

**SCSJ3483**

**WEB TECHNOLOGY**

**ASSIGNMENT 4**

GROUP NAME: TYPO

GROUP MEMBER: PANG WEN JIE (A18CS0232)

TAN CHONG LIM (A18CS0255)

ONG SHI BING (A18CS0230)

YEOH KAI XIANG (A18CS0278)

SECTION: 02

LECTURER: MR NORIZAM KATMON

Table of Contents

[1.0 Prerequisites 3](#_Toc77084619)

[2.0 Prepare the Environment 4](#_Toc77084620)

[2.1 MySQL Installation and Configuration 4](#_Toc77084621)

[2.2 PHP Installation and Configuration (Including phpMyAdmin) 6](#_Toc77084622)

[2.3 Node.js Installation and Configuration 10](#_Toc77084623)

[3.0 Setup the Project in Production Mode (CRAS) 12](#_Toc77084624)

[3.1 Setup the Database (MySQL) 12](#_Toc77084625)

[3.2 Setup the Backend (Node.js) 14](#_Toc77084626)

[3.3 Setup the Frontend (Angular) 15](#_Toc77084627)

[4.0 References 16](#_Toc77084628)

# 1.0 Prerequisites

* Ubuntu 20.04 x64 server
* Nginx installed and running
* Domain name (and subdomain) pointing to the server’s IP as follows

Graphical user interface, text, application

Description automatically generated

# 2.0 Prepare the Environment

In this manual, the server will be configured to handle multiple services under one server, with different subdomain name. Nginx will be used as reverse proxy to manage the subdomain. Table below summarize the details.

|  |  |  |
| --- | --- | --- |
| Domain/Subdomain | Service | Folder Path |
| typo.ninja  www.typo.ninja | Default landing page | /var/www/html |
| admin.typo.ninja | phpMyAdmin | /var/www/admin.typo.ninja |
| api.typo.ninja | Backend for CRAS (node.js) | /var/www/api.typo.ninja |
| cras.typo.ninja | Frontend for CRAS (angular) | /var/www/cras.typo.ninja |

## 2.1 MySQL Installation and Configuration

1. Run sudo apt-get install mysql-server to download mysql server

Text

Description automatically generated

1. Run mysql\_secure\_installation to start the installation

Text

Description automatically generated

1. Choose No for the first question, then enter new password for MySQL root user.

Text

Description automatically generated

1. Enter y for the following questions until the end.

Text

Description automatically generated

1. Run sudo service mysql status to check MySQL status, make sure it is **active** and **running**.

Text

Description automatically generated

## 2.2 PHP Installation and Configuration (Including phpMyAdmin)

Note: phpMyAdmin will be used only for managing database in this project.

1. Run sudo apt-get install php-fpm php-mysql to install php processor.

Text

Description automatically generated

1. Run sudo nano /etc/php/7.4/fpm/php.ini , then uncomment and change 1 to 0 at line cgi.fix\_pathinfo=0.

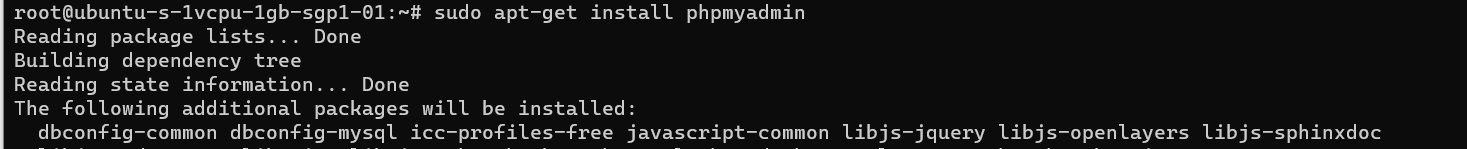
Text

Description automatically generated

1. Save and restart php processor by running sudo systemctl restart php7.4-fpm
2. Run sudo nano /etc/nginx/sites-available/default. Remove all the default configuration and add the following lines of codes.



1. Make a new directory by typing mkdir /var/www/admin.typo.ninja .
2. Run sudo systemctl restart nginx to restart nginx server.
3. Run sudo apt-get install phpmyadmin to install phpMyAdmin



1. Press TAB to skip web server configuration

Graphical user interface, text, application, email

Description automatically generated

1. Choose YES to configure database.

Graphical user interface, text

Description automatically generated

1. Input new password for phpMyAdmin

A picture containing text

Description automatically generated

1. Run sudo ln -s /usr/share/phpmyadmin /var/www/admin.typo.ninja to link the php files to nginx directory.
2. Run sudo systemctl restart php7.4-fpm to restart PHP processor.
3. Go to <https://admin.typo.ninja> . You should see the following results, meaning that phpMyAdmin server is up and running.

