HOMEWORK 5

APPLICATION. VALIDATION. SECURITY

Tasks

1. Add `/auth` route to your application.
2. Implement standard login and password authentication:
    a. POST `/auth` login and password should check user login and password (may be hardcoded somewhere in application for now).
    b. If user exists, generate **JWT token** (you can use any package for generating it, e.g. `jsonwebtoken`) and send the following response:

```
{
   "code": 200,
   "message": "OK",
   "data": {
       "user": {
           "email": "...",
           "username": "..."
       }
   },
   "token": "..."
}
```

    c. If user does not exist or credentials do not much - send error response with the proper error code:

```
{
   "code": 404,
   "message": "Not Found",
   "data": { ... additional error response data if needed ... }
}
```

3. Write a **middleware** to verify **JWT token** for all `/products` and `/users` routes from the Homework 4 (`/auth` route should be excluded from verification).
4. Add `passport` package into your application.
5. Implement local authentication strategy using `passport` to allow login with user's **credentials** (hardcoded credentials from *2b* may be used).

6. Implement following authentication strategies using `passport`:

    a. **Facebook** strategy.
    b. **Twitter** strategy.
    c. **Google OAuth** strategy.

Evaluation Criteria

1. All required routes is added and passport is installed (*tasks 1, 4*).
2. Standard login and password authentication is implemented (*task 2*).
3. JWT token is verified for all required routes (*task 3*).
4. Local authentication strategy is implemented via passport (*task 5*).
5. All authentication strategies is implemented via passport (*task 6*).