

ПО “Платформастер”

Руководство по развертыванию

Оглавление

1. Подготовка инфраструктуры.....	2
2. Установка ПО на сервере.....	3
2.1. Настройка конфигурационных файлов приложения.....	3
2.2. Запуск контейнеров приложения	3
2.3. Настройка СУБД.....	4
2.4. Установка пакетов composer.....	4
2.5. Фронтенд.....	4

1. Подготовка инфраструктуры

Для развертывания ПО необходим сервер баз данных (БД) и сервер приложений (для тестовых инсталляций могут быть на одном сервере).

В качестве операционной системы на серверах БД и приложений должна использоваться ОС, Oracle Linux Server не ниже версии 8.7.

В качестве СУБД рекомендуется использовать PostgreSQL, с расширением PostGIS из состава ОС. Системные требования к серверу БД: RAM 8ГБ, свободное место на жестком диске 40ГБ.

В качестве сервера приложений рекомендуется использовать web-сервер nginx из состава ОС с установленным модулем php, в который добавлено расширение pgsql для работы с СУБД PostgreSQL. Системные требования к серверу приложений: RAM 8ГБ, свободное место на жестком диске 20ГБ.

MongoDB: RAM 2ГБ, свободное место на жестком диске 20ГБ.

RabbitMQ: RAM 2ГБ, свободное место на жестком диске 20ГБ.

CDN: RAM 2ГБ, свободное место на жестком диске 120ГБ.

2. Установка ПО на сервере

Предварительно установить на сервере docker & docker-compose

Например для Oracle Linux будет так.

```
sudo yum install -y yum-utils
sudo yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-
ce.repo
sudo yum install docker-ce docker-ce-cli containerd.io
sudo systemctl enable docker
sudo systemctl start docker
sudo yum install docker-compose-plugin
docker compose version
```

Официальная инструкция по установке <https://docs.docker.com/engine/install/>

2.1. Настройка конфигурационных файлов приложения

.env для docker-compose

```
POSTGRES_USER=master
POSTGRES_PASSWORD=secret
POSTGRES_DB=master

MONGODB_USER: root
MONGODB_PASSWORD: root
```

.env в каталоге backend

```
DATABASE_URL=pgsql://master:secret@postgres:5432/master
REPORT_DATABASE_URL=pgsql://master:secret@postgres:5432/report
```

.env в каталоге frontend

```
VUE_APP_API_URL=http://127.0.0.1:8080
```

2.2. Запуск контейнеров приложения

docker compose up -d

В docker-compose также присутствуют контейнеры для rabbitmq \ webdav , но на практике вместо них используются внешние сервисы (cdn и прочее)

2.3. Настройка СУБД

Пользователь БД для PostgreSQL создается автоматически, согласно конфигурации (см. пункт 2.1).

Дамп базы закидывается в `docker/postgres/initdb.sql` и разворачивается также автоматом (при первой инициализации контейнера)

Альтернативно, можно поднять дампы вручную (параметры для `psql` соответствуют значениям в `.env`)

```
docker ps -q -f ancestor=mastersibsetiru-postgres
```

```
docker exec -i <container_id> psql -U app -c "CREATE DATABASE master;"
```

```
cat initdb.sql | docker exec -i <container_id> psql -U master -d master
```

После этого, при необходимости, можно применить миграции

```
docker compose exec php-fpm sh
```

2.4. Установка пакетов composer

```
docker compose exec -u root php-fpm sh -c "composer i -a"
```

2.5. Фронтенд

Фронтенд собирается в виде статических файлов на этапе подготовки контейнера. Если нужна пересборка статических файлов, то

```
docker compose build --no-cache nginx && docker compose up -d nginx
```