

Mise en place d'un environnement de développement Docker & Symfony

Partie 1– Préparation de l'environnement Docker:

docker/dev/Dockerfile:

FROM php:8.2-cli

Installer les extensions nécessaires pour Symfony et MySQL

RUN apt-get update && apt-get install -y \

git \

unzip \

zip \

libzip-dev \

libonig-dev \

libxml2-dev \

libpq-dev \

libpng-dev \

&& docker-php-ext-install pdo pdo_mysql zip

Installer Composer

COPY --from=composer:2 /usr/bin/composer /usr/bin/composer

Installer Symfony CLI

COPY --link \

--from=ghcr.io/symfony-cli/symfony-cli:latest \

/usr/local/bin/symfony /usr/local/bin/symfony

WORKDIR /var/www/html

CMD ["tail", "-f", "/dev/null"]

Pourquoi php:cli et pas php-fpm ?

⇒ Parce qu'on utilise symfony serve, pas Nginx/Apache.

docker-compose.yml

services:

php:

build:

context: .

dockerfile: ./docker/dev/Dockerfile

container_name: mon_projet_php

volumes:

- ./var/www/html

ports:

- "8000:8000" # pour Symfony Serve

depends_on:

- db

db:

image: mysql:8.0

container_name: mon_projet_db

environment:

MYSQL_ROOT_PASSWORD: root

MYSQL_DATABASE: symfony

MYSQL_USER: symfony

MYSQL_PASSWORD: symfony

ports:

- "3306:3306"

volumes:

- db_data:/var/lib/mysql

phpmyadmin:

image: phpmyadmin/phpmyadmin

container_name: mon_projet_phpmyadmin

environment:

PMA_HOST: db

PMA_USER: symfony

PMA_PASSWORD: symfony

ports:

- "8080:80"

depends_on:

- db

volumes:

db_data:

⇒ **Lancer l'environnement Docker:**

docker-compose up -d

⇒ **Vérifier que tout est lancé :**

docker-compose ps #php, db et phpmyadmin en Up

Partie 2 – Installation de Symfony Demo:

Pour exécuter des commandes Symfony ou Composer, vous devez être "à l'intérieur" du conteneur PHP :

docker exec -it mon_projet_php bash

Toutes les commandes suivantes doivent être tapées à l'intérieur du conteneur:

- Vérifie que Symfony CLI est installé

symfony -v

- Augmenter le timeout de Composer (si besoin)

composer config --global process-timeout 2000

- Créer l'application Demo sans GIT

symfony new demo --demo --no-git

demo sera le nom du dossier de l'application

--demo indique que tu veux le Symfony Demo Application

--no-git: pas de configuration Git requise

- Configurer la base de données dans .env.local

`DATABASE_URL="mysql://symfony:symfony@db:3306/symfony?serverVersion=8.0&charset=utf8mb4"`

Explication :

`symfony` → user MySQL

`symfony` → password

`db` → nom du service MySQL (docker-compose)

`symfony` → nom de la base

`8.0` → version MySQL

`utf8mb4` → recommandé

- Entrer dans le dossier du projet Symfony

cd demo

- Créer le schéma Doctrine

php bin/console doctrine:schema:create

- Charger les fixtures

php bin/console doctrine:fixtures:load

- Lancer Symfony

symfony serve --no-tls --allow-all-ip