



Institute of Technology of Cambodia

Department of Information and
communication Engineering



Assignment TP08

TP08: VueJS, NodeJS (Authentication frontend and APIs)

Group: I4 GIC(C)

Lecturers

HOK TIN

Student's Name:

ROTHA DAPRAVITH

Subjects

TP Internet-Programming II (IP)

ID

e20190915

Academic Year: 2022 ~ 2023

Table of Contents

GitHub's Repository: https://github.com/Dapravith/TP_Internet_Programming/tree/TP08	1
TP08.1: Auth screens in VueJS	1
TP08.2: NodeJS with ExpressJS	2
TP08.3: UI and APIs Integration	4


GitHub's Repository: https://github.com/Dapravith/TP_Internet_Programming/tree/TP08

TP08.1: Auth screens in VueJS

Create a simple login and register form in VueJS

By using VueJS router, we can access by

- Result of Exercise1:
- Login: <http://localhost:3000/>



Username

Password

[Forget Password?](#)

Login

- Register: <http://localhost:3000/register>



Sign Up

Please fill in this form to create an account.

Email

Username

First name

Last name

Password

Login

- Home: <http://localhost:3000/home>



Welcome

Authentication App~

[Login](#) | [Register](#)

You're in Home page

well done 🎉

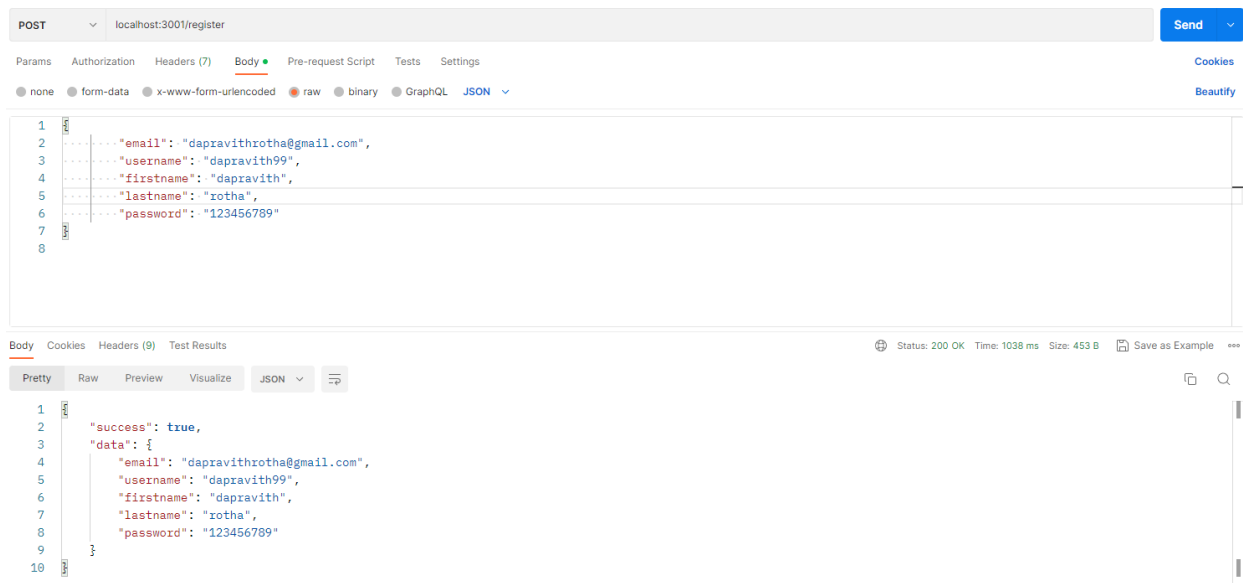
[back to login](#)

TP08.2: NodeJS with ExpressJS

Create a nodejs APIs for authentication with ExpressJS. Use Json file as a database to store user's information.

Result of Ex02:

- register success



- register existed account

The screenshot shows a REST client interface with a POST request to `localhost:3001/register`. The request body is a JSON object with the following fields: `email`, `username`, `firstname`, `lastname`, and `password`. The response status is 200 OK, and the response body is a JSON object indicating that the user already exists.

```
POST localhost:3001/register
```

Params Authorization Headers (7) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "email": "dapravithrotha@gmail.com",
3   "username": "dapravith99",
4   "firstname": "dapravith",
5   "lastname": "rotha",
6   "password": "123456789"
7 }
8
```

Body Cookies Headers (9) Test Results

Status: 200 OK Time: 13 ms Size: 351 B Save as Example

Pretty Raw Preview Visualize JSON

```
1 {
2   "success": false,
3   "err": "User is already existed~"
4 }
```

- login success

The screenshot shows a REST client interface with a POST request to `localhost:3001/login`. The request body is a JSON object with the following fields: `email` and `password`. The response status is 200 OK, and the response body is a JSON object indicating a successful login with user details.

