

GABRIELE ROMANATO

Menu

Usare PHP con Docker

Docker è diventato uno strumento essenziale per sviluppatori e amministratori di sistema, permettendo di creare, distribuire e gestire applicazioni in container leggeri e portatili. PHP, uno dei linguaggi di scripting più popolari per lo sviluppo web, può essere facilmente integrato con Docker per creare ambienti di sviluppo e produzione isolati e consistenti. In questo articolo, vedremo come configurare un ambiente PHP utilizzando Docker.

Iniziamo creando una semplice struttura di progetto:

```
my-php-app/  
├── src/  
│   └── index.php  
├── Dockerfile  
└── docker-compose.yml
```

Il `Dockerfile` definisce l'immagine Docker personalizzata per la nostra applicazione PHP. Creiamo un file `Dockerfile` nella directory principale del progetto con il seguente contenuto:

```
# Usare l'immagine ufficiale di PHP con Apache  
FROM php:8.3.7-apache  
  
# Copiare i file del progetto nella directory di lavoro dell'immagine  
COPY src/ /var/www/html/  
  
# Esporre la porta 80 per il server web  
EXPOSE 80
```

Questo `Dockerfile` utilizza l'immagine ufficiale di PHP con Apache, copia i file del progetto nella directory appropriata e espone la porta 80 per il server web.

Il `docker-compose.yml` definisce i servizi Docker necessari per eseguire l'applicazione. Creiamo un file `docker-compose.yml` nella directory principale del progetto con il seguente contenuto:

```
version: '3.8'

services:
  web:
    build: .
    ports:
      - "8080:80"
    volumes:
      - ./src:/var/www/html
```

Questo file `docker-compose.yml` definisce un servizio chiamato `web`, costruito utilizzando il `Dockerfile` nella directory corrente. Il servizio mappa la porta 80 del container alla porta 8080 dell'host e monta la directory `src` del progetto nella directory `/var/www/html` del container per supportare il live-reloading durante lo sviluppo.

Creiamo un semplice file PHP per verificare che tutto funzioni correttamente. Nel file `src/index.php`, aggiungi il seguente codice:

```
<?php
phpinfo();
```

Questo script visualizza informazioni dettagliate sulla configurazione di PHP.

Con la struttura del progetto e i file di configurazione pronti, possiamo avviare l'applicazione con Docker Compose. Esegui il seguente comando nella directory principale del progetto:

```
docker-compose up
```

