

supawit kaennak 66543210031-1

LAB4 back end server

```
devuser@backend031:~$ sudo apt update
```

ภาพที่ 1 คำสั่ง update ubuntu

```
devuser@backend031:~$ sudo apt list --upgradable
```

ภาพที่ 2 คำสั่งดู list upgrade

```
devuser@backend031:~$ sudo apt upgrade
```

ภาพที่ 3 คำสั่ง upgrade

```
devuser@backend031:~$ apt-cache search mysql-server
```

ภาพที่ 4 คำสั่งดูหา mysql server

```
devuser@backend031:~$ apt info -a mysql-server-8.0
```

ภาพที่ 5 คำสั่งหา ดู info ของ mysql

```
devuser@backend031:~$ sudo apt install mysql-server-8.0
```

```
Do you want to continue? [Y/n] y
```

ภาพที่ 6 คำสั่ง ติดตั้ง mysql 8.0



ภาพที่ 7 กด ok

```
devuser@backend031:~$ sudo apt update && sudo apt upgrade
```

ภาพที่ 8 update กับ upgrade อีกครั้ง

```
devuser@backend031:~$ sudo apt install nginx certbot python3-certbot-nginx
```

ภาพที่ 9 ติดตั้ง nginx



ภาพที่ 10 กด ok

supawit kaennak 66543210031-1

```
devuser@backend031:~$ sudo ufw allow 'Nginx Full'  
Rules updated  
Rules updated (v6)
```

ภาพที่ 11 เพิ่ม nginx เข้า firewall

```
devuser@backend031:~$ sudo apt-get update  
Hit:1 http://ports.ubuntu.com/ubuntu-ports jammy InRelease  
Hit:2 http://ports.ubuntu.com/ubuntu-ports jammy-updates InRelease  
Hit:3 http://ports.ubuntu.com/ubuntu-ports jammy-backports InRelease  
Hit:4 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease  
Reading package lists... Done
```

ภาพที่ 12 update อีกครั้ง

```
devuser@backend031:~$ sudo apt-get install -y ca-certificates curl gnupg  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).  
ca-certificates set to manually installed.  
curl is already the newest version (7.81.0-1ubuntu1.15).  
curl set to manually installed.  
gnupg is already the newest version (2.2.27-3ubuntu2.1).  
gnupg set to manually installed.  
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
```

ภาพที่ 13 ติดตั้ง certificate

```
devuser@backend031:~$ sudo mkdir -p /etc/apt/keyrings  
  
curl -fsSL https://deb.nodesource.com/gpgkey/nodesource-repo.gpg.key | sudo gpg --dearmor -o /etc/apt/keyrings/nodesource.gpg
```

ภาพที่ 14 สร้าง directory keyring

```
devuser@backend031:~$ NODE_MAJOR=20  
  
echo "deb [signed-by=/etc/apt/keyrings/nodesource.gpg] https://deb.nodesource.com/node_${NODE_MAJOR}.x nodistro main" | sudo tee /etc/apt/sources.list.d/nodesource.list  
deb [signed-by=/etc/apt/keyrings/nodesource.gpg] https://deb.nodesource.com/node_20.x nodistro main
```

ภาพที่ 15 กำหนด node ที่จะลงเป็น version 20

```
devuser@backend031:~$ sudo apt-get update
```

ภาพที่ 16 update อีกครั้ง

```
devuser@backend031:~$ sudo apt-get install nodejs -y
```

ภาพที่ 17 ติดตั้ง nodejs

```
devuser@backend031:~$ node -v  
v20.10.0  
devuser@backend031:~$ npm -v  
10.2.3
```

ภาพที่ 18 ตรวจสอบ version node และ npm

```
devuser@webserver031:~$ sudo mkdir -p /var/www/lab4
```

ภาพที่ 19 สร้าง directory lab4

```
devuser@webserver031:/var/www/lab4$ sudo mkdir -p nodelogin
```

ภาพที่ 20 สร้าง directory nodelogin ข้างใน directory lab4

```
devuser@webserver031:/var/www/lab4$ sudo chown -R $USER:$USER /var/www/lab4
```

ภาพที่ 21 คำสั่งอนุญาตให้ user ธรรมดา แก้ไขไฟล์ต่างๆใน directory lab4 ได้

```
devuser@webserver031:/var/www/lab4$ cd nodelogin
```

ภาพที่ 22 เข้า directory nodelogin

```
devuser@webserver031:/var/www/lab4/nodelogin$ npm init
```

ภาพที่ 23 คำสั่งสร้าง package nodejs โดยใช้ npm

```
{
  "name": "nodelogin",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}
```

ภาพที่ 24 (ต่อ) หลังจากใช้คำสั่ง npm init มันจะให้ใส่ข้อมูลต่างๆ ถ้าไม่ใส่กด enter ไปให้หมด

```
devuser@webserver031:/var/www/lab4/nodelogin$ npm install express --save
added 62 packages, and audited 63 packages in 2s

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

found 0 vulnerabilities
devuser@webserver031:/var/www/lab4/nodelogin$ npm install express-session --save
added 5 packages, and audited 68 packages in 893ms

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

found 0 vulnerabilities
devuser@webserver031:/var/www/lab4/nodelogin$ npm install mysql --save
added 12 packages, and audited 80 packages in 996ms
```

ภาพที่ 25 ติดตั้ง express , expression-session และ mysql

```
devuser@webserver031:~$ sudo mysql -u example_user -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 20
Server version: 8.0.35-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

ภาพที่ 26 เข้า mysql ด้วย user exmaple\_user

```
mysql> CREATE DATABASE IF NOT EXISTS 'nodelogin' DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
```

ภาพที่ 27 สร้าง database nodelogin

```
mysql> USE nodelogin;
Database changed
mysql> CREATE TABLE IF NOT EXISTS `accounts` (
  ->   `id` int(11) NOT NULL AUTO_INCREMENT,
  ->   `username` varchar(50) NOT NULL,
  ->   `password` varchar(255) NOT NULL,
  ->   `email` varchar(100) NOT NULL,
  ->   PRIMARY KEY (`id`)
  -> ) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;
Query OK, 0 rows affected, 2 warnings (0.02 sec)
```

ภาพที่ 28 สร้าง ใช้ database แล้ว สร้าง ตารางข้างในโดยมีข้อมูลตามรูปตัวอย่าง

