

ИСП 1.1

Задача: провести инсталляцию программного обеспечения.

На основе материалов, опубликованных в примечаниях к заданию, установить необходимое ПО. В отчете поэтапно продемонстрировать процесс установки (в виде набора скриншотов) с ключевыми моментами установки.

Примечание 1: <https://disk.yandex.ru/i/bHPfXyufLTK0og>

Выполнение:

1. <https://github.com/getgrav/docker-grav>

Dockerfile:

```
FROM php:8.3-apache-bullseye
LABEL maintainer="Andy Miller <rhuk@getgrav.org> (@rhukster)"

# Enable Apache Rewrite + Expires Module
RUN a2enmod rewrite expires && \
    sed -i 's/ServerTokens OS/ServerTokens ProductOnly/g' \
    /etc/apache2/conf-available/security.conf

# Install dependencies
RUN apt-get update && apt-get install -y --no-install-recommends \
    unzip \
    libfreetype6-dev \
    libjpeg62-turbo-dev \
    libpng-dev \
    libyaml-dev \
    libzip4 \
    libzip-dev \
    zlib1g-dev \
    libicu-dev \
    g++ \
    git \
    cron \
    vim \
    && docker-php-ext-install opcache \
    && docker-php-ext-configure intl \
    && docker-php-ext-install intl \
    && docker-php-ext-configure gd --with-freetype --with-jpeg \
    && docker-php-ext-install -j$(nproc) gd \
    && docker-php-ext-install zip \
    && rm -rf /var/lib/apt/lists/*
```

```
# set recommended PHP.ini settings
# see https://secure.php.net/manual/en/opcache.installation.php
RUN { \
    echo 'opcache.memory_consumption=128'; \
    echo 'opcache.interned_strings_buffer=8'; \
    echo 'opcache.max_accelerated_files=4000'; \
    echo 'opcache.revalidate_freq=2'; \
    echo 'opcache.fast_shutdown=1'; \
    echo 'opcache.enable_cli=1'; \
    echo 'upload_max_filesize=128M'; \
    echo 'post_max_size=128M'; \
    echo 'expose_php=off'; \
} > /usr/local/etc/php/conf.d/php-recommended.ini

RUN pecl install apcu \
    && pecl install yaml-2.2.4 \
    && docker-php-ext-enable apcu yaml

# Set user to www-data
RUN chown www-data:www-data /var/www
USER www-data

# Define Grav specific version of Grav or use latest stable
ARG GRAV_VERSION=latest

# Install grav
WORKDIR /var/www
RUN curl -o grav-admin.zip -SL https://getgrav.org/download/core/grav-admin/${GRAV_VERSION} && \
    unzip grav-admin.zip && \
    mv -T /var/www/grav-admin /var/www/html && \
    rm grav-admin.zip

# Create cron job for Grav maintenance scripts
RUN (crontab -l; echo "* * * * * cd /var/www/html;usr/local/bin/php bin/grav scheduler 1>> /dev/null 2>&1") | crontab -

# Return to root user
USER root

# Copy init scripts
# COPY docker-entrypoint.sh /entrypoint.sh

# provide container inside image for data persistence
VOLUME ["/var/www/html"]
```

```
# ENTRYPOINT ["/entrypoint.sh"]
# CMD ["apache2-foreground"]
CMD ["sh", "-c", "cron && apache2-foreground"]
```

docker-compose.yml:

```
volumes:
  grav-data:
    driver: local
    driver_opts:
      type: none
      device: ./grav
      o: bind

services:
  grav:
    build: ./
    ports:
      - 8080:80
    volumes:
      - grav-data:/var/www/html
```

<https://learn.getgrav.org/17/basics/installation>

Option 1: Install from ZIP package

The easiest way to install Grav is to download the ZIP package and extract it:

1. Download the latest-and-greatest [Grav](#) or [Grav + Admin](#) package.
2. Extract the ZIP file in the [webroot](#) of your web server, e.g. `~/webroot/grav`

There are [Skeleton](#)-packages available, which include the core Grav system, sample pages, plugins, and configuration. They are a great way to get started; all you have to do is [download the Skeleton](#)-package you prefer, and follow the steps above.

