

Инструкция по установке ExrasyS QuestionnaireStudioPro и необходимых компонентов для её развёртывания на примере OS Debian11 (bullseye), reverse-proxy Caddy и СУБД Postgres Pro

- 1) Скачиваем актуальный дистрибутив Debian11 с официального сайта <https://www.debian.org/CD/http-ftp/#stable>
- 2) Устанавливаем ОС на любой из доступных гипервизоров (например, Microsoft Hyper-V, VMwareESXI, Proxmox) (Рекомендованные системные ресурсы: 2 ядра процессора, 2ГБ ОЗУ, 40 гб свободного места на жёстком диске или SSD)
- 3) После установки операционной системы выполняем команды `apt update` и `apt upgrade`
- 4) Устанавливаем reverse-proxy Caddy (<https://caddyserver.com/docs/install#debian-ubuntu-raspbian>)

4.1) Выполняем команды (от имени пользователя root, если не root, то добавляем `sudo` вначале):

```
apt install curl
```

```
apt install -y debian-keyring debian-archive-keyring apt-transport-https
```

```
curl -1sLf 'https://dl.cloudsmith.io/public/caddy/stable/gpg.key' | gpg --dearmor -o /usr/share/keyrings/caddy-stable-archive-keyring.gpg
```

```
curl -1sLf 'https://dl.cloudsmith.io/public/caddy/stable/debian.deb.txt' | tee /etc/apt/sources.list.d/caddy-stable.list
```

```
apt update
```

```
apt install caddy
```

4.2) После установки reverse-proxy Caddy редактируем конфигурационный файл командой

```
nano /etc/caddy/CaddyFile
```

и приводим его к виду:

```
:80 {  
    # Set this path to your site's directory.  
    #root * /usr/share/caddy  
  
    # Enable the static file server.  
    #file_server  
  
    # Another common task is to set up a reverse proxy:  
    reverse_proxy localhost:5000  
  
    # Or serve a PHP site through php-fpm:  
    # php_fastcgi localhost:9000  
}
```

и перезапускаем сервис командой
service caddy restart

- 5) Устанавливаем dotnet-sdk-6
(<https://docs.microsoft.com/ru-ru/dotnet/core/install/linux-debian>)

5.1) Выполняем команды:

```
wget https://packages.microsoft.com/config/debian/11/packages-microsoft-prod.deb -O packages-microsoft-prod.deb
```

```
dpkg -i packages-microsoft-prod.deb
```

```
rm packages-microsoft-prod.deb
```

```
apt update
```

```
apt install -y dotnet-sdk-6.0
```

- 6) Устанавливаем СУБД Postgres Pro (вместе с балансировщиком нагрузки pgbouncer)

6.1) Выполняем команды:

```
curl -o pgpro-repo-add.sh https://repo.postgrespro.ru/pgpro-14/keys/pgpro-repo-add.sh
```

```
sh pgpro-repo-add.sh
```

```
apt install postgrespro-std-14
```

```
systemctl enable postgrespro-std-14
```

```
systemctl start postgrespro-std-14
```

6.2) Выполняем конфигурацию Postgres Pro

6.2.1) Выполняем команду:

```
nano /var/lib/pgpro/std-14/data/pg_hba.conf
```

и приводим содержимое файла pg_hba.conf к следующему виду:

```
# TYPE      DATABASE      USER      ADDRESS      METHOD
# "local" is for Unix domain socket connections only
local      all          all              trust
# IPv4 local connections:
host       all          all          0.0.0.0/0    md5
# IPv6 local connections:
host       all          all          ::1/128     md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local      replication  all          md5
host       replication  all          127.0.0.1/32 md5
host       replication  all          ::1/128     md5
```

6.2.2) Выполняем команду:

```
nano /var/lib/pgpro/std-14/data/postgresql.conf
```

и редактируем следующие строки:

```
#-----
# CONNECTIONS AND AUTHENTICATION
#-----

# - Connection Settings -

listen_addresses = '*'          # what IP address(es) to listen on;
                                # comma-separated list of addresses;
                                # defaults to 'localhost'; use '*' for all
                                # (change requires restart)
port = 5432                     # (change requires restart)
max_connections = 100           # (change requires restart)
```

6.2.3) Выполняем команду:

```
psql -U postgres
```

после чего попадаем в консоль psql, внутри которой выполняем следующие команды:

```
CREATE DATABASE qspro; (создаём базу с именем qspro)
```

```
ALTER USER postgres WITH PASSWORD 'test12345';
```

(примечание: это ненадёжный пароль для демонстрации выполнения команды, Вам следует указать свой надёжный пароль в одинарных кавычках)

```
GRANT ALL PRIVILEGES ON DATABASE "qspro" to postgres;
```

и выходим из консоли psql командой \q

6.3) Устанавливаем балансировщик pgbouncer

6.3.1) Выполняем команду apt install pgbouncer

6.3.2) Выполняем команду nano /etc/pgbouncer/userlist.txt и приводим содержимое текстового файла к виду:

```
"postgres" "test12345"
```

6.3.3) Выполняем команду nano /etc/pgbouncer/pgbouncer.ini и редактируем/добавляем строки:

```
[databases]
;; foodb over Unix socket
;foodb =

;; redirect bardb to bazdb on localhost
;bardb = host=localhost dbname=bazdb

;; access to dest database will go with single user
;forcedb = host=localhost port=300 user=baz password=foo client_encoding=UNICODE datestyle=ISO connect_query='SELECT 1'

;; use custom pool sizes
;nondefaultdb = pool_size=50 reserve_pool=10

;; use auth_user with auth_query if user not present in auth_file
;; auth_user must exist in auth_file
; foodb = auth_user=bar

;; fallback connect string
;* = host=testserver

qspro = host=127.0.0.1 dbname=qspro port=5432 password=test12345
;; User-specific configuration
```

```
[pgbouncer]

;;;
;;; Administrative settings
;;;

logfile = /var/log/pgbouncer/pgbouncer.log
pidfile = /run/pgbouncer/pgbouncer.pid

;;;
;;; Where to wait for clients
;;;

;; IP address or * which means all IPs
listen_addr = *
listen_port = 6432
```

```
;; When server connection is released back to pool:
;; session      - after client disconnects (default)
;; transaction  - after transaction finishes
;; statement    - after statement finishes
pool_mode = transaction
```

6.3.4) Выполняем команды:

```
systemctl enable pgbouncer
```

```
systemctl start pgbouncer
```

7) Редактируем файл /etc/sysctl.conf командой:

```
nano /etc/sysctl.conf
```

и в конец файла добавляем следующие строки:

```
fs.inotify.max_user_instances=524288
```

```
fs.inotify.max_user_watches=1048576
```

```
fs.inotify.max_queued_events=163840
```

8) Выполняем команду

```
touch /etc/systemd/system/qspro.service
```

8.1) Выполняем команду

```
nano /etc/systemd/system/qspro.service
```

и вставляем следующие строки в файл:

[Unit]

Description=Expasys Questionnaire Studio Pro

[Service]

WorkingDirectory=/opt/qspro/

ExecStart=/usr/bin/dotnet ./QuestionnaireStudioPro.dll

Restart=always

RestartSec=10

SyslogIdentifier=ExpasysQSPRO

User=root

Environment=ASPNETCORE_ENVIRONMENT=Production

[Install]