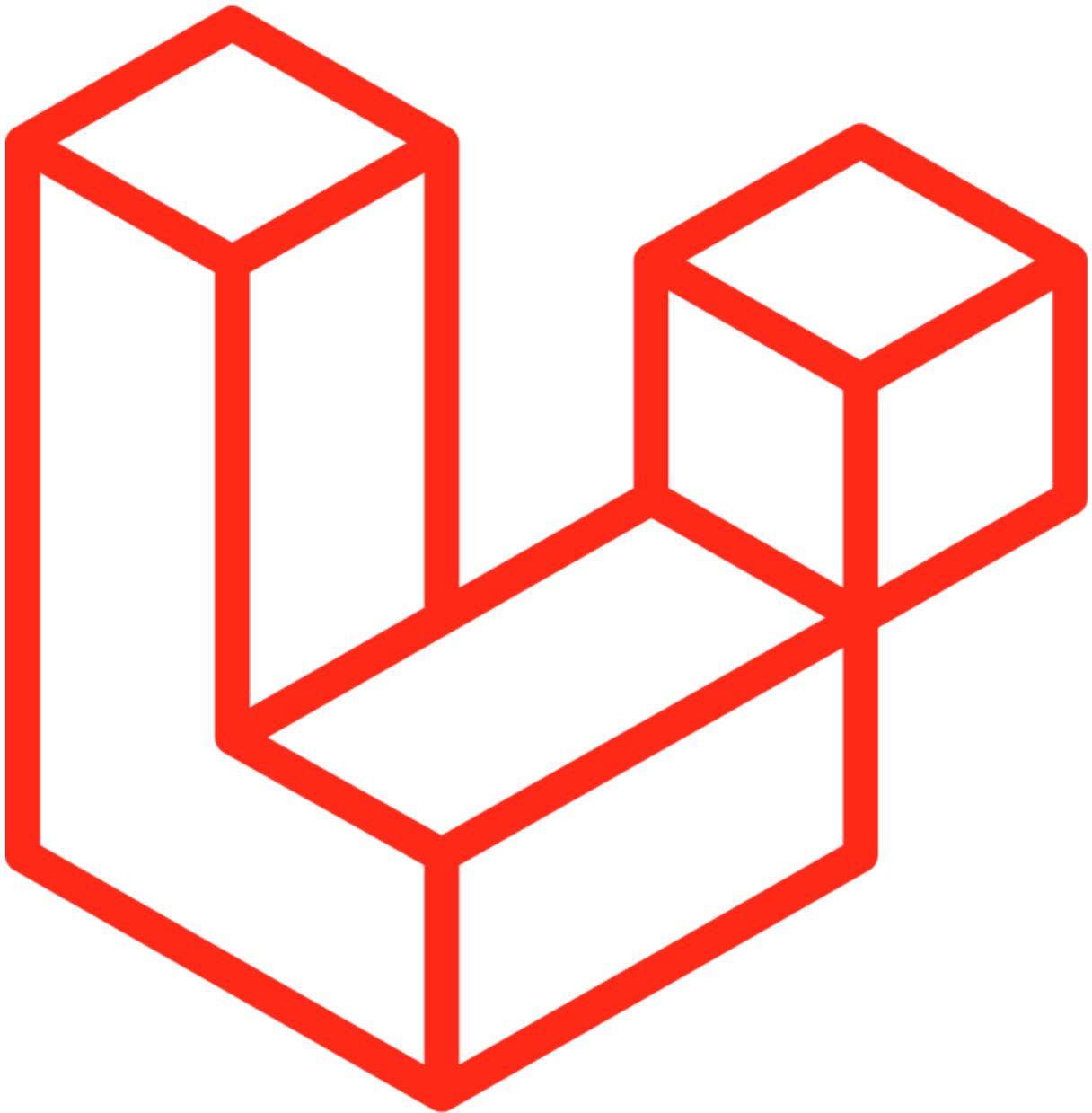
Docker Compose costruirà l'immagine Docker e avvierà il container. Una volta avviato, l'applicazione PHP sarà accessibile all'indirizzo `http://localhost:8080` nel browser. Dovresti vedere la pagina di informazioni di PHP.

Conclusione

In questo articolo, abbiamo visto come configurare un ambiente PHP con Docker utilizzando un `Dockerfile` e `docker-compose.yml`. Questa configurazione consente di creare ambienti di sviluppo e produzione consistenti e isolati, facilitando lo sviluppo e la distribuzione di applicazioni PHP.

Docker semplifica la gestione delle dipendenze e dell'ambiente di esecuzione, rendendo il processo di sviluppo più efficiente e meno incline a errori dovuti a differenze di configurazione tra ambienti diversi.

