

Общество с ограниченной ответственностью «АКТИОН 360 УРАЛ ИНФО»

Корпоративная система обучения Урал Инфо 360

Руководство администратора

Установка и эксплуатация

2026

Содержание

1. Назначение и состав системы
2. Варианты использования
3. Требования к серверу и рабочим местам
4. Установка на сервер
5. Параметры настройки
6. Резервное копирование и восстановление
7. Обновление
8. Работа администратора в интерфейсе
 - 8.1. Курсы и редактор курса
 - 8.2. Тесты
 - 8.3. Рубрики каталога и нормативные акты
 - 8.4. Сотрудники, учащиеся, группы
 - 8.5. Журнал действий и шаблоны писем
9. Служебные команды
10. Техническая поддержка

1. Назначение и состав системы

Корпоративная система обучения Урал Инфо 360 (далее — система) предназначена для дистанционного обучения работников организации: размещения учебных курсов, прохождения уроков, учебных игр, трёхмерных тренажёров и тестов, учёта прогресса, оформления документов об обучении, работы учебных групп с кураторами, аналитики и ведения раздела локальных нормативных актов. Работа обучающегося и куратора описана в документе «Руководство пользователя».

Система состоит из следующих частей:

- веб-интерфейс — одностраничное приложение (React, TypeScript), которое отдаётся браузеру как набор статических файлов;
- сервер приложения (Node.js, Express, TypeScript) — программный интерфейс для веб-интерфейса, обмен сообщениями чата в реальном времени (Socket.IO), формирование PDF-документов, рассылка уведомлений и плановые задания (напоминания, сроки обучения);
- база данных PostgreSQL — учётные записи, курсы, прогресс, результаты тестов, журнал действий;
- файловое хранилище на диске сервера — изображения и модели уроков, PDF-документы, сертификаты, зашифрованные сканы личных документов;
- веб-сервер nginx — принимает запросы по HTTPS, отдаёт веб-интерфейс и передаёт запросы к программному интерфейсу серверу приложения.

В системе три роли: обучающийся, куратор учебной группы и администратор. Администратор имеет доступ ко всем функциям, включая управление курсами, сотрудниками, нормативными актами, журналом действий и шаблонами писем.

2. Варианты использования

Работа на сервере правообладателя. Система размещена на сервере ООО «АКТИОН 360 УРАЛ ИНФО» и доступна по адресу <https://uralinfo360.ru/app/>. Организации-заказчику ничего устанавливать не нужно: правообладатель заводит администратора организации, дальше работа идёт через браузер. Разделы 4–7 этого документа в таком варианте выполняет правообладатель.

Установка на сервер заказчика. Правообладатель передаёт дистрибутив (исходный код в репозитории git), заказчик устанавливает систему на свой сервер по разделу 4 и обслуживает её по разделам 5–7.

3. Требования к серверу и рабочим местам

Сервер:

Параметр	Требование
Операционная система	Linux. Эксплуатационный экземпляр работает на Ubuntu 24.04 LTS; инструкция ниже написана для неё
Node.js	версия 20 или новее (эксплуатационный экземпляр — 22), вместе с npm
PostgreSQL	версия 16 или новее
Веб-сервер	nginx (эксплуатационный экземпляр — 1.24)
Процессор, память, диск	рекомендуется не меньше 4 ядер, 8 ГБ оперативной памяти и 80 ГБ на диске (на такой конфигурации работает эксплуатационный экземпляр); место на диске растёт вместе с числом курсов и загруженных файлов
Прочее	доменное имя и сертификат TLS (например, Let's Encrypt через certbot); почтовый сервер SMTP для писем пользователям

Рабочее место пользователя: компьютер, планшет или телефон с актуальной версией Яндекс Браузера, Google Chrome, Microsoft Edge, Mozilla Firefox или Safari и доступом к серверу системы. Для трёхмерных тренажёров браузер должен поддерживать WebGL. Установка программ на рабочее место не требуется.

4. Установка на сервер

Команды ниже выполняются от имени администратора сервера (через `sudo`). Каталог установки — `/var/www/ural-info360`, служебный пользователь — `deploy`; их можно заменить своими, поправив пути в файлах из каталога `deploy/`.

Шаг 1. Установить пакеты.

```
sudo apt update
sudo apt install -y nginx postgresql git certbot python3-certbot-nginx
# Node.js 22 — из репозитория NodeSource или другим принятым в организации
способом
node -v    # должно быть v20 или новее
```

Шаг 2. Завести пользователя и получить дистрибутив.

```
sudo adduser --system --group --home /var/www deploy
sudo mkdir -p /var/www/ural-info360 && sudo chown deploy:deploy
/var/www/ural-info360
sudo -u deploy git clone <адрес репозитория, выданный правообладателем>
/var/www/ural-info360
```

Шаг 3. Создать базу данных.

```
sudo -u postgres psql -c "CREATE USER lms WITH PASSWORD '<пароль>';"
sudo -u postgres psql -c "CREATE DATABASE lms OWNER lms;"
```

Шаг 4. Заполнить файл настроек. Скопируйте `.env.example` в `.env` в корне каталога установки и заполните параметры по разделу 5. Обязательно задайте строку подключения к базе, два секрета JWT и ключ шифрования личных документов:

```
cd /var/www/ural-info360
sudo -u deploy cp .env.example .env
openssl rand -hex 32          # ключ для PERSONAL_DOCS_ENCRYPTION_KEY
openssl rand -base64 48      # по одному значению для JWT_ACCESS_SECRET и
JWT_REFRESH_SECRET
sudo -u deploy nano .env
sudo chmod 600 .env
```

Шаг 5. Установить зависимости, создать таблицы и собрать систему.

```
cd /var/www/ural-info360
sudo -u deploy -H npm ci
sudo -u deploy -H npm run generate -w backend
sudo -u deploy -H npm run migrate:deploy -w backend
sudo -u deploy -H npm run build -w shared
sudo -u deploy -H npm run build -w backend
sudo -u deploy -H npm run build -w frontend
```

Перед сборкой веб-интерфейса проверьте, что в `frontend/.env.production` или в окружении сборки не задано `VITE_USE MOCKS=true`: этот режим подменяет сервер тестовыми данными и нужен только разработчикам.

Шаг 6. Запустить сервер приложения как службу. В репозитории есть готовый файл службы `systemd`:

```
sudo mkdir -p /var/www/ural-info360/backend/storage
sudo chown deploy:deploy /var/www/ural-info360/backend/storage
sudo cp deploy/systemd/lms-backend.service /etc/systemd/system/
sudo systemctl daemon-reload
sudo systemctl enable --now lms-backend
curl http://127.0.0.1:4000/health      # ответ означает, что сервер
приложения работает
```

Служба работает от пользователя `deploy`, перезапускается при сбое и пишет журнал в `journald` (`journalctl -u lms-backend`).

Шаг 7. Настроить nginx и HTTPS. Образец конфигурации — `deploy/nginx/ural-info360.conf`: веб-интерфейс отдаётся по адресу `/app/` из `frontend/dist`, запросы `/api/v1/` и `/socket.io/` передаются серверу приложения на порт 4000. Замените в файле доменное имя на своё. Сертификат удобно сначала получить с упрощённой конфигурацией `deploy/nginx/ural-info360-bootstrap.conf`:

```
sudo cp deploy/nginx/ural-info360-bootstrap.conf /etc/nginx/sites-available/ural-info360.conf
sudo ln -s /etc/nginx/sites-available/ural-info360.conf /etc/nginx/sites-enabled/
sudo nginx -t && sudo systemctl reload nginx
sudo certbot certonly --webroot -w /var/www/certbot -d <домен>
sudo cp deploy/nginx/ural-info360.conf /etc/nginx/sites-available/ural-info360.conf
sudo nginx -t && sudo systemctl reload nginx
```

Шаг 8. Завести первого администратора. Пользователь сам регистрируется только с ролью обучающегося и только по коду группы, роль администратора выдаётся командой на сервере. На пустой базе групп ещё нет, поэтому сначала запустите начальное наполнение — оно создаёт демонстрационный курс, две группы и демонстрационные учётные записи с паролем Password123:

```
sudo -u deploy -H npm run seed -w backend
```

Затем откройте <https://<домен>/app/register>, зарегистрируйте будущего администратора по коду группы URAL-KTM234 и выдайте ему роль:

```
sudo -u deploy -H npm run grant:role -w backend -- --email
admin@organization.ru --role ADMIN
```

После этого смените пароли демонстрационных учётных записей (раздел 9) и снимите демонстрационный курс с публикации в разделе «Курсы».

Шаг 9. Включить резервное копирование — по разделу 6.

Проверка установки. Откройте <https://<домен>/app/>, войдите администратором: после пароля на почту придёт код подтверждения. Если письмо не пришло, проверьте параметры SMTP и MAIL_DRY_RUN=false (раздел 5).

5. Параметры настройки

Все параметры задаются в файле `.env` в корне каталога установки, после изменения нужно перезапустить службу: `sudo systemctl restart lms-backend`. Полный перечень с пояснениями — в файле `.env.example`. Основные параметры:

Параметр	Назначение
DATABASE_URL	строка подключения к PostgreSQL
PORT, CORS_ORIGIN	порт сервера приложения (по умолчанию 4000) и адрес сайта, с которого открывается веб-интерфейс
JWT_ACCESS_SECRET, JWT_REFRESH_SECRET	секреты для подписи сеансов пользователей, не короче 32 символов, у каждой установки

	свои
PERSONAL_DOCS_ENCRYPTION_KEY	ключ шифрования сканов личных документов (AES-256-GCM), 64 шестнадцатеричных символа. Без него сервер не запустится. Потерянный ключ не восстанавливается: без него зашифрованные файлы прочитать нельзя. Храните ключ вместе с паролем базы данных
STORAGE_ROOT, STORAGE_MAX_FILE_SIZE	каталог файлового хранилища и предельный размер загружаемого файла (по умолчанию 50 МБ)
SMTP_HOST, SMTP_PORT, SMTP_SECURE, SMTP_USER, SMTP_PASSWORD, MAIL_FROM	почтовый сервер для писем пользователям (коды входа, приглашения, уведомления)
MAIL_DRY_RUN	true — письма не отправляются, а пишутся в журнал. На рабочей установке должно быть false
TELEGRAM_BOT_TOKEN, TELEGRAM_BOT_USERNAME, TELEGRAM_WEBHOOK_SECRET	уведомления в Telegram. Пустой токен отключает канал
VK_GROUP_TOKEN, VK_GROUP_SLUG, VK_WEBHOOK_SECRET, VK_CONFIRMATION_TOKEN	уведомления через сообщество ВКонтакте. Пустой токен отключает канал
NOTIFICATIONS_DRY_RUN	true — сообщения в мессенджеры не отправляются
INACTIVITY_THRESHOLD_DAYS, INACTIVITY_REMINDER_COOLDOWN_DAYS, INACTIVITY_CRON	через сколько дней без занятий напоминать обучающемуся, как часто повторять напоминание и когда проводить проверку (по умолчанию 7 дней, раз в 7 дней, ежедневно в 09:00)
DEADLINE_CRON	время ежедневной проверки сроков обучения: обучающемуся — напоминания за 14, 7 и 1 день, куратору — сообщение о просрочке
CERTIFICATE_ORGANIZATION, CERTIFICATE_FONT_REGULAR, CERTIFICATE_FONT_BOLD	наименование организации в сертификатах и шрифты с кириллицей для PDF

6. Резервное копирование и восстановление

Резервная копия состоит из двух частей: дампа базы данных и копии файлового хранилища. Нужны обе: в базе хранятся ссылки на файлы, а сами файлы — в хранилище.

Скрипт `deploy/scripts/backup.sh` делает обе копии, проверяет, что дамп читается, и удаляет копии старше 14 дней. Чтобы он запускался каждую ночь в 03:30:

```
sudo cp deploy/cron/lms-backup /etc/cron.d/lms-backup
sudo chmod 644 /etc/cron.d/lms-backup
```

Журнал копирования — `/var/log/lms-backup.log`. Копии желательно регулярно переносить на другой сервер или носитель.

Восстановление: остановите службу, восстановите базу из дампа и верните каталог хранилища, затем запустите службу.

```
sudo systemctl stop lms-backend
sudo -u postgres pg_restore --clean --if-exists -d lms <файл дампа>.dump
# вернуть каталог backend/storage из той же копии
sudo systemctl start lms-backend
```

7. Обновление

Перед обновлением сделайте резервную копию базы (раздел 6). Затем получите новую версию и повторите сборку:

```
cd /var/www/ural-info360
sudo -u deploy git pull
sudo -u deploy -H npm ci
sudo -u deploy -H npm run generate -w backend
sudo -u deploy -H npm run migrate:deploy -w backend
sudo -u deploy -H npm run build -w shared
sudo -u deploy -H npm run build -w backend
sudo -u deploy -H npm run build -w frontend
sudo systemctl restart lms-backend
```

Команда `migrate:deploy` применяет изменения структуры базы данных, если они есть в новой версии; данные при этом сохраняются. Все команды выполняются от пользователя `deploy`: файлы, созданные от `root`, служба потом не сможет перезаписать.

8. Работа администратора в интерфейсе

Администратор входит в систему так же, как другие пользователи, с кодом подтверждения из письма. Раздел управления открывается из меню пользователя пунктом «Перейти в управление». Слева — области «Учебный контент», «Люди», «Сайт компании» и «Отчёты». Функции куратора (сводка, группы, выпускники, загрузки) описаны в «Руководстве пользователя», раздел 4; администратору они доступны по всей организации.

8.1. Курсы и редактор курса

«Учебный контент» → «Курсы» — список курсов. Кнопка «Создать курс» создаёт пустой курс, «Импорт .md» загружает готовый курс из файлов. В меню курса —

«Опубликовать» / «Снять с публикации» и удаление. Обучающиеся видят только опубликованные курсы.

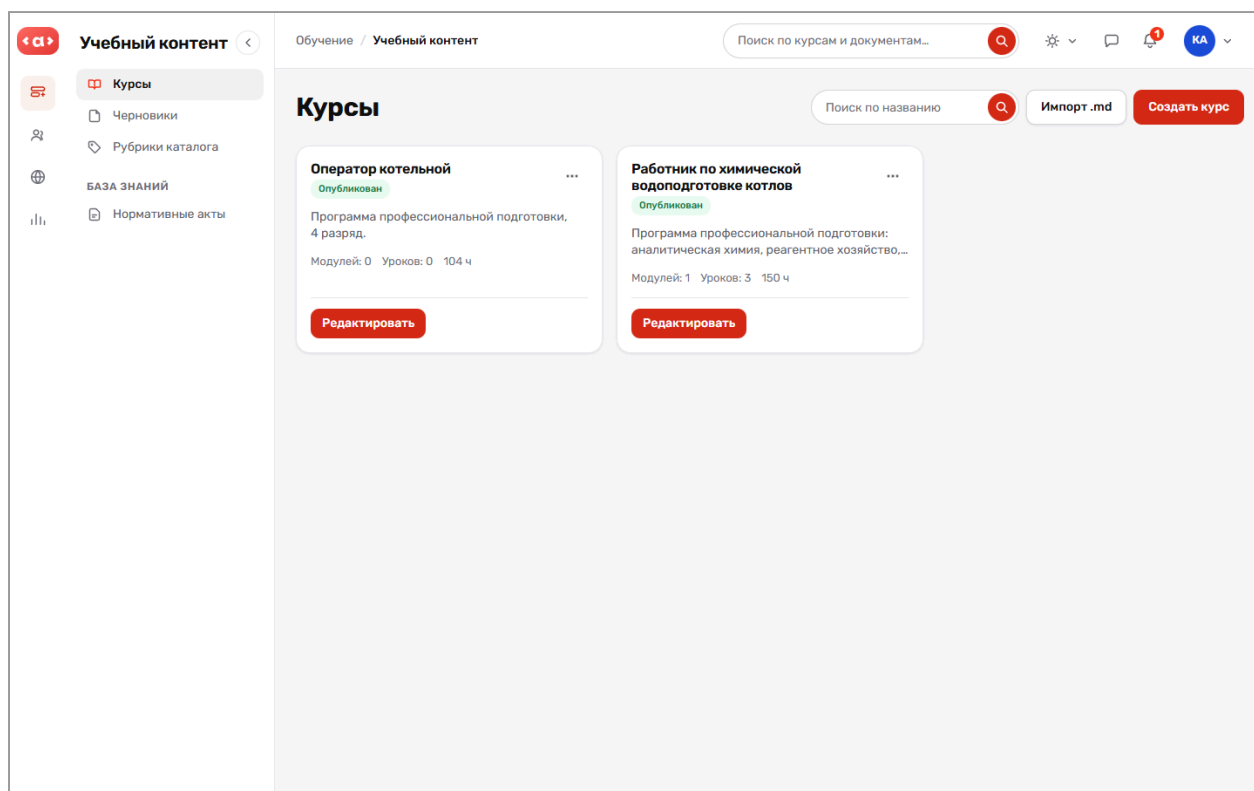


Рисунок 1 — Список курсов

Редактор курса разделён на три части. Слева — структура: модули и уроки, их добавляют кнопками «+ Добавить модуль» и «+ Урок», переименовывают и перетаскивают мышью. Урок можно сделать зачётным («Сделать зачётом»). В середине — содержимое урока из блоков: заголовок, текст, колонки, текст с картинкой, изображение, видео, файл для скачивания, таблица, формула, врезка, код, спойлер, пошаговая инструкция, цитата, параметр и значение, учебная игра. Блоки добавляются из палитры «Добавить блок» и переставляются перетаскиванием. Справа — свойства выбранного блока, оформление курса (шрифты, акцентный цвет) и сведения о курсе: описание, часы и квалификация для документа об обучении, состав итоговой аттестации, рубрика каталога.

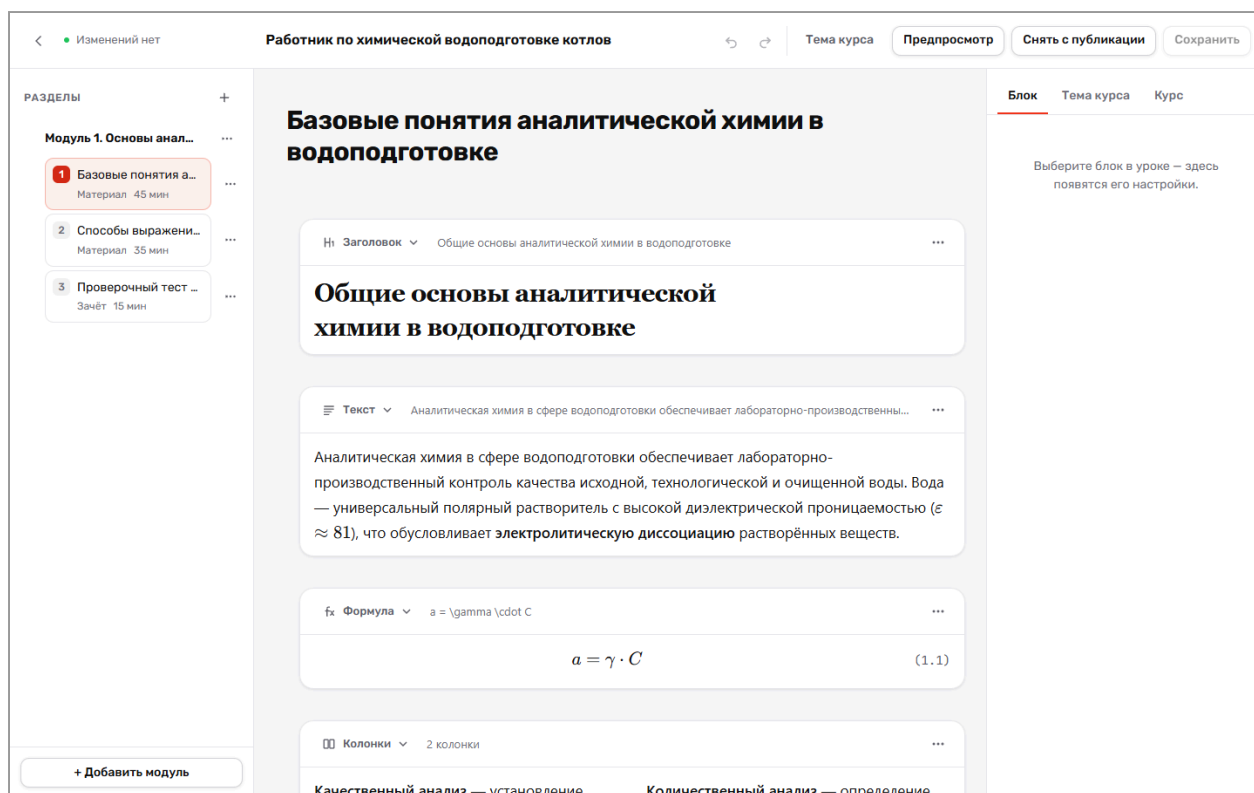


Рисунок 2 — Редактор курса

Кнопка «Предпросмотр» показывает урок так, как его увидит обучающийся (прогресс при этом не ведётся). Правки сохраняются кнопкой «Сохранить», последние действия отменяются сочетаниями **Ctrl+Z** и **Ctrl+Shift+Z**. Неопубликованные правки уроков собраны в разделе «Черновики».

Импорт из Markdown: в окне «Импорт курса из Markdown» нажмите «Выбрать папку курса». Вложенные папки станут модулями, файлы **.md** — уроками. Кнопка «Показать, что получится» показывает структуру до загрузки, «Импортировать» создаёт курс.

8.2. Тесты

Тест — урок особого типа. В редакторе теста слева список вопросов, в середине текст вопроса и варианты ответов, справа тип вопроса, баллы и пояснение к ответу. Типы вопросов: один верный ответ, несколько верных, свободный ответ, пропуски в тексте, сопоставление. На вкладке «Тест» задаются проходной балл, число попыток (или без ограничений), перемешивание, число вопросов в одной попытке (случайная выборка из всех вопросов теста) и время на попытку.

Сохранено

Проверочный тест по модулю 1

Предпросмотр

Сохранить

вопросы

1

Чему равно ионное п...

Один верный ответ

2

Какие методы относ...

Несколько верных ответ...

3

Раствор с pH называ...

Пропуски в тексте

4

Сопоставьте обозна...

Сопоставление

+ Добавить вопрос

Вопрос 1

ОДИН ВЕРНЫЙ ОТВЕТ

Текст вопроса *

Чему равно ионное произведение воды K_w при 25°C ?

Markdown: **жирный**, формулы через $\$...\$$

ТАК ВОПРОС УВИДИТ СТУДЕНТ

Чему равно ионное произведение воды K_w при 25°C ?

Отметьте один верный вариант

☐

10^{-7}

10^{-7}

x

☒

10^{-14}

10^{-14}

x

☐

10^{-21}

10^{-21}

x

Добавить вариант

Вопрос

Тест

вопрос

Тип вопроса

Один верный ответ

Введённое сохранится — сменить тип можно без потерь

Баллы

1

0 — вопрос не влияет на оценку

РАЗБОР

Пояснение

При 25°C $K_w = [\text{H}^+] \cdot [\text{OH}^-] = 10^{-14} \text{ (моль/л)}^2$.

Показывается в разборе после отправки. Markdown, формулы через $\$...\$$

ТАК ПОЯСНЕНИЕ УВИДИТ СТУДЕНТ

При 25°C $K_w = [\text{H}^+] \cdot [\text{OH}^-] = 10^{-14} \text{ (моль/л)}^2$.

Рисунок 3 — Редактор теста

8.3. Рубрики каталога и нормативные акты

«Рубрики каталога» — разделы, по которым курсы разложены в каталоге: рубрики добавляют, переименовывают, меняют местами и удаляют. Курс относится к рубрике в настройках курса.

«Нормативные акты» — приказы, положения и инструкции организации. Кнопка «Создать документ» создаёт текстовый документ в том же блочном редакторе или загружает PDF. Документ доступен всем сотрудникам после публикации; кнопки «Опубликовать» / «Снять» и удаление — в строке документа.

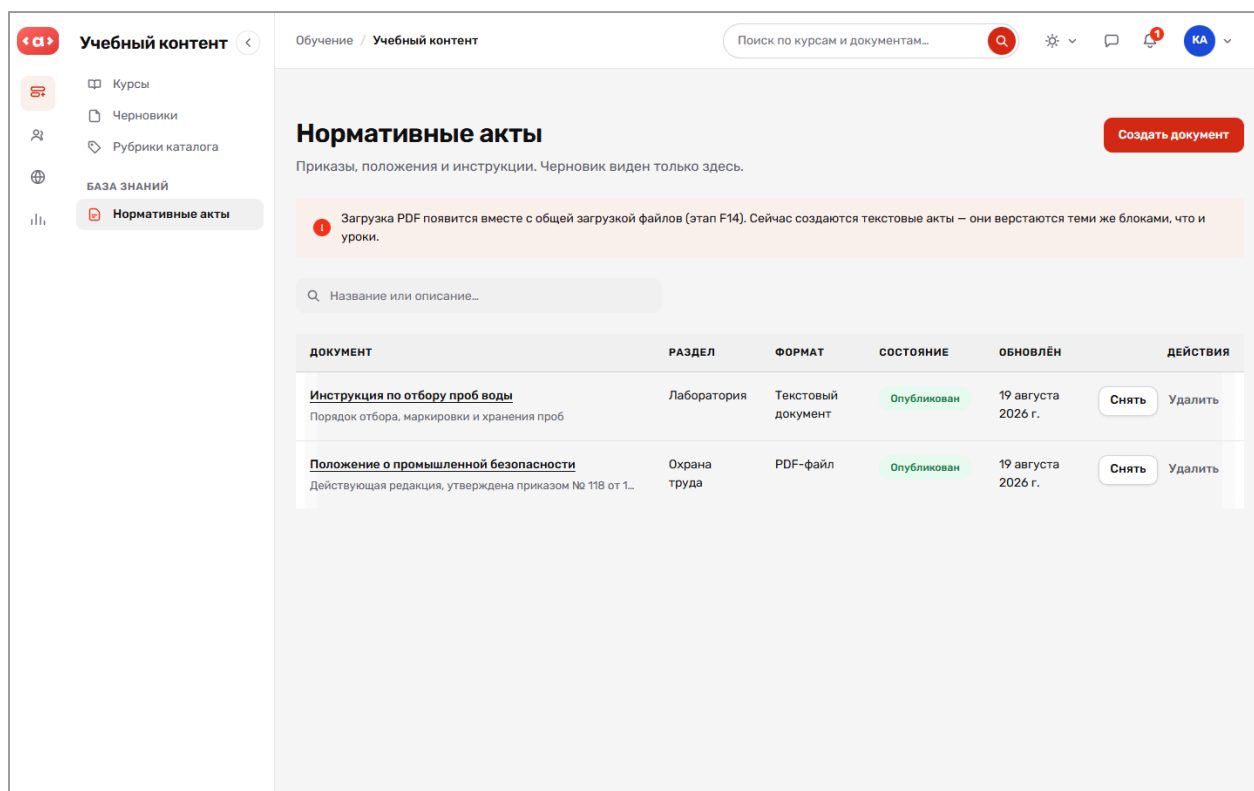


Рисунок 4 — Нормативные акты

8.4. Сотрудники, учащиеся, группы

«Люди» → «Сотрудники» — администраторы и кураторы. Кнопка «Пригласить» отправляет приглашение на почту: по ссылке из письма сотрудник задаёт пароль. «Выслать заново» повторяет письмо, «Справка» выгружает сведения о сотруднике.

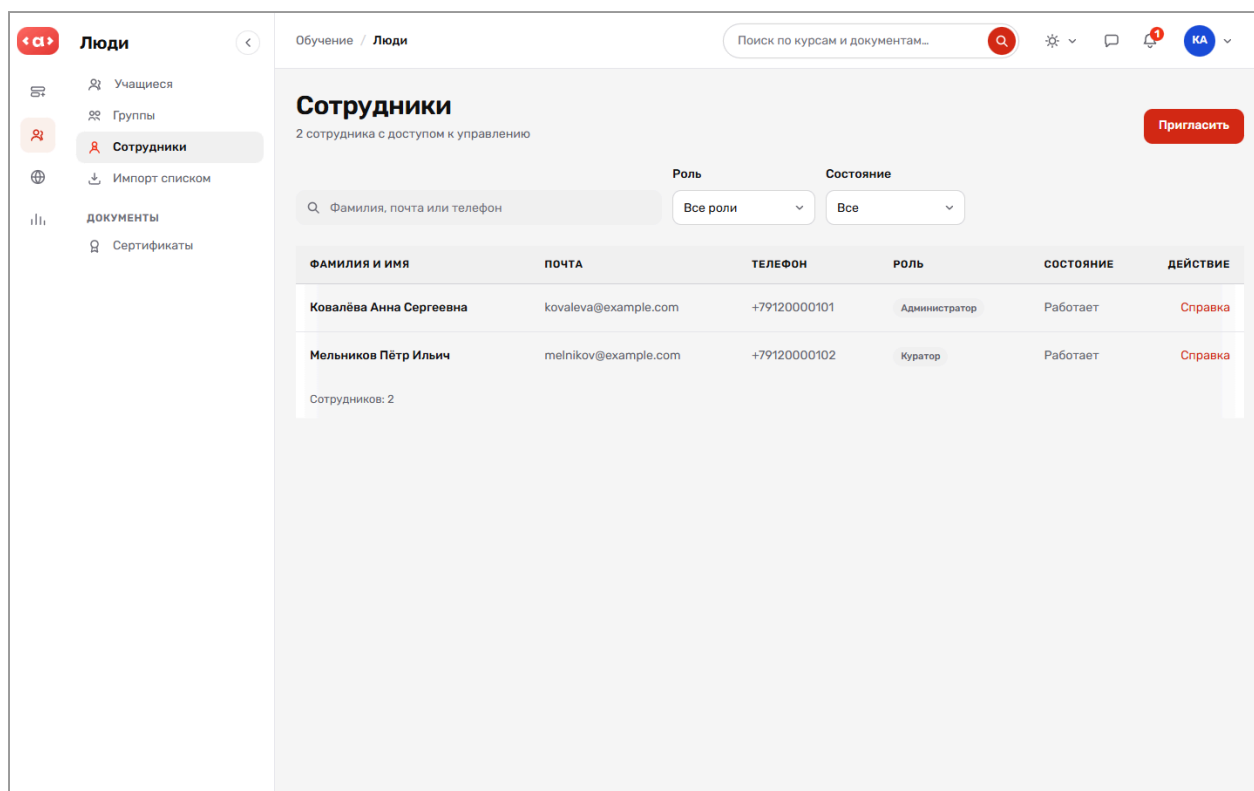


Рисунок 5 — Сотрудники

«Учащиеся», «Группы» и «Импорт списком» устроены так же, как у куратора, но администратор видит всех пользователей и все группы, создаёт группы и назначает им кураторов. У администратора в строке учащегося есть ссылка «Документы» — сканы личных документов для оформления документа об обучении. Каждый просмотр таких документов записывается в журнал доступа.

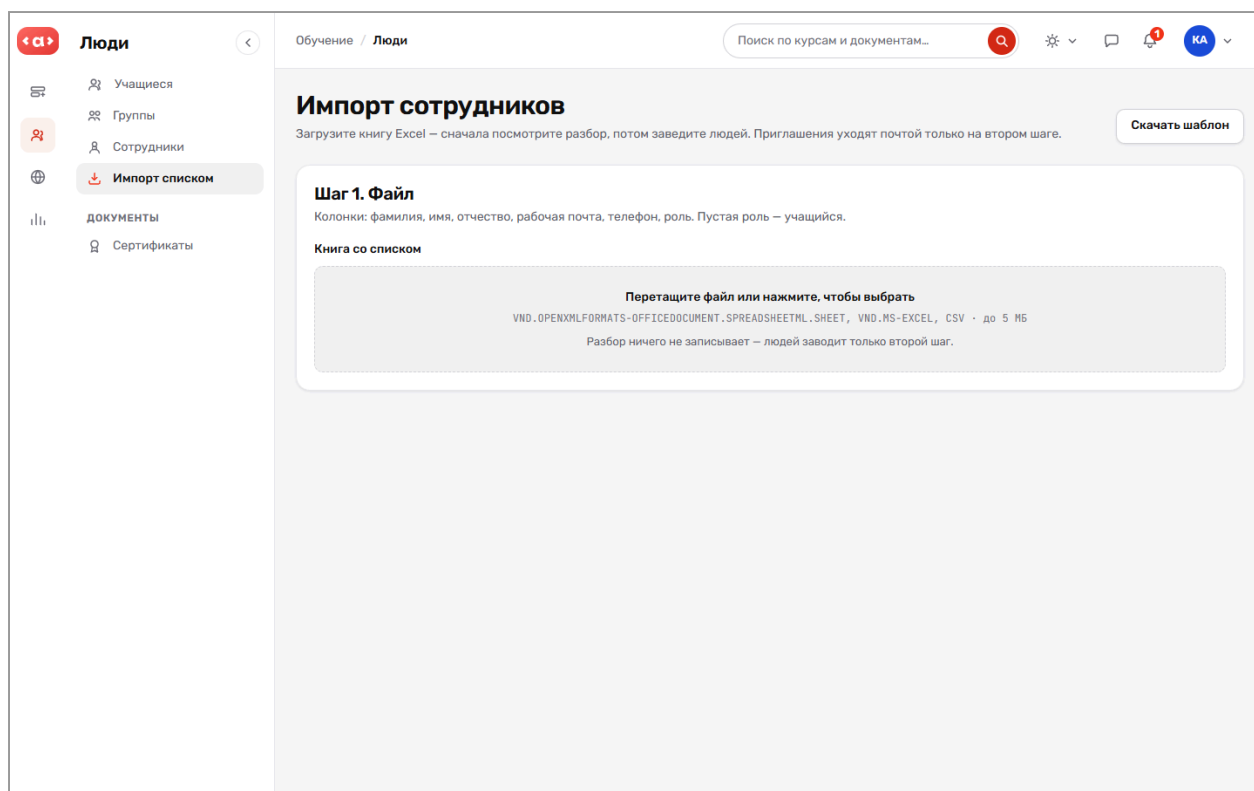


Рисунок 6 — Импорт сотрудников из Excel

8.5. Журнал действий и шаблоны писем

«Отчёты» → «Журнал действий» — кто, когда и с какого адреса выполнил действие в системе: вход, назначение курса, изменение ролей, работу с документами. Записи фильтруются по действию, объекту и периоду; кнопка «Только этого» оставляет действия одного пользователя.

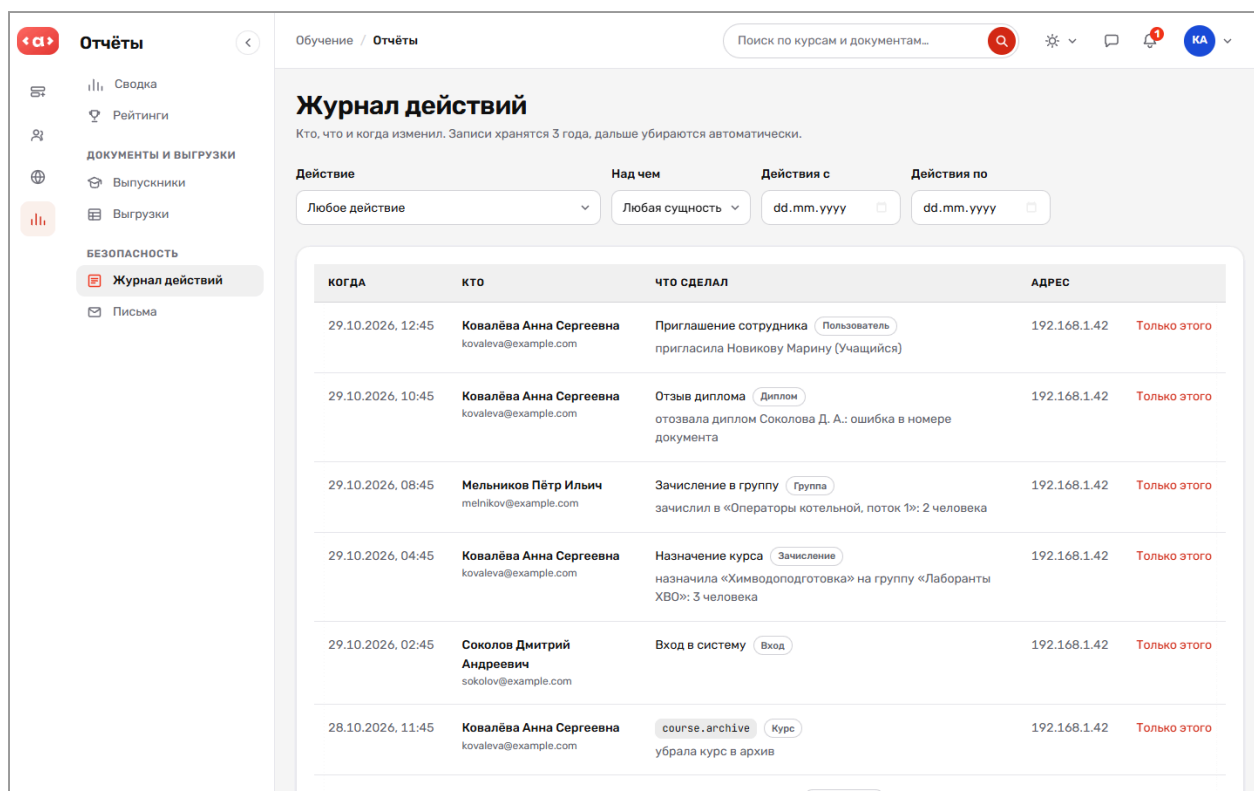


Рисунок 7 — Журнал действий

«Отчёты» → «Письма» — тексты писем, которые система отправляет пользователям: о назначении курса, напоминания, о выдаче документа и другие. Текст письма собирается из абзацев, приписок, кнопок, списков и цитат; справа видно, как письмо будет выглядеть. Кнопка «Вернуть текст из кода» восстанавливает исходный текст.

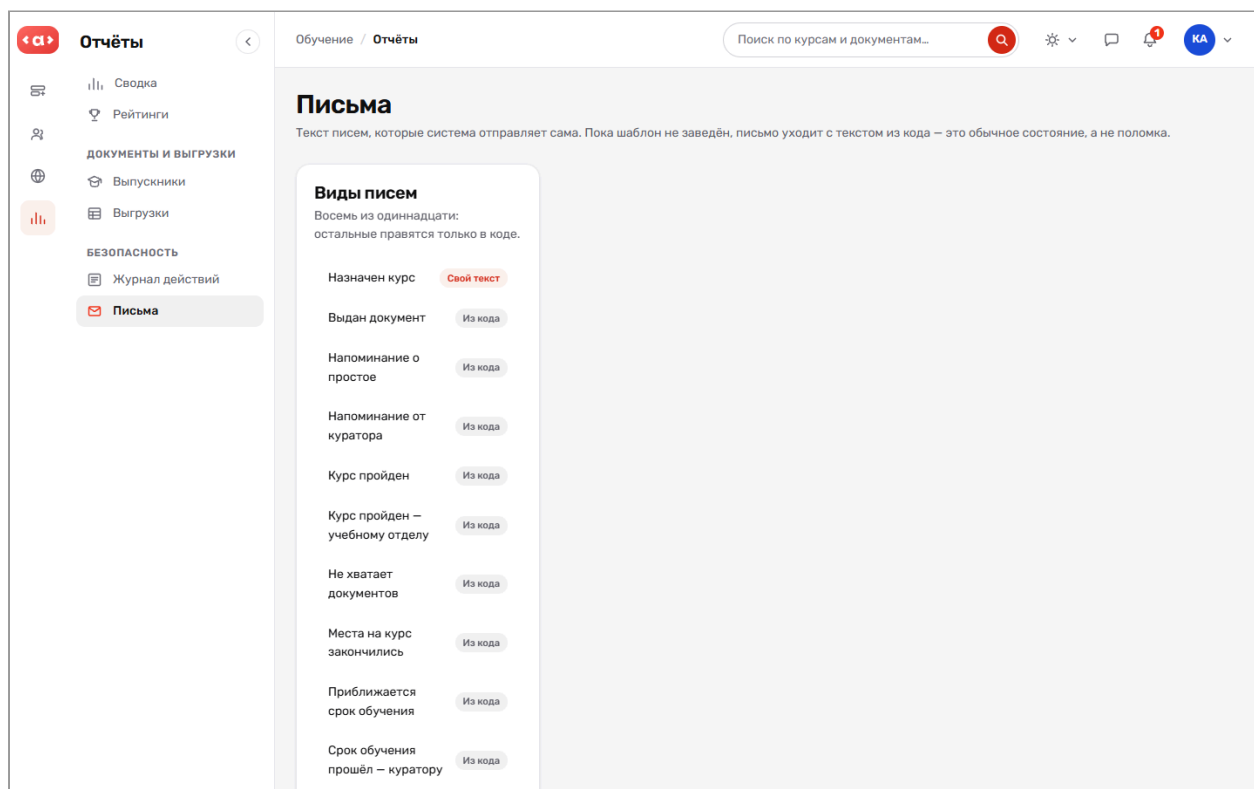


Рисунок 8 — Шаблоны писем

9. Служебные команды

Команды выполняются на сервере из каталога установки от пользователя `deploy`.

Команда	Назначение
<code>npm run grant:role -w backend -- --email <почта> --role ADMIN</code>	выдать пользователю роль (ADMIN, CURATOR или STUDENT). Понижение роли требует флага <code>-yes-i-mean-it</code>
<code>npm run reset:password -w backend - --email <почта> --password '<пароль>'</code>	задать пароль пользователю, например если администратор потерял доступ, а письма не доходят
<code>npm run search:reindex -w backend</code>	перестроить поисковый индекс по курсам и документам
<code>sudo systemctl status lms-backend, journalctl -u lms-backend</code>	состояние службы и её журнал

10. Техническая поддержка

Правообладатель: ООО «АКТИОН 360 УРАЛ ИНФО», ИНН 1109016184, ОГРН 1221100000417.

Телефон: +7 922 588-88-25, электронная почта: zhizhov1988@mail.ru. Сайт: <https://uralinfo360.ru>.