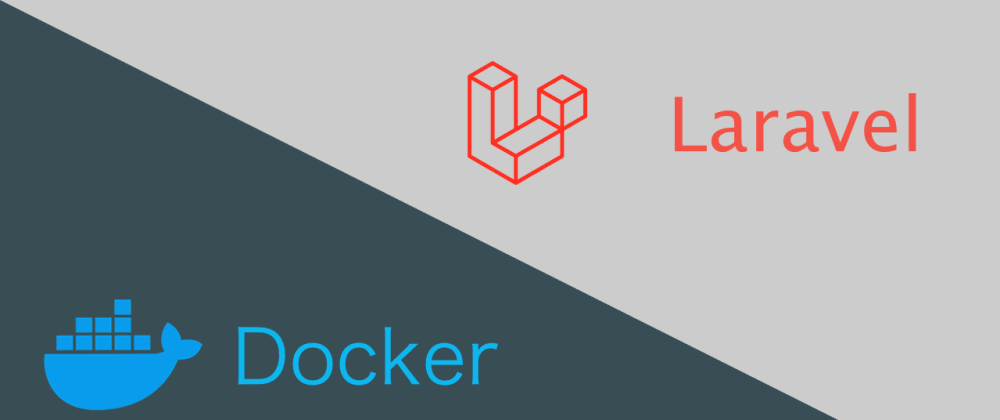
[](https://res.cloudinary.com/practicaldev/image/fetch/s--A5eu5cwL--/c_imagga_scale,f_auto,fl_progressive,h_420,q_auto,w_1000/https:/dev-to-uploads.s3.amazonaws.com/uploads/articles/is1yo8p671zivo9kkdd5.png)

[ucan_lab](https://dev.to/ucan_lab)

[ucan\_lab](https://dev.to/ucan_lab)

Posted on 7 thg 5, 2021 • Updated on 9 thg 5, 2021

102

**How to install and set up Laravel's local development environment with Docker Compose**

[#php](https://dev.to/t/php)[#laravel](https://dev.to/t/laravel)[#docker](https://dev.to/t/docker)

**Introduction**

This article explains how to build a LEMP environment for PHP, nginx, and MySQL with Docker and Docker Compose.

**Prerequisites**

* Docker
* Git

$ docker -v

Docker version 20.10.5, build 55c4c88

$ git --version

git version 2.31.1

**How to enable Docker Content Trust (DCT)**

DCT is a security feature that protects your Docker image from spoofing and tampering.  
It works automatically when the push, build, create, pull, or run command is executed.

$ echo export DOCKER\_CONTENT\_TRUST=1 >> ~/.zshrc

* <https://docs.docker.com/engine/security/trust>

**Repository**

* <https://github.com/ucan-lab/docker-laravel>

**Container Structures**

It has a three-tier structure consisting of an application server, a web server, and a database server.

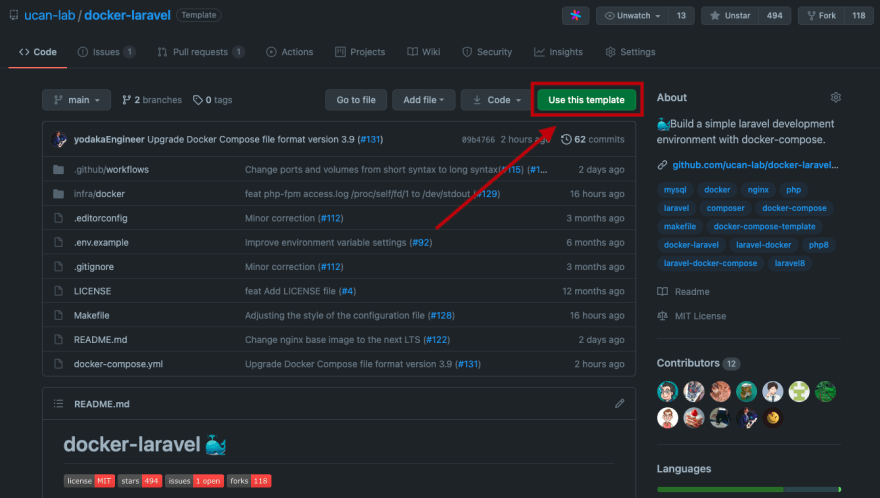
├── app - service running php-fpm

├── web - service running nginx

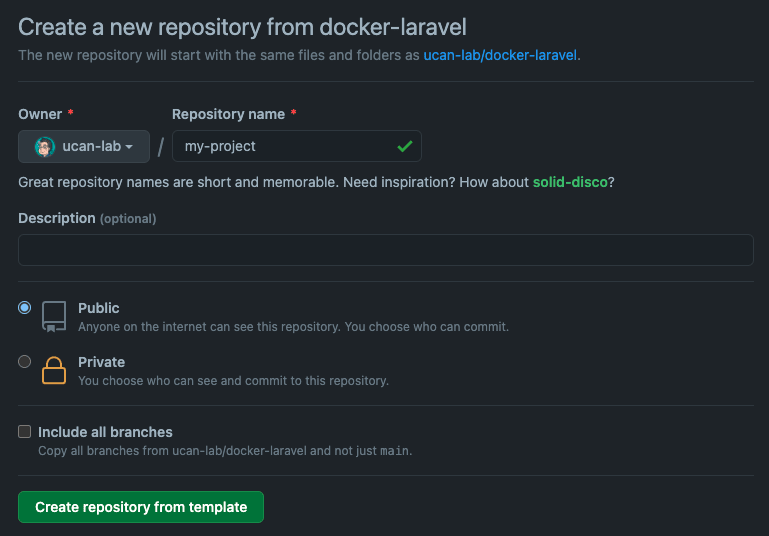
└── db - service running mysql

**How to use a new project**

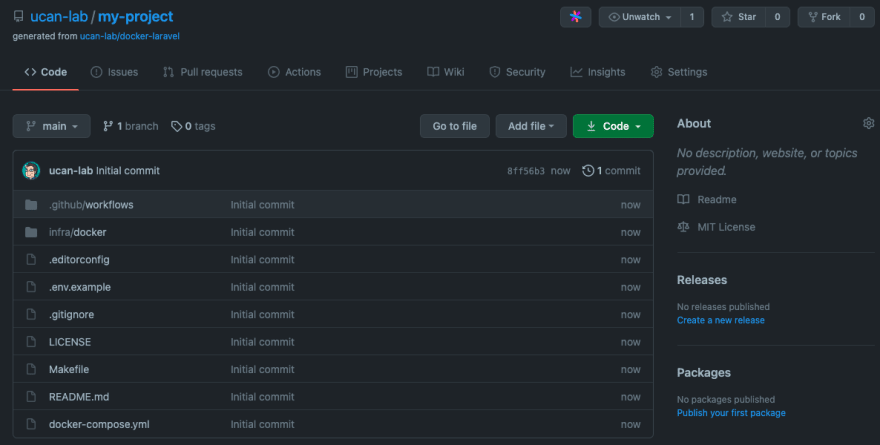
**Step1. Create a new repository from template**

[](https://res.cloudinary.com/practicaldev/image/fetch/s--oQLjQndx--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/i.imgur.com/1UuNYLH.png)

Click Use this template.

[](https://res.cloudinary.com/practicaldev/image/fetch/s--evry_gAe--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/i.imgur.com/Dc7n5d6.png)

Create a repository for my-project.

[](https://res.cloudinary.com/practicaldev/image/fetch/s--_H3m64_u--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/i.imgur.com/btHLRsF.png)

If you use the template repository, all commits will be combined into one.

**Step2. Clone your GitHub repository**

$ git clone git@github.com:ucan-lab/my-project.git

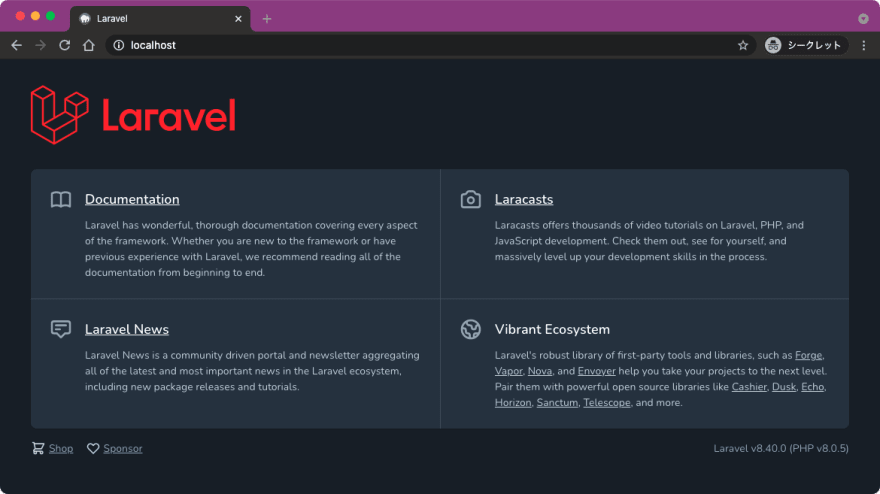
$ cd my-project

**Step3. Create a Laravel Project**

$ make create-project

**Step4. Show the Laravel Welcome page**

[http://localhost](http://localhost/)

[](https://res.cloudinary.com/practicaldev/image/fetch/s--qDKoco9B--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/i.imgur.com/6nS45tR.png)

**Step5. First commit and push**

$ git add .

$ git commit -m "laravel install"

$ git push -u origin HEAD

**How to use an existing project**

**(Optional) Delete the local repository**

$ docker-compose down -v --rmi all

$ cd ..

$ rm -rf my-project

**Step1. Clone your GitHub repository**

$ git clone git@github.com:ucan-lab/my-project.git

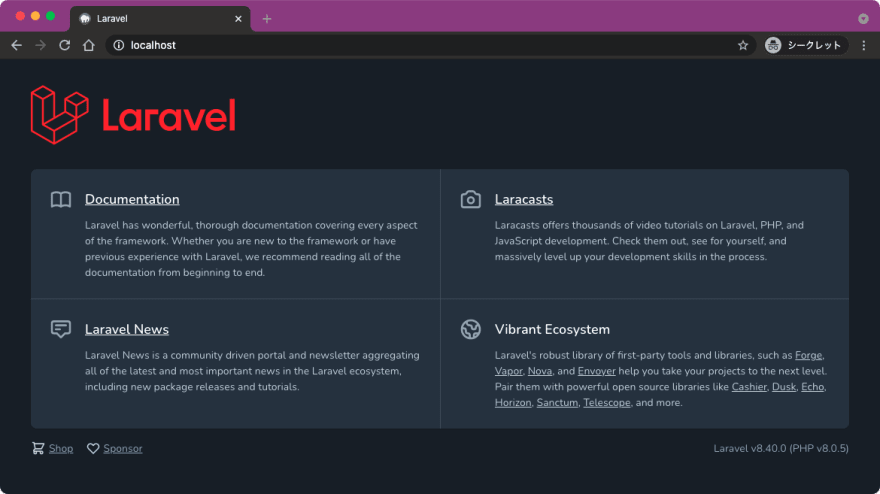
$ cd my-project

**Step2. Initialize your local environment**

$ make init

**Step3. Show Laravel Welcome page**

[http://localhost](http://localhost/)

[](https://res.cloudinary.com/practicaldev/image/fetch/s--qDKoco9B--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/i.imgur.com/6nS45tR.png)

**If you want to specify the version of Laravel**

Rewrite Makefile and execute.

laravel-install:

docker-compose exec app composer create-project --prefer-dist "laravel/laravel=6.\*" .

**Remarks**

**docker-compose.yml**

version: "3.9"

volumes:

php-fpm-socket:

db-store:

services:

app:

build:

context: .

dockerfile: ./infra/docker/php/Dockerfile

volumes:

- type: volume

source: php-fpm-socket

target: /var/run/php-fpm

volume:

nocopy: true

- type: bind

source: ./backend

target: /work/backend

environment:

- DB\_CONNECTION=mysql

- DB\_HOST=db

- DB\_PORT=3306

- DB\_DATABASE=${DB\_NAME:-laravel\_local}

- DB\_USERNAME=${DB\_USER:-phper}

- DB\_PASSWORD=${DB\_PASS:-secret}

web:

build:

context: .

dockerfile: ./infra/docker/nginx/Dockerfile

ports:

- target: 80

published: ${WEB\_PORT:-80}

protocol: tcp

mode: host

volumes:

- type: volume

source: php-fpm-socket

target: /var/run/php-fpm

volume:

nocopy: true

- type: bind

source: ./backend

target: /work/backend

db:

build:

context: .

dockerfile: ./infra/docker/mysql/Dockerfile

ports:

- target: 3306

published: ${DB\_PORT:-3306}

protocol: tcp

mode: host

volumes:

- type: volume

source: db-store

target: /var/lib/mysql

volume:

nocopy: true

environment:

- MYSQL\_DATABASE=${DB\_NAME:-laravel\_local}

- MYSQL\_USER=${DB\_USER:-phper}

- MYSQL\_PASSWORD=${DB\_PASS:-secret}

- MYSQL\_ROOT\_PASSWORD=${DB\_PASS:-secret}

**version**

Docker Compose file version

* <https://docs.docker.com/compose/compose-file>
* <https://docs.docker.com/compose/compose-file/compose-versioning/#version-3>

Note: When specifying the Compose file version to use, make sure to specify both the major and minor numbers. If no minor version is given, 0 is used by default and not the latest minor version. As a result, features added in later versions will not be supported. For example:

version: "3"

version: "3"

version: "3.0"

**volumes**

* <https://docs.docker.com/compose/compose-file/compose-file-v3/#volumes>

Define named volumes with the top-level volumes key to reuse volumes across multiple services.

**services.\*.build**

* <https://docs.docker.com/compose/compose-file/compose-file-v3/#build>

**services.\*.ports**

* <https://docs.docker.com/compose/compose-file/compose-file-v3/#long-syntax-1>

**services.\*.volumes**

* <https://docs.docker.com/compose/compose-file/compose-file-v3/#long-syntax-3>

**services.\*.environment**

* <https://docs.docker.com/compose/compose-file/compose-file-v3/#environment>

**app service**

**./infra/docker/php/Dockerfile**

FROM php:8.0-fpm-buster

LABEL maintainer="ucan-lab <yes@u-can.pro>"

SHELL ["/bin/bash", "-oeux", "pipefail", "-c"]

# timezone environment

ENV TZ=UTC \

# locale

LANG=en\_US.UTF-8 \

LANGUAGE=en\_US:en \

LC\_ALL=en\_US.UTF-8 \

# composer environment

COMPOSER\_ALLOW\_SUPERUSER=1 \

COMPOSER\_HOME=/composer

COPY --from=composer:2.0 /usr/bin/composer /usr/bin/composer

RUN apt-get update && \

apt-get -y install git libicu-dev libonig-dev libzip-dev unzip locales && \

apt-get clean && \

rm -rf /var/lib/apt/lists/\* && \

locale-gen en\_US.UTF-8 && \

localedef -f UTF-8 -i en\_US en\_US.UTF-8 && \

mkdir /var/run/php-fpm && \

docker-php-ext-install intl pdo\_mysql zip bcmath && \

composer config -g process-timeout 3600 && \

composer config -g repos.packagist composer https://packagist.org

COPY ./infra/docker/php/php-fpm.d/zzz-www.conf /usr/local/etc/php-fpm.d/zzz-www.conf

COPY ./infra/docker/php/php.ini /usr/local/etc/php/php.ini

WORKDIR /work/backend

* <https://hub.docker.com/_/php>
  + FROM <image>:<verion>-<variant>-<os>
* <https://laravel.com/docs/8.x/deployment#server-requirements>
  + PHP >= 7.3
  + BCMath PHP Extension
  + Ctype PHP Extension
  + Fileinfo PHP Extension
  + JSON PHP Extension
  + Mbstring PHP Extension
  + OpenSSL PHP Extension
  + PDO PHP Extension
  + Tokenizer PHP Extension
  + XML PHP Extension

**./infra/docker/php/php.ini**

zend.exception\_ignore\_args = off

expose\_php = on

max\_execution\_time = 30

max\_input\_vars = 1000

upload\_max\_filesize = 64M

post\_max\_size = 128M

memory\_limit = 256M

error\_reporting = E\_ALL

display\_errors = on

display\_startup\_errors = on

log\_errors = on

error\_log = /dev/stderr

default\_charset = UTF-8

[Date]

date.timezone = ${TZ}

[mysqlnd]

mysqlnd.collect\_memory\_statistics = on

[Assertion]

zend.assertions = 1

[mbstring]

mbstring.language = Neutral

**./infra/docker/php/php-fpm.d/zzz-**[**www.conf**](http://www.conf/)

[www]

listen = /var/run/php-fpm/php-fpm.sock

listen.owner = www-data

listen.group = www-data

listen.mode = 0666

access.log = /dev/stdout

**web service**

**./infra/docker/nginx/Dockerfile**

FROM node:16-alpine as node

FROM nginx:1.20-alpine

LABEL maintainer="ucan-lab <yes@u-can.pro>"

SHELL ["/bin/ash", "-oeux", "pipefail", "-c"]

ENV TZ=UTC

RUN apk update && \

apk add --update --no-cache --virtual=.build-dependencies g++

# node command

COPY --from=node /usr/local/bin /usr/local/bin

# npm command

COPY --from=node /usr/local/lib /usr/local/lib

# yarn command

COPY --from=node /opt /opt

# nginx config file

COPY ./infra/docker/nginx/\*.conf /etc/nginx/conf.d/

WORKDIR /work/backend

* <https://hub.docker.com/_/node>
* <https://hub.docker.com/_/nginx>

**./infra/docker/nginx/default.conf**

access\_log /dev/stdout main;

error\_log /dev/stderr warn;

server {

listen 80;

root /work/backend/public;

add\_header X-Frame-Options "SAMEORIGIN";

add\_header X-XSS-Protection "1; mode=block";

add\_header X-Content-Type-Options "nosniff";

index index.html index.htm index.php;

charset utf-8;

location / {

try\_files $uri $uri/ /index.php?$query\_string;

}

location = /favicon.ico { access\_log off; log\_not\_found off; }

location = /robots.txt { access\_log off; log\_not\_found off; }

error\_page 404 /index.php;

location ~ \.php$ {

fastcgi\_pass unix:/var/run/php-fpm/php-fpm.sock;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME $realpath\_root$fastcgi\_script\_name;

include fastcgi\_params;

}

location ~ /\.(?!well-known).\* {

deny all;

}

}

* <https://laravel.com/docs/8.x/deployment#nginx>

**db service**

**./infra/docker/mysql/Dockerfile**

FROM mysql/mysql-server:8.0

LABEL maintainer="ucan-lab <yes@u-can.pro>"

ENV TZ=UTC

COPY ./infra/docker/mysql/my.cnf /etc/my.cnf

* <https://hub.docker.com/r/mysql/mysql-server>

**./infra/docker/mysql/my.cnf**

[mysqld]

# default

skip-host-cache

skip-name-resolve

datadir = /var/lib/mysql

socket = /var/lib/mysql/mysql.sock

secure-file-priv = /var/lib/mysql-files

user = mysql

pid-file = /var/run/mysqld/mysqld.pid

# character set / collation

character\_set\_server = utf8mb4

collation\_server = utf8mb4\_0900\_ai\_ci

# timezone

default-time-zone = SYSTEM

log\_timestamps = SYSTEM

# Error Log

log-error = mysql-error.log

# Slow Query Log

slow\_query\_log = 1

slow\_query\_log\_file = mysql-slow.log

long\_query\_time = 1.0

log\_queries\_not\_using\_indexes = 0

# General Log

general\_log = 1

general\_log\_file = mysql-general.log

[mysql]

default-character-set = utf8mb4

[client]

default-character-set = utf8mb4

**Top comments (3)**

Subscribe

Top of Form





Bottom of Form

Tuan Anh Nguyen

•[29 thg 3 '22](https://dev.to/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf#comment-1n586)

[@ucan\_lab](https://dev.to/ucan_lab)  
Hi, I'm learning docker and I clone your repo to try it but I have an issue with app source code permission. Docker create the laravel app with the files/directories owner is root and on the host machine, I'm using a non-root user so I can't edit files or directories. What should i do?

Like[Reply](https://dev.to/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf#/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf/comments/new/1n586)

Swapnil V. Patil

•[2 thg 6](https://dev.to/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf#comment-27416)

[@ucan\_lab](https://dev.to/ucan_lab) Time to update the article to match with the latests updates?

1 likeLike[Reply](https://dev.to/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf#/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf/comments/new/27416)

calvez

•[26 thg 1 '22](https://dev.to/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf#comment-1lhf2)

good stuff

1 likeLike[Reply](https://dev.to/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf#/ucan_lab/how-to-install-and-set-up-laravel-s-local-development-environment-with-docker-compose-5bcf/comments/new/1lhf2)

[Code of Conduct](https://dev.to/code-of-conduct) • [Report abuse](https://dev.to/report-abuse)

[[](https://dev.to/platformsh)](https://dev.to/platformsh" \t "_blank)

[Platform.sh](https://dev.to/platformsh" \t "_blank)

PROMOTED

**[Experience effortless application deployment 🚀](https://platform.sh/try-it-now/?utm_source=devto&utm_medium=paid_social&utm_campaign=Dev.to-Free-trial-2023)**

**Deploy your apps seamlessly on an all-in-one PaaS.**  
🚀 Flexible, automated infrastructure provisioning.  
🎯 Multicloud and multistack.  
👾 Safe, secure and reliable around-the-clock.  
👉 Get a 30-day free trial to build and deploy your way.[Claim your free trial](https://platform.sh/try-it-now/?utm_source=devto&utm_medium=paid_social&utm_campaign=Dev.to-Free-trial-2023)

**Read next**

**[Convert PHP Project to Android App](https://dev.to/atreep/convert-php-project-to-android-app-56fb)**

[Treep - Jul 27](https://dev.to/atreep/convert-php-project-to-android-app-56fb)

**[What is PHP's Future with PHP 8?](https://dev.to/devteam/what-is-phps-future-with-php-8-2615)**

[dev.to staff - Jul 26](https://dev.to/devteam/what-is-phps-future-with-php-8-2615)

**[Boost Your Laravel App Speed with These Optimization Tips](https://dev.to/kinsta/boost-your-laravel-app-speed-with-these-optimization-tips-48b6)**

[Kinsta - Jul 26](https://dev.to/kinsta/boost-your-laravel-app-speed-with-these-optimization-tips-48b6)

**[Laravel 101: Everything You Need to Know to Get Started with This Popular PHP Framework](https://dev.to/kazemmdev/laravel-101-everything-you-need-to-know-to-get-started-with-this-popular-php-framework-2gal)**

[Kazem - Jul 26](https://dev.to/kazemmdev/laravel-101-everything-you-need-to-know-to-get-started-with-this-popular-php-framework-2gal)