# **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 <code>Dockerfile</code> and a apache virtual host configuration file like <code>laravel-world.conf</code>

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/list/apt/* \
    && 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=compc
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**

Tag Docker Image:

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

Login to Docker Hub:

docker login

Provide login credential of Docker hub account

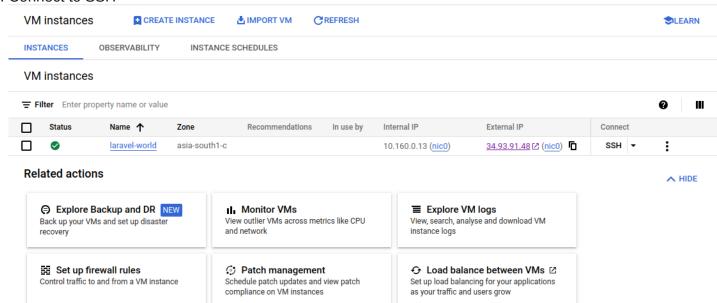
Push the Image to Docker Hub:

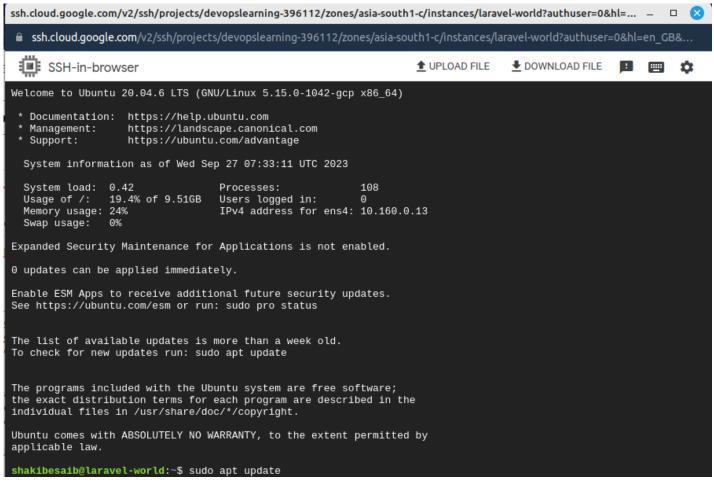
docker push shakibesaib/laravel-world:t1

## Deploy the container in GCP VM

1. Create a vm in gcp. Ref: <a href="https://dev.azure.com/fronturetech/FrontureERP/\_wiki/wikis/Fronture-ERP.wiki/225/Create-VM-Using-GCP-portal(ubunto)">https://dev.azure.com/fronturetech/FrontureERP/\_wiki/wikis/Fronture-ERP.wiki/225/Create-VM-Using-GCP-portal(ubunto)</a>.

#### 2. Connect to SSH





- 3. Install docker on vm.https://www.digitalocean.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
- Check container.

Docker ps

```
shakibesaib@laravel-world:~$ sudo docker ps

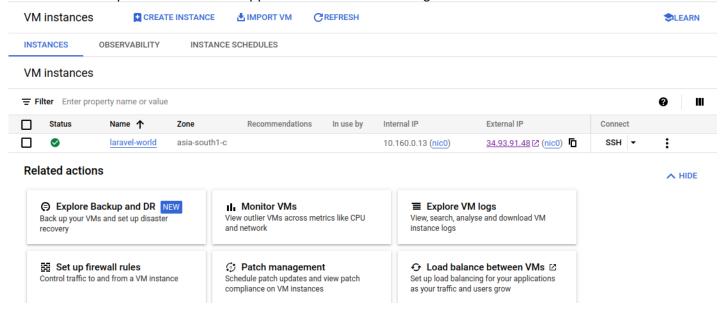
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS

NAMES

a2e25c3f3a3a shakibesaib/laravel-world:t1 "docker-php-entrypoi..." 9 seconds ago Up 7 seconds 0.0.0.0:8

0->80/tcp_:::80->80/tcp_ laravel-container
```

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



## 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.