

Лабораторна робота №6. Створення сервісу nginx. Проксірування запитів

Хід роботи

Інструкції до виконання дивимось в [лекції №5](#)

- Створіть в проекті папку *nginx*, в якій створіть файли *nginx.prod.conf* та *nginx.dev.conf*
- Створіть віртуальний хост, наприклад *websystem-docker.com*
- Створити сервіс для *nginx*, здійснити первинні налаштування докера для даного сервісу (*docker-compose.yml*) та конфігурації самого серверу (*nginx.prod.conf*)
- Використовуючи *axios* та проксірування запитів, здійснити запит між клієнтською частиною та серверним додатком
- Видалити відкриті порти в *yml*-файлі (всі запити йдуть виключно через *nginx*)

Nginx для dev-режиму

- Створіть хост для режиму розробки, наприклад *websystem-docker.local*
- В файлі *nginx.conf.dev* пропишіть *server_name* на даний хост. Всі інші налаштування поки такі ж як і в *nginx.conf.prod*
- В *yml*-файлі для *dev* в сервісі *nginx* пропишіть *volume* на *nginx.conf.dev*
- Дозволити використання протоколу WS (web-sockets) в *dev*-конфігурації *nginx*:

```
location / {  
    proxy_pass http://frontend:3000;  
    # needed for sockets  
    proxy_http_version 1.1;  
    proxy_set_header Upgrade $http_upgrade;  
    proxy_set_header Connection "upgrade";  
}
```

Як перевіряти

Для production-версії:

- Запити повинні здійснюватись за адресою <http://websystem-docker.com>
- Заборонено здійснювати запити за адресами <http://localhost:3000>, <http://localhost:3001>
- Повинен працювати запит між frontend та api сервісами

Для dev-версії

- Запити повинні здійснюватись за адресою <http://websystem-docker.local>
- Зміни в коді повинні автоматично оновлювати відповідні компоненти сторінки в браузері