

# Dockerization a Laravel App

Last updated by | shakibesaib | Sep 27, 2023 at 3:13 PM GMT+6

## Create Dockerfile

Navigate to the root directory of your Laravel application. Then create a Docker file with out any extension `Dockerfile` and a apache virtual host configuration file like `laravel-world.conf`

Open Dockerfile file :

```
FROM php:8.1-apache
USER root

RUN apt-get update && apt-get upgrade -y && apt-get install -y \
    procps \
    nano \
    git \
    unzip \
    libicu-dev \
    zlib1g-dev \
    libxml2 \
    libxml2-dev \
    libreadline-dev \
    supervisor \
    cron \
    sudo \
    libzip-dev \
    && docker-php-ext-configure pdo_mysql --with-pdo-mysql=mysqlnd \
    && docker-php-ext-configure intl \
    && docker-php-ext-install \
        pdo_mysql \
        sockets \
        intl \
        opcache \
        zip \
    && rm -rf /tmp/* \
    && rm -rf /var/lib/apt/lists/* \
    && apt-get clean

RUN a2dissite 000-default.conf
COPY laraveldock.conf /etc/apache2/sites-available/laraveldock.conf
RUN a2ensite laraveldock.conf
RUN a2enmod rewrite
RUN chown -R www-data:www-data /var/www/html
RUN curl -sS https://getcomposer.org/installer | php -- --install-dir=/usr/local/bin --filename=composer
WORKDIR /var/www/html
COPY . .
RUN chown -R www-data:www-data /var/www/html/storage
RUN chmod -R 775 /var/www/html/storage
RUN composer install
USER root
```

Open laravel-world.conf file:

```
<VirtualHost *:80>
ServerName docker23.local
DocumentRoot /var/www/html/public
<Directory /var/www/html/public>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

## Build and Run

To build the image:

```
sudo docker build -t laravel-world:t1
```

here laravel-world is the image name and t1 is tag.

Run the image:

```
sudo docker run -d -p 8097:80 --name laravel-container laravel-world:t1
```

here, specified a port 8097 on localhost, laravel container is the container name and the image we want to run.

## Push the Image to Docker hub

1. Tag Docker Image:

```
docker tag laravel-world:t1 shakibesaib/laravel-world:t1
```

2. Login to Docker Hub:

```
docker login
```

Provide login credential of Docker hub account

3. Push the Image to Docker Hub:

```
docker push shakibesaib/laravel-world:t1
```

## Deploy the container in GCP VM

1. Create a vm in gcp. Ref: [https://dev.azure.com/fronturetech/Fronture ERP/\\_wiki/wikis/Fronture-ERP.wiki/225/Create-VM-Using-GCP-portal\(ubuntu\).](https://dev.azure.com/fronturetech/Fronture ERP/_wiki/wikis/Fronture-ERP.wiki/225/Create-VM-Using-GCP-portal(ubuntu).)

2. Connect to SSH

VM instances
CREATE INSTANCE
IMPORT VM
REFRESH
LEARN

INSTANCES
OBSERVABILITY
INSTANCE SCHEDULES

VM instances

Filter Enter property name or value

| <input type="checkbox"/> | Status | Name ↑                        | Zone          | Recommendations | In use by | Internal IP        | External IP        | Connect |
|--------------------------|--------|-------------------------------|---------------|-----------------|-----------|--------------------|--------------------|---------|
| <input type="checkbox"/> | ✓      | <a href="#">laravel-world</a> | asia-south1-c |                 |           | 10.160.0.13 (nic0) | 34.93.91.48 (nic0) | SSH     |

Related actions
HIDE

Explore Backup and DR NEW
Back up your VMs and set up disaster recovery

Monitor VMs
View outlier VMs across metrics like CPU and network

Explore VM logs
View, search, analyse and download VM instance logs

Set up firewall rules
Control traffic to and from a VM instance

Patch management
Schedule patch updates and view patch compliance on VM instances

Load balance between VMs
Set up load balancing for your applications as your traffic and users grow

```

ssh.cloud.google.com/v2/ssh/projects/devopslearning-396112/zones/asia-south1-c/instances/laravel-world?authuser=0&hl=...
ssh.cloud.google.com/v2/ssh/projects/devopslearning-396112/zones/asia-south1-c/instances/laravel-world?authuser=0&hl=en_GB&...

SSH-in-browser  UPLOAD FILE  DOWNLOAD FILE  !  ⌨  ⚙

Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1042-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Sep 27 07:33:11 UTC 2023

System load:  0.42           Processes:            108
Usage of /:   19.4% of 9.51GB Users logged in:        0
Memory usage: 24%           IPv4 address for ens4: 10.160.0.13
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

shakibesaib@laravel-world:~$ sudo apt update

```

3. Install docker on vm. <https://www.digialocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-20-04>

4. Pull the docker image from Docker Hub  
 sudo docker pull shakibesaib/laravel-world:t1

5. Login to Docker Hub.

6. Run the docker image.  
 sudo docker run -d -p 80:80 --name laravel-container shakibesaib/laravel-world:t1

7. Check container.

Docker ps

```

shakibesaib@laravel-world:~$ sudo docker ps
CONTAINER ID   IMAGE                                NAMES                COMMAND                CREATED        STATUS        PORTS
a2e25c3f3a3a   shakibesaib/laravel-world:t1       "docker-php-entrypoi..."  9 seconds ago   Up 7 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp
laravel-container

```

8. Hit the external ip address and the application would be running.

VM instances [CREATE INSTANCE](#) [IMPORT VM](#) [REFRESH](#) [LEARN](#)

**INSTANCES** OBSERVABILITY INSTANCE SCHEDULES

VM instances

Filter Enter property name or value

| <input type="checkbox"/> | Status | Name ↑                        | Zone          | Recommendations | In use by | Internal IP                        | External IP  | Connect |
|--------------------------|--------|-------------------------------|---------------|-----------------|-----------|------------------------------------|--|---------|
| <input type="checkbox"/> | ✓      | <a href="#">laravel-world</a> | asia-south1-c |                 |           | 10.160.0.13 <a href="#">(nic0)</a> | <a href="#">34.93.91.48</a> <a href="#">(nic0)</a> | SSH ▾   |

**Related actions** [HIDE](#)

**Explore Backup and DR** NEW  
 Back up your VMs and set up disaster recovery

**Monitor VMs**  
 View outlier VMs across metrics like CPU and network

**Explore VM logs**  
 View, search, analyse and download VM instance logs

**Set up firewall rules**  
 Control traffic to and from a VM instance

**Patch management**  
 Schedule patch updates and view patch compliance on VM instances

**Load balance between VMs**   
 Set up load balancing for your applications as your traffic and users grow

Troubleshoot:

### 1. 403 Forbidden

- . Make sure www-data (apache) has enough permission within the docker container to read write and execute.
- . In host configuration make sure document root is perfectly set. `/var/www/html/public` .

### 2. 500 Internal server error

- . Make The .env file is available at the document root.

3. While building the docker image, make sure all the directories are defined perfectly and all the permission has given with in the docker file.