1. Установить Nginx и настроить его на работу с PHP-FPM.

sudo apt install nginx // устанавливаем Nginx

sudo apt install php-fpm // устанавливаем PHP-FPM

sudo service nginx start // запускаем Nginx

cd /etc/nginx/sites-available // переходим в каталог где лежат конфигурационные файлы

sudo touch project.local // cоздаем свой конфигурационный файл

sudo nano project.local // открываем его

// прописываем

server {

listen 80; # порт, прослушивающий nginx

server\_name project.local; # доменное имя, относящиеся к текущему виртуальному хосту

root /var/phpcode/project.local; # каталог в котором лежит проект, путь к точке входа

index index.php;

# add\_header Access-Control-Allow-Origin \*;

# serve static files directly

location ~\* \.(jpg|jpeg|gif|css|png|js|ico|html)$ {

access\_log off;

expires max;

log\_not\_found off;

}

location / {

# add\_header Access-Control-Allow-Origin \*;

try\_files $uri $uri/ /index.php?$query\_string;

}

location ~\* \.php$ {

try\_files $uri = 404;

fastcgi\_split\_path\_info ^(.+\.php)(/.+)$;

fastcgi\_pass unix:/var/run/php/php8.3-fpm.sock; # подключаем сокет php-fpm

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

include fastcgi\_params;

}

location ~ /\.ht {

deny all;

}

}

sudo nginx -t // проверяем нет ли ошибок

cd /etc/nginx/sites-enabled/ // активируем конфигурационный файл

sudo ln -s /etc/nginx/sites-available/project.local // создаем символическую ссылку

sudo nginx -t // проверяем нет ли ошибок

sudo nano /etc/hosts

127.0.0.1 project.local  // добавляем домен

sudo service php8.3-fpm start // запускаем php-fpm

cd /var/phpcode/ // переходим в каталог с проектом

sudo chmod -R 777 project.local // даем полные права

cd project.local

sudo touch index.php

sudo nano index.php

<?php

echo “Hello”; ?>

В браузере набираем доменное имя project.local

1. Установить Apache. Настроить обработку PHP. Добиться одновременной работы с Nginx.

sudo apt install apache2 -y // устанавливаем apache

sudo apt install libapache2-mod-php -y // устанавливаем php

cd /etc/apache2 // переходим в каталог

sudo nano ports.conf // открываем файл настройки

Listen 45678 // меняем порт

cd sites-enabled // переходим в каталог

sudo nano 000-default.conf // открываем файл

<VirtualHost \*:45678> // меняем порт

1. Настроить схему обратного прокси для Nginx (динамика - на Apache).

sudo nano /etc/nginx/sites-enabled/default // открываем файл

location / {

proxy\_pass <http://localhost:45678>;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Real-IP $remote\_addr;

}

location ~\* ^.+.(jpg|jpeg|gif|png|ico|css|zip|pdf|txt|tar|js)$

{

root /var/www/html;

}

1. Установить MySQL. Создать новую базу данных и таблицу в ней.

sudo apt install mysql-server -y // установка

sudo mysql // запуск

CREATE DATABASE gb; // создание новой базы данных

USE gb; // выбираем базу данных

CREATE TABLE users (name VARCHAR(15), age INT); // создаем таблицу

INSERT INTO users (age, name) VALUES (14, "Sergey"), (10, "Max"), (17, "Alex"); // записываем в таблицу данные

SELECT \* FROM users; // выводим всю таблицу

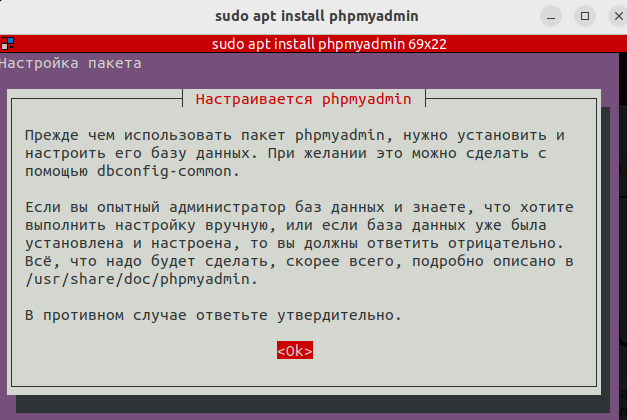
exit // выход

1. \* Установить пакет phpmyadmin и запустить его веб-интерфейс для управления MySQL.

sudo apt install php-mbstring php-zip php-gd php-json php-curl //установка расширения php

sudo apt install phpMyAdmin

Выбираем сервер apache2, устанавливаем пароль



sudo phpenmod mbstring // включаем расширение php

sudo service apache2 restart // перезагружаем сервер

<http://localhost:45678/phpmyadmin/> // в браузере

•\* Настроить схему балансировки трафика между несколькими серверами Apache на стороне Nginx с помощью модуля ngx\_http\_upstream\_module.