```
mysql> SHOW DATABASES
-> ;
+-----+
| Database |
+-----+
| information_schema |
| nodelogin |
| performance_schema |
+-----+
3 rows in set (0.00 sec)
```

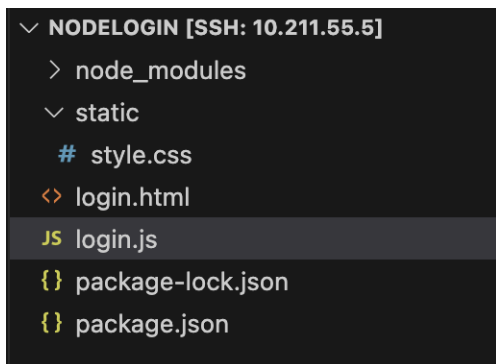
ภาพที่ 29 แสดง database nodelogin

```
mysql> INSERT INTO `accounts` (`id`, `username`, `password`, `email`) VALUES (1, 'test', 'test', 'test@test.com');
Query OK, 1 row affected (0.01 sec)
```

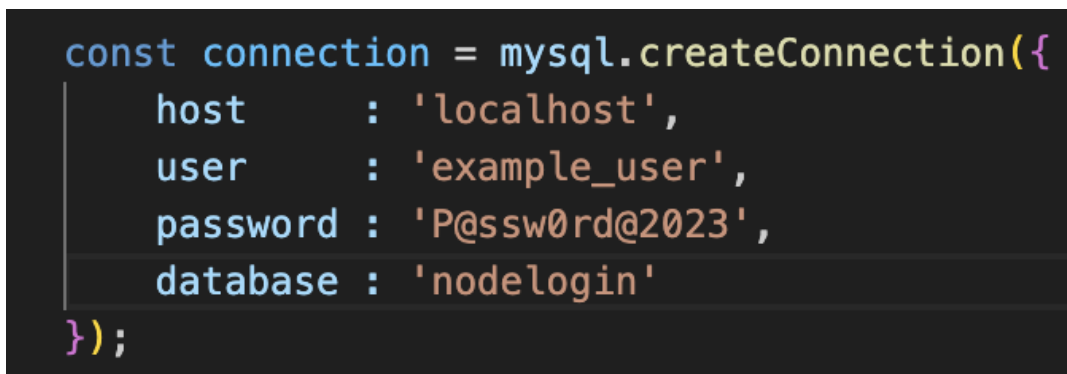
ภาพที่ 30 แสดง เพิ่มข้อมูลต่างๆ ลงในตารางตามภาพ

```
mysql> ALTER USER 'example_user'@ '%' IDENTIFIED WITH mysql_native_password BY 'P@ssw0rd@2023';
Query OK, 0 rows affected (0.01 sec)
```

ภาพที่ 31 คำสั่งแก้ไขรหัสผ่านผู้ใช้ example\_user



ภาพที่ 32 ส่วนของ vscode สร้างไฟล์ต่างๆตามภาพนี้



ภาพที่ 33 (ต่อ) ส่วนของ vscode หน้า login.js (แบบย่อขนาดภาพ)



ภาพที่ 34 (ต่อ) ส่วนของ vscode หน้า login.js port (ปัจจุบันแก้ไขเป็น 3004)

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width,minimum-scale=1">
6     <title>Login</title>
7     <!-- the form awesome library is used to add icons to our form -->
8     <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.7.1/css/all.css">
9     <!-- include the stylesheet file -->
10    <link href="/style.css" rel="stylesheet" type="text/css">
11  </head>
12  <body>
13    <div class="login">
14      <h1>Login</h1>
15      <form action="/auth" method="post">
16        <label for="username">
17          <!-- font awesome icon -->
18          <i class="fas fa-user"></i>
19        </label>
20        <input type="text" name="username" placeholder="Username" id="username" required>
21        <label for="password">
22          <i class="fas fa-lock"></i>
23        </label>
24        <input type="password" name="password" placeholder="Password" id="password" required>
25        <input type="submit" value="Login">
26      </form>
27    </div>
28  </body>
29 </html>

```

ภาพที่ 35 (ต่อ) ส่วนของ vscode หน้า login.html

```

static > # style.css > % login form input[type="submit"]:hover
1 {
2   box-sizing: border-box;
3   font-family: -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, Oxygen, Ubuntu, Cantarell, "Fira Sans", "Droid Sans", "Helvetica Neue", Arial, sans-serif;
4   font-size: 16px;
5 }
6 body {
7   background-color: #435165;
8 }
9 .login {
10   width: 400px;
11   background-color: #ffffff;
12   box-shadow: 0 0 9px 0 Crgb(0, 0, 0, 0.3);
13   margin: 100px auto;
14 }
15 .login h1 {
16   text-align: center;
17   color: #435165;
18   font-size: 24px;
19   padding: 20px 0 20px 0;
20   border-bottom: 1px solid #dee0e4;
21 }
22 .login form {
23   display: flex;
24   flex-wrap: wrap;
25   justify-content: center;
26   padding-top: 20px;
27 }
28 .login form label {
29   display: flex;
30   justify-content: center;
31   align-items: center;
32   width: 50px;
33   height: 50px;
34   background-color: #3274d6;
35   color: #ffffff;
36 }
37 .login form input[type="password"], .login form input[type="text"] {
38   width: 310px;
39   height: 50px;
40   border: 1px solid #dee0e4;
41   margin-bottom: 20px;
42   padding: 0 15px;
43 }
44 .login form input[type="submit"] {
45   width: 100%;
46   padding: 15px;
47   margin-top: 20px;
48   background-color: #3274d6;
49   border: 0;
50   cursor: pointer;
51   font-weight: bold;
52   color: #ffffff;
53   transition: background-color 0.2s;
54 }
55 .login form input[type="submit"]:hover {
56   background-color: #2866c7;
57   transition: background-color 0.2s;
58 }

