PollyApp2
Documentation

GITHUB REPOSITORY LINK

https://github.com/MohsinJKhan/PollyApp/tree/develop

Submitted To	Paolo Cremonasi Riccardo Medana	Submitted By	Pragati Gupta Mohsin Khan

0.1 Abstract

The project aims to extend the PollyApp1 project developed by Ali Emre Soyluçiçek and Laura Schiatti Sisó.
The motive is to use the PollyApp1 to convert the footer notes in the slides, converting them to audios and then embedding them into slides.

A Nodejs based application that exploits Text-to-Speech service provided by Amazon Polly to perform the following transformations-

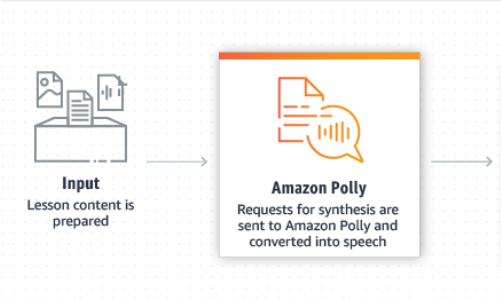
- Convert text file (.txt format) to audio file (.mp3) and multiple text files simultaneously present in a folder to multiple audio files.
- In addition to that the main functionality is to read the footer notes in the Microsoft PowerPoint Presentation (.pptx format), converting them to audios using Amazon Polly and then inserting those audios in the corresponding slides in the deck.

1 Contents

2 Amazon Web Services(AWS)

2.1 Introduction to Amazon Polly

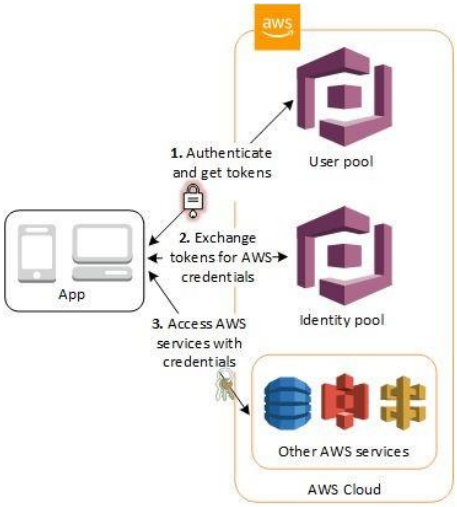
Amazon Polly converts input text into life-like speech. Provide parameters and Amazon Polly then synthesizes the provided text into a high-quality speech audio stream.
The main flow can be described as follows



More details about SynthesizeSpeech method
https://docs.aws.amazon.com/polly/latest/dg/API_SynthesizeSpeech.html

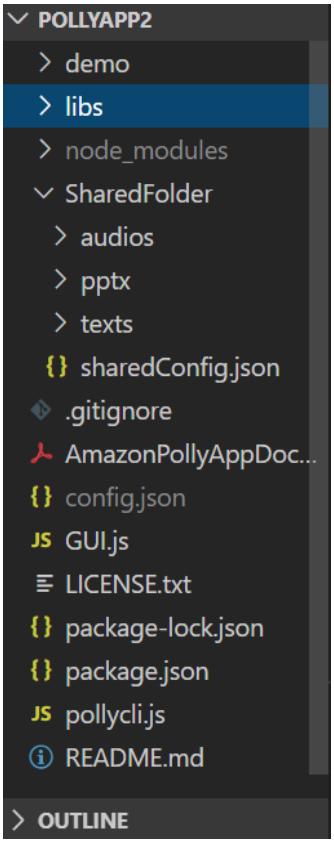
2.2 Authentication with Amazon Cognito

Use Amazon Cognito to deliver temporary, limited-privilege credentials to your application, so that your users can access AWS resources.
Amazon identity pool



3 Project File Structure

3.1 Shared Folder Structure.



Some of the configuration has to be present in the shared folder for the application to be able to perform properly. This shared folder will be provided by the Professor. The folder includes:

- sharedConfig.json
- Folder that includes the text files (.\PollyApp2\SharedFolder\texts)
- Folder that audio files will be saved to (.\PollyApp2\SsharedFolder\audios)
- Folder which includes the pptx files (.\PollyApp2\SharedFolder\pptx)

sharedConfig.js file will include several configuration parameters:

- aws_pool_id: Identity pool id for the AWS Cognito Service
- text_folder: Name of the text folder
- audio_folder: Name of the audio folder
- default_voice: Default name of the voice id for the speech
- extra_voice_character: Character to be used to identify if there is an extra voice parameter within the text file.

