

**Автоматизированная информационная система «Тренировочное тестирование для
подготовки к НОК в сфере ЖКХ»**

Версия 1.0.1

Инструкция по установке

Правообладатель: Ряполов Егор Денисович, гражданин Российской Федерации,
зарегистрированный в качестве индивидуального предпринимателя (ОГРНИП
324508100055816)

12 октября 2026 г.

Состав программы. Программа «Автоматизированная информационная система „Тренировочное тестирование для подготовки к НОК в сфере ЖКХ“» (альтернативное наименование — АИС «Тренировочное тестирование для подготовки к НОК в сфере ЖКХ»), версия 1.0.1, состоит из двух частей, которые поставляются вместе в одном дистрибутиве:

1. **Серверная часть** — веб-приложение, планировщик заданий, схема базы данных, сценарии установки и обслуживания. Устанавливается на сервер под управлением Linux.

2. **Модуль обмена** (aiszhk_export) — передаёт серверной части сведения из внешней системы АИС ОК ЖКХ. Устанавливается на сервер системы АИС ОК ЖКХ её владельцем или с его согласия и обращается к её базе данных только на чтение.

В состав программы не входят банк экзаменационных вопросов и ответов, настройки экзаменационных тестов и сведения о соискателях. Это данные лицензиата, которые он получает из системы АИС ОК ЖКХ на основании собственных прав. Не входят также операционная система, СУБД, веб-сервер и интерпретатор Python: они устанавливаются из репозитория операционной системы.

Способы предоставления: интернет-сервис и установка на сервер лицензиата.

Для промышленной эксплуатации нужны: сервер Linux с СУБД MariaDB и веб-сервером nginx; доменное имя и сертификат TLS (работа только по HTTPS); модуль обмена на сервере системы АИС ОК ЖКХ и защищённый канал до него; почтовый сервер для уведомлений.

Демонстрационный режим: модуль обмена работает со встроенным набором вымышленных данных вместо базы системы АИС ОК ЖКХ; письма не отправляются. В этом режиме можно проверить все функции программы без доступа к системе АИС ОК ЖКХ.

1. Состав поставки

Платформа поставляется архивом дистрибутива `oqzhkh-<версия>.tar.gz` с контрольной суммой `oqzhkh-<версия>.tar.gz.sha256`. В архиве:

Каталог / файл	Назначение
<code>webapp/, services/, importers/, templates/, static/</code>	Программный код и интерфейс Платформы
<code>aiszkh_export/</code>	Модуль обмена (устанавливается на сервере системы АИС ОК ЖКХ)
<code>migrations/, migrate.py</code>	Схема базы данных и её обновление
<code>wheelhouse/</code>	Все библиотеки Python, необходимые для работы, — устанавливаются из комплекта поставки, без обращения к репозиторию PyPI
<code>install.sh</code>	Сценарий установки
<code>scripts/demo_stand.sh</code>	Развёртывание демонстрационного стенда с демо-данными
<code>scripts/backup.sh, aiszkh-backup.*</code>	Резервное копирование
<code>nginx_test_aiszkh.conf, nginx_ratelimit.conf, nginx_demo.conf</code>	Настройки веб-сервера
<code>test_aiszkh.service, test_aiszkh_scheduler.service, aiszkh_tunnel.service</code>	Службы systemd
<code>THIRD_PARTY_LICENSES.md, LICENSES/</code>	Перечень сторонних компонентов и тексты их лицензий
<code>CHANGELOG.md</code>	История версий

Проверка целостности архива:

```
sha256sum -c oqzhkh-1.0.1.tar.gz.sha256
```

2. Требования к серверу

Параметр	Значение
Процессор	2 виртуальных ядра x86_64 и более
Оперативная память	4 ГБ и более
Диск	20 ГБ и более
Операционная система	Ubuntu 24.04 LTS (с Python 3.12)
Сеть	Доступ пользователей по HTTPS (порт 443; порт 80 — только перенаправление на HTTPS и выпуск сертификата); доступ сервера к репозиториям операционной системы на время установки
Права	Учётная запись с правами root (sudo)

Системные компоненты (MariaDB, nginx, Python, шрифты DejaVu) не входят в поставку: сценарий устанавливает их из репозитория операционной системы. Библиотеки Python ставятся из каталога `wheelhouse/` комплекта поставки.

3. Установка

1. Скопировать архив на сервер и распаковать:

```
tar -xzf oqzhkh-1.0.1.tar.gz cd oqzhkh-1.0.1
```

2. Запустить сценарий установки:

```
sudo bash install.sh
```

Сценарий выполняет:

- установку системных пакетов: `python3`, `python3-venv`, `nginx`, `mariadb-server`, `fonts-dejavu-core`;
- копирование Платформы в каталог `/opt/test_aiszh_app`;
- создание виртуального окружения Python и установку библиотек из `wheelhouse/`;
- создание файла настроек `/opt/test_aiszh_app/.env` из `.env.example` с генерацией секретного ключа, ключей шифрования персональных данных и пароля базы данных;
- создание базы данных `test_aiszh` и её пользователя в MariaDB;
- создание таблиц базы данных (миграции);
- регистрацию служб `systemd` `test_aiszh` (веб-приложение), `test_aiszh_scheduler` (планировщик) и таймера резервного копирования `aiszh-backup.timer`.

3. Создать учётную запись администратора:

```
cd /opt/test_aiszh_app sudo -u www-data bash -c 'set -a; . ./env; set +a; venv/bin/python manage.py create-admin admin admin@example.ru'
```

Пароль администратора вводится с клавиатуры (не короче 8 символов).

4. Запустить службы:

```
sudo systemctl start test_aiszkh test_aiszkh_scheduler
```

5. Настроить веб-сервер (раздел 5) и открыть Платформу в браузере.

4. Настройка

Настройки хранятся в файле `/opt/test_aiszkh_app/.env`. После изменения — перезапуск служб: `sudo systemctl restart test_aiszkh test_aiszkh_scheduler`.

Параметр	Назначение
BASE_URL	Адрес Платформы для ссылок в письмах, например <code>https://test.aiszkh.ru</code>
SUPPORT_EMAIL	Адрес службы поддержки, указывается в письмах и на странице «О программе»
AISZKH_DB_*	Подключение к базе данных Платформы (заполняются при установке)
PDN_ENCRYPTION_KEY, EMAIL_HMAC_KEY	Ключи шифрования персональных данных (генерируются при установке; утрата ключа делает данные нечитаемыми — хранить копию)
AISZKH_API_URL, AISZKH_API_KEY	Адрес и ключ доступа модуля обмена
AISZKH_BUNDLE_URL, AISZKH_BUNDLE_TOKEN	Адрес и ключ выгрузки банка вопросов
AISZKH_SYNC_MINUTES	Период получения заявлений, минут (по умолчанию 5)
AISZKH_DICT_SYNC_HOUR, AISZKH_BANK_SYNC_HOUR	Час ежесуточного обновления справочников и банка вопросов
LOGIN_MAX_FAILURES, LOGIN_MAX_FAILURES_IP, LOGIN_WINDOW_MINUTES	Защита от подбора пароля
SESSION_COOKIE_SECURE	1 — cookie сеанса передаются только по HTTPS. В промышленной эксплуатации обязательно 1 ; 0 допускается только для изолированного демонстрационного стенда по HTTP
MAIL_RECIPIENT_ALLOWLIST	Ограничение адресатов писем (пусто — без ограничения)

Параметры почтового сервера (SMTP) задаются администратором в интерфейсе: «Почта → Настройки».

5. Веб-сервер

Промышленная эксплуатация — только по HTTPS с действующим сертификатом TLS; в

```
.env BASE_URL=https://... и SESSION_COOKIE_SECURE=1.
```

```
sudo cp /opt/test_aiszh_app/nginx_ratelimit.conf /etc/nginx/conf.d/ratelimit.conf
sudo cp /opt/test_aiszh_app/nginx_test_aiszh.conf
/etc/nginx/sites-available/test_aiszh
sudo ln -s /etc/nginx/sites-available/test_aiszh /etc/nginx/sites-enabled/
```

В файле /etc/nginx/sites-available/test_aiszh указать доменное имя (server_name) и пути к сертификату и ключу TLS (ssl_certificate, ssl_certificate_key). Затем выполнить `sudo nginx -t` и `sudo systemctl reload nginx`.

Работа по HTTP допускается только для демонстрационного стенда в изолированной сети или виртуальной машине без реальных данных (раздел 7).

6. Подключение к системе АИС ОК ЖКХ

Модуль обмена устанавливается на сервере системы АИС ОК ЖКХ (Windows Server, PostgreSQL):

1. Установить Python 3.12 и скопировать каталог `aiszh_export/` в `C:\app\aiszh_export`.
2. Выполнить от имени администратора сценарий `aiszh_export\deploy_windows.ps1`. Он создаёт виртуальное окружение, устанавливает зависимости и регистрирует автозапуск модуля. Модуль слушает только адрес `127.0.0.1:8789`.
3. Создать в PostgreSQL учётную запись только для чтения и указать её в файле `.env.export` модуля. Порядок описан в `aiszh_export/README.md`.
4. На сервере Платформы настроить зашифрованный канал (SSH-туннель):

```
sudo apt install autossh sudo ssh-keygen -t ed25519 -f /root/.ssh/aiszh_tunnel -N '' # открытый ключ /root/.ssh/aiszh_tunnel.pub добавить на сервере АИС ОК ЖКХ echo 'AISZH_TUNNEL_TARGET=<пользователь>@<адрес сервера АИС ОК ЖКХ>' | sudo tee /etc/default/aiszh_tunnel echo 'AISZH_TUNNEL_KEY=/root/.ssh/aiszh_tunnel' | sudo tee -a /etc/default/aiszh_tunnel sudo cp /opt/test_aiszh_app/aiszh_tunnel.service /etc/systemd/system/ sudo systemctl daemon-reload && sudo systemctl enable --now aiszh_tunnel
```

5. В `.env` Платформы указать `AISZH_API_URL=http://127.0.0.1:8789/api/v1`, `AISZH_BUNDLE_URL=http://127.0.0.1:8789/bundle/` и ключи доступа, заданные в модуле.
6. Выполнить первичную загрузку:

```
cd /opt/test_aiszh_app sudo -u www-data bash -c 'set -a; . ./env; set +a; \
venv/bin/python manage.py aiszh-dict-sync && \ venv/bin/python manage.py aiszh-
bank-sync --force && \ venv/bin/python manage.py aiszh-sync'
```

Далее обмен выполняется планировщиком автоматически.

7. Демонстрационный стенд

Для ознакомления и проверки Платформа разворачивается с демонстрационными данными, без подключения к системе АИС ОК ЖКХ. Модуль обмена работает в демонстрационном

режиме на том же сервере и отдаёт вымышленные заявления, ЦОК, СПК и вопросы. Письма не отправляются.

```
sudo bash install.sh
sudo bash scripts/demo_stand.sh
```

Пароль администратора стенда не передаётся в командной строке (он сохранился бы в истории команд и был бы виден в списке процессов): сценарий запрашивает его с клавиатуры. Для запуска без терминала пароль передаётся переменной окружения — `sudo --preserve-env=ADMIN_PASSWORD bash scripts/demo_stand.sh` — либо, если она не задана, генерируется и выводится один раз в конце работы сценария.

По завершении сценарий выводит адрес стенда и учётные записи для входа:

Роль	Логин	Пароль
Администратор	admin	введённый или сгенерированный при запуске
Тестируемый (доступ открыт)	petrova_as, kuznetsov_dd	Демо-2026
Тестируемый (экзамен через 30 дней, доступ ещё закрыт)	ivanov_ii	Демо-2026
Оператор ЦОК	op.stolichny@demo.aiszh.ru	Демо-2026
Оператор СПК	op.spk@demo.aiszh.ru	Демо-2026

Второй экземпляр на том же сервере. Демонстрационный стенд можно развернуть рядом с рабочим экземпляром: отдельный каталог, база данных, порты и службы задаются переменными окружения.

```
sudo APP_DIR=/opt/test_zhkh_demo SERVICE=test_zhkh_demo APP_PORT=5100 WORKERS=2 \
  DB_NAME=test_zhkh_demo DB_USER=test_zhkh_demo_user bash install.sh
sudo APP_DIR=/opt/test_zhkh_demo SERVICE=test_zhkh_demo EXPORT_PORT=8799 \
  MOCK_SERVICE=test_zhkh_demo_export BASE_URL=https://demo.<домен> NGINX=0 \
  bash scripts/demo_stand.sh
```

Веб-сервер для такого экземпляра настраивается отдельным блоком `server` с его доменным именем (пример — `nginx_demo_test_zhkh.conf`).

8. Проверка работоспособности

```
systemctl status test_aiszh test_aiszh_scheduler
curl -s http://127.0.0.1:5000/health # {"status": "ok"}
```

Открыть в браузере адрес Платформы, войти под администратором, открыть «Отчёты → Платформа → Сводка по платформе».

9. Резервное копирование и восстановление

Таймер `aiszkh-backup.timer` ежедневно в 02:30 сохраняет в `/var/backups/test_aiszkh` дампы базы данных (`db-<дата>.sql.gz`) и копию файла настроек `.env` (`env-<дата>`) с ключами шифрования. Копии хранятся 30 дней. Рекомендуется дополнительно переносить их на внешнее хранилище.

Восстановление:

```
sudo systemctl stop test_aiszkh test_aiszkh_scheduler
sudo cp /var/backups/test_aiszkh/env-<дата> /opt/test_aiszkh_app/.env
set -a; . /opt/test_aiszkh_app/.env; set +a
gunzip -c /var/backups/test_aiszkh/db-<дата>.sql.gz | mysql -u "$AISZKH_DB_USER" -
p"$AISZKH_DB_PASSWORD" "$AISZKH_DB_NAME"
sudo systemctl start test_aiszkh test_aiszkh_scheduler
```

10. Обновление версии

1. Распаковать архив новой версии.
2. Скопировать файлы поверх `/opt/test_aiszkh_app`, сохранив `.env` и каталог `media/`.
3. Установить библиотеки: `sudo -u www-data /opt/test_aiszkh_app/venv/bin/pip install --no-index --find-links wheelhouse -r requirements.txt`.
4. Применить миграции: `venv/bin/python migrate.py up`.
5. Перезапустить службы.