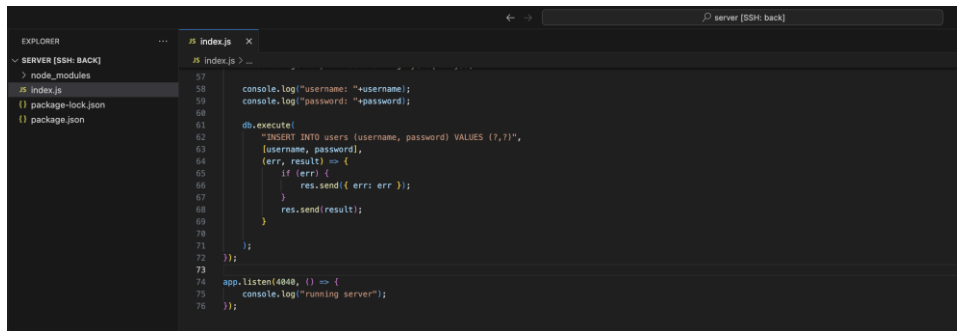


Step 9

ทำการเข้าไปที่ ubuntu backend และทำการกำหนด port



```
57
58 console.log("username: " + username);
59 console.log("password: " + password);
60
61 @.execute(
62   "INSERT INTO users (username, password) VALUES (?,?)",
63   [username, password],
64   (err, result) => {
65     if (err) {
66       res.send({ err: err });
67     }
68     res.send(result);
69   });
70
71 });
72
73 app.listen(4040, () => {
74   console.log("running server");
75 });
76
```

เปิด port 4040

```
lertpisit@backend001:/var/www/lab4$ sudo ufw allow 4040
```

แก้ไข nginx.conf

```
lertpisit@backend001:/etc/nginx$ sudo nano nginx.conf
```

ก่อน

```
include /etc/nginx/conf.d/*.conf;
include /etc/nginx/sites-enabled/*;
```

หลัง

```
include /etc/nginx/conf.d/*.conf;
include /etc/nginx/sites-available/*;
```

สร้างไฟล์ apimterm ที่ sites-available

```
lertpisit@backend001:/etc/nginx/sites-available$ sudo nano apimidterm
```

```
GNU nano 6.2 apimidterm *
server {
    listen 80;
    listen [::]:80;

    server_name api.midterm-engse202-001.com;

    location / {
        proxy_pass http://127.0.0.1:4040;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

ไปที่ เครื่อง mac แก้ไข hosts

```
admin@adminins-iMac ~ % sudo nano /etc/hosts
```

```

UW PICO 5.09 File: /etc/hosts
##
# Host Database
#
# localhost is used to configure the loopback interface
# when the system is booting. Do not change this entry.
##
127.0.0.1 localhost engse203.lab1
255.255.255.255 broadcasthost
::1 localhost
10.211.55.3 expressjs.se-rmutl.net myapp.se-rmutl.net react.se-rmutl.net weblogin.se-rmutl.net
10.211.55.3 my-react-app.se-rmutl.net ubuntu-next-app.se-rmutl.net nodelogin.se-rmutl.net
10.211.55.5 web1.midterm-engse202-001.com web2.midterm-engse202-001.com react1.midterm-engse202-001.com react2.midterm-engse202-001.com
10.211.55.4 api.midterm-engse202-001.com

```

รัน nodemon