```

ภาพที่ 36 (ต่อ) ส่วนของ vscode หน้า styles.css



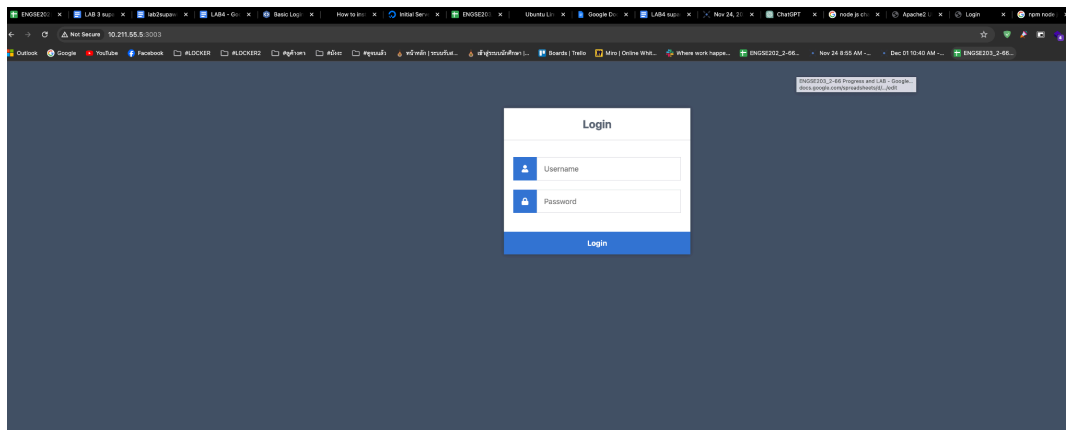
supawit kaennak 66543210031-1

```
devuser@webserver031:/var/www/lab4/nodelogin$ sudo ufw allow 3003
[sudo] password for devuser:
Rule added
Rule added (v6)
```

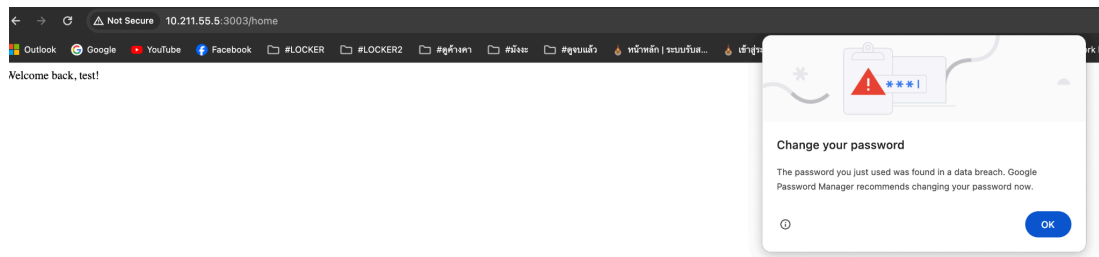
ภาพที่ 37 อนุญาต port 3003 เข้า firewall

```
devuser@webserver031:/var/www/lab4/nodelogin$ node login.js
```

ภาพที่ 38 ทำการ run login.js โดยใช้ node



ภาพที่ 39 ใช้ ip และ port ในการเข้าเว็บ



ภาพที่ 40 ทดสอบ login

```
devuser@webserver031:~$ cd /etc/nginx/sites-available
```

ภาพที่ 41 cd เข้าไป ใน sites-available

```
devuser@webserver031:/etc/nginx/sites-available$ sudo cp default nodelogin
```

ภาพที่ 42 copy ไฟล์ default เป็นไปเป็น nodelogin

```
devuser@webserver031:/etc/nginx/sites-available$ sudo nano nodelogin
```

ภาพที่ 43 แก้ไขไฟล์ nodelogin

```
GNU nano 6.2 nodelogin
##
# You should look at the following URL's in order to grasp a solid understanding
# of Nginx configuration files in order to fully unleash the power of Nginx.
# https://www.nginx.com/resources/wiki/start/
# https://www.nginx.com/resources/wiki/start/topics/tutorials/config_pitfalls/
server {
    listen 80;
    listen [::]:80;
    server_name nodelogin.se-rmutl.net;

    location / {
        proxy_pass http://127.0.0.1:3003; # !!! - change to your app port
        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;
    }
}
```

ภาพที่ 43 ข้างใน nodelogin เขียนตามรูปภาพนี้

```
[admin@admins-iMac-2 ~ % sudo nano /etc/hosts
[Password:
```

ภาพที่ 44 terminal ของ computer ให้ใช้ nano แก้ไข hosts

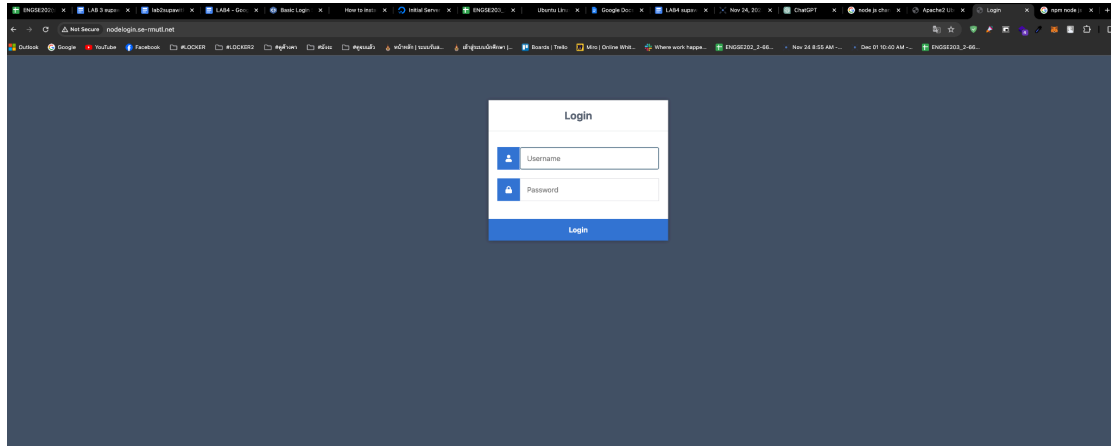
```
##
# Host Database
#
# localhost is used to configure the loopback interface
# when the system is booting. Do not change this entry.
##
127.0.0.1 localhost engse203.lab1
255.255.255.255 broadcasthost
::1 localhost
10.211.55.5 myapp.se-rmutl.net expressjs-example.se-rmutl.net my-react-app.se-r$
10.211.55.5 nodelogin.se-rmutl.net
```

