

## Описание технических средств хранения исходного текста и объектного кода программного обеспечения, а также технических средств компиляции исходного текста в объектный код программного обеспечения

Программное обеспечение разработано на языке программирования Python (версия 3.12.1) с использованием фреймворка Django (версия 5.2.9).

Данное программное обеспечение является интерпретируемым. Компиляция исходного текста в объектный код не производится. Исполнение кода осуществляется непосредственно интерпретатором Python на сервере.

Исходный код программного обеспечения хранится в системе контроля версий Git. Серверы хранения исходного кода и серверы развертывания (продакшн) физически расположены на территории Российской Федерации.

Разработка программного обеспечения осуществляется на автоматизированных рабочих местах разработчиков со следующим программным обеспечением:

ОС: Windows 10/11 Pro

IDE: PyCharm Professional / VS Code

Система контроля версий: Git

Для функционирования ПО на серверах хранения и исполнения кода используется следующее программное обеспечение:

Операционная система: Linux Debian 12

Веб-сервер: Nginx

WSGI-сервер: Gunicorn

СУБД: PostgreSQL 16

### 2.1.2.3

Настройка и запуск

1. Скопировать на локальную машину архив с проверочным экземпляром <https://disk.yandex.ru/d/7LxgwO7QnK0wDg>

Пароль для доступа в архив можно связавшись по телефону +7 (960) 795-96-97

Лазарев Александр Андреевич (Генеральный директор)

2. Залогиниться в Реестре образов СтройкаAI, выполнив команду «docker login registry.git.stroykai.com». Данные для авторизации см. в Приложении 1

3. Выполнить команду «`docker compose up -d`», после чего начнется скачивание docker образов и запуск контейнеров
4. Backend доступен на «`http://localhost:8200`», Frontend доступен на «`http://localhost:3000`»

Требования к запуску:

1. Свободные порты: 8200, 3000, 8254, 15672, 15692, 8472. Если нет возможности освободить данные порты, то необходимо изменить конфигурации в файлах «`.env`» и «`compose.yaml`». Если меняются порты у Backend и Frontend, то данное изменение следует учитывать в п.4 раздела «Настройка и запуск» при формировании URL для доступа из браузера
2. Свободные 5 Гб (гигабайта) на жестком диске
3. Свободные 2 Гб (гигабайта) оперативной памяти
4. 2 ядра процессора

Приложение 1

Login (логин): demo

Password (пароль): glpat-48Tr74hTBLB9F6T855Es

Данная учетная запись активна до 16.05.2026, после чего необходимо будет запросить новый пароль

Дополнительные команды:

1. Для создания супер-администратора, чтобы получить доступ к админ панели backend «`http://localhost:8200/admin`» следует выполнить команду «`docker exec -it stroykai_backend python manage.py createsuperuser`» после чего ввести почту и пароль.

Эти данные можно использовать только для входа в админ панель, в пользовательском веб интерфейсе эти данные работать не будут

### 2.1.3

#### **Архитектурное описание**

Программное обеспечение построено по клиент-серверной архитектуре с разделением на пользовательский интерфейс (Frontend) и серверную логику (Backend). Взаимодействие между компонентами осуществляется через стандартизированные API (Application Programming Interface).

Система разворачивается в контейнеризированной среде (Docker) на базе ОС Linux Debian 12

Серверная часть реализована на языке Python с использованием фреймворка Django и надстройки Django Rest Framework.

Основные функции:

- Аутентификация, идентификация, авторизация и управление учетными записями пользователей
- Анализ поступающих пользовательских данных и информации из внешних источников
- Управление доступом к файлам, хранение шаблонов
- Создание Excel-таблиц на основе данных системы

Клиентская часть представляет собой Single Page Application (SPA), построенное на библиотеке React.js с использованием фреймворка Next.js. Обеспечивает динамическую подгрузку контента без перезагрузки страницы.

Взаимодействует с Backend через JSON API.

Требования к клиенту: браузер на базе движка Chromium последней версии.

Логи хранятся локально на жестком диске сервера в виде .log файлов. Настроена ротация логов: при достижении размера файла 10 МБ создается новый файл (старые удаляются).

Сбор метрик в реальном времени через связку Prometheus (хранение) + Grafana (визуализация).