You can also download any pre-packaged installation of a [tagged release](#) from [getgrav.org](#). Use the format

<https://getgrav.org/download/TYPE/PACKAGE/VERSION> .

- getgrav.org/download/core/grav/1.7.0 downloads Grav Core v1.7.0
- getgrav.org/download/core/grav/1.7.0-rc.10?testing=true downloads Grav Core v1.7.0-rc.10, a testing release
- getgrav.org/download/core/grav-admin/1.7.0 downloads Grav Core with the Admin plugin, at Core v1.7.0
- getgrav.org/download/core/grav-admin/1.7.0-rc.10?testing=true downloads Grav Core v1.7.0-rc.10 with the Admin plugin, a testing release
- getgrav.org/download/core/grav-update/1.7.0 downloads the update package for Grav Core
- getgrav.org/download/plugins/flex-objects-json/0.1.0 downloads the Flex Objects JSON plugin at v0.1.0
- getgrav.org/download/themes/quark/2.0.3 downloads the Quark theme at v2.0.3

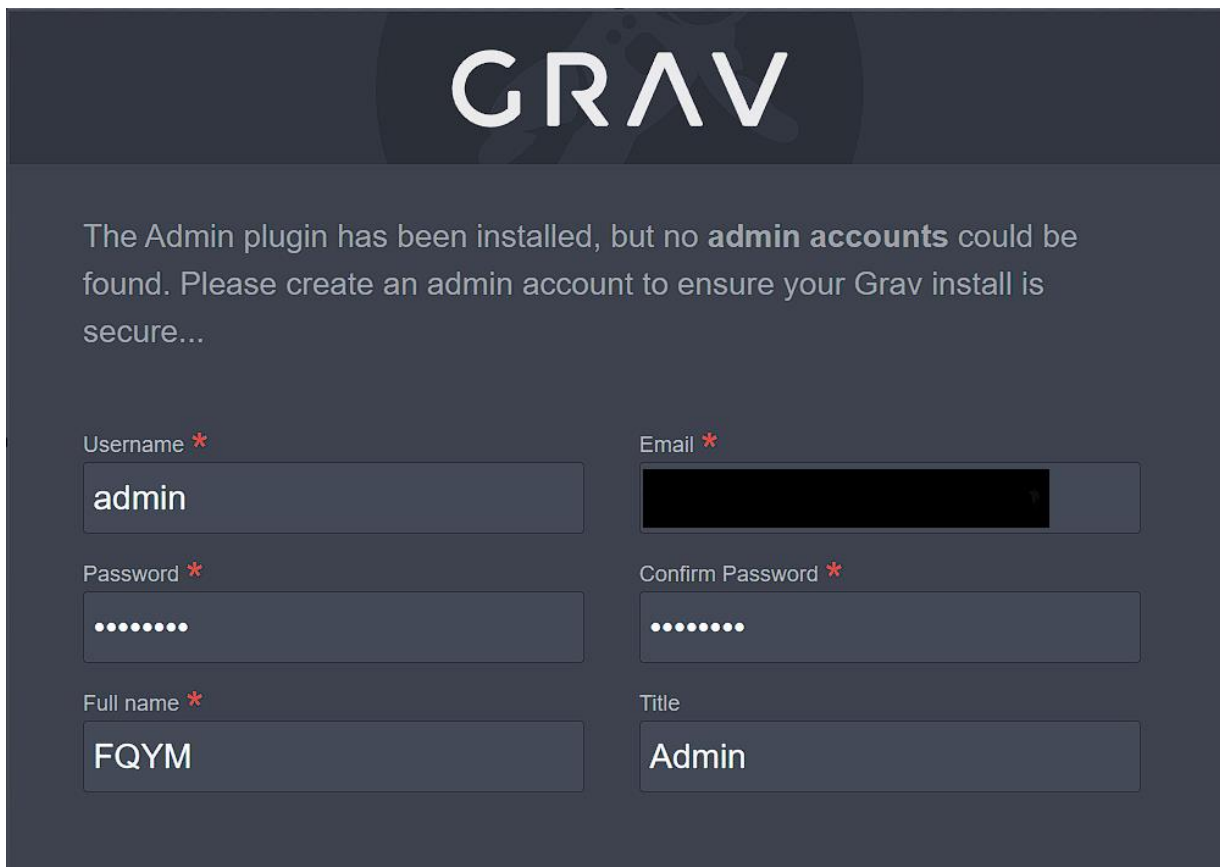
В итоге получаем директорию:

```
C:\Users\4me\OneDrive\
docker-compose.yml
Dockerfile
grav
```