ภาพที่ 45 ข้างในไฟล์ hosts แก้ตามรูปภาพนี้

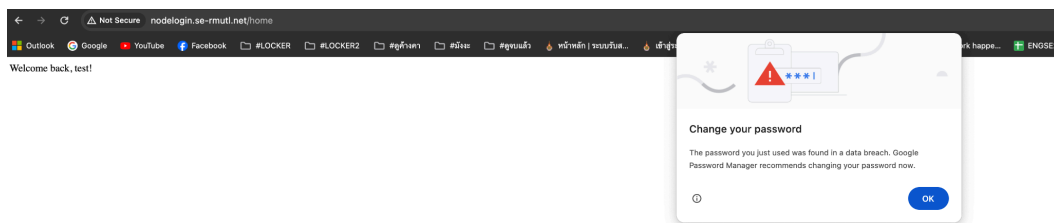
```
devuser@webserver031:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
devuser@webserver031:~$ sudo systemctl restart nginx
```

ภาพที่ 46 ตรวจสอบ syntax nginx

supawit kaennak 66543210031-1



ภาพที่ 47 ทดลอง ใช้ domain name เข้าเว็บดู



ภาพที่ 48 ลอง login

```
devuser@webserver031:/var/www/lab4/client$ mkdir client
```

ภาพที่ 49 สร้าง directory client

```
devuser@webserver031:/var/www/lab4/client$ cd client
```

ภาพที่ 50 cd เข้าไป directory client

```
devuser@webserver031:/var/www/lab4/client$ npx create-react-app .
```

ภาพที่ 51 ใช้ npx สร้าง react app

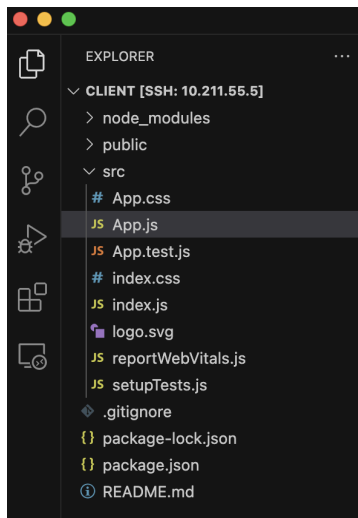
supawit kaennak 66543210031-1

```
devuser@webserver031:/var/www/lab4/client$ npm i --save axios
```

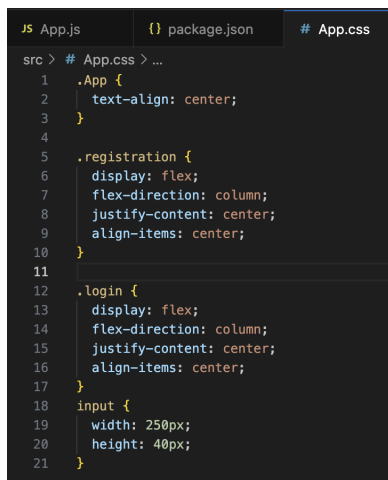
ภาพที่ 52 ติดตั้ง axios

```
devuser@webserver031:/var/www/lab4/client$ npm install express --save
```

ภาพที่ 52 ติดตั้ง express



ภาพที่ 53 ส่วนของ vscode ไฟล์จะมีตามภาพนี้

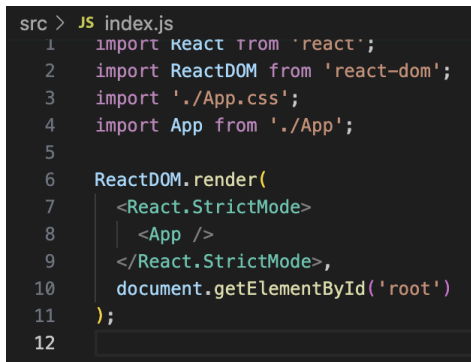


ภาพที่ 54 (ต่อ) ส่วนของ vscode โค้ดหน้า app.css มีดังภาพ



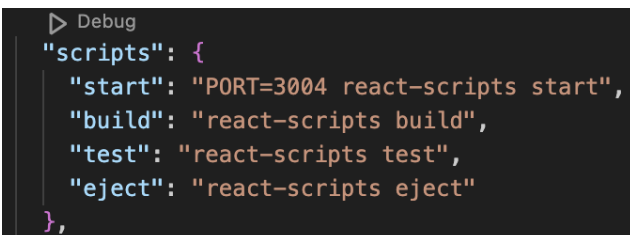
```
src > JS App.js > App > login > then() callback
1 import React, { useEffect, useState } from 'react';
2 import Axios from 'axios';
3 import './App.css';
4
5 function App() {
6
7   const [usernameReg, setUsernameReg] = useState("");
8   const [passwordReg, setPasswordReg] = useState("");
9
10  const [username, setUsername] = useState("");
11  const [password, setPassword] = useState("");
12
13  const [loginStatus, setLoginStatus] = useState("");
14
15  const register = () => {
16    Axios.post("http://localhost:3001/register", {
17      username: usernameReg,
18      password: passwordReg,
19    }).then((response) => {
20      console.log(response);
21    });
22  };
23
24  const login = () => {
25    Axios.post("http://localhost:3001/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
37 }
```

ภาพที่ 55 (ต่อ) ส่วนของ vscode โค้ดหน้า app.js มีดังภาพ



```
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 );
12
```

ภาพที่ 56 (ต่อ) ส่วนของ vscode โค้ดหน้า index.js มีดังภาพ



```
Debug
"scripts": {
  "start": "PORT=3004 react-scripts start",
  "build": "react-scripts build",
  "test": "react-scripts test",
  "eject": "react-scripts eject"
},
```

ภาพที่ 57 (ต่อ) ส่วนของ vscode โค้ดหน้า package.json แก้ไขดังภาพ

supawit kaennak 66543210031-1

```
devuser@webserver031:/var/www/lab4/client$ cd /etc/nginx/sites-available
```