Graphical user interface

Description automatically generated

1. Create new superuser for phpMyAdmin.
2. Run sudo mysql -p -u root and type your password to login as root user.

Text

Description automatically generated

1. Run the following SQL statement to create new user.



1. Login phpMyAdmin using admin and new\_password. You should see the following result.

Graphical user interface, text, email

Description automatically generated

## 2.3 Node.js Installation and Configuration

1. Run curl -sL https://deb.nodesource.com/setup\_14.x -o nodesource\_setup.sh to download version 14.x node setup file.
2. Then, run sudo bash nodesource\_setup.sh.

Text

Description automatically generated

1. Run sudo apt-get install nodejs to install nodejs.
2. Run sudo apt-get install build-essential to install other dependencies.

Text

Description automatically generated

1. Run node --version and npm --version to verify the installation.

Text

Description automatically generated

1. Run sudo npm i -g pm2 (process manager for Node.js applications)

Text

Description automatically generated

1. Run pm2 startup systemd to enable run on boot.

Text

Description automatically generated

# 3.0 Setup the Project in Production Mode (CRAS)

## 3.1 Setup the Database (MySQL)

1. Go to [https://admin.typo.ninja](https://admin.typo.ninja/phpmyadmin) and login as admin.
2. Create new database by click on Database tab

Graphical user interface, text, application, email

Description automatically generated

1. Create new database named college\_application

Graphical user interface, text, email

Description automatically generated

1. Press SQL tab

Graphical user interface, text, application, email

Description automatically generated

1. Copy and paste the SQL commands from college\_application.sql into the field, and press Go

Graphical user interface, text, application

Description automatically generated

1. The results should be similar to this.

Graphical user interface, application

Description automatically generated

## 3.2 Setup the Backend (Node.js)

1. Download the repository: git clone <https://github.com/tanchonglim/CRAS-Angular.git>

Text

Description automatically generated

1. Run cp -r CRAS-Angular/backend/ /var/www/api.typo.ninja to copy the code to new directory.
2. Run cd /var/www/api.typo.ninja to change current working directory.
3. Run npm i to install the dependencies.
4. Set the environment variable for database username and password by running

export DATABASE\_USER=admin

export DATABASE\_PASSWORD=new\_password

1. Run node index.js . If the response is as below, meaning that setup is correct.



1. Press ctrl+c to close the service and run pm2 start index.js to start the server as background process.

A screenshot of a computer

Description automatically generated

1. Run sudo nano /etc/nginx/sites-available/default and add the following codes below the current code. Then, save and exit.



1. Run systemctl restart nginx to restart nginx server.
2. Go to <https://api.typo.ninja/> . You should see the following output.

Graphical user interface, text, application, chat or text message

Description automatically generated

## 3.3 Setup the Frontend (Angular)

1. At your own PC, download the repository from <https://github.com/tanchonglim/CRAS-Angular>
2. Run npm i in the frontend/ directory
3. At /src/environments/environments.prod.ts , change the apiUrl to <https://api.typo.ninja>
4. Run ng build in frontend/ directory
5. Upload the files under /dist/cras/ to hosting server under directory /var/www/cras.typo.ninja by using FileZilla or any FTP client.
6. Run sudo nano /etc/nginx/sites-available/default and add the following codes below the current code. Then, save and exit.



1. Run systemctl restart nginx to restart nginx server.
2. Go to <https://cras.typo.ninja/> . You should see the home page of CRAS.

# 4.0 References

<https://www.digitalocean.com/community/tutorials/how-to-install-linux-nginx-mysql-php-lemp-stack-in-ubuntu-16-04>

<https://www.digitalocean.com/community/tutorials/how-to-install-and-secure-phpmyadmin-with-nginx-on-ubuntu-16-04>

<https://www.digitalocean.com/community/tutorials/how-to-set-up-a-node-js-application-for-production-on-ubuntu-16-04>

<https://nicknetvideos.com/blog/post/how-to-run-a-website-in-a-subdomain-in-digital-ocean>