



## **Инструкция по установке**

**Комплекс расчетов оценки стоимости  
строительства (КРОСС)**



## Содержание

1	Начальные требования .....	3
2	Установка и развертывание КРОСС без использования Docker .....	3
2.1	Подготовка к установке компонентов.....	3
2.2	Загрузка файлов КРОСС на сервер.....	3
2.3	Настройка web сервера .....	3
2.4	Установка базы данных PostgreSQL.....	5
2.5	Проверка работоспособности КРОСС .....	5
3	Запуск PostgreSQL в Docker (раздельный вариант) .....	6
3.1	PostgreSQL image.....	6
3.2	phpPgAdmin.....	6
3.3	Запуск PostgreSQL в Docker (вместе с Adminer).....	6
3.4	Проверка работоспособности КРОСС .....	7

## 1 Начальные требования

Для развертывания КРОСС необходимы:

- Ubuntu 20.04.2.0
- PostgreSQL
- .NET 5.0

## 2 Установка и развертывание КРОСС без использования Docker

Инструкция для установки предполагает, что действия в Ubuntu ведутся под пользователем, имеющем права суперпользователя.

### 2.1 Подготовка к установке компонентов

Установить обновления ПО и перезагрузить:

```
sudo apt-get update           #Fetches the list of available updates
sudo apt-get upgrade         #Strictly upgrades the current packages
sudo apt-get dist-upgrade    #Installs updates (new ones)
sudo reboot
```

Установить .NET Runtime:

```
sudo apt-get install -y dotnet-runtime-5.0
```

### 2.2 Загрузка файлов КРОСС на сервер

1. Создать директорию `/var/www/kross/`
2. Загрузить в нее файл `publish.zip`
3. Распаковать скаченные файлы в созданную директорию

```
sudo unzip /var/www/kross/publish.zip -d /var/www/kross/
```

4. Переместить файл лицензии с расширением `.clf` в директорию `/var/www/kross/`

### 2.3 Настройка web сервера

1. Создать конфигурацию сайта Kestrel:

```
sudo vim /etc/systemd/system/kestrel-kross.service
```

2. Вставить содержимое:

```
[Unit]
```

```
    Description=KROSS
```

```
[Service]
```

```
WorkingDirectory=/var/www/kross
```

```
ExecStart=/usr/bin/dotnet /var/www/kross/KROSS_Core.dll
```

```
Restart=always
```

```
RestartSec=10 # Restart service after 10 seconds if dotnet  
service crashes
```

```
SyslogIdentifier=dotnet-kross
```

```
User=www-data
```

```
Environment=ASPNETCORE_ENVIRONMENT=Development
```

```
[Install]
```

```
WantedBy=multi-user.target
```

### 3. Запустить демон:

```
sudo systemctl enable kestrel-kross.service # register the service
```

```
sudo systemctl start kestrel-kross.service # start the service
```

```
sudo systemctl status kestrel-kross.service # check the service is  
running
```

### 4. Установить веб-сервер nginx:

```
sudo apt-get install nginx
```

### 5. Отредактировать конфигурацию:

```
sudo vim /etc/nginx/sites-available/default
```

Содержимое:

```
server {  
    listen 80;  
    location / {  
        proxy_pass http://localhost:5000;  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection keep-alive;  
        proxy_set_header Host $host;  
        proxy_cache_bypass $http_upgrade;  
    }  
}
```

6. Проверить конфигурацию:

```
sudo nginx -t
```

7. Применить изменения:

```
sudo nginx -s reload
```

## 2.4 Установка базы данных PostgreSQL

1. Установить PostgreSQL:

```
sudo apt install postgresql postgresql-contrib
```

2. Создать пользователя, базу данных и назначить права:

```
sudo -u postgres psql
```

```
CREATE USER sa2 WITH PASSWORD 'sa2'
```

```
CREATE DATABASE ASEZ
```

```
GRANT ALL PRIVILEGES ON DATABASE ASEZ TO sa2
```

3. Загрузить дампы:

```
psql -h localhost -d ASEZ -U sa2 -W -f /var/www/kross/dump-ASEZ-  
20210220.sql
```

## 2.5 Проверка работоспособности КРОСС

Запустить в браузере ссылку:

<http://<адрес сервера>/ASEZ/GlobalSearchAsez>

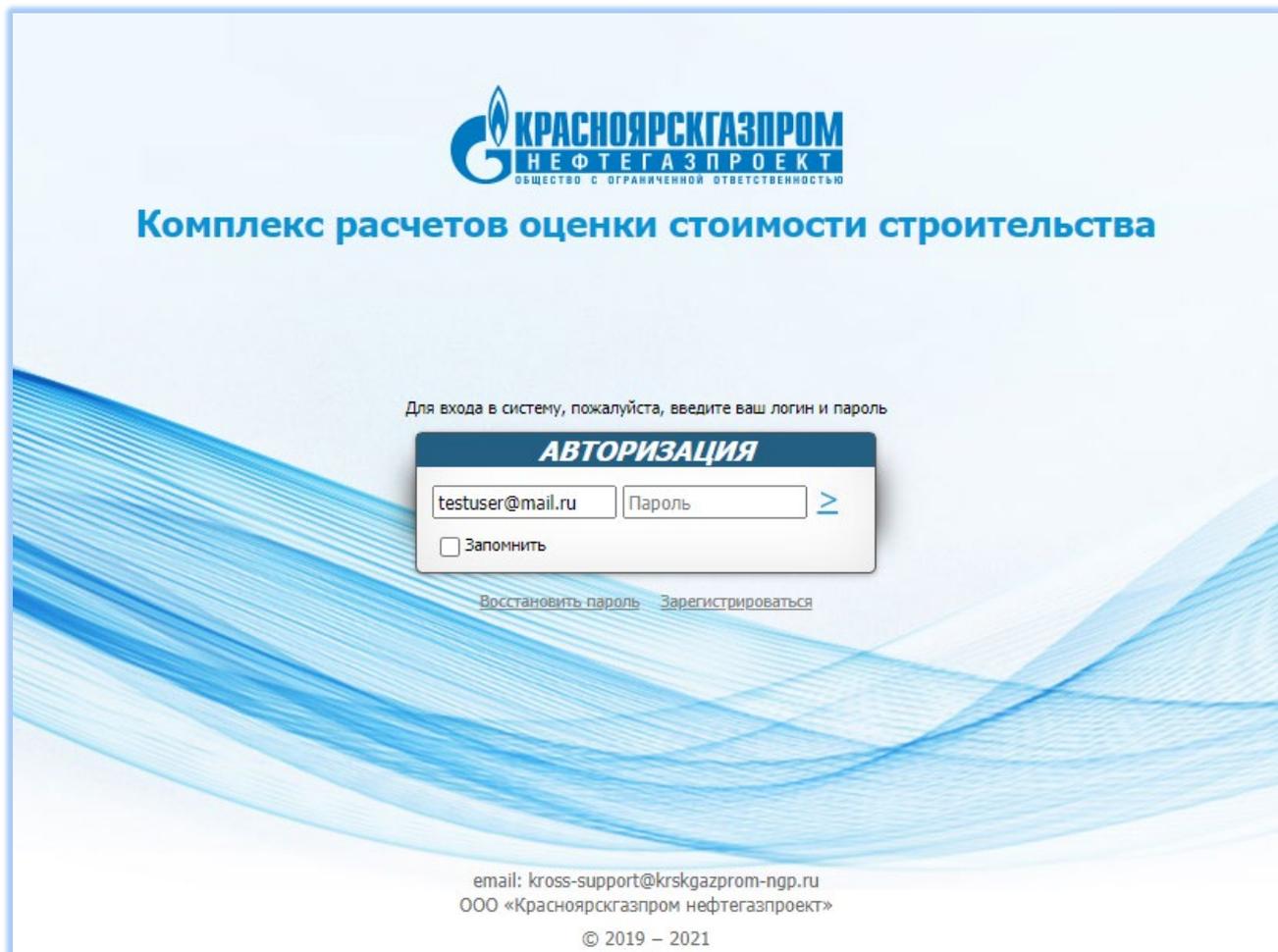


Рисунок 1. Окно авторизации КРОСС

### 3 Запуск PostgreSQL в Docker (раздельный вариант)

#### 3.1 PostgreSQL image

1. Скачать образ PostgreSQL из Docker-hub:

```
$ docker pull postgres
```

2. Запустить образ с заданным паролем суперпользователя:

```
$ docker run --name postgresqldb -e POSTGRES_PASSWORD=password -d postgres
```

#### 3.2 phpPgAdmin

1. Скачать образ из Docker-hub:

```
$ docker pull dockage/phpPgAdmin:latest
```

#### 3.3 Запуск PostgreSQL в Docker (вместе с Adminer)

1. Создать файл dbstack.yml:

```
# Use postgres/example user/password credentials
```

```
version: '3.1'
```

```
services:
```

```
  db:
```

```
    image: postgres
```

```
    restart: always
```

```
    ports:
```

```
      - "5433:5432" # HOST:CONTAINER
```

```
    environment:
```

```
      POSTGRES_PASSWORD: sa2
```

```
      POSTGRES_USER: sa2
```

```
      POSTGRES_DB: ASEZ
```

```
  adminer:
```

```
    image: adminer
```

```
    restart: always
```

```
    ports:
```

```
      - 8080:8080
```

2. Выполнить команду в этом каталоге:

```
$ docker-compose -f dbstack.yml up
```

PostgreSQL запущен и принимает подключения на 5433 порту, создана база ASEZ с суперпользователем sa2 и паролем sa2.

### 3.4 Проверка работоспособности КРОСС

1. Запустить в браузере ссылку:

<http://<адрес сервера>/ASEZ/GlobalSearchAsez>