3.2 Modules

In Order to make the project more reader, We have exploited the functionality of the “Modules” so that the program has independent, interchangeable modules, such that each contains everything necessary to execute only one aspect of the desired functionality. A module interface expresses the elements that are provided and required by the module.

Slno	Library Name	Functionality
1	getNotes.js	The module supports fetching the notes from the xml file contained in the zip folder.
2	compress.js	The module supports compressing the PPTX into the Zip File.
3	generateAllAudios.js	The module supports feeding multiple text files to Amazon Polly
4	getPollyParams.js	This is utility function that generate pollyParams for AWS Polly service call
5	manageConfig.js	Other relevant configuration calls for the file system
6	polly.js	The module to instantiate Polly and to feed single text file and to generate audio
7	processPPTXAudioHelper.js	The module helps perform audio conversion for processPPTXFile
8	processPPTXFile.js	Perform functions on PPTX like zip conversion, extraction, Fetching Configs, unzipping etc
9	questions.js	Handles the question asked to user while using GUI interface
10	readFile.js	Module to read the contents of the file
11	removeDir.js	Module to remove the directoty

4 Prerequisites to run the Application

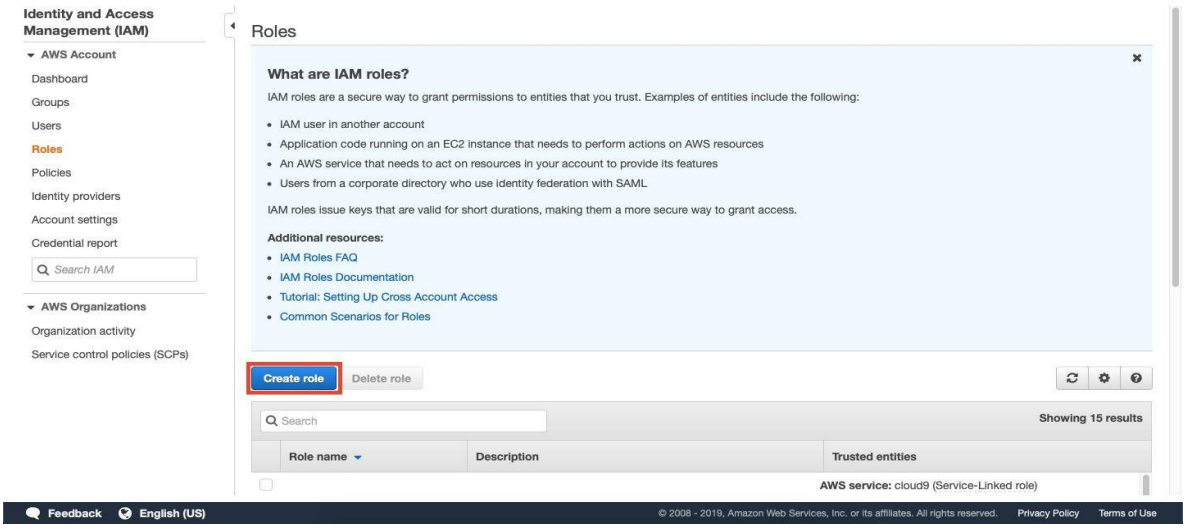
4.1 Other Standard Libraries

1. **Inquirer.js**
   A collection of common interactive command line user interfaces. Inquirer ease the process of asking end user questions , parsing, validating answers, managing hierarchical prompts and providing error feedback. Inquirer provide the user interface , and the inquiry session flow.