2. Запускаем в командной строке команду docker-compose up

```
[+] Building 3.3s (15/15) FINISHED                                docker:desktop-linux
=> [grav internal] load build definition from Dockerfile          0.0s
=> => transferring dockerfile: 2.44kB                             0.0s
=> [grav internal] load metadata for docker.io/library/php:8.3-apache-bullseye 1.1s
=> [grav auth] library/php:pull token for registry-1.docker.io   0.0s
=> [grav internal] load .dockerignore                             0.0s
=> => transferring context: 2B                                     0.0s
=> [grav 1/9] FROM docker.io/library/php:8.3-apache-bullseye@sha256:6415657dc5b12c6da256716eeeff9e5ac3a373d37def 0.0s
=> => resolve docker.io/library/php:8.3-apache-bullseye@sha256:6415657dc5b12c6da256716eeeff9e5ac3a373d37def57ccd 0.0s
=> CACHED [grav 2/9] RUN a2enmod rewrite expires && sed -i 's/ServerTokens OS/ServerTokens ProductOnly/g' 0.0s
=> CACHED [grav 3/9] RUN apt-get update && apt-get install -y --no-install-recommends unzip libfreetype6 0.0s
=> CACHED [grav 4/9] RUN { echo 'opcache.memory_consumption=128'; echo 'opcache.interned_strings_buffer= 0.0s
=> CACHED [grav 5/9] RUN pecl install apcu && pecl install yaml-2.2.4 && docker-php-ext-enable apcu yaml 0.0s
=> CACHED [grav 6/9] RUN chown www-data:www-data /var/www        0.0s
=> CACHED [grav 7/9] WORKDIR /var/www                            0.0s
=> CACHED [grav 8/9] RUN curl -o grav-admin.zip -SL https://getgrav.org/download/core/grav-admin/latest && u 0.0s
=> CACHED [grav 9/9] RUN (crontab -l; echo "* * * * * cd /var/www/html;/usr/local/bin/php bin/grav scheduler 1>> 0.0s
=> [grav] exporting to image                                     2.1s
=> => exporting layers                                           0.0s
=> => exporting manifest sha256:47dba73372927f938a78d2807bb4454f6c7f4e72c652d6b4eea83ae4473d511e 0.0s
=> => exporting config sha256:cf4e5261220427384e96a8e70df58e6eeb2d3d23656b9441e43284a899b10f2f 0.0s
=> => exporting attestation manifest sha256:e366918c478ef4f737ebe843fad9742105ad6109260d9fb35a0df6626ac675a 0.0s
=> => exporting manifest list sha256:dbbb81be470dcae1074f1501df478eafeb79341b48f60877b667a9af73283a71 0.0s
=> => naming to docker.io/library/grav-dock-grav:latest         0.0s
=> => unpacking to docker.io/library/grav-dock-grav:latest      2.0s
=> [grav] resolving provenance for metadata file                 0.0s
[+] Running 4/4
✔grav Built 0.0s
✔Network grav-dock_default Created 0.3s
✔Volume "grav-dock_grav-data" Created 0.0s
✔Container grav-dock-grav-1 Created 0.3s
Attaching to grav-1
```

3. Заходим на localhost:8080, создаём администратора:



The screenshot shows the Grav Admin installation interface. At the top, the Grav logo is displayed. Below it, a message states: "The Admin plugin has been installed, but no **admin accounts** could be found. Please create an admin account to ensure your Grav install is secure...". The form consists of six input fields arranged in two columns. The left column contains fields for Username (filled with 'admin'), Password (masked with dots), and Full name (filled with 'FQYM'). The right column contains fields for Email (redacted with a black box), Confirm Password (masked with dots), and Title (filled with 'Admin'). Each field is preceded by its label and a red asterisk indicating a required field.

Username *	Email *
admin	
Password *	Confirm Password *
.....
Full name *	Title
FQYM	Admin

4. Интерфейс Grav, глазами администратора:

