

GABRIELE ROMANATO

Come usare PostgreSQL in Docker con Docker Compose e PHP

Docker e Docker Compose semplificano la configurazione e la gestione di database come PostgreSQL. Questa guida ti mostrerà come configurare un ambiente di sviluppo con PostgreSQL e PHP utilizzando Docker Compose.

Requisiti

- Docker installato sul tuo sistema
- Docker Compose installato

Passo 1: Creazione della Struttura dei File

Inizia creando una cartella per il tuo progetto con la seguente struttura:

```
.
|-- docker-compose.yml
|-- php/
|   |-- Dockerfile
|-- app/
|   |-- index.php
```

Passo 2: Configurare `docker-compose.yml`

Il file `docker-compose.yml` definirà i servizi necessari, come PostgreSQL e PHP. Ecco un esempio:

```
version: '3.9'

services:
  db:
    image: postgres:15
    container_name: postgres-container
    environment:
      POSTGRES_USER: user
      POSTGRES_PASSWORD: password
      POSTGRES_DB: mydb
    ports:
      - "5432:5432"
    volumes:
      - postgres_data:/var/lib/postgresql/data
```

```
php:
  build:
    context: ./php
  container_name: php-container
  volumes:
    - ./app:/var/www/html
  ports:
    - "8080:80"
  depends_on:
    - db

volumes:
  postgres_data:
```

Passo 3: Creare il Dockerfile per PHP

Il Dockerfile nella cartella php/ configurerà l'ambiente PHP. Ecco un esempio:

```
FROM php:8.2-apache

RUN docker-php-ext-install pdo pdo_pgsql

COPY . /var/www/html
```

Passo 4: Creare un File PHP

Nella cartella app/, crea un file index.php con il seguente contenuto:

```
<?php
$host = 'db';
$dbname = 'mydb';
$user = 'user';
$password = 'password';

try {
    $dsn = "pgsql:host=$host;port=5432;dbname=$dbname;";
    $pdo = new PDO($dsn, $user, $password);
    echo "Connessione al database riuscita!";
} catch (PDOException $e) {
    echo "Errore: " . $e->getMessage();
}
?>
```

Passo 5: Avviare i Servizi

Ora puoi avviare i container con il comando:

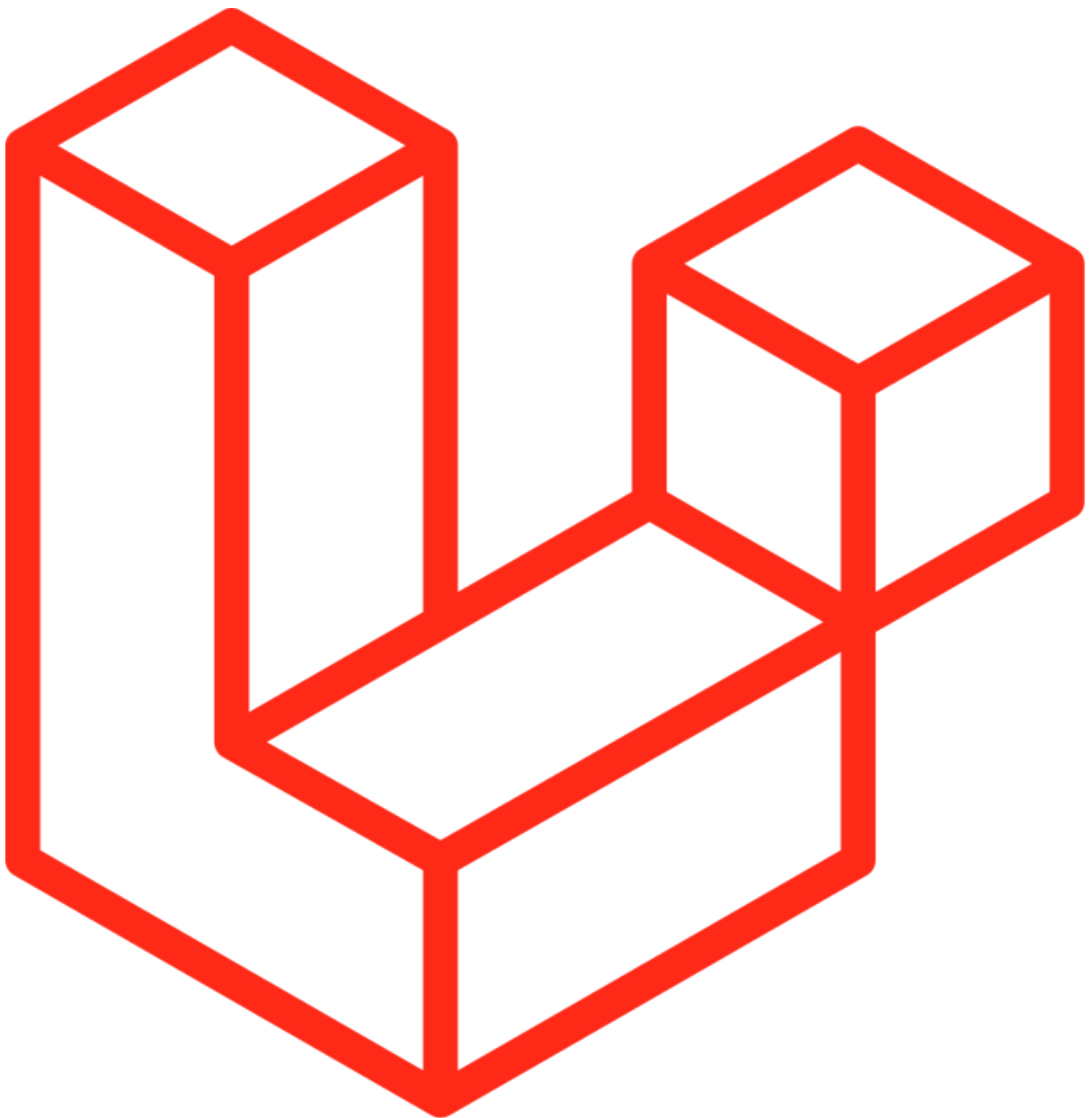
```
docker compose up -d
```

Accedi al tuo browser e visita <http://localhost:8080> per verificare la connessione al database.

Conclusione

Hai configurato con successo PostgreSQL e PHP utilizzando Docker e Docker Compose. Questo approccio è utile per sviluppare e testare applicazioni in un ambiente isolato.

Applicazioni Correlate

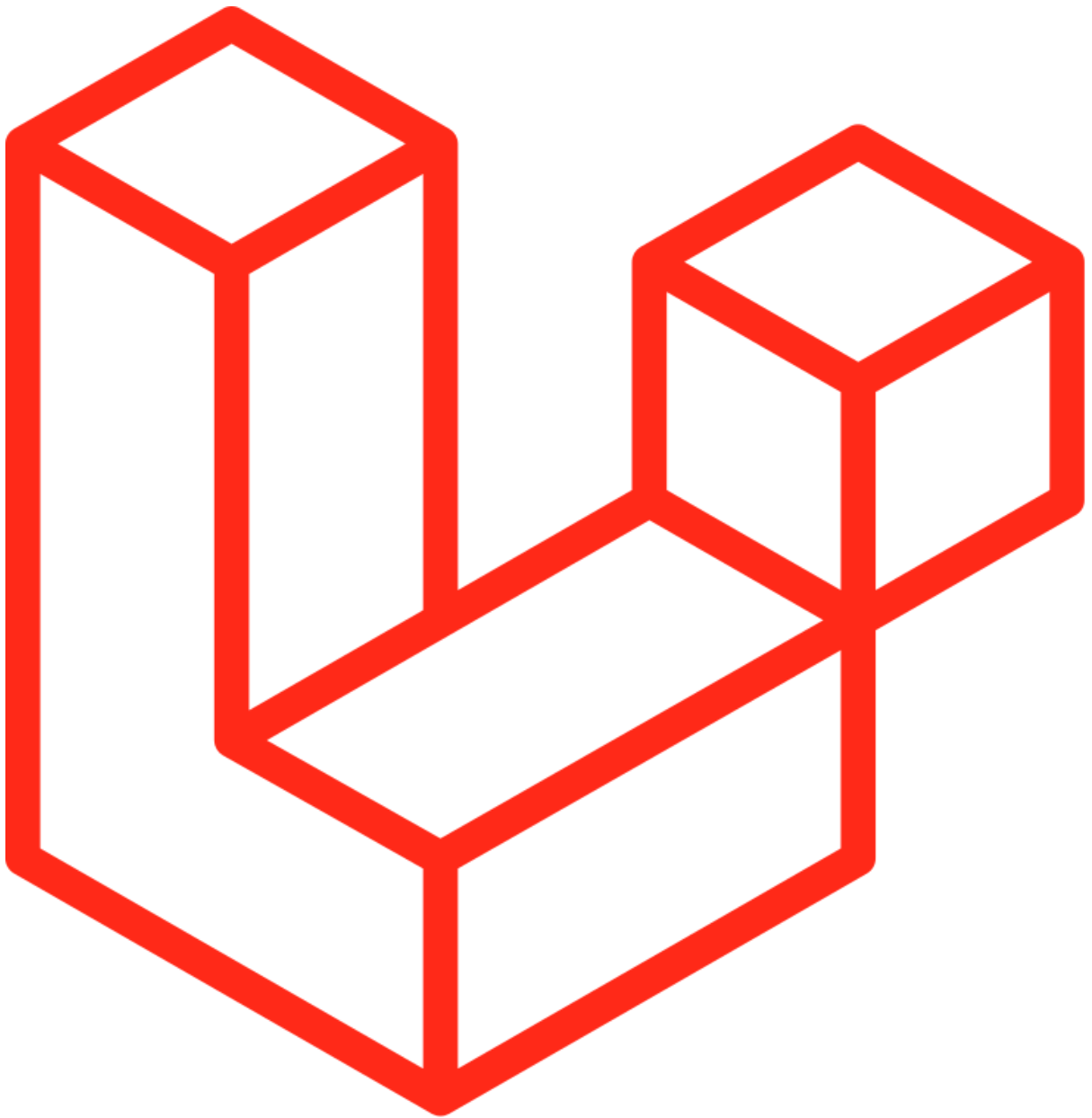


•

Laravel Placeholder Image

Applicazione in Laravel per creare immagini segnaposto.

Docker Docker Compose Composer PHP Laravel JavaScript

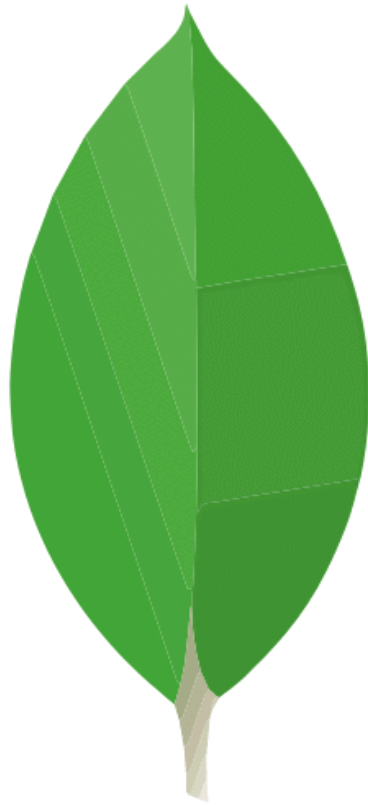


•

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.
DockerDocker ComposeComposerPHPMongoDB