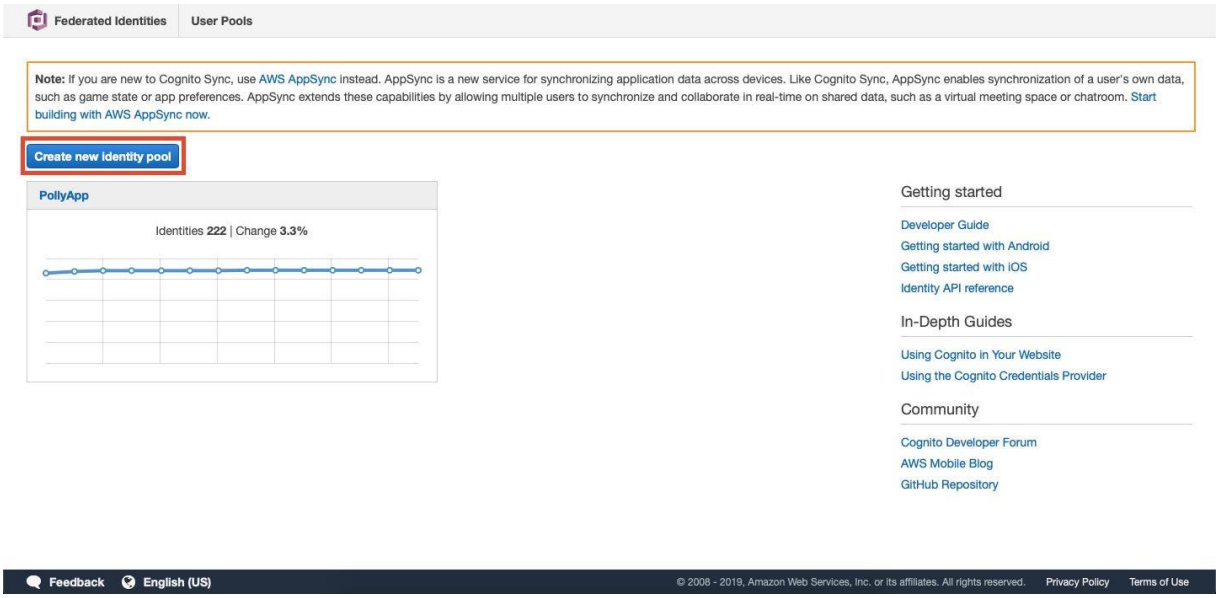
2. **cli_init (CLI handler)**
   The library is used to build the command line calls for the project.

4.2 Steps to use Cognito Service

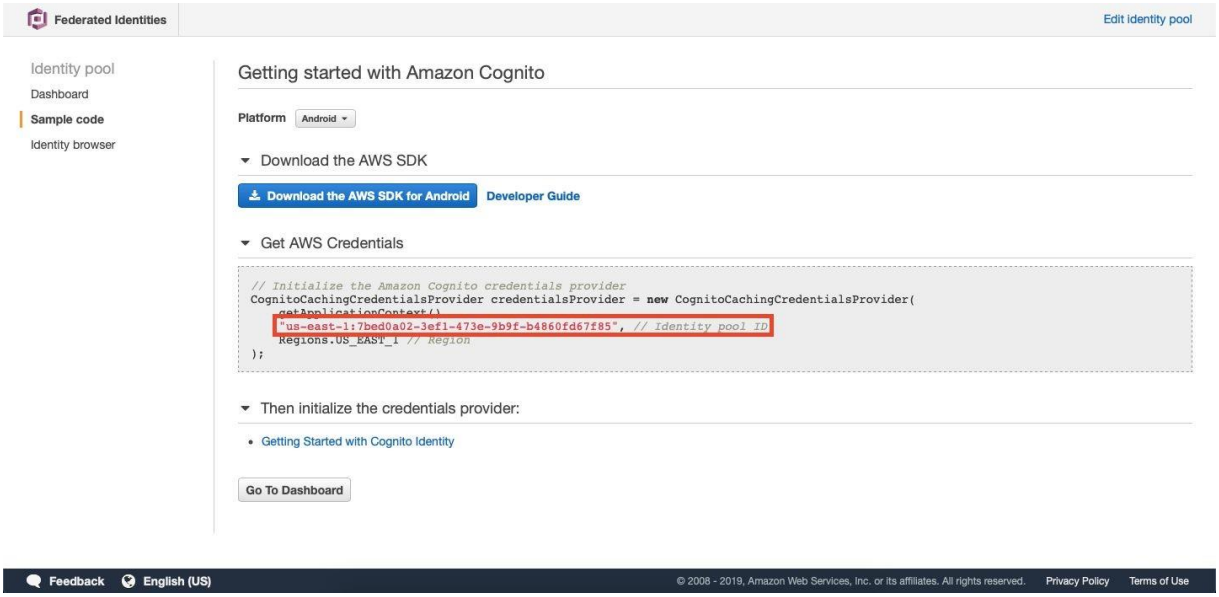
1. Create a Role from AWS IAM service and assign AWS Polly full access for that specific role.
2. Create an Identity Pool from AWS Cognito Service.
3. Copy the application to a desired folder.
4. Open terminal/cmd in the same folder as the application
5. Write npm install, wait for it to finish.
6. Write node app.js, application will guide you.



Creating a Role in IAM Service



Creating an Identity Pool from Cognito Service



Identity Pool Id Location

5 Project Functionality

The core functionality involves handling three types of transformation scenarios

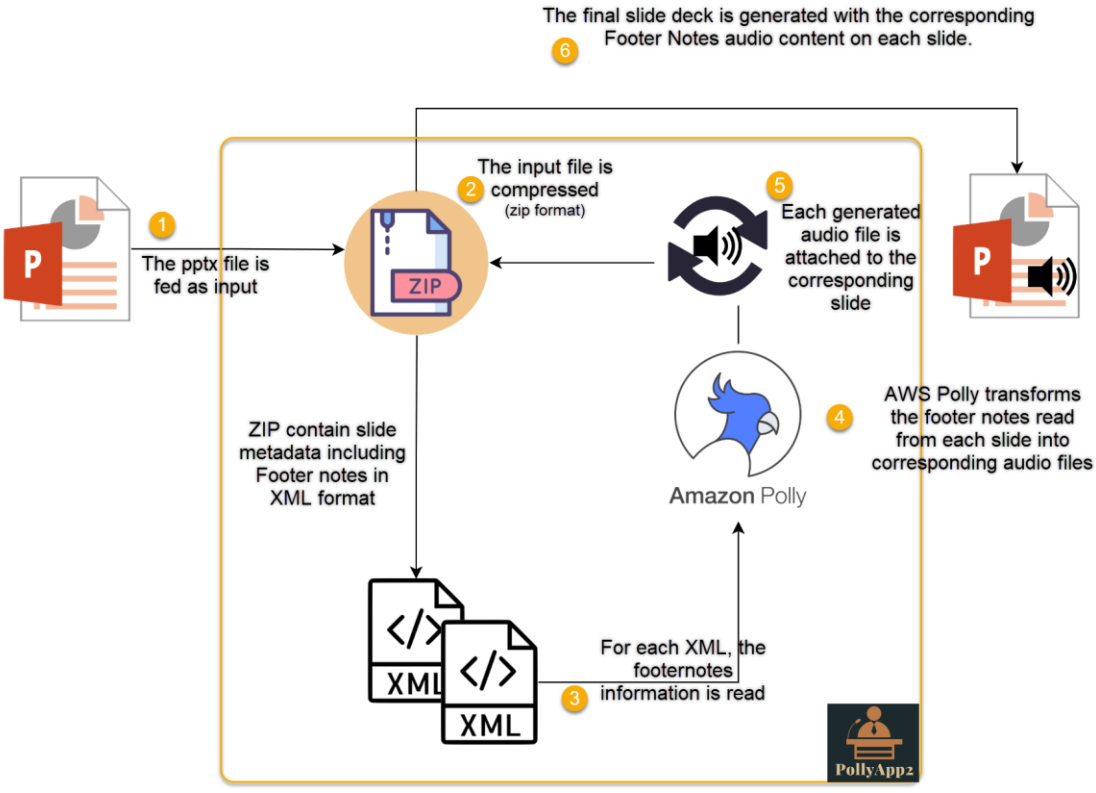
- Converting the footer notes in the pptx and embed as audio file in the slide
- Converting Multiple Text files into Multiple Audio files
- Single Text File into Audio File

5.1 Converting the footer notes in pptx and embed as audio file in the slide

> ***NOTE***
> In PollyApp2 the original pptx file contains the dummy audio files in the slides. The project creates the audio files from the footer notes and then replaces this dummy audio files with the actual audios. The functionality of insertion of the audio file without any dummy will be looked upon in the next phase of the project.

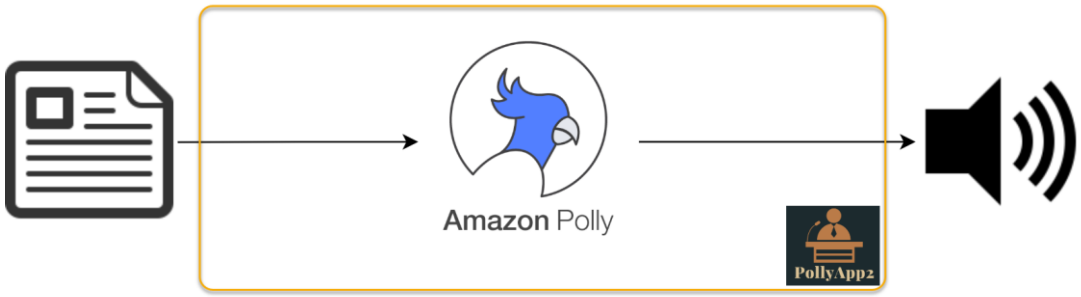
The functionality is a follows

1. First the pptx is converted into the zip folder. This is because zipped format of the pptx converts the PPT into a folder like structure with each slide having its own notation as per the formatting done on the slide.
2. The footer notes of the respective slides can be easily fetched considering it to be a file structure using the predefined IO libraries.
3. These notes are then fed to AWS Polly.
4. The converted file will be replacing the audio files already present in the zip.
5. The zipped format will be unzipped into .pptx structure again



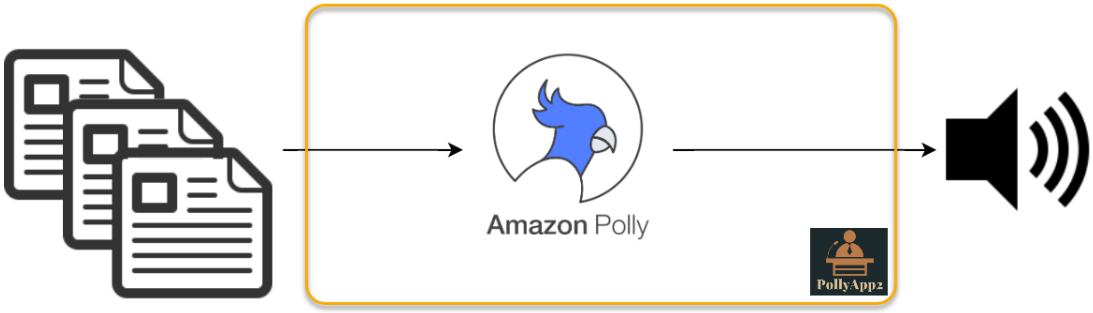
5.2 Single Text File into Audio File

The functionality included is to convert the text file (.txt format) to the Audio File. The file needs to be placed at texts subfolder of the Shared Folder (.\PollyApp2\SharedFolder\texts)



5.3 Converting Multiple Text files into Multiple Audio files

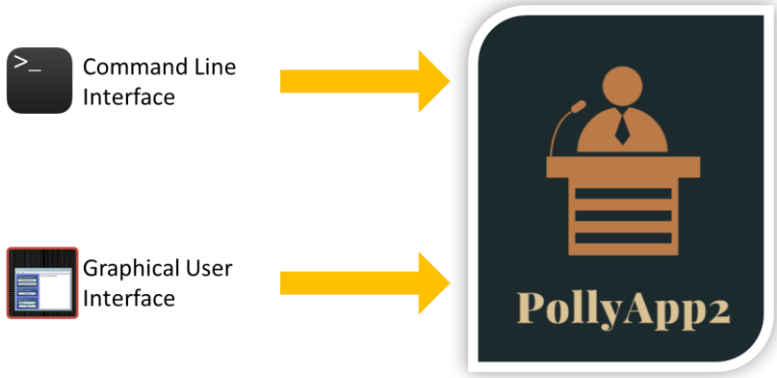
The functionality covered is to convert the multiple text file (.txt format) present in the texts folder in one go to the multiple Audio Files. The file needs to be placed at texts subfolder of the Shared Folder (.\PollyApp2\SharedFolder\texts)



6 Project Usage

The input media can be fed to the PollyApp2 via two methods

- As a Command Line Interface(CLI)
- As a Graphical User Interface (GUI)



6.1 Input via Command Line Interface(CLI)

The command to access the CLI is

- node pollycli

To access the details about the functionality (HelpDoc), the user can use the command

- node pollycli --help

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL
PS C:\Users\Pragati Gupta\Documents\GitHub\PollyApp2> node pollycli --help
pollycli <cmd> [args]

Commands:
  pollycli singletext [filename]  welcome to pollycli
  pollycli singlepptx [filename]  welcome to pollycli
  pollycli multitext              welcome to pollycli

Options:
  --version  Show version number                                    [boolean]
  --help     Show help                                              [boolean]
PS C:\Users\Pragati Gupta\Documents\GitHub\PollyApp2>
```

6.1.1 Single Text File Conversion

The first functionality covered in the command is to convert the text file (.txt format) to the Audio File.

The command is

- node pollycli singletext --filename *“<<file_name.pptx>>”*

The “—filename” is optional for the case the name of the file contains spaces.
The file is fetched from the texts folder and is fed to AWS Polly. The Polly service reads the text and converts it to an audio file. The configurations are being fetched from the sharedConfig.json like the speed, the voice etc. The converted file is stored in the audios folder present in the Shared Folder only.

```
PS C:\Users\Pragati Gupta\Documents\GitHub\PollyApp2> node pollycli singletext text
Authenticating AWS...

Generating Audio
```

6.1.2 Multiple Text File Conversion

The functionality covered in the command is to convert the multiple text file (.txt format) present in the texts folder in one go to the multiple Audio Files. The file needs to be placed at texts subfolder of the Shared Folder (.\PollyApp2\SharedFolder\texts)

The command is

- node pollycli multitext

The text folder is fetched and then all the audio files are fed to AWS Polly in an iterative way. The Polly service reads the text and converts it to audio files and store them into the location (.\PollyApp2\SharedFolder\audios).
The configurations are again being fetched from the sharedConfig.json.

```
PS C:\Users\Pragati Gupta\Documents\GitHub\PollyApp2> node pollycli multitext
Authenticating AWS...

Generating Audio

Generating Audio

Generating Audio
```

6.1.3 Converting Footer notes into Audios in Presentation

The pptx file is placed int the pptx folder under the SharedFolder. The command used

- node pollycli singlepptx --filename *"<<file_name>>"*

```
PS C:\Users\Pragati Gupta\Documents\GitHub\PollyApp2> node pollycli singlepptx --filename "demo 3"
Authenticating AWS...
Authenticating AWS...

Generating Audio
```

6.2 Input via Graphical User Interface (GUI)

The command to access the CLI is

- node GUI

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

PS C:\Users\Pragati Gupta\Documents\GitHub\PollyApp2> node GUI
```

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+-+
|T|e|x|t|-|t|o|-|S|p|e|e|c|h| |C|o|m|m|a|n|d|-|L|i|n|e| |T|o|o|l|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+-+

Authenticating AWS...
? Select the type of file (Use arrow keys)
> Text
  Power Point Presentation
```

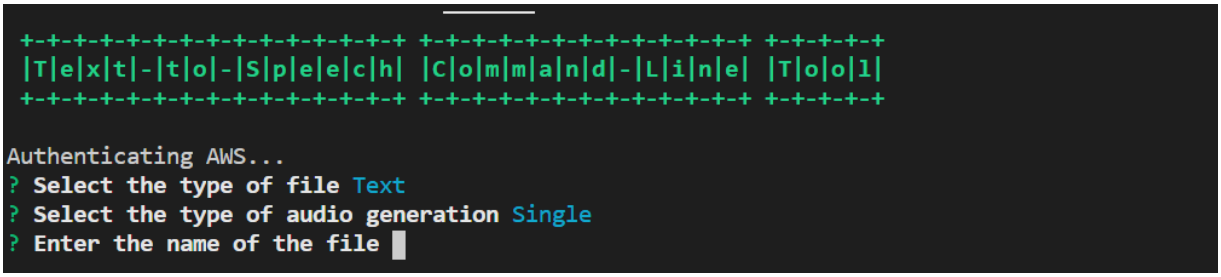
6.2.1 Single Text File Conversion

The first functionality covered in the command is to convert the text file (.txt format) to the Audio File. The file needs to be placed at texts subfolder of the Shared Folder (.\PollyApp2\SharedFolder\texts). The user can mention the name of the file which is there in the SharedFolder. No need to maintain the extension of the file. The system implicitly checks the file format.

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+-+
|T|e|x|t|-|t|o|-|S|p|e|e|c|h| |C|o|m|m|a|n|d|-|L|i|n|e| |T|o|o|l|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+-+-+-+-+-+-+-+-+-+ +-+-+-+-+

Authenticating AWS...
? Select the type of file Text
? Select the type of audio generation (Use arrow keys)
> Single
  All
```

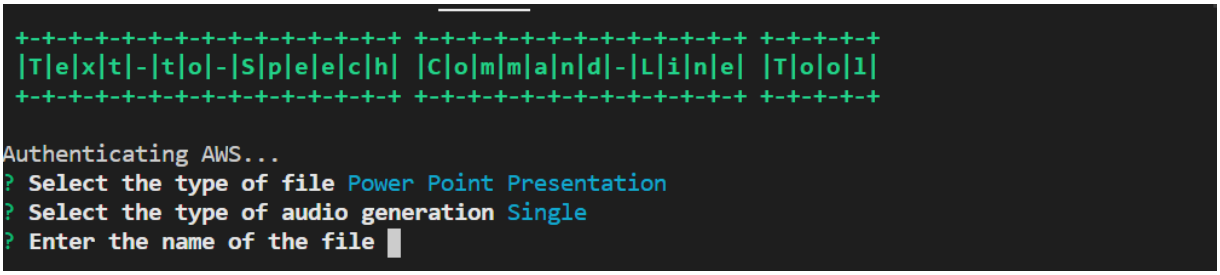


6.2.2 Multiple Text File Conversion

The functionality covered in the command is to convert the multiple text file (.txt format) present in the texts folder in one go to the multiple Audio Files. The file needs to be placed at texts subfolder of the Shared Folder (.\PollyApp2\SharedFolder\texts). The use can select text and then can select “ALL” option in the menu.
The text folder is fetched and then all the audio files are fed to AWS Polly in an iterative way. The Polly service reads the text and converts it to audio files and store them into the location (.\PollyApp2\SharedFolder\audios).

6.2.3 Converting Footer notes into Audios in Presentation

The pptx file is placed int the pptx folder under the SharedFolder. The use can select the Powen Point Presentation shown in the menu. It then asks for the name of the PPTX file. The user just need to mention the file name.



7 Bibliography

Tutorials

- https://medium.com/@anaptfox/getting-started-with-amazon-polly-using-node-js-345e84dbd23d
- https://medium.com/@smcelhinney/building-a-greeting-app-using-amazon-polly-and-nodejs-a605f29c20f5
- https://trevorsullivan.net/2016/12/01/amazon-aws-cloud-polly-nodejs/

Amazon Cognito:

- https://docs.aws.amazon.com/cognito/latest/developerguide/what-is-amazon-cognito.html

Contact Persons

Role	Name	Occupation	E-Mail
Manager	Paolo Cremonesi	Faculty, Polimi	paolo.cremonesi@polimi.it
Coordinator	Riccardo Medana	Faculty, Polimi	riccardo.medana@polimi.it
Coordinator	Federico Schiepatti	Faculty, Polimi	federico.schiepatti@polimi.it
Developer	Pragati Gupta	Student, Polimi	pragatigupta.97@gmail.com
Developer	Mohsin Khan	Student, Polimi	mohsin.j.khan310@gmail.com