

# **MAM English Project**

## **Backend Server (API) Documentation**

**This API is hosted using Google's Firebase Service**

**This Service is Also Used to Host the Database and Image/Audio Files.**

**All Responses Are Sent in JSON (JavaScript Object Notation) Format.**

## 1- API Routes:

This backend server has multiple routes (links), and they are:

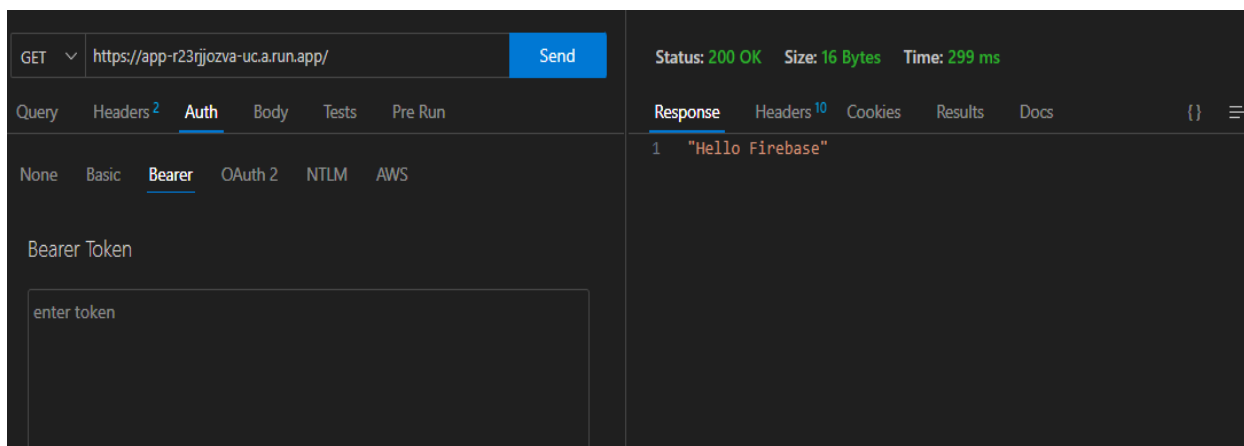
1- <https://app-r23rjjozva-uc.a.run.app/>

This link (route) is the default endpoint of the API and is just there to test if the API server is working or not.

Valid HTTP Request Method: GET

Valid Response: a message saying “Hello Firebase”

Valid Response Status Code: 200



2- <https://app-r23rjjozva-uc.a.run.app/api/categories>

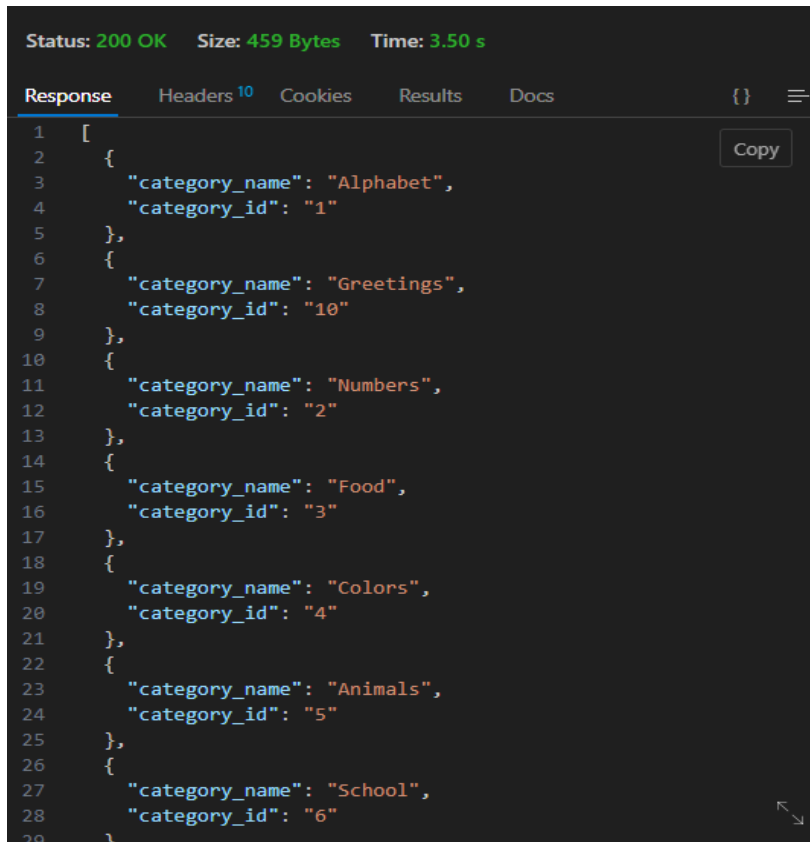
This route will get all the categories currently available in the database

Valid HTTP Request Method: GET

Valid Response: an array of objects, each object will have the category\_id and category\_name in it.

Valid Response Status Code: 200

Response Status Code When There is an Error: 400



```
Status: 200 OK Size: 459 Bytes Time: 3.50 s
Response Headers 10 Cookies Results Docs {} ≡
1  [
2  {
3    "category_name": "Alphabet",
4    "category_id": "1"
5  },
6  {
7    "category_name": "Greetings",
8    "category_id": "10"
9  },
10 {
11  "category_name": "Numbers",
12  "category_id": "2"
13 },
14 {
15  "category_name": "Food",
16  "category_id": "3"
17 },
18 {
19  "category_name": "Colors",
20  "category_id": "4"
21 },
22 {
23  "category_name": "Animals",
24  "category_id": "5"
25 },
26 {
27  "category_name": "School",
28  "category_id": "6"
29 }
```

### 3- <https://app-r23rjjozva-uc.a.run.app/api/auth/signup>

This is the route where the username, email, and password of the user will be submitted through a POST request, after the user successfully signs up, a jwt token will be created.

Valid HTTP Request Method: POST

Valid Response: a message saying that the user has successfully signed up, and the authorization header being set in the response, this authorization header will contain a JWT token that allows the user to access protected routes that require authentication (like /api/words)

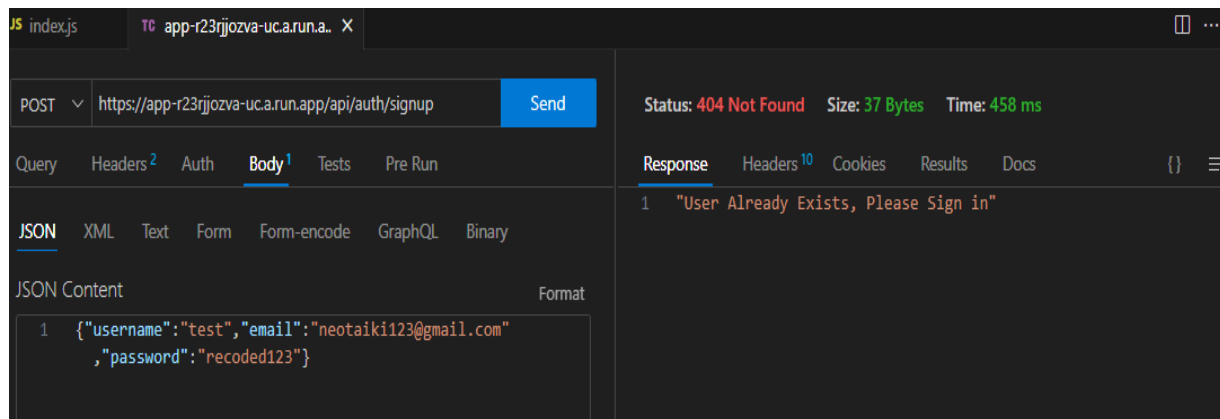
Valid Response Status Code: 200, successful signup

Response Status Code When There is an Error:

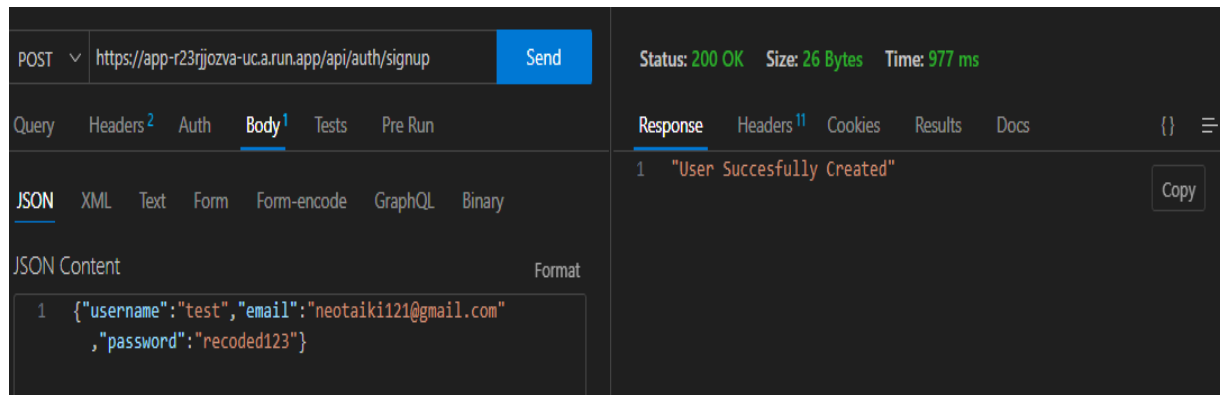
1- 404, meaning the user tried to sign up with an already existing email address.

2- 500, for any other kind of error, for example if you do not provide an email, password, or username, it will cause an error

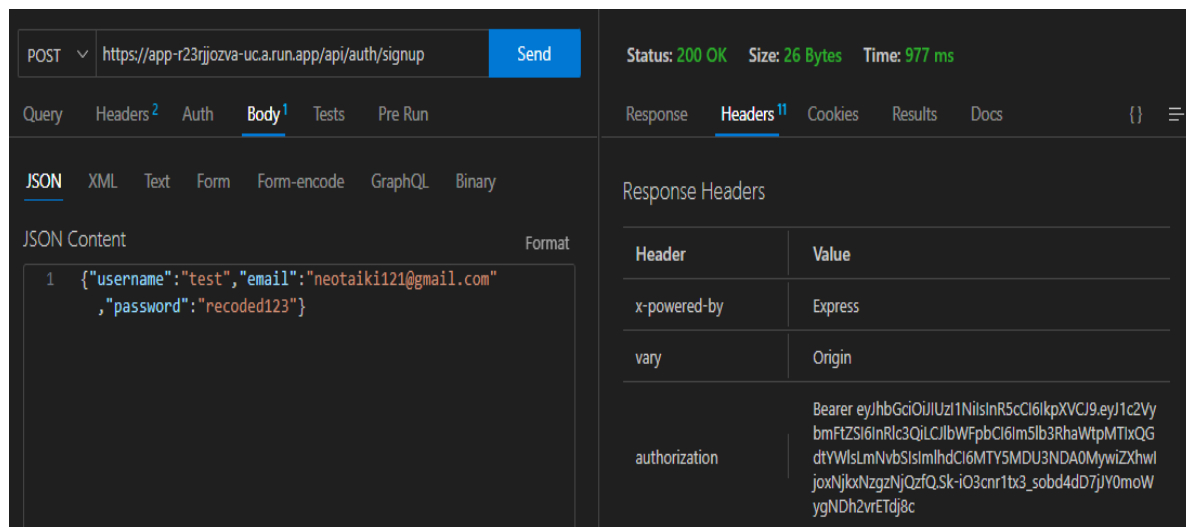
For example, the server responds with a message saying that user should sign in because the email account already exists in the database:



Now for an example of a successful signup:



And the authorization header is set, it will contain the jwt token:



#### 4- <https://app-r23rjjozva-uc.a.run.app/api/auth/login>

This is the endpoint where the user's email and password will be submitted through an HTTP POST request, and it will be checked if they exist in the database.

Valid HTTP Request Method: POST

Valid Response: a message saying that the user has successfully logged in, and like when the user signs up, the authorization header is also being set in the response, this authorization header will contain a JWT token that allows the user to access protected routes that require authentication (like /api/words)

Valid Response Status Code: 200, successful login

Response Status Code When There is an Error:

1- 404, meaning the unique combination of the submitted email and password doesn't exist in the database.

2- 500, for any other kind of error, for example if you do not provide an email or password, it will cause an error

Example of an unsuccessful login:

The screenshot displays a REST client interface with a dark theme. The top bar shows a POST request to the URL `https://app-r23rjjozva-uc.a.run.app/api/auth/login` with a 'Send' button. Below the URL bar, tabs for 'Query', 'Headers', 'Auth', 'Body', 'Tests', and 'Pre Run' are visible, with 'Body' selected. The 'Body' tab shows a JSON request: `{ "email": "neotaiki120@gmail.com", "password": "recoded123" }`. On the right, the response status is `404 Not Found` (red text), with a size of `16 Bytes` and a time of `445 ms`. Below the status bar, tabs for 'Response', 'Headers', 'Cookies', 'Results', and 'Docs' are shown, with 'Response' selected. The response body is `1 "User Not Found"`. A 'Copy' button is located to the right of the response text. The bottom left section is labeled 'JSON Content' and 'Format'.

Example of successful login:

The screenshot shows a REST client interface with a POST request to `https://app-r23rjjozva-uc.a.run.app/api/auth/login`. The request body is a JSON object: `{"email": "neotaiki121@gmail.com", "password": "recoded123"}`. The response status is `200 OK` with a size of `27 Bytes` and a time of `512 ms`. The response body is `"You Logged In Successfully"`.

Like the signup process, after the successful login, the authorization header will be set, and it will have the token in it:

The screenshot shows the same REST client interface, but the response headers are displayed. The headers are:

Header	Value
x-powered-by	Express
vary	Origin
authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFTZSI6InRlc3QjLCJlbWVpbGl6Im5lb3RhaWtpMTIxQGdtYWVsLmNvbSIsImhhdCI6MTY5MDU3NDY0MCwiZXhwIjoxNjkxNzg0MjQwfQ.LWXo0cJrbS11TE4d42Vdk013eUCnckwnV12tL2olc0
content-type	application/json; charset=utf-8

5- <https://app-r23rjjozva-uc.a.run.app/api/words?categoryId=1>

This route will get all the words that belong to a certain categoryId.

**NOTE 1: This route is a protected route, meaning it needs a token to be accessed first you will get a token by either signing up or logging in first.**

**NOTE 2: If you want to get the words from another category, just change the categoryId in the link to the category\_id that you want to get, for example if you want to get the words in category number 2 (which is the Numbers category), then the link will be:**

6- <https://app-r23rjjozva-uc.a.run.app/api/words?categoryId=2>

Valid Request Method: GET

Valid Response: An array of objects, each object will contain the category\_id, the image\_path, audio\_path, and the word itself

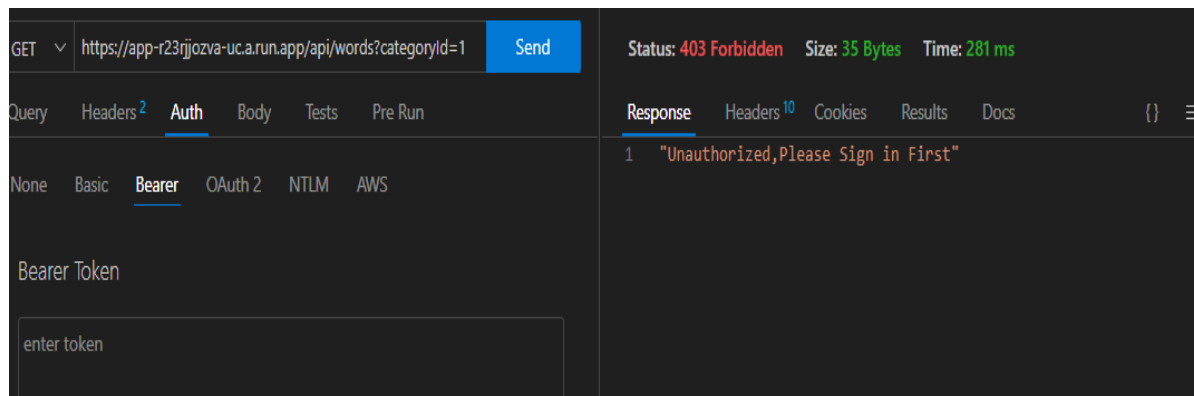
Valid Response Status Code: 200, success

Response Status Code When There is an Error:

1-403, Unauthorized access, user must sign up or login first

2- 400, for any other kind of error

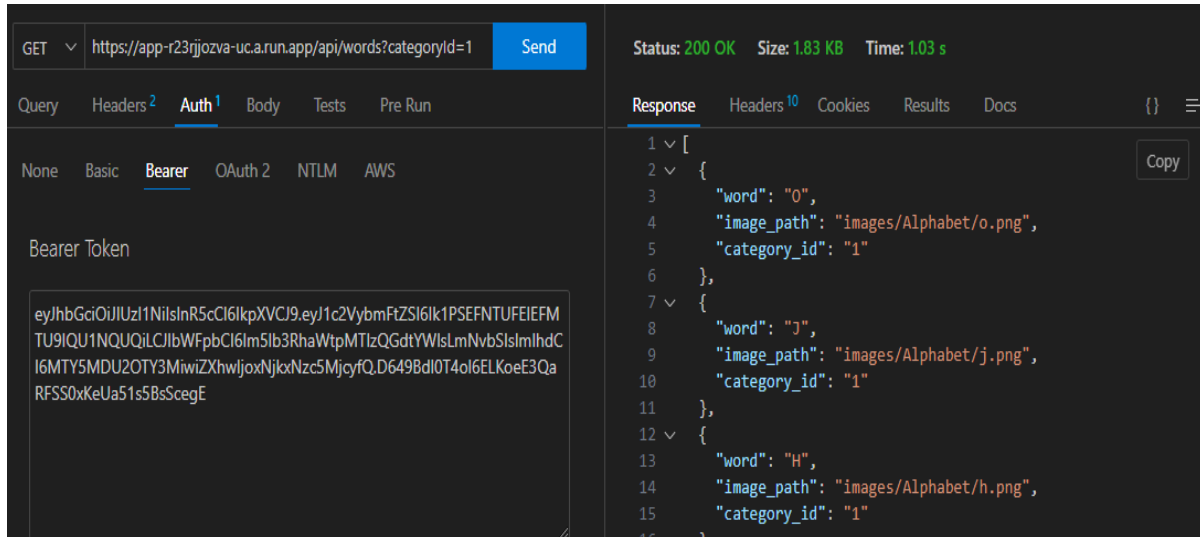
Example of invalid request:





The server is displaying this error message because no authorization header that contains the token was sent with the request to this route.

Example of successful request:



This is successful because an authorization header that had the token was sent with the request to this route, and the server responded with an array of objects.

7- <https://app-r23rjjozva-uc.a.run.app/api/me>

This route is just to get the username and email of the user who just signed in/signed up, to access this route, you need to have a token first. This token must be sent in the authorization header of the request.

Valid HTTP Request Method: GET

Valid Response: an object containing the username and password

Valid Response Status Code: 200, success

Response Status Code When There is an Error:

1- 404, a JWT token was not provided in the Authorization header of the request.

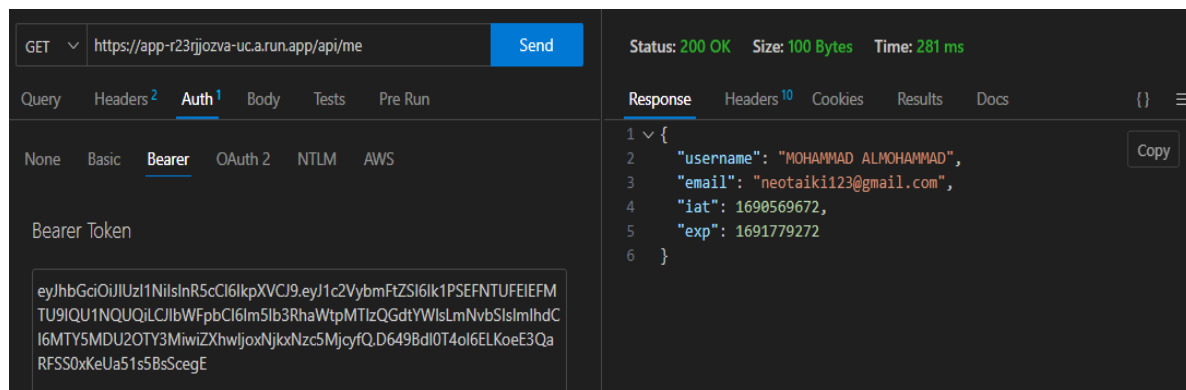
2- 500, for any other kind of error, for example if you do not provide an email or password, it will cause an error

Example of an invalid request:

The screenshot shows a REST client interface with the following details:

- Request Method:** GET
- URL:** <https://app-r23rjjozva-uc.a.run.app/api/me>
- Auth:** Bearer (selected)
- Response Status:** 404 Not Found
- Response Size:** 56 Bytes
- Response Time:** 282 ms
- Response Body:** 1 "You need to have a JWT token first to access this page"

Example of successful request:



## End of Routes Section

---

## 2- Some Additional Information And Notes:

**1-** you can get the images/audios files of the words by first getting the image/audio path from /api/words, and then getting the file from this link:

<https://firebasestorage.googleapis.com/v0/b/mam-english.appspot.com/o/IMAGE%2FFILE%2FPATH.png?alt=media>

After you get the image file path of the word from the route /api/words/, you can just replace the “IMAGE%2FFILE%2FPATH.png” with the image path you just got. So, for example if I want the image for the letter “a”:

Image\_path is: “/images/Alphabet/a.png”

So, the link to the image will be:

<https://firebasestorage.googleapis.com/v0/b/mam-english.appspot.com/o/images%2FAlphabet%2Fa.png?alt=media>

The same thing for audio file happens:

audio\_path: “/audio/Alphabet/a.mp3”

So, the link to the image will be:

<https://firebasestorage.googleapis.com/v0/b/mam-english.appspot.com/o/audio%2FAlphabet%2Fa.mp3?alt=media>

<https://firebasestorage.googleapis.com/v0/b/mam-english.appspot.com/o/>, this part of the link will always be the same.

2- after you signup/login, you should get the token from the authorization header, and save it somewhere, this can be “localStorage” for React and “SharedPreferences” in Flutter, if you don’t save the token somewhere you will have to go to the signup/login route and get a token every time to access the protected routes, if you save the token somewhere, you can just retrieve it without having to do login/signup every time.

-----

**END OF API DOCUMENTATION**