Php App Fine tuning and Deployment(Docker, Php-Fpm, Nginx, Supervisor, Cronjobs)

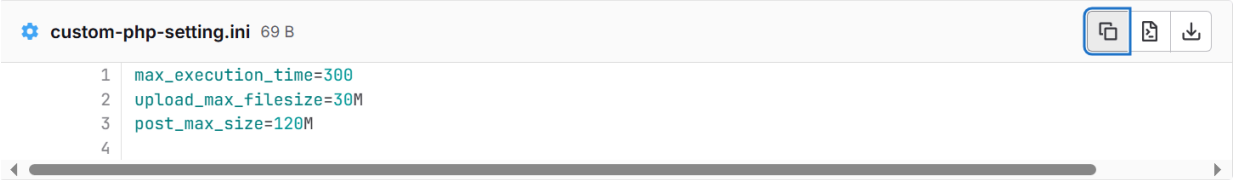
Step 1: Create a custom PHP settings file:

Touch custom-php-setting.ini

max_execution_time=300

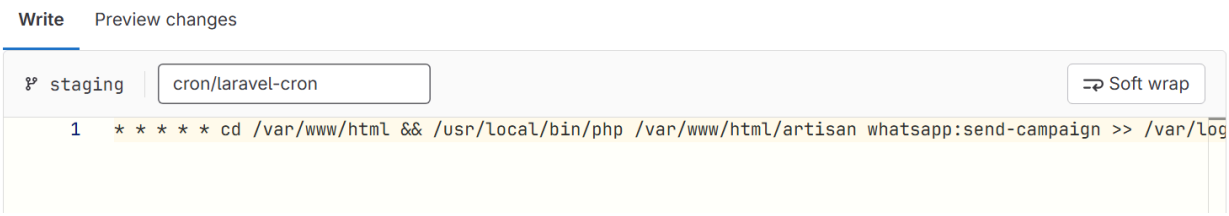
upload_max_filesize=30M

post_max_size=120M



Step 2: Create a custom File for cronjobs in cron folder:

Touch laravel-cron



Step 3: Create a custom File for php-fpm

Touch custom-fpm-setting.conf

[www]

; Unix user/group of the child processes.

user = www-data

group = www-data

; The address on which to accept FastCGI requests.

listen = 127.0.0.1:9000

; Process management settings

pm = dynamic

pm.max_children = 20

pm.start_servers = 5

pm.min_spare_servers = 3

pm.max_spare_servers = 10

Step4: Create a supervisor configuration file(php-artisan.conf inside supervisor folder):

php-artisan.conf 2.00 KiB

```
[supervisord]
nodaemon=true
logfile=/var/log/supervisord.log
pidfile=/var/run/supervisord.pid
childlogdir=/var/log/
user=root

[program:php-fpm]
process_name=%(program_name)s
command=php-fpm --nodaemonize
autostart=true
autorestart=true
user=root
redirect_stderr=true
stdout_logfile=/var/log/php-fpm.log
priority=5
```

```
[program:nginx]
process_name=%(program_name)s
command= /bin/sh -c "sleep 3 && nginx -g 'daemon off;'"
autostart=true
autorestart=true
user=root
redirect_stderr=true
stdout_logfile=/var/log/nginx.log
priority=10

[program:php-horizon]
process_name=%(program_name)s
command=/usr/local/bin/php /var/www/html/artisan horizon
autostart=true
autorestart=true
user=root
redirect_stderr=true
stdout_logfile=/var/log/horizon.log
priority=20

[program:kafka-message-event]
process_name=%(program_name)s
command=/usr/local/bin/php /var/www/html/artisan kafka:message-event
autostart=true
autorestart=true
user=root
redirect_stderr=true
stdout_logfile=/var/log/kafka-message-event.log
priority=20
```

```
[program:kafka-chat-event]
process_name=%(program_name)s
command=/usr/local/bin/php /var/www/html/artisan kafka:chat-event
autostart=true
autorestart=true
user=root
redirect_stderr=true
stdout_logfile=/var/log/kafka-chat-event.log
priority=20

[program:kafka-integration-event]
process_name=%(program_name)s
command=/usr/local/bin/php /var/www/html/artisan kafka:integration-event
autostart=true
autorestart=true
user=root
redirect_stderr=true
stdout_logfile=/var/log/kafka-integration-event.log
priority=20


[program:laravel-cron]
command=crond -f -L /var/log/cron.log crond
autostart=true
autorestart=true
stdout_logfile=/var/log/cron.stdout.log
stderr_logfile=/var/log/cron.stderr.log
user=root
priority=30
```

Step5: create nginx configuration file:

Touch nginx.conf

nginx.conf 1.58 KiB

```
user  nginx;
worker_processes  auto;

error_log /var/log/nginx/error.log warn;
pid        /var/run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include       /etc/nginx/mime.types;
    default_type  application/octet-stream;
    sendfile        on;
    keepalive_timeout  65;

    client_max_body_size 30M;

    server {
        listen 80;
        root /var/www/html/public;
        index index.php index.html;

        add_header 'Access-Control-Allow-Origin' '*' always;
        add_header 'Access-Control-Allow-Methods' 'GET, POST, OPTIONS' always;
        add_header 'Access-Control-Allow-Headers' 'Content-Type' always;
        add_header Strict-Transport-Security "max-age=31536000; includeSubDomains" always;
        add_header Content-Security-Policy "frame-ancestors https://*.mybigcommerce.com https://*.bigcommerce.
```

```
        location / {
            try_files $uri $uri/ /index.php?$query_string;
        }

        location ~ \.php$ {
            include fastcgi_params;
            fastcgi_pass 127.0.0.1:9000;
            fastcgi_index index.php;
            fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
            fastcgi_param DOCUMENT_ROOT $document_root;
            fastcgi_read_timeout 300;



            add_header X-Frame-Options "SAMEORIGIN" always;
            add_header X-Content-Type-Options "nosniff" always;
            add_header X-XSS-Protection "1; mode=block" always;

        }

        location ~ /\.ht {
            deny all;
        }
    }
}
```