WantedBy=multi-user.target

8.3) выполняем команду `systemctl daemon-reload`

9) Выполняем команду `mkdir /opt/qspro`

9.1) Выполняем команду `apt install unzip`

9.2) Выполняем команду `cd /opt/qspro/`

9.3) Выполняем команду:

wget https://ваша_ссылка_для_скачивания_QuestionnaireStudioPro

9.4) Выполняем команду `unzip QuestionnaireStudioPro.zip`

9.5) После разархивирования выполняем команду

`rm QuestionnaireStudioPro.zip`

9.6) Выполняем команду

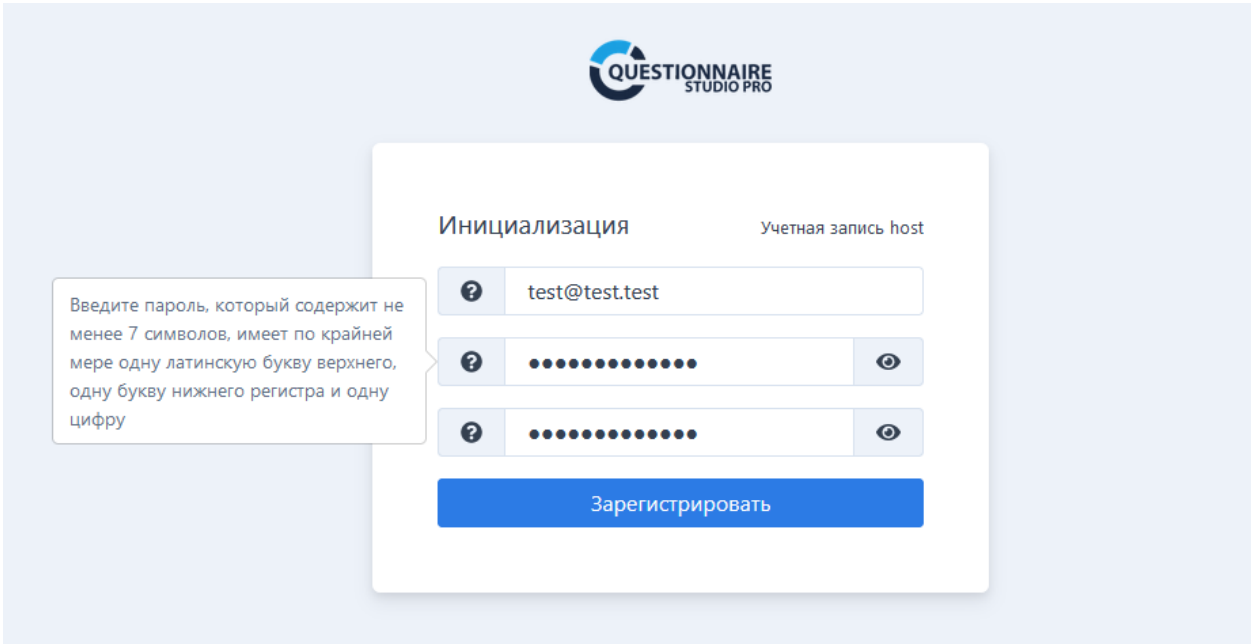
`nano /opt/qspro/appsettings.json`

и приводим файл к виду:

```
{
  "Data": {
    "DefaultConnection": {
      "ConnectionString": "Server=IP_АДРЕС_ВАШЕЙ_МАШИНЫ;Port=6432;Database=qspro;User Id=postgres;Password=test12345;Pooling=false;No Reset On Close=true;"
    },
    "DataProtectionKey": "69d631616ce94dd3b13f40ff5ba34718",
    "CookieDomain": "IP_АДРЕС_ВАШЕЙ_МАШИНЫ",
    "ViewMode": "B-Data_copp",
    "CRMIntegration": {
      "RequestUrl": "",
      "WebFormKey": "",
      "NotifyList": "",
      "ManagersList": "",
      "Tag": "",
      "Source": ""
    },
    "Environment": "notContainer",
    "Logging": {
      "IncludeScopes": false,
      "LogLevel": {
        "Default": "Information",
        "System": "Information",
        "Microsoft": "Information"
      }
    }
  }
}
```