```

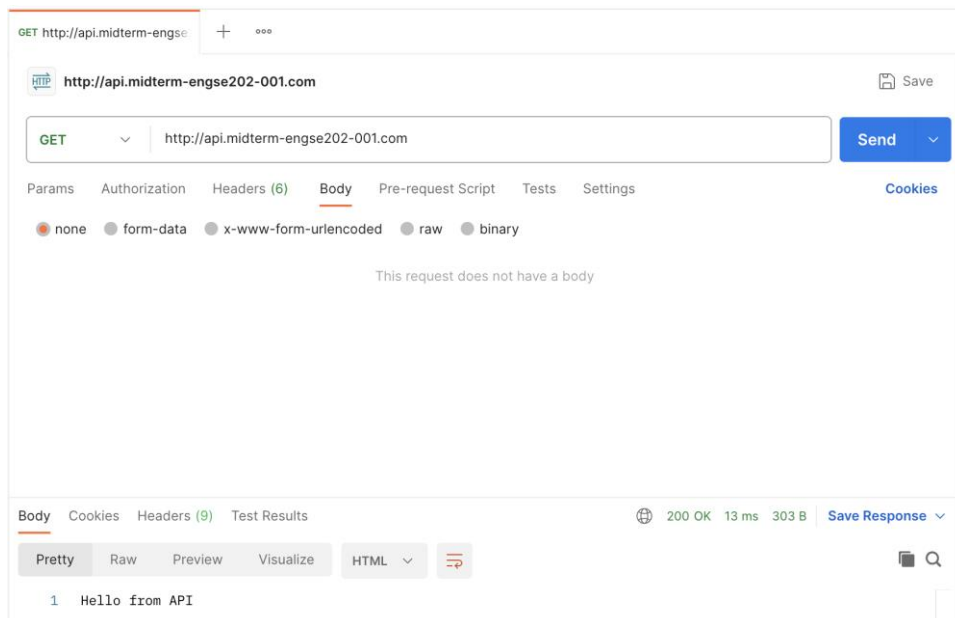
lertpsit@backend001:/var/www/lab4/server$ nodemon index.js

```

Step 10

Test postman

get



post

The screenshot shows a REST client interface. The top bar displays the URL `http://api.midterm-engse202-001.com/register` and a `POST` method. The body tab is selected, showing a JSON payload: `{ "username": "pain", "password": "1234" }`. Below the request, the response is shown with a status of `200 OK`, a response time of `24 ms`, and a size of `405 B`. The response body is a JSON object: `{ "fieldCount": 0, "affectedRows": 1, "insertId": 0, "info": "", "serverStatus": 2, "warningStatus": 0, "changedRows": 0 }`.

Step 11

สร้างไฟล์ login

```
lertpisit@webserver-66543210001:/var/www/nginx$ mkdir login
```

สร้าง react

```
lertpisit@webserver-66543210001:/var/www/nginx/login$ npx create-react-app .
Need to install the following packages:
create-react-app@5.0.1
Ok to proceed? (y) y
```

แก้ไข app.css

```
# App.css ×
login > src > # App.css > input
1  .App {
2    text-align: center;
3  }
4
5  .registration {
6    display: flex;
7    flex-direction: column;
8    justify-content: center;
9    align-items: center;
10 }
11
12 .login {
13   display: flex;
14   flex-direction: column;
15   justify-content: center;
16   align-items: center;
17 }
18 input {
19   width: 250px;
20   height: 40px;
21 }
22
```

แก้ไข index.js

```
# App.css  JS index.js  X

login > src > JS index.js
1  import React from 'react';
2  import ReactDOM from 'react-dom';
3  import './App.css';
4  import App from './App';
5
6  ReactDOM.render(
7    <React.StrictMode>
8    |   <App />
9    </React.StrictMode>,
10   document.getElementById('root')
11 );
```

แก้ไข app.js

```
# App.css  JS index.js  JS App.js  X

login > src > JS App.js > App > register
1  import React, { useEffect, useState } from "react";
2  import Axios from 'axios';
3  import './App.css';
4
5  function App() {
6
7    const [usernameReg, setUsernameReg] = useState("");
8    const [passwordReg, setPasswordReg] = useState("");
9
10   const [username, setUsername] = useState("");
11   const [password, setPassword] = useState("");
12
13   const [loginStatus, setLoginStatus] = useState("");
14
15   const register = () => {
16     Axios.post("http://api.midterm-engse202-001.com/register", {
17       username: usernameReg,
18       password: passwordReg,
19     }).then((response) => {
20       console.log(response);
21     });
22   };
23
24   const login = () => {
25     Axios.post("http://api.midterm-engse202-001.com/login", {
26       username: username,
27       password: password,
28     }).then((response) => {
29       if (!response.data.message) {
30         setLoginStatus(response.data.message);
31       } else {
32         setLoginStatus(response.data[0].message);
33       }
34     });
35   };
36 }
```

```

36
37   return (
38     <div className="App">
39       <div className="registration">
40         <h1>Registration</h1>
41         <label>Username</label>
42         <input
43           type="text"
44           onChange={(e) => {
45             setUsernameReg(e.target.value);
46           }}
47         /><br/>
48         <label>password</label>
49         <input
50           type="text"
51           onChange={(e) => {
52             setPasswordReg(e.target.value);
53           }}
54         /> <br />
55         <button onClick={register} > Register</button>
56       </div>
57       <div className="login">
58         <h1>Login</h1>
59         <input
60           type="text"
61           placeholder="Username_"
62           onChange = { (e) => {
63             setUsername (e.target.value);
64           }}
65         /> <br/>
66         <input
67           type="password"
68           placeholder="Password_"
69           onChange = { (e) => {
70             setPassword (e.target.value);
71           }}
72         />
73         <button onClick={login}>Login</button>
74       </div>
75       <h1> {loginStatus}</h1>
76     </div>
77   );
78 }
79
80
81   export default App;

```

ลง axios

```

lertpisit@webserver-66543210001:/var/www/nginx/login$ npm i --save axios

added 3 packages, and audited 1556 packages in 2s

254 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

```

กำหนด port 3012

```

    "scripts": {
      "start": "PORT=3012 react-scripts start",
      "build": "react-scripts build",
      "test": "react-scripts test",
      "eject": "react-scripts eject"
    },

```

เปิด port 3012

```

lertpisit@webserver-66543210001:/var/www/nginx/login$ sudo ufw allow 3012

```

Copy react1 to login

```

lertpisit@webserver-66543210001:/etc/nginx/sites-available$ sudo cp react1 login

```

แก้ไข login

```

lertpisit@webserver-66543210001:/etc/nginx/sites-available$ sudo nano login

```

```

GNU nano 6.2 login
server {
    listen 81;
    listen [::]:81;

    server_name login.midterm-engse202-001.com ;

    location / {
        proxy_pass http://127.0.0.1:3012;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}

```

ไปที่เครื่อง mac แก้ไข hosts

```

admin@admins-iMac ~ % sudo nano /etc/hosts

```

```

10.211.55.3 web1.midterm-engse202-001.com web2.midterm-engse202-001.com react1.midterm-engse202-001.com react2.midterm-engse202-001.com login.midterm-engse202-001.com

```

รัน login

Compiled successfully!

You can now view **login** in the browser.

Local: http://localhost:3012

On Your Network: http://10.211.55.5:3012

Note that the development build is not optimized.
To create a production build, use `npm run build`.

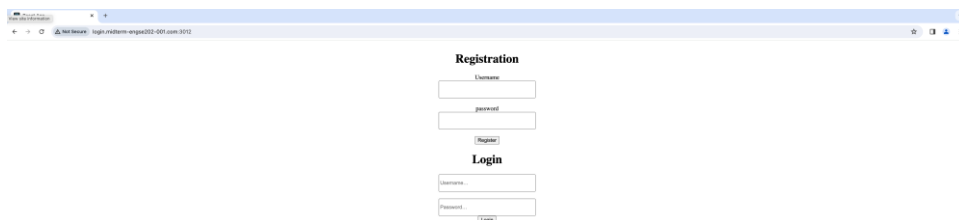
webpack compiled **successfully**



รัน server

```
lertpisit@backend001:/var/www/lab4/server$ nodemon index.js
[nodemon] 3.0.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: *.js
```

ผลลัพธ์

A screenshot of a web browser displaying a simple web application. The browser's address bar shows 'http://localhost:3012'. The page has a white background and a light blue header. The main content area contains two sections: 'Registration' and 'Login'. The 'Registration' section has two input fields for 'Username' and 'password', followed by a 'Register' button. The 'Login' section has two input fields for 'Username' and 'Password', followed by a 'Login' button. The text is in a clean, sans-serif font.

React App

login.middleware-express202-001.com:3012

Registration

Username

mid

Password

mid

Register

Login

Username

mid

Password

mid

Login

'mid' was inserted

React App

login.middleware-express202-001.com:3012

Registration

Username

mid

Password

mid

Register

Login

Username

mid

Password

mid

Login

'mid' was logged in