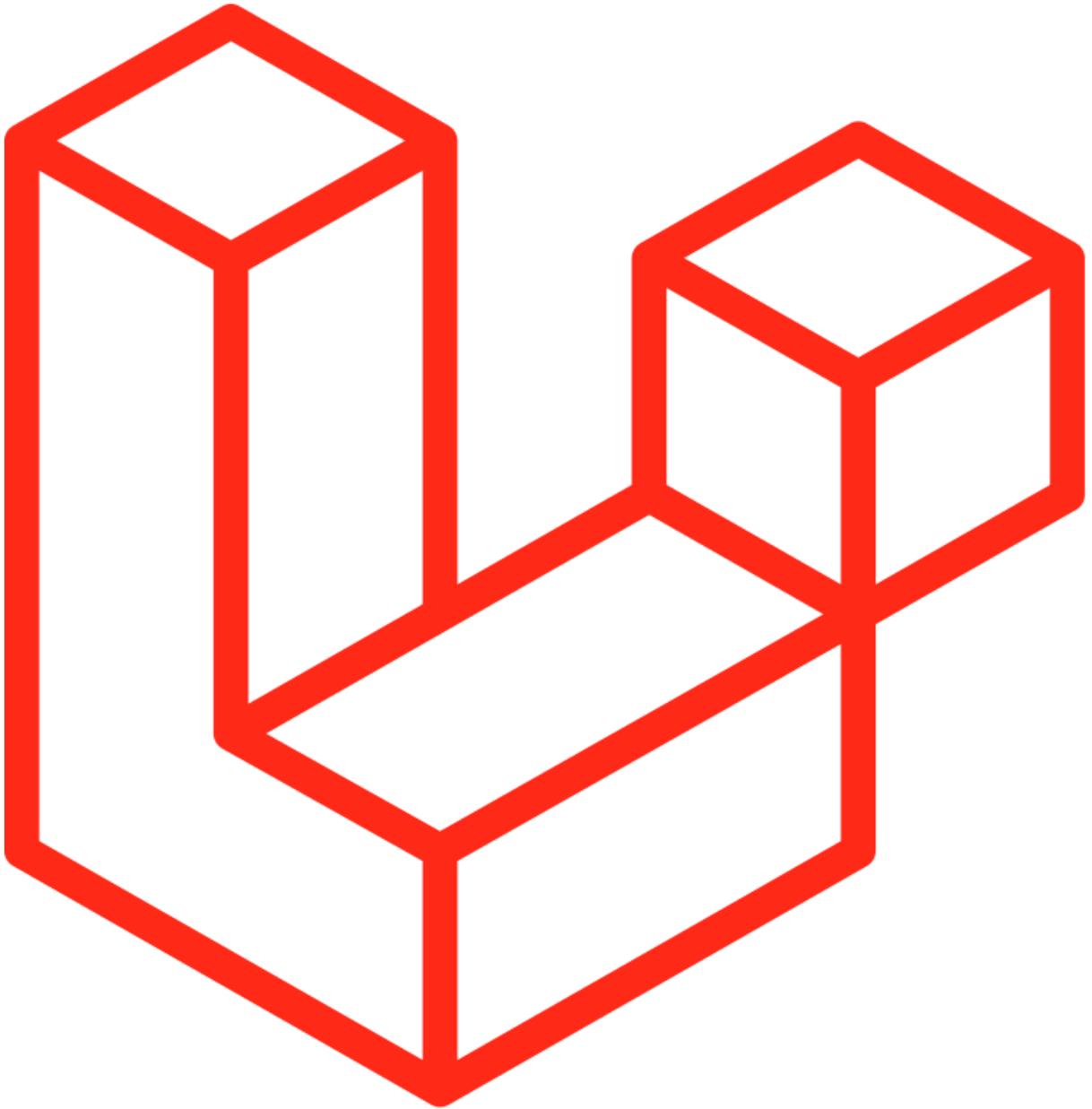
Applicazioni Correlate



-

Laravel Placeholder Image

Applicazione in Laravel per creare immagini segnaposto.
Docker Docker Compose Composer PHP Laravel JavaScript

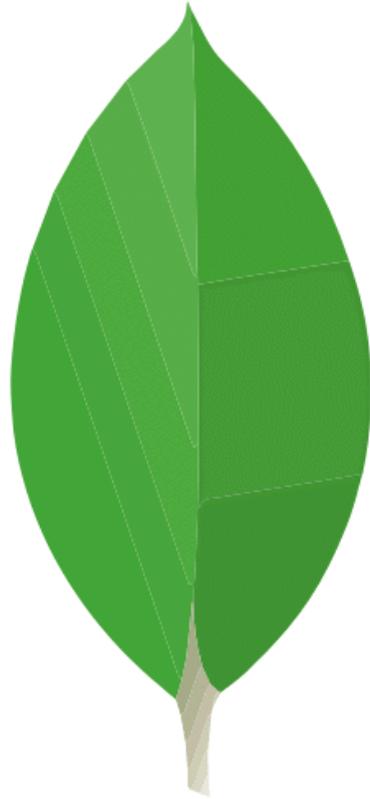


•

Laravel Shopping Cart

Applicazione che fa parte del progetto Laravel E-commerce e implementa la gestione del carrello in Laravel.

Docker Docker Compose Composer PHP Laravel



-

PHP MongoDB App

Applicazione basata su MongoDB con il driver PHP ufficiale.
Docker Docker Compose Composer PHP MongoDB