

Azure Text to Speech

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

Introduction:

In this project, NodeJs express is used to construct APIs that can interface with the Microsoft Azure text to speech service while also requiring authentication. Before, you can proceed, you must first create or log into an Azure account. If you already have an account, you may be able to get the service for free.

How to subscribe in Azure portal:

1. Create a subscription or use school email address for free credit.
2. Create a resource group. Select region as “useast”.
3. In the resource group, search and create a “Speech Services” resource to connect the file path between Azure text to speech service
4. Key, EndPoint will be generated which is ready to use.

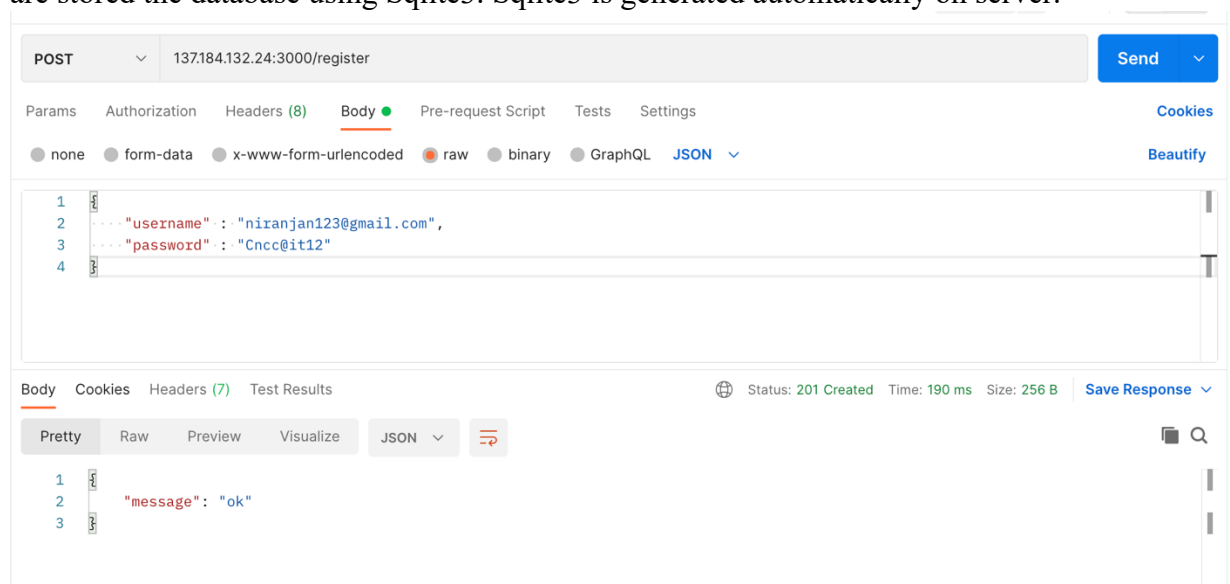
Import Json file in postman:

Import the JSON file which is shared in the supporting documents folder and 4 requests will be imported which are showed below.

Features:

1. Register:

Users can register if they didn’t create the account yet. Input parameters are username and password which are sent through raw JSON format. Once they registered the details are stored the database using Sqlite3. Sqlite3 is generated automatically on server.

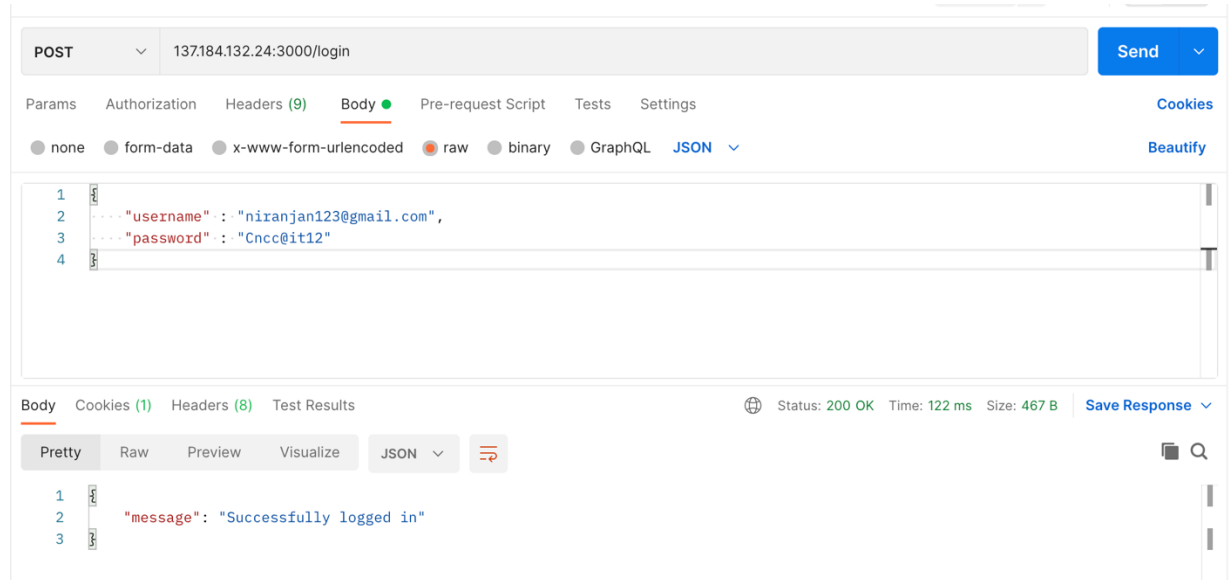


ITIS 6177 – SYSTEM INTEGRATION FINAL PROJECT

Niranjan Rao Deshineni

2. Login:

User can login once they are registered. Input parameters are username and password which are sent through raw JSON format. It checks the values in the Sqlite3 and authenticates the user.



3. Text to Speech API

Users can pass the text to API and get stream speech audio of their text. Text is passed as input param.

ITIS 6177 – SYSTEM INTEGRATION FINAL PROJECT

Nirnanjan Rao Deshineni

GET 137.184.132.24:3000/text2speech?text=my name is nirnanjan Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	text	my name is nirnanjan			
	Key	Value	Description		

Body Cookies (1) Headers (6) Test Results Status: 200 OK Time: 1058 ms Size: 18.18 KB Save Response

0:02 / 0:02

4. Logout:

User can logout using this API. No input parameters.

GET 137.184.132.24:3000/logout Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (8) Test Results Status: 200 OK Time: 124 ms Size: 346 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Successfully logged out"
3 }
```

**ITIS 6177 – SYSTEM INTEGRATION
FINAL PROJECT**

Niranjan Rao Deshineni

Dependencies used in this project:

- NodeJs - The compiler of javascript
- Express js - Running web server and API framework.
- Cookies - To save session logged in users
- Jwt - To encrypt saved user data for security
- Micorosoft Azure - To convert users text to speech
- npm - Nodejs package manager

Installation:

My project requires [Node.js](https://nodejs.org/) to run.
Install the dependencies using npm i and start the server.

Routes:

There are 4 routes written for my project that users can utilize

1. Register:

http
POST /register

Body:

```
raw"{\n \"username\" : \"ndeshine@uncc.edu\", \n \"password\" : \"12345678\" \n}"
```

username: string (required)
password: string (required)

Response: Json format
{message: string}

2. Login:

http
POST /login

Body:

```
raw"{\n \"username\" : \"ndeshine@uncc.edu\", \n \"password\" : \"12345678\" \n}"
```

username: string (required)
password: string (required)

Response: Json format
{message: string}

3. Register:

http

**ITIS 6177 – SYSTEM INTEGRATION
FINAL PROJECT**

Niranjan Rao Deshineni

GET / text2speech?text=my name is niranjan"

Response: audio

4. Logout:

http

GET /login

Response: Json format

{message: string}

Status codes:

200 – OK

201 – CREATED

400 – BAD REQUEST

404 – NOT FOUND

500 – INTERNAL SERVER ERROR

Host Address: 137.184.132.24

Port: 3000