

Инструкция по установке системы

В инструкции будут использованы обозначения параметров, например

```
<путь_до_папки_проекта>
```

Означает, что надо указывать параметр, не используя знаки `<` и `>`, т.е. для `<путь_до_папки_проекта>` значение может быть, например

```
/home/user/rvravo
```

Первоначальная настройка сервера. Установка apt пакетов

1. Если ваша система - Astra Linux. Проверьте, что в файле `/etc/apt/sources.list` доступны удаленные репозитории и закомментирован `cdrom`. Или воспользуйтесь оптическим диском. [Документация](#)
2. Обновите `apt`

```
sudo apt update
```

3. Установите необходимые библиотеки. Во время установки `firebird-server` откроется окно для ввода пароля администратора. Оставьте поле пустым или введите свой пароль, в таком случае его необходимо будет указать в настройках приложения (см. [Настройка параметров проекта](#)).

```
sudo apt install build-essential libpq-dev firebird-server apache2 postgresql
```

Для того, чтобы работал экспорт отчетных форм в формат pdf, необходимо установить `libreoffice`

```
sudo apt install libreoffice
```

Создание базы данных

1. Введите команду

```
sudo -u postgres psql
```

2. Создайте базу данных

```
CREATE DATABASE <имя_базы_данных>;
```

3. Создайте пользователя

```
CREATE USER <имя_пользователя> WITH PASSWORD '<пароль_пользователя>';
```

4. Подключитесь к созданной базе данных

```
\с <имя_базы_данных>;
```

5. Введите команду, команда выведет путь

```
SHOW data_directory;
```

6. Скопируйте файл инициализации базы (выполните команду в отдельном окне терминала)

```
sudo cp <путь_до_папки_проекта>/01-init-postgres.sql <путь_из_5_пункта>
```

7. Примените скрипт инициализации

```
\i <путь_из_5_пункта>/01-init-postgres.sql
```

8. Настройте пользователя

```
ALTER ROLE <имя_пользователя> SET client_encoding TO 'utf8';  
ALTER ROLE <имя_пользователя> SET default_transaction_isolation TO 'read  
committed';  
ALTER ROLE <имя_пользователя> SET timezone TO 'UTC';  
ALTER ROLE <имя_пользователя> SET standard_conforming_strings = on;  
GRANT ALL PRIVILEGES ON DATABASE <имя_базы_данных> TO <имя_пользователя>;  
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO <имя_пользователя>;  
GRANT ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public TO <имя_пользователя>;  
GRANT ALL PRIVILEGES ON ALL FUNCTIONS IN SCHEMA public TO <имя_пользователя>;
```

9. Выйдите из psql консоли

```
\q
```

10. У созданного пользователя (`<имя_пользователя>`) должны быть настроены все права для подключения к базе данных. Если ваша система Astra Linux имеет мандатное управление, у пользователя должны быть настроены мандатные атрибуты для доступа к базе данных.

Настройка параметров проекта

1. Перейдите по пути

```
cd <папка_проекта>/core
```

2. Измените файл `.env`. Например, можно воспользоваться консольным редактором `nano`

```
sudo nano .env
```

3. Заполните параметры (**Важно:** если параметр не обязателен к заполнению, и не меняется значение по умолчанию, удалите или закомментируйте параметр. Не должно остаться строки вида `PARAM_NAME=`)

```
# Порт для запуска python backend.
# (Не обязателен к заполнению, по умолчанию PORT=8000)
PORT=

# Кол-во потоков для апуска python приложения. (Не обязателен к заполнению, по
умолчанию THREADS=4)
THREADS=

# Имя базы данных PostgreSQL. Обязаталельно к заполнению
# (Пункт 2 "Создание базы данных")
POSTGRES_DATABASE=

# (Не обязателен к заполнению, по умолчанию POSTGRES_HOST=127.0.0.1)
POSTGRES_HOST=

# Порт, на котором запущен PostgreSQL. (Не обязателен к заполнению, по
умолчанию POSTGRES_PORT=5432)
POSTGRES_PORT=

# Пользователь PostgreSQL. Обязательно к заполнению
# (Пункт 3 "Создание базы данных")
POSTGRES_USER=

# Пароль пользователя PostgreSQL. Обязаталельно к заполнению
# (Пункт 3 "Создание базы данных")
POSTGRES_PASSWORD=

# Пользователь Firebird.
# (Не обязателен к заполнению, по умолчанию FIREBIRD_USER=SYSDBA)
FIREBIRD_USER=

# Пароль пользователя Firebird.
# (Не обязателен к заполнению, по умолчанию FIREBIRD_PASSWORD не задан.
# Указывать только в том случае, если на этапе утановки
# Firebird был введен пароль.)
FIREBIRD_PASSWORD=
```

Настройка Systemctl

1. Введите команду для редактирования файла

```
sudo nano /etc/systemd/system/rvrao.service
```

2. Заполните файл содержимым

```
[Service]
Type=simple
Restart=always
ExecStart=/usr/bin/python3 <путь_до_папки_проекта>/core/run.py
StandardOutput=<путь_до_папки_проекта>/logs/logs.txt
StandardError=<путь_до_папки_проекта>/logs/logs.txt
User=root

[Install]
WantedBy=multi-user.target
```

3. Выполните команды

```
sudo systemctl daemon-reload
sudo systemctl start rvrao.service
sudo systemctl enable rvrao.service
```

4. Посмотреть статус systemctl.

```
sudo systemctl status rvrao.conf
```

Если процесс успешно запущен, выведется сообщение

```
...
Active: active (running) since ...
...
```

Если произошла ошибка, смотрите логи по пути `<папка_проекта>/logs/logs.txt`.

Настройка Apache

1. Выполните команды

```
sudo a2enmod rewrite
sudo a2enmod proxy_http
sudo service apache2 restart
```

- Измените файл `/etc/apache2/apache2.conf`. Раскомментируйте строку `AstraMode on`, измените на `AstraMode off`
- Перейдите по пути `/etc/apache2/sites-available`
- Создайте файл `rvrao.conf` с содержимым

```
# Справка по параметрам
# <порт_python_backend> - параметр PORT из пункта 3 "Настройка проекта"
# <адрес_порта_для_frontend> - произвольный свободный порт

listen <адрес_порта_для_frontend>

<VirtualHost *:<адрес_порта_для_frontend>>
    ServerName <server_name>
    Alias / <путь_до_папки_проекта>/frontend/
    ProxyPass "/api" "http://localhost:<порт_python_backend>/"
    ProxyPassReverse "/api" "http://localhost:<порт_python_backend>/"
    <Directory <путь_до_папки_проекта>/core/>
        Require all granted
    </Directory>
    <DirectoryMatch "<путь_до_папки_проекта>/frontend/">
        Require all granted
        <IfModule mod_rewrite.c>
            RewriteEngine On
            RewriteBase /
            RewriteRule ^index\.html$ - [L]
            RewriteCond %{REQUEST_FILENAME} !-f
            RewriteCond %{REQUEST_FILENAME} !-d
            RewriteRule . /index.html [L]
        </IfModule>
    </DirectoryMatch>
</VirtualHost>
```

5. Выполните команду

```
sudo a2ensite rvrao.conf
```

6. Задайте необходимые права на папку. (Пример, где <путь_до_папки_проекта> = /home/user/project)

```
sudo chmod +rx /home /home/user /home/user/project /home/user/project/frontend
```

7. Перезапустите Apache

```
sudo service apache2 restart
```

8. В случае возникновения ошибок, смотрите логи по пути /var/log/apache2/error.log.