ЕДИНАЯ ЦИФРОВАЯ АНТИКОРРУПЦИОННАЯ КОМПЛАЕНС-ПЛАТФОРМА

ПРАВООБЛАДАТЕЛЬ: АНО «МОСКОВСКАЯ ДИРЕКЦИЯ ТРАНСПОРТНОГО ОБСЛУЖИВАНИЯ»

Инструкция по установке

Профиль
Единая цифровая антикоррупционная комплаенс-платформа

Версия 1.1.

ОГЛАВЛЕНИЕ

1 ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ 3
2 ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ 4
2.1 Назначение и функции 4
2.2 Технические и программные средства, обеспечивающие выполнение программы 4
3 СТРУКТУРА ПРОГРАММЫ 5
4 НАСТРОЙКА ПРОГРАММЫ 6
4.1 Общие сведения 6
4.2 Установка платформы 6
5 ПРОВЕРКА ПРОГРАММЫ 10
6 КОНТАКТЫ СЛУЖБЫ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ 11

1 ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Термины, сокращения, их определения и расшифровки, используемые в этом документе, приведены в таблице ниже (Таблица 1).

Таблица 1 – Термины, сокращения, определения и расшифровки

Термин/сокращение	**Определение/расшифровка**
Платформа	Единая цифровая антикоррупционная комплаенс-платформа
СУБД	Система управления базами данных
API	Application programming interface, описание способов взаимодействия одной компьютерной программы с другими
HTTP/HTTPS	Hypertext Transfer Protocol, протокол прикладного уровня передачи данных. HyperText Transfer Protocol Secure, расширение протокола HTTP для поддержки шифрования в целях повышения безопасности.
REST	Representational state transfer, архитектурный стиль взаимодействия компонентов распределённого приложения в сети. RESTful API, способ взаимодействия сайтов и веб-приложений с сервером.
SSH	Secure Shell, сетевой протокол прикладного уровня, позволяющий производить удалённое управление операционной системой и туннелирование TCP-соединений. Схож по функциональности с протоколами Telnet и rlogin, но, в отличие от них, шифрует весь трафик, включая и передаваемые пароли.
TCP/IP	Transmission Control Protocol, протокол управления передачей является одним из основных протоколов набора интернет-протоколов. Internet Protocol, маршрутизируемый протокол сетевого уровня стека TCP/IP.

2 ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ

2.1 Назначение и функции

Платформа является инструментом, позволяющим аккумулировать и анализировать информацию о потенциальных и реальных коррупционных рисках, о факторах и источниках их возникновения, а также осуществлять контроль за их устранением.

Использование Платформы направлено на достижение следующих целей:

– выявление коррупционных рисков;
– профилактика возникновения коррупционных рисков;
– минимизация последствий возникновения коррупционных рисков;
– совершенствование механизмов контроля и мониторинга коррупционных рисков;
– снижение трудозатрат на работу по выявлению, оценке и минимизации коррупционных рисков;
– повышение эффективности мер по устранению коррупционных рисков.

Функциональные возможности Платформы направлены на решение следующих задач:

– автоматизация процессов выявления реальных факторов и источников возникновения коррупционных рисков;
– автоматизация процессов выявления потенциальных факторов и источников возникновения коррупционных рисков;
– автоматизация процессов предупреждения возникновения коррупционных рисков;
– автоматизация оценки коррупционных рисков;
– формирование единой базы данных, позволяющей анализировать коррупционные риски.

2.2 Технические и программные средства, обеспечивающие выполнение программы

Платформа размещается на сервере – физическом или виртуальном, который должен удовлетворять следующим минимальным требованиям:

– тип и объем накопителя 100 Гб SSD;
– оперативная память 32 Гб;
– операционная система Debian 11;
– база данных PostgreSQL 13;
– обеспечение доступа по SSH с правами sudo.