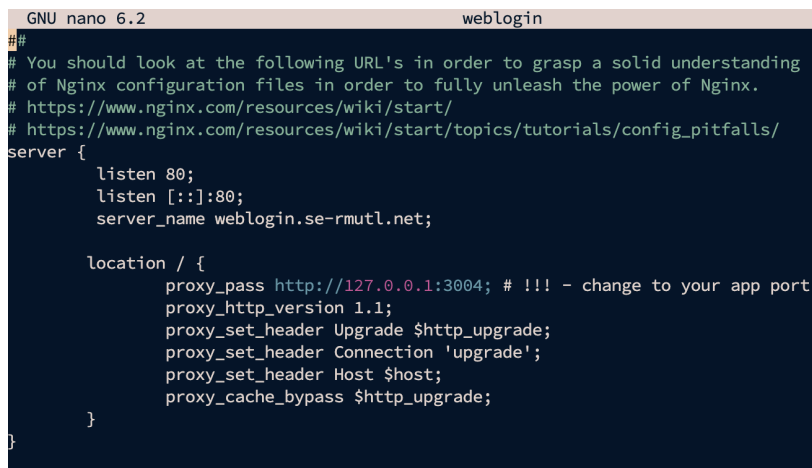
ภาพที่ 58 ใช้ cd เข้า directory sites-available

```
devuser@webserver031:/etc/nginx/sites-available$ sudo cp nodelogin weblogin
```

ภาพที่ 59 copy ไฟล์ nodelogin เป็น weblogin

```
devuser@webserver031:/etc/nginx/sites-available$ sudo nano weblogin
```

ภาพที่ 60 nano ไปแก้ไขไฟล์ weblogin



```
GNU nano 6.2 weblogin
##
# You should look at the following URL's in order to grasp a solid understanding
# of Nginx configuration files in order to fully unleash the power of Nginx.
# https://www.nginx.com/resources/wiki/start/
# https://www.nginx.com/resources/wiki/start/topics/tutorials/config_pitfalls/
server {
    listen 80;
    listen [::]:80;
    server_name weblogin.se-rmutl.net;

    location / {
        proxy_pass http://127.0.0.1:3004; # !!! - change to your app port
        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;
    }
}
```

ภาพที่ 61 ข้างในไฟล์ weblogin แก้ไขตามภาพ

```
10.211.55.5 nodelogin.se-rmutl.net weblogin.se-rmutl.net
```

ภาพที่ 62 ไปที่ terminal ของ pc แล้ว nano เข้าไปแก้ไขไฟล์ hosts ตามภาพ

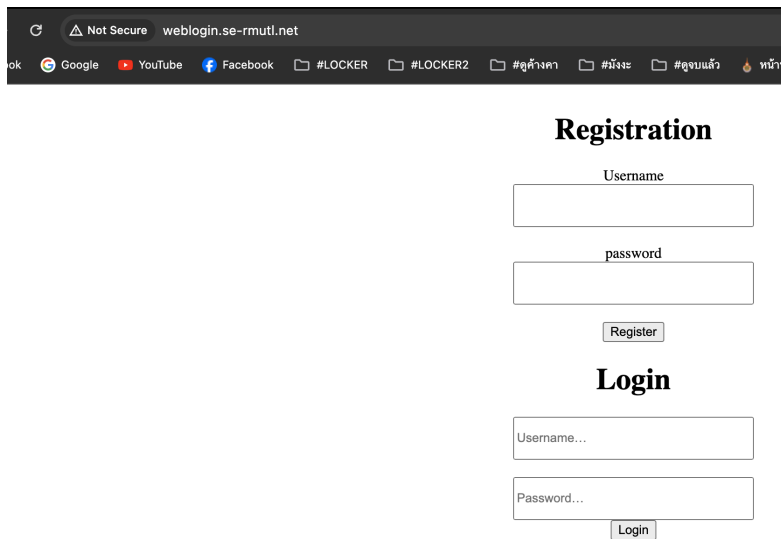
```
devuser@webserver031:/etc/nginx/sites-available$ sudo ufw allow 3004
```

ภาพที่ 63 อนุญาต port 3004 เข้า firewall

```
devuser@webserver031:/var/www/lab4/client$ npm start
```

ภาพที่ 64 กลับมา directory client แล้ว npm start ให้ react app

supawit kaennak 66543210031-1



The screenshot shows a web browser window with the address bar displaying "weblogin.se-rmutl.net". The page has a dark header with navigation links: "Google", "YouTube", "Facebook", "#LOCKER", "#LOCKER2", "#ดูคำสั่ง", "#มีอะไร", "#ดูจบแล้ว", and "หน้า". The main content area is white and contains two sections: "Registration" and "Login". The "Registration" section has a "Username" input field, a "password" input field, and a "Register" button. The "Login" section has a "Username..." input field, a "Password..." input field, and a "Login" button.

ภาพที่ 65 ทดสอบเข้าหน้าเว็บ weblogin

```
devuser@webserver031:/var/www/lab4$ mkdir server
```

ภาพที่ 66 สร้าง directory server

```
devuser@webserver031:/var/www/lab4$ cd server
```

ภาพที่ 67 cd เข้า directory server

```
devuser@webserver031:/var/www/lab4/server$ npm init
```

ภาพที่ 68 npm init สร้าง package

```
devuser@webserver031:/var/www/lab4/server$ npm i --save cors
```

ภาพที่ 69 ติดตั้ง cors

```
devuser@webserver031:/var/www/lab4/server$ install express --save
```

ภาพที่ 70 ติดตั้ง express

supawit kaennak 66543210031-1

```
index.js node_modules package.json package-lock.json
devuser@webserver031:/var/www/lab4/server$ npm install --save mysql2
```

ภาพที่ 71 ติดตั้ง mysql2

```
devuser@webserver031:/var/www/lab4/server$ sudo npm install -g nodemon
```

ภาพที่ 72 ติดตั้ง nodemon

```
mysql> CREATE TABLE IF NOT EXISTS `users` (
  ->   `username` varchar(50) NOT NULL,
  ->   `password` varchar(500) NOT NULL,
  ->   PRIMARY KEY (`username`)
  -> ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

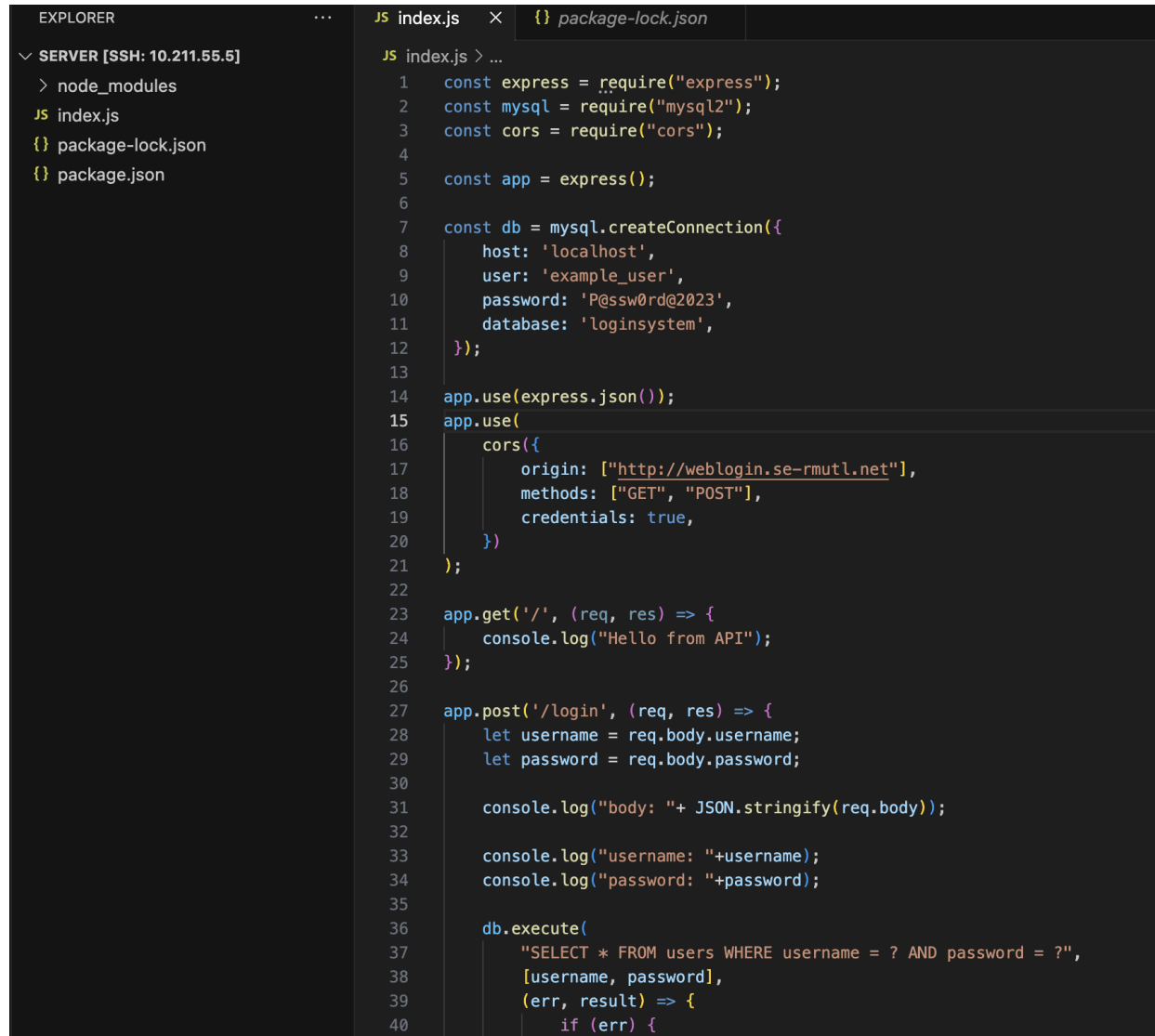
ภาพที่ 73 เข้า mysql แล้ว สร้าง table ตามภาพ

```
mysql> USE loginsystem; INSERT INTO `users` (`username`, `password`) VALUES ('test2', 'test2');
select * from users;
Database changed
Query OK, 1 row affected (0.00 sec)

+-----+-----+
| username | password |
+-----+-----+
| test     | test     |
| test1    | test1    |
| test2    | test2    |
+-----+-----+
4 rows in set (0.00 sec)
```

ภาพที่ 74 เลือกตารางที่สร้างไว้ แล้ว insert ข้อมูลตามภาพ





```
JS index.js > ...
1  const express = require("express");
2  const mysql = require("mysql2");
3  const cors = require("cors");
4
5  const app = express();
6
7  const db = mysql.createConnection({
8    host: 'localhost',
9    user: 'example_user',
10    password: 'P@ssw0rd@2023',
11    database: 'loginsystem',
12  });
13
14  app.use(express.json());
15  app.use(
16    cors({
17      origin: ["http://weblogin.se-rmutl.net"],
18      methods: ["GET", "POST"],
19      credentials: true,
20    })
21  );
22
23  app.get('/', (req, res) => {
24    console.log("Hello from API");
25  });
26
27  app.post('/login', (req, res) => {
28    let username = req.body.username;
29    let password = req.body.password;
30
31    console.log("body: " + JSON.stringify(req.body));
32
33    console.log("username: " + username);
34    console.log("password: " + password);
35
36    db.execute(
37      "SELECT * FROM users WHERE username = ? AND password = ?",
38      [username, password],
39      (err, result) => {
40        if (err) {
```

ภาพที่ 75 ส่วนของ vscode หน้า index.js แก่ตามภาพ

```
devuser@backend031:/var/www$ sudo mkdir lab4
[sudo] password for devuser:
devuser@backend031:/var/www$ cd lab4
devuser@backend031:/var/www/lab4$ sudo mkdir server
devuser@backend031:/var/www/lab4$ cd server
devuser@backend031:/var/www/lab4/server$ npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.
```

ภาพที่ 76 ส่วนนี้จะเป็นของ backend เป็นต้นไป สร้าง directory lab4, เข้าไปใน directory lab4, สร้าง directory server ข้างใน directory lab4 จากนั้น npm init สร้างเพื่อ package

