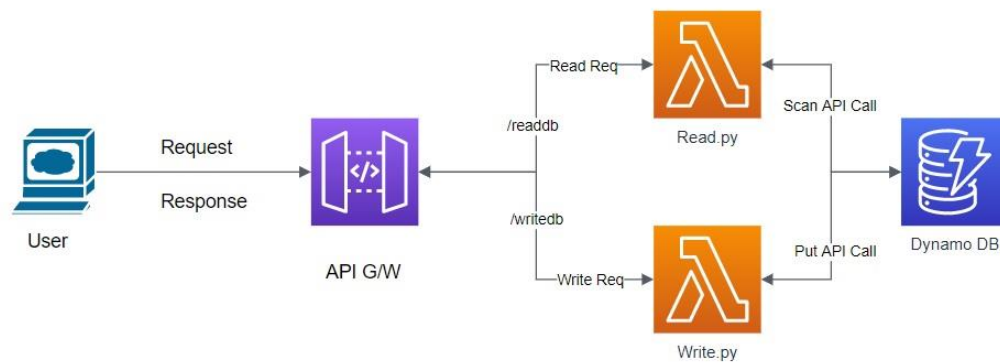


Announcement-Microservice

Author: Naveen Suthar

Github: <https://github.com/naveensuthar07/Announcement-MicroService/tree/main/apiLambdaDdbSAM>

Architecture:



Announcement Microservice

Pre-requisite:

- AWS SAM
- Docker
- Python3.8
- Git
- Postman

Process:

- This project creates a serverless microservice for Announcement Read and Write API's
- To start this setup the local environment for DynamoDB table using docker at path\apiLambdaDdbSAM using command "Docker-compose up"
- Build and Deploy SAM template(This will run AWS serverless components locally)
 - sam build
 - sam local start-api

Testing:

- Write

Request:

send the below details in post request at path "http://localhost:3000/writedb"

```
{"title": "Title Details",  
  "description": "Title Description",
```

"date":"date of the Announcement"}

Response: It will return response code 200 with appropriate response message

b. Read:

i. Full table scan:

Request : send the post request at path "http://localhost:3000/readdb"

Response : It will return response code 200 with all Announcement details

ii. Pagination :

Request : send the post request at path
"http://localhost:3000/readdb?limit=5&page=1"

Limit : It is used to limit the number of response

Page: To get the requested page

Error handling

Write:

In case of error API will return error code 400 with all error details.

Read :

In case of error API will return error code 400 with all error details.If given page doesn't Exist it will return the error code 404.