Программные средства необходимые для выполнения серверной части Платформы:

– СУБД PostgreSQL 13;
– HTTP-сервер Nginx;
– интерпретатор Python 3.9;
– программная платформа Node.js 18;
– почтовый сервер с учетной записью для работы почтовых уведомлений Платформы[1].

Для работы с Платформой необходим компьютер, обеспечивающий возможность использования актуальной версии любого из следующих браузеров:

– Google Chrome;
– Microsoft Edge;
– Mozilla Firefox;
– Opera;
– Яндекс Браузер.

3 СТРУКТУРА ПРОГРАММЫ

Платформа состоит из серверной части (back-end) и внешнего пользовательского интерфейса (front-end).

Серверная часть обеспечивает работу Платформы – хранение данных в базе, авторизацию пользователей, выполнение бизнес-процессов, реализацию общей логики.

Пользователи Платформы используют ее функциональные возможности через внешний графический пользовательский интерфейс.

Компоненты взаимодействуют между собой через REST API.

Информационный обмен между компонентами Платформы выполняется в составе единой сети с использованием набора протоколов TCP/IP. Доступ к пользовательскому интерфейсу реализован с использованием протокола HTTPS, в зависимости от настроек серверной части.

Программная структура Платформы включает в себя следующие функциональные элементы:

– модуль «Библиотека знаний»;
– модуль «Обучение;
– модуль «Декларации;
– модуль «Внутренние проверки;
– модуль «Оценка рисков;
– модуль «Проверка деловых партнеров;

[1] Допускается использование адреса электронной почты на почтовых сервисах «Яндекс», «mail» и т. п.

– модуль «Горячая линия;
– модуль «Администрирование»;
– личный кабинет.

Назначение функциональных элементов следует из их названий.

Взаимодействие Платформы с внешним программным обеспечением не предусмотрено, однако такая возможность может быть реализована в рамках ее дальнейшего развития.

4 НАСТРОЙКА ПРОГРАММЫ

4.1 Общие сведения

Перед установкой Платформы убедитесь в соответствии сервера требованиям из подраздела 2.2 выше.

4.2 Установка платформы

Порядок установки и настройки Платформы:

1 Первичная настройка сервера:

1.1 Создайте пользовательскую группу: «sudo groupadd compliance_production».

1.2 Создайте пользователя приложения: «sudo useradd -m compliance_production».

1.3 Установите следующие пакеты ОС: libpq-dev, nginx, postgresql, postgresql-contrib, python3-dev, python3-psycopg2, python3-venv, supervisor, certbot. Команда для установки: «sudo apt install -y libpq-dev nginx postgresql postgresql-contrib python3-dev python3-psycopg2 python3-venv supervisor certbot».

2 Настройка СУБД:

2.1 Убедитесь, что сервис СУБД запущен: «sudo service postgresql status». В случае, если сервис не работает, запустите его вручную: «sudo service postgresql start».

2.2 Создайте БД и пользователя.

2.2.1 Подключитесь к консоли СУБД: «sudo -u postgres psql».

2.2.2 Создайте пользователя БД: «create user compliance_production with password 'ПАРОЛЬ_ДЛЯ_БД_ЖЕЛАТЕЛЬНО_50_100_СИМВОЛОВ';»

2.2.3 Создайте БД и назначьте ее владельцем этого пользователя: «CREATE DATABASE compliance_production LC_COLLATE 'ru_RU.UTF-8' lc_ctype 'ru_RU.UTF-8' OWNER compliance_production template template0;».

2.2.4 Выйдете из консоли СУБД, набрав команду «\q».

3 Развертывание приложения:

3.1 Создайте каталог приложения: «mkdir /home/compliance_production/compliance_production/».

3.2 Распакуйте архив приложения в каталог, созданный на предыдущем шаге: «cd /home/compliance_production/compliance_production && tar -xvf {НАЗВАНИЕ ФАЙЛА С АРХИВОМ}.tar.xz».

3.3 Создайте файл под названием «.env», заполните в нем переменные и разместите в корневом каталоге приложения. Для этого заполните значения следующим образом: «

DATABASE_URL="postgres://compliance_production:ПАРОЛЬ_ПОЛЬЗОВАТЕЛЯ_БД@localhost:5432/compliance_production"

DEBUG=False

DEFAULT_FROM_EMAIL="АДРЕС_ПОЧТЫ_С_КОТОРОЙ_ПОЙДЕТ_РАССЫЛКА"

DJANGO_SETTINGS_MODULE="pltcompliance.settings"

EMAIL_URL="smtp+ssl://"smtp+ssl://comrobot@mdto.ru:jkABQpCJxaaLcjQjHuWd@smtp.mail.ru:465":ПАРОЛЬ_ОТ_ЭТОЙ_ПОЧТЫ@smtp.{yandex или mail}.ru:465"

MEDIA_URL=https://ДОМЕН_ПЛАТФОРМЫ/media/

NOTICE_EMAIL=True

SECRET_KEY="{{ secret_key }}"

SERVER_EMAIL="ПОСЛЕДОВАТЕЛЬНОСТЬ_ИЗ_50_СИМВОЛОВ"

STATIC_URL="https://ДОМЕН_ПЛАТФОРМЫ/static/"».

3.4 Настройте владельца файлов каталога приложения: «sudo chown -R compliance_production:compliance_production /home/compliance_production/compliance_production».

3.5 Обновите установщик пакетов pip до последней версии: «/home/compliance_production/compliance_production/venv/bin/pip install --upgrade pip».

3.6 Настройте виртуальное окружение Python: «python[2] -m venv».

3.7 Установите зависимости из файла «requirements.txt», поставляемого вместе с исходным кодом Платформы: «

/home/compliance_production/compliance_production/venv/bin/pip install -r /home/compliance_production/compliance_production/requirements.txt».

3.8 Настройте каталогу приложения права доступа: «

chmod u+rwx "/home/compliance_production/compliance_production/"

chmod go+rx "/home/compliance_production/compliance_production/"

».

3.9 Настройте права доступа к файлу «/home/compliance_production/compliance_production/manage.py»: «

chmod u+rwx "/home/compliance_production/compliance_production/manage.py"

chmod go+rx "/home/compliance_production/compliance_production/manage.py"

».

3.10 Настройте права доступа к статическим файлам, расположенным в каталогах «static» и «media»:

chmod -R u=rwx "/home/compliance_production/compliance_production/static"

[2] Вместо слова «python» может потребоваться указать полный путь к интерпретатору.

chmod -R go=rx "/home/compliance_production/compliance_production/media"
».

3.11 Выполните миграцию базы данных Django: «/home/compliance_production/compliance_production/venv/python /home/compliance_production/compliance_production/manage.py makemigrations».

3.12 Установите необходимые разрешения для каталога приложения: «/home/compliance_production/compliance_production/venv/python /home/compliance_production/compliance_production/manage.py setupperms».

3.13 Настройте параметры сервера приложений в файле «gunicorn.cfg», расположенном в каталоге приложения. Для этого заполните значения следующим образом: «

bind = "unix:/tmp/gunicorn_compliance_production.sock"
chdir = "/home/compliance_production/compliance_production"
workers = 12
threads = 16
».

3.14 Создайте файл: «
cat > "/etc/supervisor/conf.d/compliance_production.conf"
».

Заполните файл следующим образом: «
[program:compliance_production]
command = /home/compliance_production/compliance_production/venv/bin/gunicorn /bin/gunicorn -c /home/compliance_production/compliance_production/ gunicorn.cfg pltcompliance.wsgi
redirect_stderr = true
user = compliance_production
group = compliance_production
».

3.15 Запустите приложение: «sudo supervisorctl start compliance_production».

3.16 Настройте планировщик задач:

3.16.1 Запустите редактор планировщика: «crontab -e». При первом запуске потребуется выбрать подходящий редактор.

3.16.2 Заполните файл следующим образом: «
0 0 * * * /home/compliance_production/compliance_production/venv/bin/python /home/compliance_production/compliance_production/manage.py edufinish
0 1 * * * /home/compliance_production/compliance_production/venv/bin/python /home/compliance_production/compliance_production/manage.py compliance_production
0 8 * * * /home/compliance_production/current/venv/bin/python /home/compliance_production/compliance_production/manage.py sendreminders
».

3.16.3 Выйдите из редактора сохранив файл.

4 Настройка веб-сервера:

4.1 Настройте параметры в файле «/etc/nginx/conf.d/compliance_production_app.conf». Для этого заполните значения следующим образом: «

```
server {
listen 443;
server_name ДОМЕН_В_КОДИРОВКЕ_PUNICODE;

# IMPORTANT check this server is not available from public internet

client_max_body_size 1000m;
set_real_ip_from 0.0.0.0/0;
keepalive_timeout 5s;
proxy_buffers 24 8k;
proxy_connect_timeout 1s;
proxy_redirect off;
include proxy_params;

error_page 404 /static/404.html;
error_page 500 502 503 504 =503 /static/50x.html;
root {{ project_path }}/static;

location = / {
add_header Cache-Control "no-store, max-age=0";
try_files /index.html =404;
}

location = /favicon.ico {
access_log off;
add_header Cache-Control "public, max-age=3600";
}
location / {
# all random /foobar urls fall here without attempt to open file("static/foobar")
return 404;
}

# "deep links into SPA", must be exactly same config as `location = /`
location /a/ {
add_header Cache-Control "no-store, max-age=0";
try_files /index.html =404;
}

location /admin/ {
proxy_pass http://unix:/tmp/gunicorn_compliance_production.sock;
}
```

```
location /api/ {
proxy_pass http://unix:/tmp/gunicorn_compliance_production.sock;
}

location /b/ {
alias /home/compliance_production/compliance_production/static/b/;
add_header Cache-Control "public, immutable";
expires max;
}

location /media/ {
alias /home/compliance_production/compliance_production/media/;
}

location /static/ {
alias /home/compliance_production/compliance_production/static/;
}

location /swagger/ {
proxy_pass http://unix:/tmp/gunicorn_compliance_production.sock;
}
}
```

».

4.2 Включите HTTPS для доступа к Платформе: «certbot -n --agree-tos --nginx -d ДОМЕН_В_КОДИРОВКЕ_PUNICODE».

4.3 Перезагрузите nginx: «systemctl restart nginx».

5 Добавьте супер-администратора Платформы: «/home/compliance_production/compliance_production/manage.py createsuperuser». В ходе диалога укажите адрес электронной почты супер-администратора[3] и пароль.

Для проверки корректности установки и настройки Платформы воспользуйтесь инструкцией из раздела 5 этого документа.

5 ПРОВЕРКА ПРОГРАММЫ

Перед проверкой работоспособности Платформы убедитесь, что на рабочем месте установлен браузер из вышеприведенного списка (подраздел 2.2).

Для проверки откройте в браузере главную страницу Платформы и пройдите аутентификацию. После успешного входа будет открыт раздел супер-администратора Платформы.

[3] Важно: роль супер-администратора несовместима с иными ролями Платформы, поэтому для нее должен быть выделен отдельный адрес электронной почты.

При корректной работе разделы должны открываться, действия не должны приводить к техническим ошибкам.

В случае неработоспособности Платформы или возникновения ошибок, свяжитесь со службой технической поддержки по контактам, приведенным ниже (раздел 6).

6 КОНТАКТЫ СЛУЖБЫ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ

Для получения консультации по установке и настройке свяжитесь со службой технической поддержки по адресу электронной почты compliance-mdto@transport.mos.ru или номеру телефона +7 (495) 620 20 00 (доб. 31356).