```
POST localhost:3001/login
```

Params Authorization Headers (7) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "email": "dapravithrotha@gmail.com",
3   "password": "123456789"
4 }
5
```

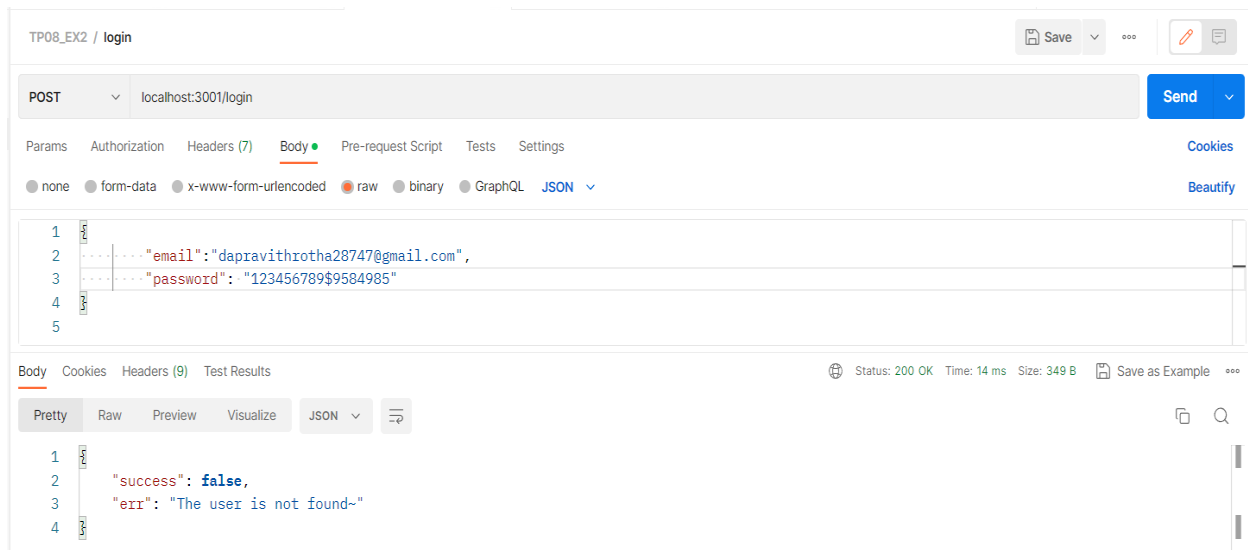
Body Cookies Headers (9) Test Results

Status: 200 OK Time: 323 ms Size: 453 B Save as Example

Pretty Raw Preview Visualize JSON

```
1 {
2   "success": true,
3   "data": {
4     "email": "dapravithrotha@gmail.com",
5     "username": "dapravith99",
6     "firstname": "dapravith",
7     "lastname": "rotha",
8     "password": "123456789"
9   }
10 }
```

- Login wrong email and password



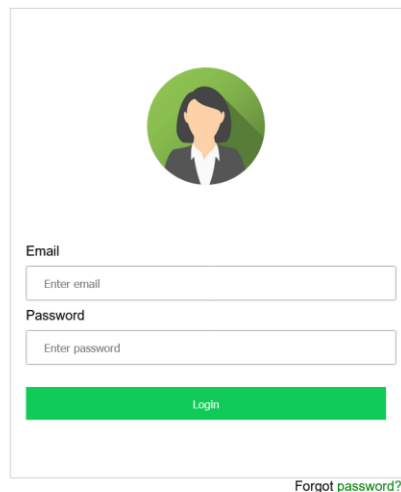
The screenshot shows a REST client interface for a POST request to `localhost:3001/login`. The request body is a JSON object: `{ "email": "dapravithrotha28747@gmail.com", "password": "123456789$9584985" }`. The response status is 200 OK, and the response body is a JSON object: `{ "success": false, "err": "The user is not found~" }`.

TP08.3: UI and APIs Integration

Integrate Login/Register app with APIs.

Result of Ex3:

- Login: <http://localhost:3001/login>



The image shows a login form with a circular profile picture placeholder. Below the profile picture are two input fields: 'Email' and 'Password'. Below the input fields is a green 'Login' button. At the bottom right of the form is a link that says 'Forgot password?'.

- Register: <http://localhost:3001/register>



Sign up

Please in this form to create an account.

Email

Enter email

Username

Username

First name

First name

Last name

Last name

Password

Create your password

By creating an account you agree to our [Term](#) & [Privacy](#)

Sign up

- Home: <http://localhost:3001/home>

[ABOUT US](#)[ACADEMIC](#)[WHY GIC](#)[LIFESTYLE](#)[CONTACT](#)[Logout](#)

Deep Learning Through Deep connection.

Taking a deep dive

The hero's journey start here. Become a learning legend

[GET START](#)

Elevate your experience. Empower the next generation

[Join our team](#)