```
devuser@backend031:/var/www/lab4/server$ sudo npm install express --save

added 62 packages, and audited 63 packages in 17s

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

found 0 vulnerabilities
devuser@backend031:/var/www/lab4/server$ sudo npm install --save mysql2

added 11 packages, and audited 74 packages in 2s

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

found 0 vulnerabilities
```

ภาพที่ 77 ติดตั้ง express และ mysql2 โดยใช้ npm

```
devuser@backend031:/var/www/lab4/server$ sudo npm i --save cors

added 2 packages, and audited 76 packages in 697ms

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

found 0 vulnerabilities
```

ภาพที่ 78 ติดตั้ง cors โดยใช้ npm

```
Round 5 - vulnerabilities
devuser@backend031:/var/www/lab4/server$ sudo npm install -g nodemon

added 33 packages in 7s

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

ภาพที่ 79 ติดตั้ง nodemon โดยใช้ npm

```
devuser@backend031:/var/www/lab4/server$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.35-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'P@ssw0rd@2023';
Query OK, 0 rows affected (0.01 sec)

mysql> CREATE DATABASE IF NOT EXISTS `loginsystem` DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
Query OK, 1 row affected, 2 warnings (0.00 sec)

mysql> use loginsystem;
Database changed
mysql> CREATE TABLE IF NOT EXISTS `users` (
  ->   `username` varchar(50) NOT NULL,
  ->   `password` varchar(500) NOT NULL,
  ->   PRIMARY KEY (`username`)
  -> ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Query OK, 0 rows affected, 1 warning (0.02 sec)

mysql> use loginsystem;
Database changed
mysql> INSERT INTO `users` (`username`, `password`) VALUES ('test', 'test');
Query OK, 1 row affected (0.01 sec)
```

ภาพที่ 79 เข้า mysql ,สร้าง database loginsystem, พิมพ์ user loginsystem จากนั้น สร้างตาราง users เพิ่มข้อมูลตามภาพ และ insert ข้อมูลตามภาพ

```
mysql> select * from users;
+-----+-----+
| username | password |
+-----+-----+
| test     | test     |
+-----+-----+
1 row in set (0.00 sec)
```

ภาพที่ 80 เลือกตาราง users เพื่อตรวจสอบข้อมูลข้อใน

```
mysql> CREATE USER 'example_user'@'%' IDENTIFIED BY 'P@ssw0rd@2023';  
Query OK, 0 rows affected (0.02 sec)  
  
mysql> ALTER USER 'example_user'@'%' IDENTIFIED WITH mysql_native_password BY 'P@ssw0rd@2023';  
Query OK, 0 rows affected (0.00 sec)
```

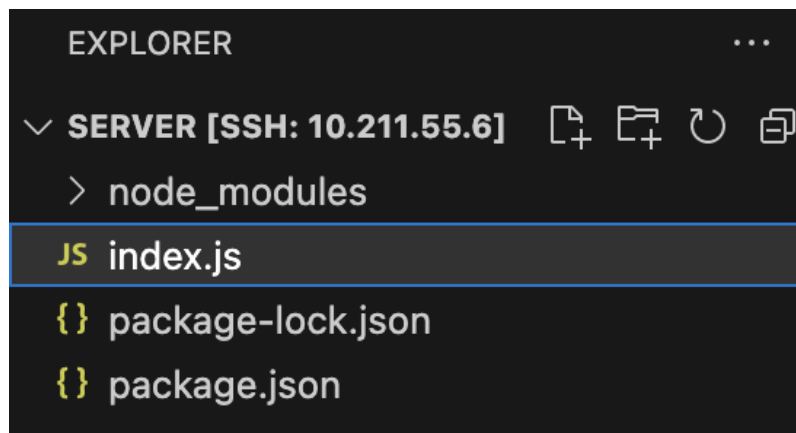
ภาพที่ 81 สร้างผู้ใช้ example\_user และกำหนดรหัส P@ssw0rd@2023 และใช้ alter ให้กับ example\_user

```
mysql> GRANT ALL ON loginsystem.* TO 'example_user'@'%;  
Query OK, 0 rows affected (0.02 sec)  
  
mysql> exit
```

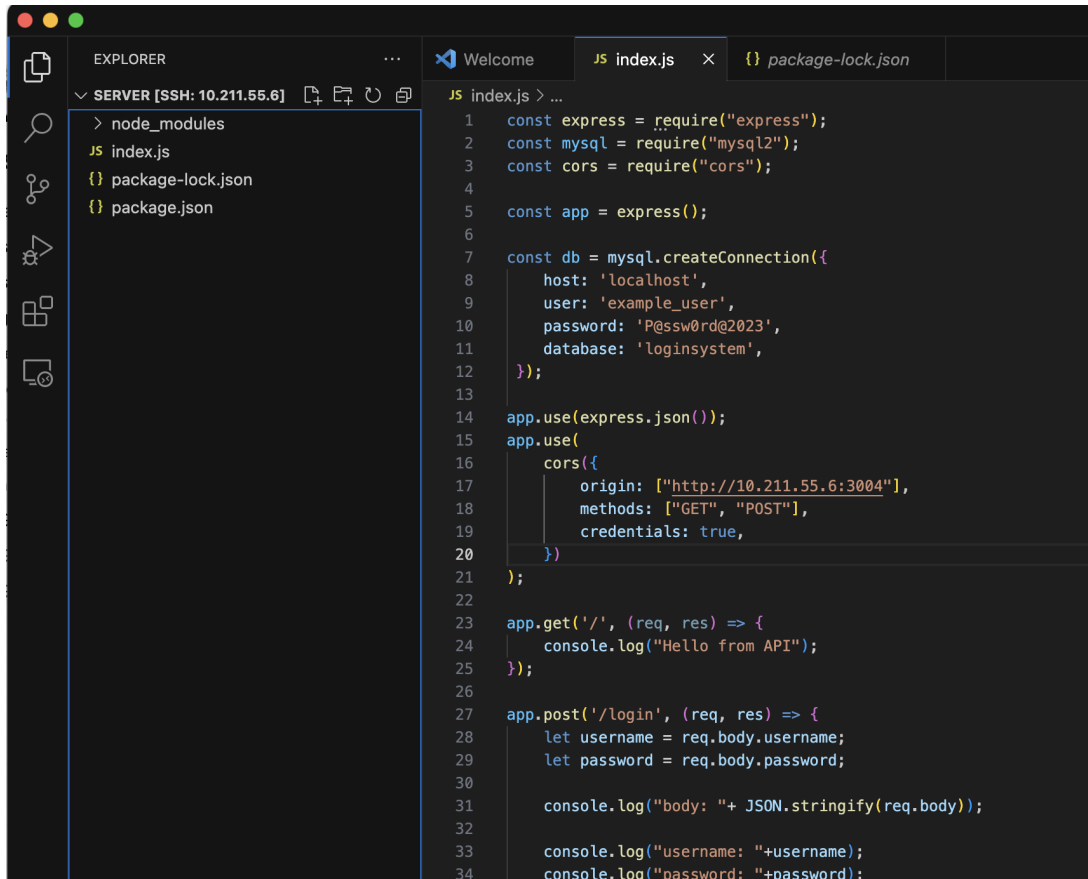
ภาพที่ 82 ใช้ grant all on ในการให้สิทธิ์ user ในการแก้ไข databases

```
devuser@backend031:/var/www/lab4/server$ sudo chown -R $USER:$USER /var/www/lab4
```

