoy!