

ИП Сысоев Алексей Евгеньевич, ИНН 280125708388

Программа для ЭВМ «ФотоОбъект»

Руководство по установке

Документ 03

Версия программы: веб-кабинет 2026.10, мобильное приложение 2026.3.4

Дата: 13.10.2026

1. Общая схема

На одном сервере работают три части, каждая в своих контейнерах Docker:

1. **Сервер приёма отчётов** (ODK Central) — принимает отчёты с телефонов, слушает только внутренний адрес 127.0.0.1:8080.
2. **Веб-кабинет ФотоОбъект** — слушает 127.0.0.1:8000.
3. **Caddy** — принимает подключения из интернета по HTTPS (порты 80 и 443) и передаёт их кабинету и серверу отчётов.

Используются два доменных имени: `app.<домен>` — веб-кабинет, `odk.<домен>` — адрес для телефонов.

2. Подготовка

Что	Требование
Сервер	x86-64, от 2 ядер, от 4 ГБ памяти, диск от 100 ГБ с учётом фото
ОС	Linux с Docker (проверено: Ubuntu Server 24.04/26.04, РЕД ОС 8 — для кабинета)
DNS	А-записи <code>app.<домен></code> и <code>odk.<домен></code> на IP-адрес сервера
Сеть	Открыты порты 22 (администрирование), 80 и 443
Почта	Адрес администратора для уведомлений

Перед выпуском сертификатов проверьте, что имена указывают на сервер: `getent hosts app.<домен>` .

3. Установка Docker

Установите Docker Engine и плагин Compose из официального репозитория для вашей ОС (Ubuntu — пакеты `docker-ce`, `docker-compose-plugin`; РЕД ОС — пакеты `docker-ce` из репозитория РЕД ОС). Включите ограничение журналов Docker (`/etc/docker/daemon.json` : `"log-opts": {"max-size": "10m", "max-file": "5"}`).

4. Установка сервера приёма отчётов (ODK Central)

```
cd /opt
git clone https://github.com/getodk/central && cd central
git checkout v2026.3.1 && git submodule update --init
cp .env.template .env
```

В файле `.env` задайте:

```
DOMAIN=odk.<домен>
SYSADMIN_EMAIL=<адрес администратора>
SSL_TYPE=upstream
HTTP_PORT=8080
HTTPS_PORT=443
```

Скопируйте настройки локализации из комплекта ФотоОбъект и подготовьте шаблон веб-сервера:

```
cp /opt/fotoobject/deploy/central/docker-compose.override.yml /opt/central/
python3 /opt/fotoobject/deploy/central/make_odk_conf.py
```

Файл `docker-compose.override.yml`: сервер отчётов слушает только внутренний адрес `127.0.0.1:8080`; кэш Enketo — Valkey (BSD-3-Clause) вместо Redis; отчёты об ошибках во внешние сервисы (Sentry) выключены; веб-сервер использует шаблон без обращений к внешним адресам (создаёт `make_odk_conf.py`, запускать заново после каждого обновления ODK Central).

```
docker compose build && docker compose up -d
docker compose exec service odk-cmd --email <адрес администратора> user-create
docker compose exec service odk-cmd --email <адрес администратора> user-promote
docker compose exec service odk-cmd --email <служебный адрес кабинета> user-create
```

Служебной учётной записи кабинета назначьте роль администратора (кабинет сам создаёт проекты филиалов). Создайте первый проект и запомните его номер.

5. Установка веб-кабинета и Caddy

```
cd /opt
git clone <адрес репозитория ФотоОбъект> fotoobject
cd fotoobject/deploy
cp .env.example .env
chmod 600 .env
```

Заполните `.env`:

Параметр	Значение
<code>CENTRAL_DOMAIN</code> , <code>CABINET_DOMAIN</code> , <code>SITE_DOMAIN</code>	доменные имена (кириллические домены — в записи punycode)
<code>CENTRAL_URL</code> , <code>CENTRAL_PUBLIC_URL</code>	<code>https://odk.<домен></code>
<code>CENTRAL_EMAIL</code> , <code>CENTRAL_PASSWORD</code>	служебная учётная запись кабинета
<code>CENTRAL_PROJECT_ID</code>	номер первого проекта
<code>SECRET_KEY</code>	случайная строка: <code>openssl rand -base64 48</code>
<code>COLLECT_ADMIN_PW</code>	пароль, закрывающий настройки приложения на телефонах

Параметр	Значение
DEFAULT_COMPANY	название первой компании

```
docker compose up -d --build
docker compose exec cabinet python -m app.cli check
docker compose exec cabinet python -m app.cli setup-central
docker compose exec cabinet python -m app.cli create-admin <e-mail> "<Имя Фамилия>"
```

Команда `create-admin` создаёт владельца сервиса и первую компанию с филиалом «Основной филиал». Caddy автоматически получает сертификаты HTTPS. Проверьте, что открываются `https://app.<домен>` и `https://odk.<домен>`.

Установка на РЕД ОС. Образ кабинета можно собрать на официальном базовом образе РЕД ОС: `docker build -f deploy/compat/redos8.Dockerfile -t fotoobject-cabinet .` (протокол испытаний — документ 07).

6. Установка мобильного приложения

1. Скачайте файл APK со страницы `https://фотообъект.рф/download.html` или установите приложение из RuStore.
2. При установке файла APK разрешите установку из этого источника.
3. Обновления устанавливаются поверх предыдущей версии, данные сохраняются (все версии подписаны одним ключом правообладателя).

7. Первичная настройка

1. Войдите в кабинет `https://app.<домен>` учётной записью владельца.
2. **Филиалы** — при необходимости создайте филиалы. **Настройки** → **Пользователи** — добавьте сотрудников и роли.
3. **Шаблоны фото** — добавьте готовые шаблоны или создайте свои, нажмите «Опубликовать».
4. **Бригады** → «Добавить бригаду» — откроется окно с QR-кодом и адресом подключения.
5. На телефоне: «Подключить по QR-коду из офиса» → навести камеру на QR-код (или «Выбрать QR-код из галереи»).
6. **Заказы** → «Новый заказ» — выберите объект, шаблон и бригаду. Заказ появится на телефоне в «Новый отчёт».
7. Пробный выезд: бригада снимает и отправляет отчёт, через 2 минуты он виден в кабинете («Фото от бригад»).
8. По желанию: **Настройки** → «Куда пересылать фото» — сетевая папка или Яндекс Диск.

8. Резервное копирование и обновление

Скрипт `deploy/backup.sh` сохраняет базу кабинета, дампы базы сервера отчётов и папки с фото. Запуск по расписанию:

```
30 3 * * * /opt/fotoobject/deploy/backup.sh >> /var/log/fotoobject-backup.log 2>&1
```

Копии хранятся 7 дней; рекомендуется копировать их на другой сервер. Обновление:

```
cd /opt/fotoobject && git pull && cd deploy && ./backup.sh && docker compose up -d --build
```

9. Работа без иностранных сервисов

Программа не требует активации, лицензионных ключей и подключения к серверам правообладателя или иностранных компаний. Рекомендуемые настройки для полностью автономной работы:

- настройки сервера отчётов из `deploy/central/` (раздел 4): отчёты об ошибках в Sentry выключены, веб-сервер без внешних адресов;
- не задавать `TELEGRAM_BOT_TOKEN` (уведомления о заявках с сайта);
- для импорта реестра использовать файлы Excel/CSV, а не ссылку на онлайн-таблицу;
- при необходимости — HTTPS-сертификат российского удостоверяющего центра вместо автоматического (раздел `tls` в `deploy/Caddyfile`).