

ITIS - 6177 System Integration

FINAL PROJECT

Microsoft Azure Text to Speech API Documentation

Name: Abhishekji Dumala

Student ID: 801307903

Email-ID: Abhishekji Dumala

Introduction:

This is a document which explains the total process of generating the text to speech using Microsoft Azure API key. Deep neural networks are used in this process to generate the texts into the speech format.

Text to Speech Requirements:

This is built by using Microsoft Azure Text to Speech API. For this initially we have to create an account in the Azure portal.

<https://azure.microsoft.com/en-us/products/cognitive-services/text-to-speech/>

1. Setup:

- Creating the speech resource in the azure microsoft account for getting the subscription key and endpoint.
- Installation the code through git clone link
- cd ITIS-6177-Final-Project.
- Create .env file in this directory and include the following data:
 - API_AUTH_URL = <https://eastus.api.cognitive.microsoft.com/sts/v1.0/issuetoken>
 - API_URL = <https://eastus.tts.speech.microsoft.com/cognitiveservices/v1>
 - API_KEY = <your-api-key>
 - SERVICE_NAME_NEURAL = (en-US, JessaNeural)
 - VOICE_URL = <https://eastus.tts.speech.microsoft.com/cognitiveservices/voices/list>
 - VOICE_HOST = eastus.tts.speech.microsoft.com

2. Installation dependencies:

- npm install - nodejs package manager.
- node app.js - JavaScript run engine.

3. Applications:

- Node js for running the js program.
- Terminal for running the code.
- Postman for testing purposes.

4. Testing process:

- We can test this the postman workspace api by giving this link:
<http://167.71.14.20:4004/>
- The main two endpoints for this API are:
 - /voices
 - /create

1. **GET Method:** This is the link used for getting the list of voices to be viewed.

<http://167.71.14.20:4004/voices>

The screenshot shows a Postman interface with a GET request to `http://167.71.14.20:4004/voices`. The request is successful, returning a 200 OK status with a response time of 1755 ms and a body size of 155.08 KB. The response is displayed in JSON format, showing details for a voice named 'Adri'.

KEY	VALUE	DESCRIPTION
text	how are you?	
Key	Value	Description

```
{  "Name": "Microsoft Server Speech Text to Speech Voice (af-ZA, AdriNeural)",  "DisplayName": "Adri",  "LocalName": "Adri",  "ShortName": "af-ZA-AdriNeural",  "Gender": "Female",  "Locale": "af-ZA",  "LocaleName": "Afrikaans (South Africa)",  "SampleRateHertz": "48000",  "VoiceType": "Neural"}
```

Response:

Code Description

200 Successful

400 Bad data

500 Internal Server Error

2. POST Method: This is the link used for posting our request to convert text to speech in json format in Postman application.

<http://167.71.14.20:4004/create>

POST > Body > raw > JSON

```
{  
  "TEXT": "Enter the text which you want to convert to speech? "  
}
```

POST ⌵ http://167.71.14.20:4004/create Send ⌵

Params ● Auth Headers (7) Body ● Pre-req. Tests Settings Cookies

raw ⌵ JSON ⌵ Beautify

```
1 {  
2   "TEXT": "THIS IS MY FINAL SYSTEM INTEGRATION PROJECT TEXT TO SPEECH"  
3 }
```

Body ⌵ 🌐 200 OK 2.68 s 201.3 KB Save Response ⌵

0:00 / 0:04 🔊 ⋮

🍪 Cookies ✅ Auto-select agent 🏃 Runner 🗑️ Trash ⛶

Here Default given voices in the app.js file.

Response:

Code Description

200 Successful

400 Bad data

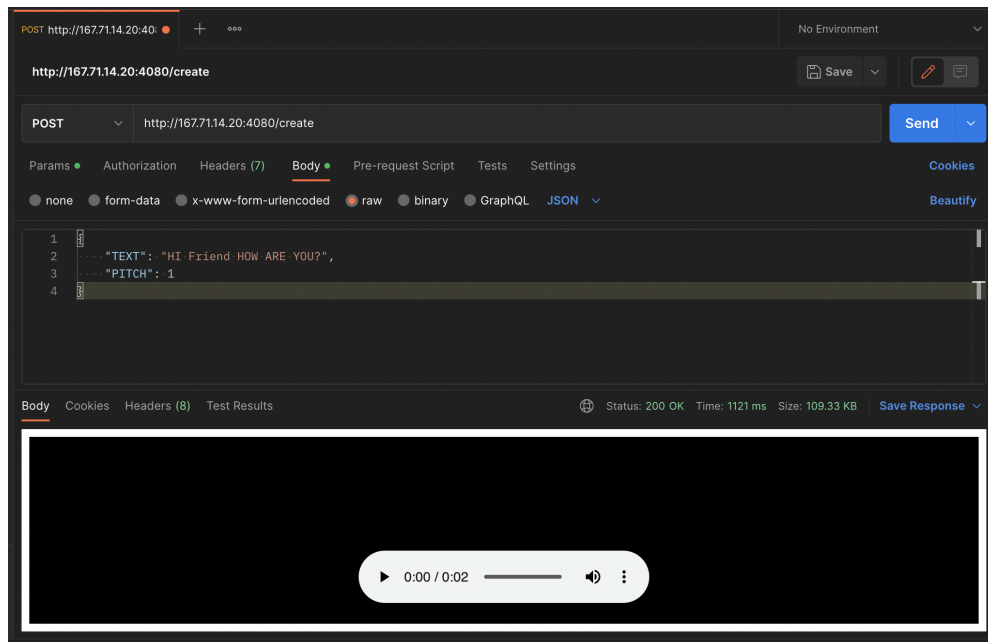
500 Internal Server Error

PITCH:

In addition to this, there are a few features like Pitch to the voice. Where it takes the pitch value and converts it into accordingly.

POST > Body > raw > JSON

```
{
  "TEXT": "Enter the text which you want to convert to speech? ",
  "PITCH": 1
}
```



SPEEキング_SPEED:

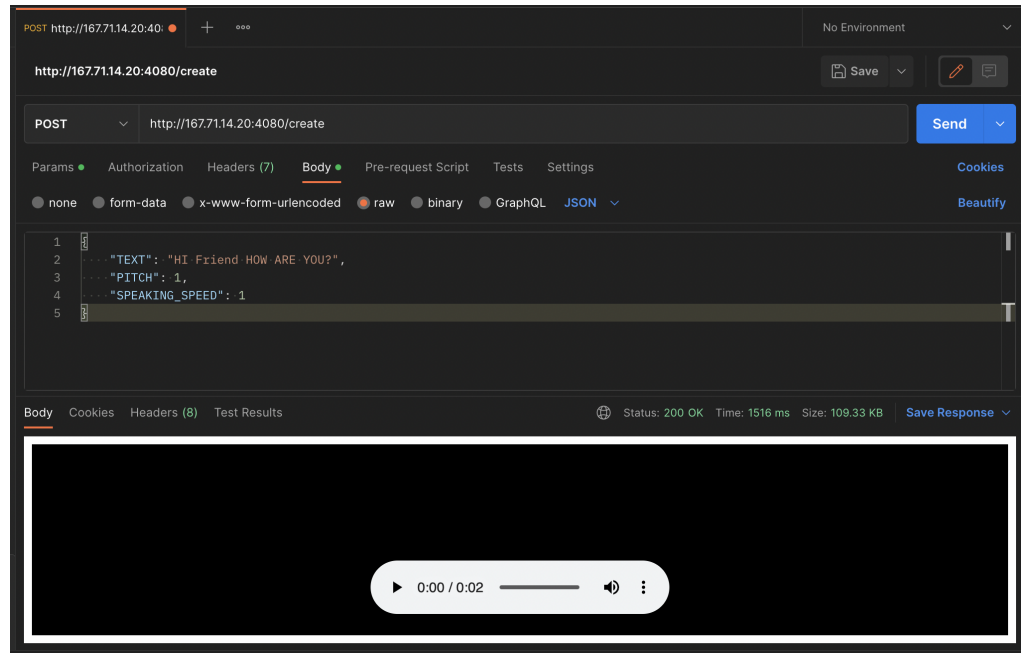
In addition to this, there are a few features like SPEEキング_SPEED to the voice.

Where it takes the speaking speed value and converts the voice speed according to the given value.

POST > Body > raw > JSON

```
{
```

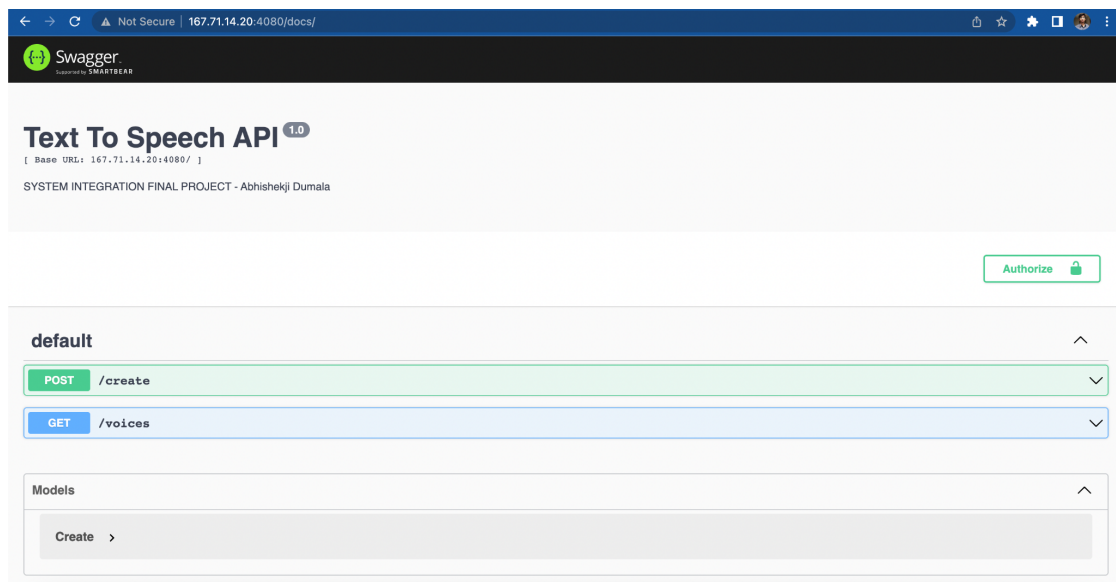
```
    "TEXT": "Enter the text which you want to convert to speech? ",  
    "PITCH": 1,  
    "SPEAKING_SPEED": 1  
  }  
}
```



5.Swagger:

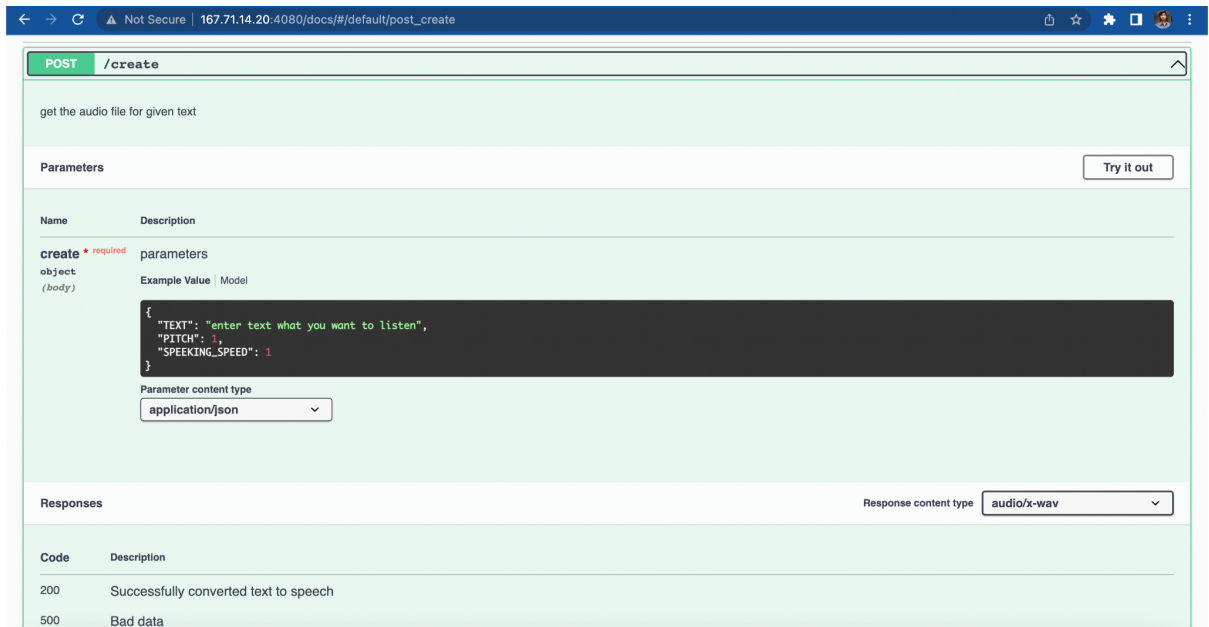
Follow the below link to access swagger application:

<http://167.71.14.20:4080/docs/>



1. /create

Click on try it out to post the values for getting the text to speech output of all the create.



Swagger UI for the `POST /create` endpoint. The interface shows the endpoint description, parameters, and responses.

Endpoint: `POST /create`

Description: get the audio file for given text

Parameters:

Name	Description
create * <small>required</small>	parameters
object (body)	Example Value Model

```
{  "TEXT": "enter text what you want to listen",  "PITCH": 1,  "SPEAKING_SPEED": 1}
```

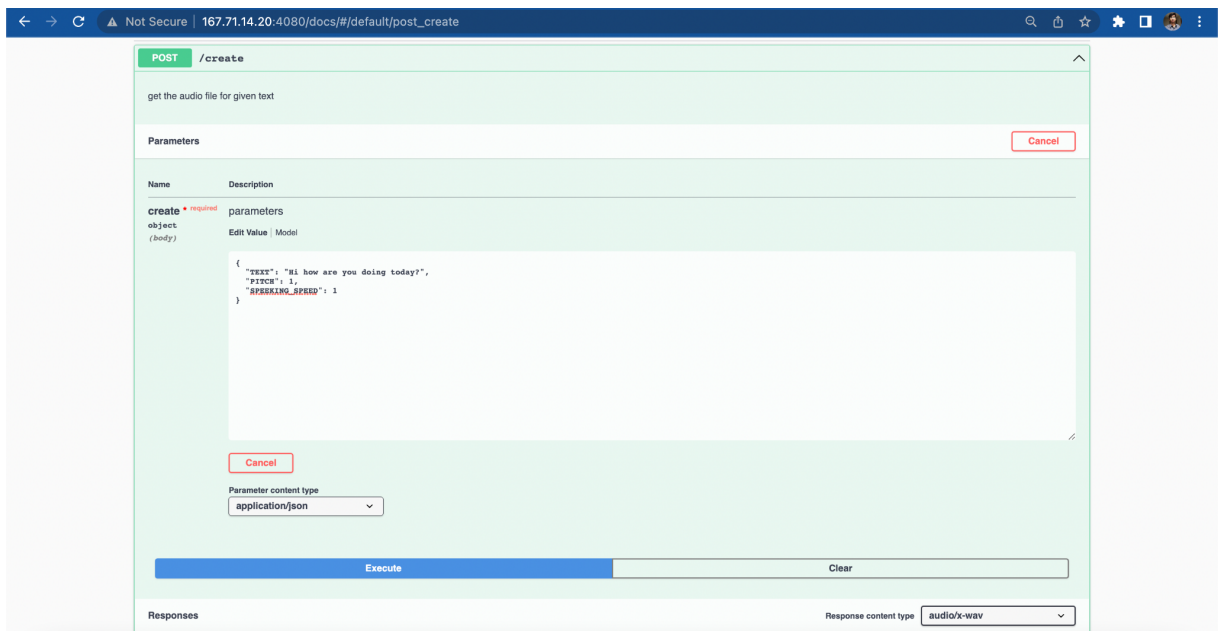
Parameter content type: `application/json`

Responses:

Code	Description
200	Successfully converted text to speech
500	Bad data

Response content type: `audio/x-wav`

Enter the text whatever you want to listen to in the speech format.



Swagger UI for the `POST /create` endpoint. The interface shows the endpoint description, parameters, and responses. A text input field is present for entering the text to be converted to speech.

Endpoint: `POST /create`

Description: get the audio file for given text

Parameters:

Name	Description
create * <small>required</small>	parameters
object (body)	Edit Value Model

```
{  "TEXT": "Hi how are you doing today?",  "PITCH": 1,  "SPEAKING_SPEED": 1}
```

Parameter content type: `application/json`

Buttons: `Execute` (blue), `Clear` (grey), `Cancel` (red)

Responses:

Code	Description
200	Successfully converted text to speech
500	Bad data

Response content type: `audio/x-wav`

Click on execute to view the result.

The screenshot shows a REST client interface with a browser address bar at `167.71.14.20:4080/docs/#/default/post_create`. The interface has a top bar with "Execute" and "Clear" buttons. Below is a "Responses" section with a dropdown for "audio/x-wav". The "Curl" section contains the following command:

```
curl -X 'POST' \
  http://167.71.14.20:4080/create' \
  -H 'accept: audio/x-wav' \
  -H 'Content-Type: application/json' \
  -d '{
    "TEXT": "enter text what you want to listen",
    "PITCH": 1,
    "SPEAKING_SPEED": 1
  }'
```

The "Request URL" is `http://167.71.14.20:4080/create`. The "Server response" section shows a "200" status code and a "Response body" with an audio player. The "Response headers" are:

```
access-control-allow-origin: *
connection: close
content-type: audio/x-wav
date: Fri, 09 Dec 2022 16:43:02 GMT
strict-transport-security: max-age=15724800; includeSubDomains
transfer-encoding: chunked
x-aws-idempotency-token: 84
x-powered-by: Express
```

At the bottom, a "Responses" table lists:

Code	Description
200	Successfully converted text to speech
500	Bad data

2. /voices

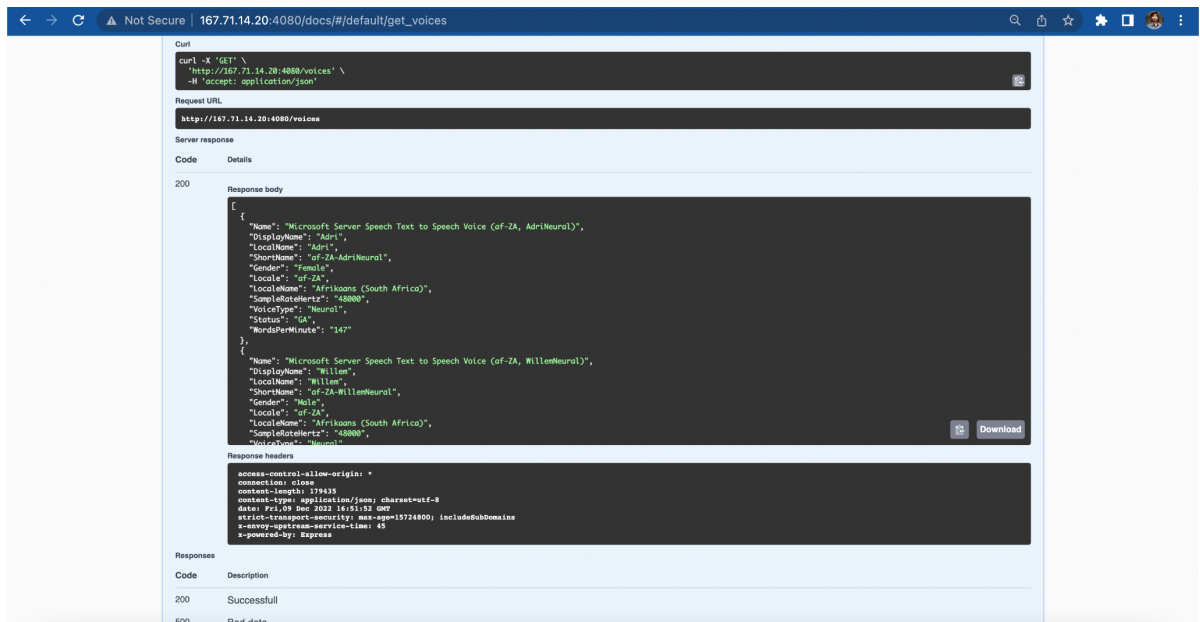
Click on try it out to get the list of values under voice to view which is getting retrieved from the microsoft azure.

The screenshot shows a REST client interface with a browser address bar at `167.71.14.20:4080/docs/#/default/get_voices`. The interface has a top bar with "POST /create" and "GET /voices" tabs. The "GET /voices" tab is active, showing the description "get the list of supported Voices". The "Parameters" section is empty, with a "Try it out" button. The "Responses" section has a dropdown for "application/json". The "Responses" table lists:

Code	Description
200	Successfull
500	Bad data

At the bottom, a "Models" section shows a "Create" button with a right arrow.

Click on Execute to view the list of voices available in this API.



Response:

Code Description

200	Successful
400	Bad data
500	Internal Server Error

References:

<https://azure.microsoft.com/en-us/products/cognitive-services/text-to-speech/>

<https://learn.microsoft.com/en-us/azure/cognitive-services/speech-service/index-text-to-speech>

<https://learn.microsoft.com/en-us/azure/cognitive-services/speech-service/get-started-text-to-speech>