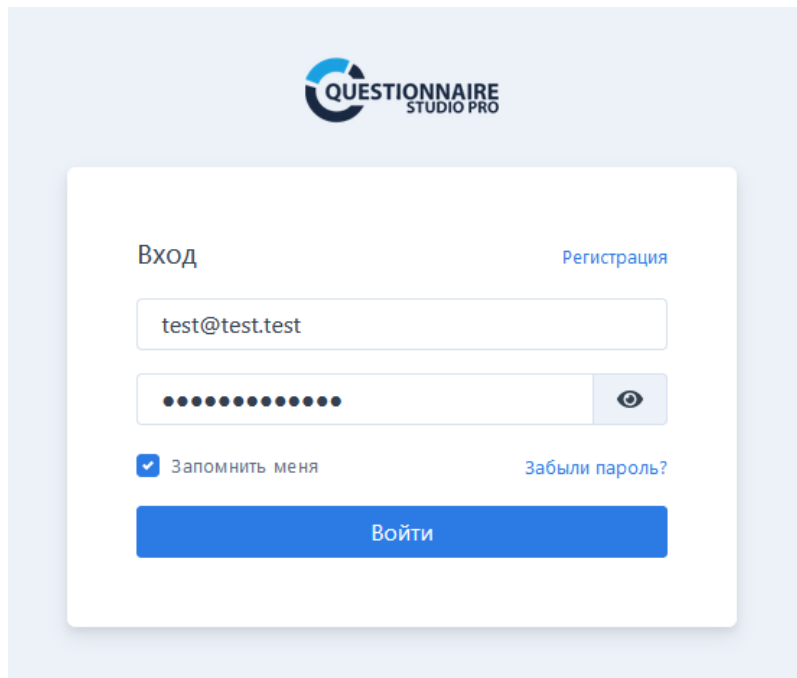
9.6) Выполняем команды:
systemctl enable qspro.service
systemctl start qspro.service
reboot

10) После перезагрузки машины в адресной строке браузера переходите по ссылке http://IP_Адрес_Вашей_Машины_С_QuestionnaireStudioPro/install и заполняете данные для Вашего аккаунта



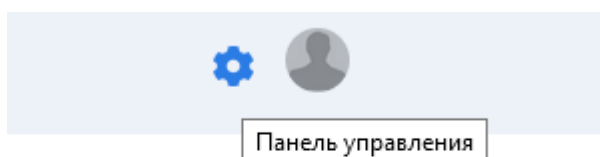
The screenshot shows the 'Инициализация' (Initialization) page of Questionnaire Studio Pro. The page title is 'Инициализация' and the subtitle is 'Учетная запись host'. There are three input fields: the first contains 'test@test.test', the second and third are masked with dots. A blue button labeled 'Зарегистрировать' (Register) is at the bottom. A callout box on the left contains the text: 'Введите пароль, который содержит не менее 7 символов, имеет по крайней мере одну латинскую букву верхнего, одну букву нижнего регистра и одну цифру' (Enter a password that contains at least 7 characters, has at least one uppercase Latin letter, one lowercase Latin letter, and one digit).

10.1) Затем на главной странице нажимаете на кнопку Вход в правом верхнем углу и вводите данные только что созданного пользователя

