Rest Api With Lumen & Docker!

Let’s Gooo…..

Create File a PHP

Make sure you have set up your laravel application with the Lumen framework. Here is the IndexController.php file:

```php
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Http\Response;
use Illuminate\Support\Str;
use Illuminate\Http\JsonResponse;
use Mail;

use App\Helpers\ResponseService;

class IndexController extends Controller
{

    public function __construct()
    {

    }

    public function index() : JsonResponse
    {
        $message =  env('APP_ENV')." - Welcome Rest Api Bayu Widia Santoso...";

        return ResponseService::produceCode(200
                                    , "success"
                                    , date("Y-m-d H:i:s")
                                    , $message);
    }
}
```

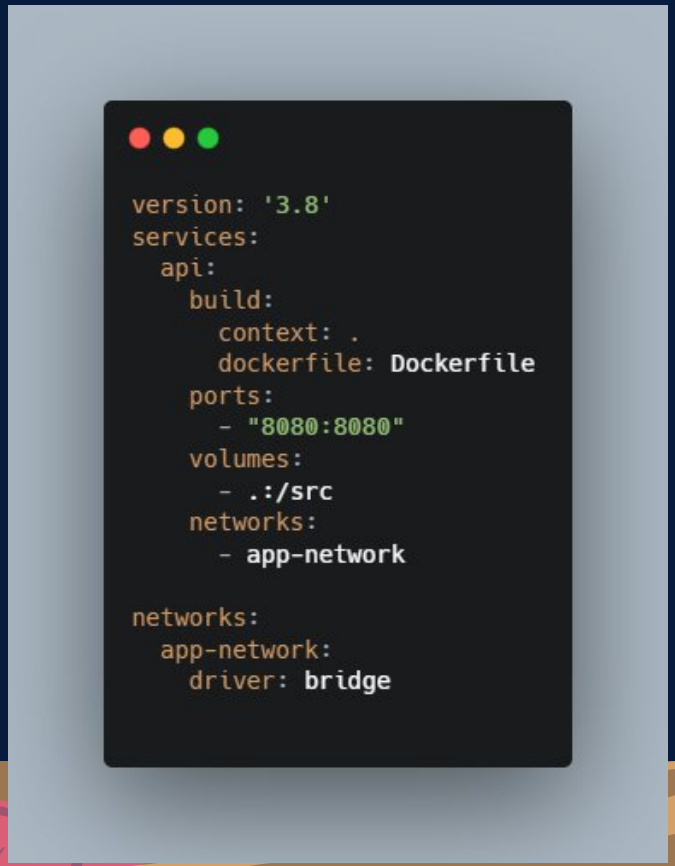
Create a Dockerfile

Create a docker file in the same directory as your app folder :

```
# Use a PHP 8.1 base image
FROM php:8.1-fpm-alpine

# Install necessary system dependencies and PHP extensions
RUN apk --no-cache add \
    libpng-dev \
    libjpeg-turbo-dev \
    libwebp-dev \
    zlib-dev \
    libxml2-dev \
    oniguruma-dev \
    curl \
    bash\
    && docker-php-ext-configure gd --with-jpeg --with-webp

# Install Composer
RUN curl -sS https://getcomposer.org/installer | php -- --install-dir=/usr/local/bin --
filename=composer

# Copy all of the files in the current folder to /src
COPY . /src
WORKDIR /src

# Install PHP dependencies using Composer
RUN composer install --no-dev --optimize-autoloader

# Expose port 8080
EXPOSE 8080

# Run the PHP built-in server
CMD ["php", "-S", "0.0.0.0:8080", "-t", "public"]
```

Create a docker-compose

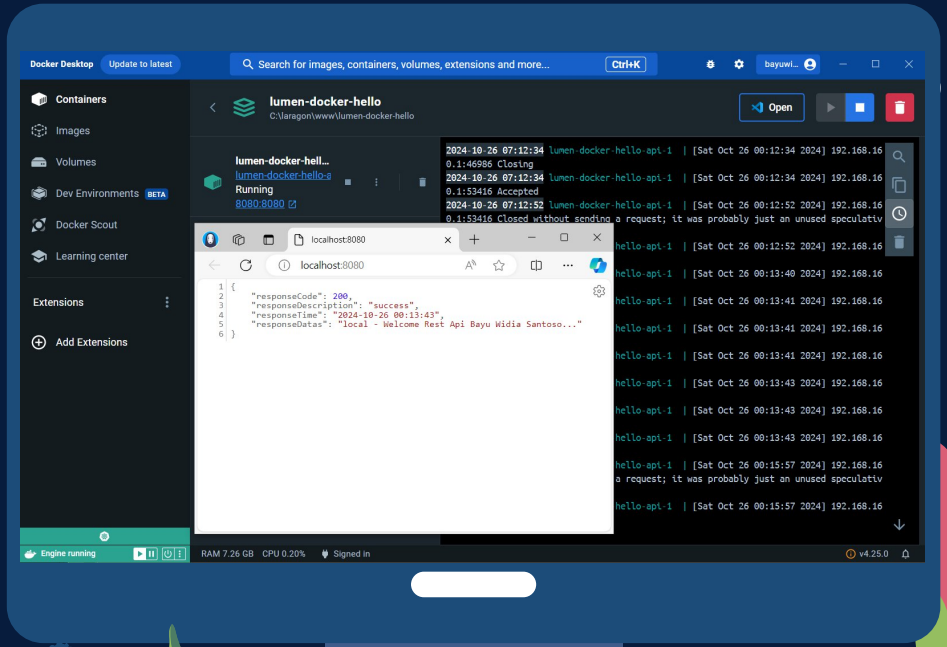
Create a docker-compose in the same directory as your app folder :



```yaml
version: '3.8'
services:
  api:
    build:
      context: .
      dockerfile: Dockerfile
    ports:
      - "8080:8080"
    volumes:
      - .:/src
    networks:
      - app-network

networks:
  app-network:
    driver: bridge
```

Docker Desktop

```
# Rebuild the Docker image
docker-compose down
docker-compose build --no-cache
docker-compose up
```



SEE YOU...