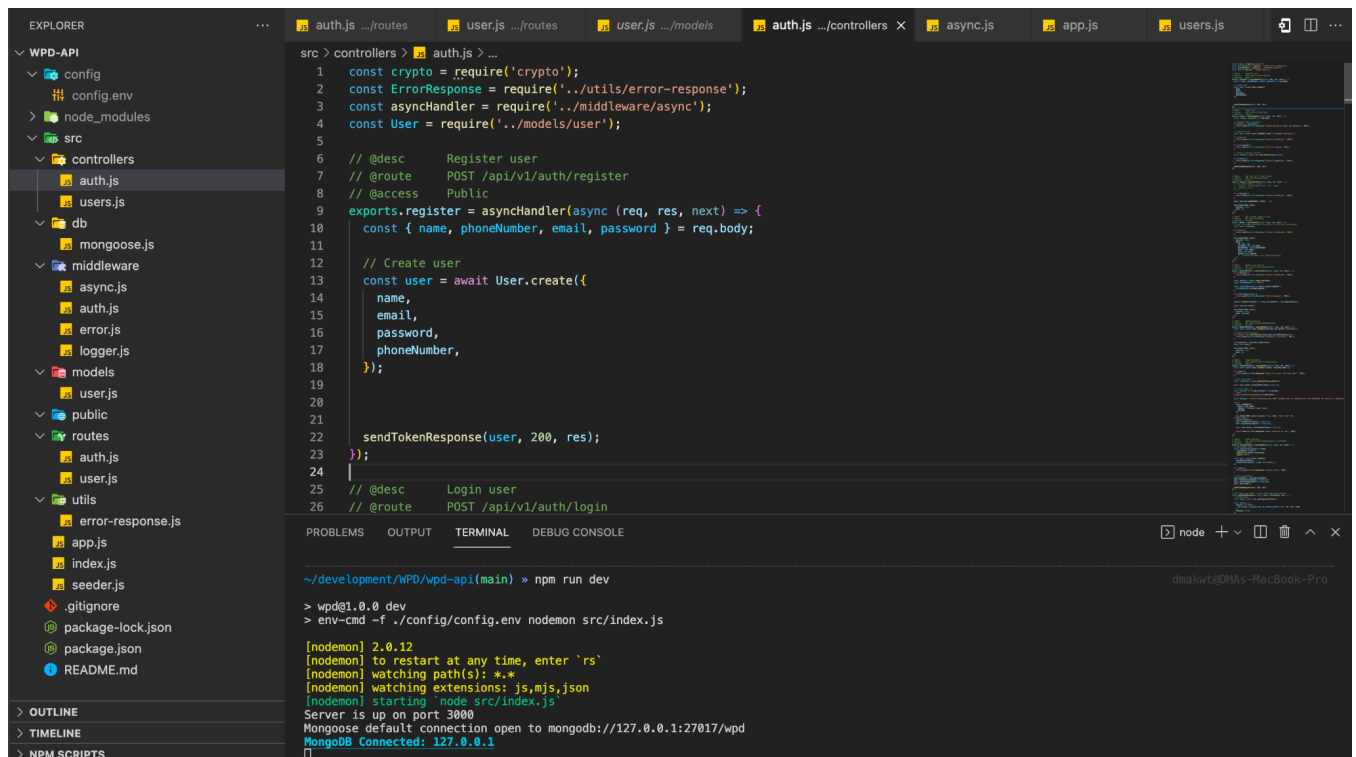


Week of September. 12th

This week, I finished creating the files and folders for the backend side. Also, I was developing the backend side, so I finished the authentication part, such as the sign-in, sign-up, and reset password.

In addition, I was testing the authentication part by using the Postman program to make sure that the backend side is secure and works correctly.

Also, I was using the VS Code IDE to develop the backend side.



```
src > controllers > auth.js > ...
1  const crypto = require('crypto');
2  const ErrorResponse = require('../utils/error-response');
3  const asyncHandler = require('../middleware/async');
4  const User = require('../models/user');
5
6  // @desc    Register user
7  // @route    POST /api/v1/auth/register
8  // @access   Public
9  exports.register = asyncHandler(async (req, res, next) => {
10     const { name, phoneNumber, email, password } = req.body;
11
12     // Create user
13     const user = await User.create({
14       name,
15       email,
16       password,
17       phoneNumber,
18     });
19
20     sendTokenResponse(user, 200, res);
21   });
22
23   // @desc    Login user
24   // @route    POST /api/v1/auth/login
```

```
~/development/WPD/wpd-api(main) > npm run dev
> wpd@1.0.0 dev
> env-cmd -f ./config/config.env nodemon src/index.js

[nodemon] 2.0.12
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting node src/index.js
Server is up on port 3000
Mongoose default connection open to mongodb://127.0.0.1:27017/wpd
MongoDB Connected: 127.0.0.1
```

Testing the endpoints by using Postman program

The screenshot displays the Postman application interface. On the left, the 'My Workspace' sidebar shows a collection of API endpoints under the 'WPD' environment. The main panel shows a 'POST' request to the endpoint '({URL})/api/v1/auth/register ...'. The request body is a JSON object with the following fields:

```
1 {
2   "name": "Test",
3   "email": "test@gmail.com",
4   "password": "123456"
5 }
```

The response is a JSON object with the following fields:

```
1 {
2   "success": true,
3   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI1MTYwOTY4NzNjZjY0IjE2MzE0ODV9.5cTZV0Irr1MSQP1ESkV8Iug_gaNUfqTPU3f44tv0ZSc",
4   "data": {
5     "id": "613eb1596f56873ef7668368",
6     "name": "Test",
7     "email": "test@gmail.com",
8     "role": "police"
9   }
10 }
```

The status of the response is '200 OK' with a time of '276 ms' and a size of '750 B'. The response is displayed in the 'Body' tab, which is currently set to 'Pretty' view.