Step6: Create php-fpm dockerfile (Dockerfile copy all above files)

Dockerfile_staging 1.98 KiB

```
# Base image
FROM php:8.2-fpm-alpine

# Set working directory
WORKDIR /var/www/html

# Install system dependencies + PHP extensions
RUN apk add --no-cache \
    nginx supervisor gcc g++ make  curl zip unzip git autoconf \
    libpng-dev libjpeg-turbo-dev libwebp-dev freetype-dev librdkafka-dev \
    oniguruma-dev libxml2-dev icu-dev zlib-dev libzip-dev \
    && docker-php-ext-configure gd \
        --with-freetype \
        --with-jpeg \
        --with-webp \
    && docker-php-ext-install -j$(nproc) \
        gd pdo pdo_mysql mbstring xml intl zip bcmath opcache pcntl \
    && pecl install rdkafka \
    && docker-php-ext-enable rdkafka

# Install Composer
COPY --from=composer:2 /usr/bin/composer /usr/bin/composer
RUN ls -la

# Copy and activate development php.ini
RUN cp /usr/local/etc/php/php.ini-production /usr/local/etc/php/php.ini
COPY ./custom-php-setting.ini /usr/local/etc/php/conf.d/custom-php-setting.ini

# Copy your custom nginx configuration
COPY ./nginx.conf /etc/nginx/nginx.conf

# Copy Supervisor configuration file
COPY ./supervisor/php-artisan.conf /etc/supervisor/conf.d/php-artisan.conf

# Copy Laravel cron file
COPY ./cron/laravel-cron  /etc/crontabs/root
RUN chmod 0644 /etc/crontabs/root

# Copy application code
COPY . .

# Mark /var/www/html as safe for git
RUN git config --global --add safe.directory /var/www/html

# Install PHP dependencies
RUN composer install --no-interaction --prefer-dist --optimize-autoloader

# Cache Laravel config/routes/views
RUN php artisan config:clear && \
    php artisan migrate --force && \
    php artisan db:seed --class=PlansTableSeeder --force && \
    php artisan optimize

# Set correct permissions
RUN chown -R www-data:www-data /var/www/html && \
    chmod -R 775 /var/www/html/storage /var/www/html/bootstrap/cache


# Switch to non-root user
USER root

# Expose PHP-FPM port
EXPOSE 80

# Start PHP-FPM
CMD ["/usr/bin/supervisord", "-c", "/etc/supervisor/conf.d/php-artisan.conf"]
```

Step 7: Run Docker file

```
stage:
  stage: stage
  tags:
    - staging
  script:
    - |
    - cp /home/gitlab-runner/reflys-laravel-server/.env .
    - docker build -t reflys-laravel-server-stage -f Dockerfile_staging .
    - docker stop reflys-laravel-server-stage || true
    - docker rm reflys-laravel-server-stage  || true
    - docker run -d --restart unless-stopped --name reflys-laravel-server-stage -p 9000:80 reflys-laravel-serve
  only:
    - staging
```

Step 8: Test the Supervisor process inside Dockerfile

Supervisor is running everything

```
D   USER     TIME  COMMAND
  1 root      0:01 {supervisord} /usr/bin/python3 /usr/bin/supervisord -c /etc/supervisor/conf.d/php-artisan.conf
  8 root      0:00 php-fpm: master process (/usr/local/etc/php-fpm.conf)
  9 root      0:00 nginx: master process nginx -g daemon off;
 10 root      0:03 /usr/local/bin/php /var/www/html/artisan kafka:chat-event
 11 root      0:03 /usr/local/bin/php /var/www/html/artisan kafka:integration-event
 12 root      0:03 /usr/local/bin/php /var/www/html/artisan kafka:message-event
 14 root      0:03 /usr/local/bin/php /var/www/html/artisan kafka:user-event
 15 root      0:03 /usr/local/bin/php /var/www/html/artisan horizon
 16 root      0:00 crond -f -L /var/log/cron.log crond
 17 www-data  0:00 php-fpm: pool www
 18 www-data  0:00 php-fpm: pool www
 65 root      0:05 /usr/local/bin/php artisan horizon:supervisor ca5ee97d576c-YyaC:supervisor-1 redis --workers-name=defaul
 71 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 72 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 73 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 74 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 75 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 76 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 77 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 78 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 79 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 80 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 81 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 82 root      0:01 /usr/local/bin/php artisan horizon:work redis --name=default --supervisor=ca5ee97d576c-YyaC:supervisor-1
 86 nginx     0:00 nginx: worker process
 87 nginx     0:00 nginx: worker process
 88 nginx     0:00 nginx: worker process
 89 nginx     0:00 nginx: worker process
294 root      0:00 sh
300 root      0:00 [php]
```

Step 9: Edit Dockerfile and copy fpm setting

```
25  # Copy and activate production php.ini and custom php.ini
26  RUN cp /usr/local/etc/php/php.ini-production /usr/local/etc/php/php.ini
27  COPY ./custom-php-setting.ini /usr/local/etc/php/conf.d/custom-php-setting.ini
28
29  # Copy and activate custom fpm config
30  COPY ./custom-fpm-setting.conf /usr/local/etc/php-fpm.d/www.conf
31
```