ภาพที่ 83 คำสั่งอนุญาตให้ user ธรรมดา แก้ไขไฟล์ต่างๆใน directory lab4 ได้



ภาพที่ 84 ส่วนของ vscode ไฟล์ต่าง



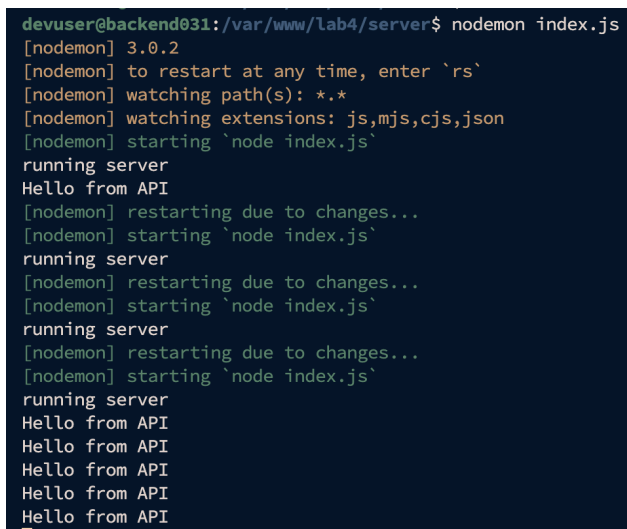
The screenshot shows the VS Code interface with a remote server connection. The Explorer sidebar on the left shows the file structure: `node_modules`, `index.js`, `package-lock.json`, and `package.json`. The main editor displays the content of `index.js`, which is a Node.js application using Express.js and MySQL. The code includes database connection details and API endpoints for GET and POST requests.

```

1  const express = require("express");
2  const mysql = require("mysql2");
3  const cors = require("cors");
4
5  const app = express();
6
7  const db = mysql.createConnection({
8    host: 'localhost',
9    user: 'example_user',
10   password: 'P@ssw0rd@2023',
11   database: 'loginsystem',
12 });
13
14 app.use(express.json());
15 app.use(
16   cors({
17     origin: ["http://10.211.55.6:3004"],
18     methods: ["GET", "POST"],
19     credentials: true,
20   })
21 );
22
23 app.get('/', (req, res) => {
24   console.log("Hello from API");
25 });
26
27 app.post('/login', (req, res) => {
28   let username = req.body.username;
29   let password = req.body.password;
30
31   console.log("body: " + JSON.stringify(req.body));
32
33   console.log("username: "+username);
34   console.log("password: "+password);

```

ภาพที่ 85 (ต่อ) ส่วนของ vscode หน้า index.js เขียนโค้ดตามภาพ



The screenshot shows a terminal window where the command `nodemon index.js` has been executed. The output shows the server starting and logging "Hello from API". The terminal also displays several "restarting due to changes..." messages, indicating that the server is running in a watch mode that automatically restarts when the code is modified.

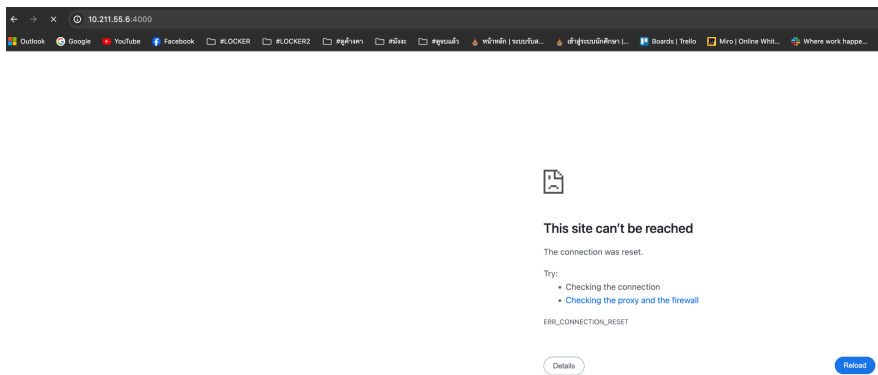
```

devuser@backend031:/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,mjs,cjs,json
[nodemon] starting `node index.js`
running server
Hello from API
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
running server
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
running server
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
running server
Hello from API
Hello from API
Hello from API
Hello from API
Hello from API

```

ภาพที่ 86 ใช้ nodemon ในการ run index.js ใน directory server

supawit kaennak 66543210031-1



ภาพที่ 87 ผลลัพธ์หน้า index.js ไม่มีหน้าเว็บเพราะเป็น api แต่หากลอง reload หน้าเว็บจะมี hello from api เข้ามาในหน้า ubuntu ดังภาพที่ 86