

Dating Apps API

Summary

Explore the heart of our Dating App with this Golang-based REST API repository

Github URL

Please check my code on : <https://github.com/panjitamzil/dating-apps>

Implementation

REST API

- a. Register : Allow users to register into our platform
- b. Login : Allow users to login into our apps
- c. Explore : Allow users to explore another people who play our apps
- d. Swipe : Feature that allow users to do swipe like or pass, which like when he interest with others profile and pass if he doesn't interest with others profile
- e. Purchase : Feature that allow users to increase their level into **PREMIUM** with unlimited Swipe feature

Tech Stacks

To implement Dating Apps API, I using several tech stacks :

- a. Golang as main programming language
- b. Gin as Framework to help me create routing
- c. Redis to caching data
- d. PostgreSQL as Database SQL
- e. JWT Token as Authorization tools to secure the API's
- f. Domain Driven Design (DDD) code structure

ERD Diagram

Users	
id	Type (PK)
email	VARCHAR(100)
password	VARCHAR(255)
fullname	VARCHAR(100)
dob	DATE
occupation	VARCHAR(100)
subscription	VARCHAR(20)
created_at	timestamp
updated_at	timestamp
deleted_at	timestamp

Test Cases and Documentation API

To see Documentation API, please download the collection on this link :

https://api.postman.com/collections/5627877-5e9abd20-6085-4b85-84b1-02911d427fbf?access_key=PMAT-01HKEFHS5Y8MRS9SM8YQ95WQMG

The screenshot shows the Postman interface for a POST request to `localhost:8080/register`. The **Tests** tab is active, displaying the following test scripts:

```
1
2 pm.test("Response status code is 200", function () {
3   pm.expect(pm.response.code).to.equal(200);
4 });
5
6 pm.test("Message field is present in the response", function () {
7   const responseData = pm.response.json();
8
9   pm.expect(responseData).to.be.an('object');
10  pm.expect(responseData.message).to.exist;
11 });
12
```

Below the scripts, the **Test Results (2/2)** section shows two passed tests:

- PASS** Response status code is 200
- PASS** Message field is present in the response

The status bar at the bottom indicates **Status: 200 OK** and **Time: 71 ms**.

The screenshot shows a Postman collection named **Dating Apps API** with the following test cases:

- POST** Register - OK
- POST** Register - FAILED Already Ex...
- POST** Login - OK
- POST** Login - FAILED Invalid Passw...
- GET** Explore - OK
- GET** Explore - FAILED Token Expir...
- POST** SWIPE - OK
- POST** SWIPE - FAILED Already Swipe
- POST** SWIPE - FAILED User Not Fo...
- POST** Purchase - OK

■ ■ ■