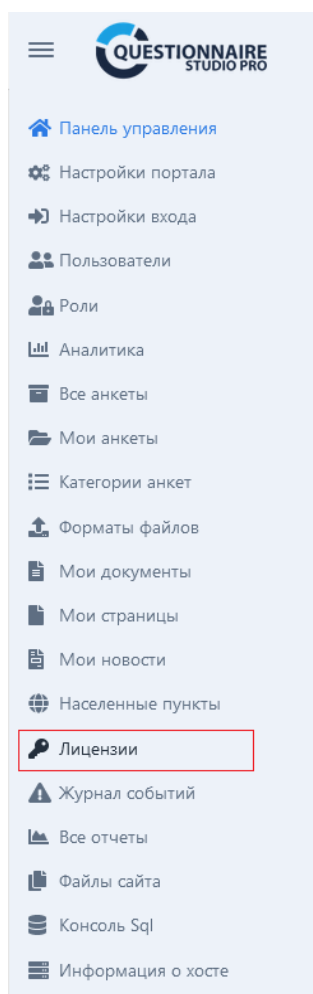


The screenshot shows the 'Вход' (Login) page of Questionnaire Studio Pro. The page title is 'Вход' and the subtitle is 'Регистрация'. There are two input fields: the first contains 'test@test.test' and the second is masked with dots. A blue button labeled 'Войти' (Login) is at the bottom. There is a checkbox labeled 'Запомнить меня' (Remember me) which is checked, and a link labeled 'Забыли пароль?' (Forgot password?).

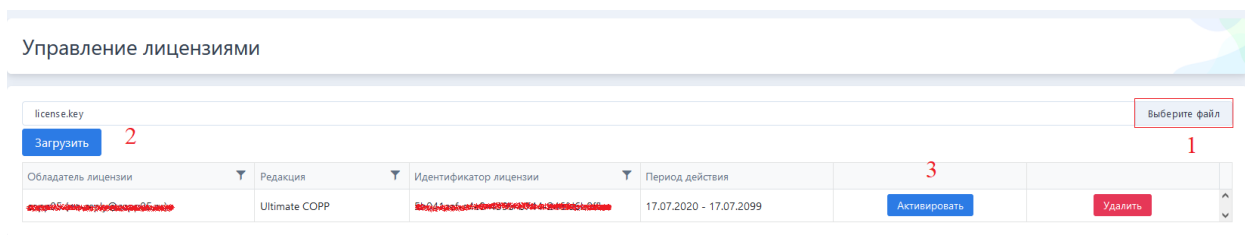
10.2) После авторизации переходим в панель управления QuestionnaireStudioPro (значок шестерёнки)



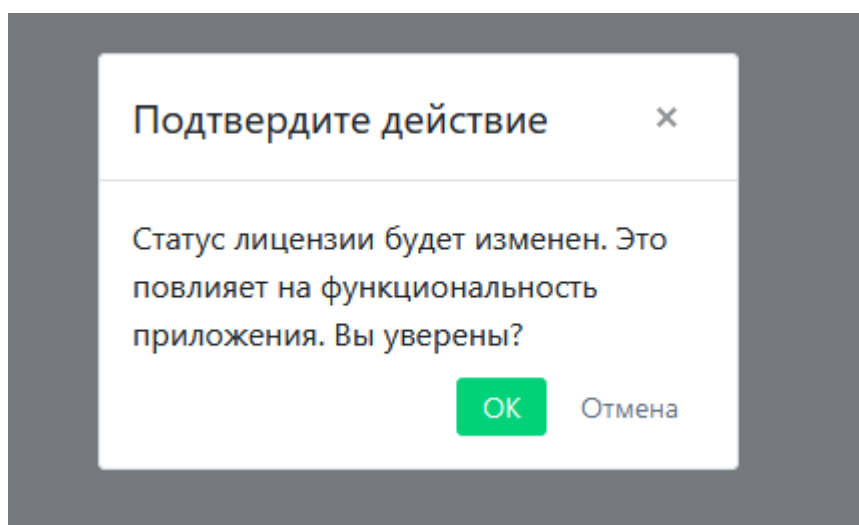
10.3) Внутри панели управления переходим на страницу Лицензии



10.4) На странице управления лицензиями сначала выбираем предоставленный Вам файл с расширением .key, затем нажимаем на кнопку загрузить, после чего появится строка с информацией об обладателе, редакции, идентификаторе и периоде действия Вашей лицензии.



После этого нажимаем на кнопку Активировать и подтверждаем действие



Лицензия станет активной, а строка с информацией о Вашей лицензии придёт к следующему виду:

