**Projet Y**Docker

PHAM HUYNH TUONG VY

NUNES-EMILIO RICARDO

LAMY EVAN 04/04/2024

Table de matières

[I. Choix Techniques 3](#_Toc163163310)

[1.1. Base de données : MySQL 3](#_Toc163163311)

[1.2. Frontend : Vue.js 3](#_Toc163163312)

[1.3. Backend : Nodejs avec express 3](#_Toc163163313)

[II. Base de données 4](#_Toc163163314)

[1.1. Base de donnée : MCD 4](#_Toc163163315)

[1.2. Base de donnée : MLD 5](#_Toc163163316)

[III. Le docker-compose 6](#_Toc163163317)

[IV. Dockerfile 7](#_Toc163163318)

[1.1. Dockerfile : backend 7](#_Toc163163319)

[1.2. Dockerfile : frontend 7](#_Toc163163320)

# Choix Techniques

## Base de données : MySQL

Nous avons choisi MySQL depuis notre fichier docker compose pour l’héberger sur phpMyAdmin, car c’est une base de données gratuite et open source, mais aussi pour sa facilité d’utilisation, ainsi que sa popularité. De plus MySQL est sécurisé puisqu’il inclut un chiffrement de données, et il est compatible avec plusieurs langages de programmations comme par exemple PHP et JavaScript.

## Frontend : Vue.js

Pour le frontend, la récupération des données a été réaliser avec axios, qui permet au Vue.js d’accéder aux données de l’API via un lien donné. Axios va récupérer les données au format json que l’on pourra par la suite stocker dans une variable.  
Le framework Vue.js, Pinia store a été utilisé afin de stocker les données, pour faciliter leurs utilisations au sein du site.

## Backend : Nodejs avec express

Afin de créer notre backend nous avons utilisé Nodejs, ce qui nous as permis de pouvoir connecter notre frontend à notre base de donnée, de plus il nous as permis de manipuler les données, que ce soit de la modification à l’insertion de données.

# Base de données

## Base de donnée : MCD

Une image contenant texte, capture d’écran, diagramme, Police

Description générée automatiquement

La base de donné a été créer sur PowerAMC, nous nous sommes inspirés de site officiel de X afin de visualiser comment créer l’arborescence des données , et facilités leurs manipulations.

## Base de donnée : MLD

Une image contenant texte, diagramme, Plan, ligne

Description générée automatiquement

Le MLD a été généré automatiquement par PowerAMC et il a été le modèle pour la création de tous les inserts composant la base. Que ce soit des ‘’posts’’ (messages apparaissant sur le home), des utilisateurs, des likes, ou encore des commentaires, tous ces insertions de données était nécessaire afin d’avoir un semblant de site internet pour la démo ainsi que le développement du site.

# Le docker-compose

Notre fichier docker compose est constitué de 5 parties.  
La 1ère partie nommé nodejs est le serveur Nodejs pour le backend, elle appelle le fichier dockerfile attribué au backend.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

La 2éme ainsi que la 3éme partie s’occupe de l’hébergement sur Phpmyadmin de la base de données ainsi que l’insertion des données dedans.

Une image contenant texte, capture d’écran, menu, Police

Description générée automatiquement

La 4éme appelle le fichier dockerfile attribué au frontend, afin de l’initialiser.  
Une image contenant texte, capture d’écran, Police

Description générée automatiquement

La 5éme permet de lancer le serveur Nginx afin de pouvoir héberger notre frontend.

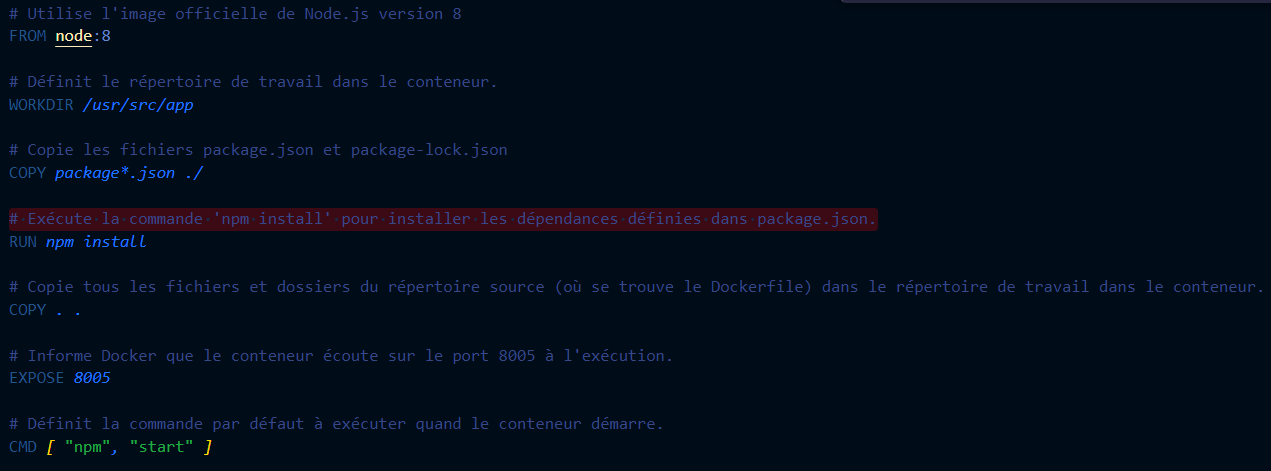
Une image contenant texte, capture d’écran, Police

Description générée automatiquement

# Dockerfile

## Dockerfile : backend

Notre fichier dockerfile attribué au backend nous permet de définir le répertoire de travail dans le conteneur, de copier les fichiers package.json et package-lock.js, d’exécuter la commande ‘npm install’ afin d’installer les dépendances dans package.json, de copier tous les fichiers et dossiers du répertoire source dans le répertoire de travail dans le conteneur, informer Docker le port et enfin d’exécuter des commandes quand le conteneur démarre.



## Dockerfile : frontend

Le fichier dockerfile attribué au frontend fais pour la première partie comme le backend, mais de plus il copie l’application du build dans le conteneur Nginx, il configure les fichiers Nginx ainsi que le port.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Voici le site final :  
