WELDCLASS SWE INTERNSHIP REPORT

By Ayaan Adil

I have selected the first component of the integration task. I have also made an attempt at doing the second component.
The repository link: https://github.com/17Ayaan28/Weldclass_Internship_task

What do I need to do ?

For the first component, I need to fetch call recording files from the 8x8 database and store them on google cloud so that they can be transcribed using google speech to text API.

Strategy -

> First make an API call to the recordings API provided in the design brief
  - https://vcc-naX.8x8.com/api/recordings/files/

The call recording are of several types - outbound, direct agent access, busy. In DAA, the customer directly connects to a particular agent and the number of the caller is saved. In outbound calls the dial out number is saved.

> This API call will fetch the data of all the file recordings. According to my understanding of this endpoint, it seems it only returns the filename and other data associated with a recording, it does not return the actual contents of the .wav file. We will have to make API calls to another endpoint to get the contents of the .wav file. I am not sure if I am right about this, but if I am not it simply involves extracting the file content from the xml output.

> Once we get the XML output from the above call, we can use regex to extract the phone number and the filename from it. Since that is all we need to store on the cloud.

> Assuming that I am correct about the above, we will have to make calls to another endpoint to get the actual file. I did some research and found this endpoint -
https://api.8x8.com/storage/ap/v3/callrecording?relativePath=filename

So I make calls to the API and get the file by putting filename as a parameter.

> Once we get this data, we simply need to add it to google cloud. This depends on the kind of database. I have two examples, one of the firebase realtime database and the other a simple google cloud bucket.

> For every subsequent call, we can use this endpoint to get the file name and add it directly to cloud -

https://vcc-naX.8x8/api/recordings/files?agentID=bill&channel=8005557780&phoneNumber=4155551234&finish-date=2007-12-19 19:16:58Z

1) The firebase realtime db is a json based, No SQL database. It stores data in a JSON format. IF this db is used then the data can be made to look like this.

```
{
    "call_recordings" : {
        "phone_number" : "recorded file data",
        "0403479857" : "recorded file data",
        "9810411159" : "recorded file data",
        "0403479857" : "recorded file"
    }
}
```

The code for the firebase solution is in the file Solution.java -

https://github.com/17Ayaan28/Weldclass_Internship_task/blob/main/Solution.java

In this case, to store the file I convert it to a byte array.

2) We can also use a google cloud bucket to store these files. This can be done by structuring the data such that the phone numbers are directories and the files inside them are the associated calls.

The code for this case is in the file Solution_2.java -

https://github.com/17Ayaan28/Weldclass_Internship_task/blob/main/Solution_2.java

I have also added some code for the transcription which I think if run in a for loop on all the files in the bucket will give the transcription.

https://github.com/17Ayaan28/Weldclass_Internship_task/blob/main/transcribe.java

Queries -

Q1) I was a bit confused as to why the recording API had a type of call for busy and why would that be recorded ?

Also, If direct agent access is when a customer directly reaches out to an agent on their channel, how is it different from an inbound call in terms of the recording?

Was I correct in assuming that a separate endpoint returns the actual data of the file and this endpoint just returns other details ?

Call Type	Channel	Phone Number	Queue ID	Queue Name
Busy (Inbound/Outbound)	Channel of the call dialing to (for inbound only)	Caller ID of the caller	Queue ID of the queue	Queue Name of the queue
Outbound	#virtual_outbound#	Dial-out number	QUEUE_OUTBOUND	QUEUE_OUTBOUND
Direct Agent Access	#virtual_daa#	Caller ID of the caller	QUEUE_DIRECT_AGE NT_ACCESS	QUEUE_DIRECT_AGE NT_ACCESS

Q2) Are there any data privacy questions that come up with recording customer conversations and transcribing them ?

Q3) What is the most efficient database out of the ones that google provides for storing these kind of audio files ? ex. CloudSQL, Realtime DB, a bucket etc.

Challenges to come up with a solution -

1) For the first component, the main challenge entails obtaining and organising the data in such a format such that it can be added to the cloud database and then can be further utilised to transcribe. Since we have to first make API calls and then filter the data, find what's required and then add it the the db, this constitutes the main challenge.

2) The second challenge in the first component of the task is to store data in an organised manner so that it is easy to access it and code up the solution for further tasks.

Thank you for giving me an opportunity to perform this task !