



2023

Цифровая платформа
управления инженерной
и коммунальной
инфраструктурой здания
Руководство администратора

ОГЛАВЛЕНИЕ

1. Общие сведения о программном продукте	3
1.1. О Системе	3
1.2. Уровень подготовки администратора	3
2. Назначение и условия применения	5
2.1. Назначение Системы	5
2.2. Программные и аппаратные требования к Системе	5
3. Установка и настройка серверного программного обеспечения	7
3.1. Установка сопутствующих программных компонентов	7
3.2. Установка и настройка PostgreSQL	7
3.3. Клонирование проекта из репозитория	7
3.4. Настройка виртуального окружения	7
3.5. Установка и настройка RabbitMQ	8
3.6. Установка и настройка компонентов пользовательского интерфейса ..	9
3.7. Создание файлов сокета и служебных файлов systemd для Gunicorn ..	9
3.8. Проверка файла сокета Gunicorn	10
3.9. Тестирование активации сокета	10
3.10. Настройка Nginx как прокси для Gunicorn	10

1. Общие сведения о программном продукте

1.1. О Системе

Данное руководство предназначено для специалистов, осуществляющих установку и администрирование серверного программного обеспечения «Цифровая платформа управления инженерной и коммунальной инфраструктурой здания» (далее — Система).

Система позволяет:

- создать «цифровой двойник» обслуживаемого объекта с историей обслуживания и перечнем предстоящих работ (техкарты, чек-листы, ППР);
- мгновенно реагировать на обращения собственников, арендаторов и посетителей, которые формируются в «два клика»;
- контролировать качество и сроки работы сотрудников и подрядчиков;
- выстроить прозрачную и понятную систему взаимоотношений с четким распределением ролей и ответственности;
- организовать мобильное рабочее место для каждого сотрудника со всей необходимой для работ информацией.

Контроль параметров работы инженерных систем здания с необходимой периодичностью и передавая их в цифровом виде в единую базу данных, позволяет получать информацию, достаточную для построения состоятельных предиктивных моделей.

1.2. Уровень подготовки администратора

Для установки и обслуживания серверного программного обеспечения Системы администратор должен уметь:

- обеспечивать функционирование компьютерных систем и компьютерных сетей;
- выполнять организацию и контроль доступа сотрудников к локальной сети и сети Интернет;

- выполнять разработку, администрирование и обновление web-приложений;
- выполнять регистрацию, назначение идентификаторов и паролей и создание учетных записей пользователей;
- определять целевое назначение программного обеспечения;
- выполнять интеграцию программного обеспечения на файл-серверах, серверах баз данных и на рабочих местах;
- выполнять установку антивирусных программ и контроль их своевременного обновления;
- обеспечивать безопасность компьютерной системы и защиту доступа к системным файлам;
- выполнять выделение места на диске для сохранности данных и резервное копирование информации;
- уметь работать с системными базами данных (PostgreSQL);
- выполнять обучение пользователей работе с компьютерными программами, разработка инструкций для пользователей;
- выполнять обновление работоспособности компьютерных систем после сбоев;
- проводить мониторинг компьютерных систем;
- осуществлять ведение технической документации.

2. Назначение и условия применения

2.1. Назначение Системы

Система позволяет управлять инженерной и коммунальной инфраструктурой здания для повышения эффективности и снижения затрат ее использования, основанных на статистических методах.

2.2. Программные и аппаратные требования к Системе

Состав технических средств, используемых во время испытаний Системы, приведен в Таблицах 1 – 2.

Таблица 1. Состав технических средств (серверное оборудование)

Компонент	Минимальная конфигурация
Количество физических ядер	Не менее 16
Тактовая частота процессора	Не менее 2 ГГц
Объем оперативной памяти	64 ГБ
Объем жесткого диска	2 ТБ
Сетевая карта	Канал предоставление доступа в Интернет 100 МБит/с
Архитектура процессора	64-битный (amd64)

Таблица 2. Состав технических средств (АРМ клиента)

Компонент	Минимальная конфигурация
Количество ядер процессора	Не менее 2
Тактовая частота процессора	Не менее 2 ГГц
Объем оперативной памяти	Не менее 4 ГБ
Объем жесткого диска	Не менее 500 ГБ
Сетевая карта	Поддержка Ethernet
Диагональ монитора	Не менее 21 дюйма

Состав программных средств, используемых во время испытаний Системы, приведен в Таблице 3.

Таблица 3. Требования к программному обеспечению

Компонент	Минимальная конфигурация
Операционная система	Ubuntu 20.04
Требование к окружению	python 3.6.4; nginx 1.16; postgresql; django 3.0.6; drf-yasg 1.17.1; wheel 0.34.2; djangorestframework 3.11.0; djangorestframework-jwt 1.11.0; sentry-sdk 0.14.0; psycopg2-binary 2.8.5; pytest 6.2.2; pytest-django 4.1.0; gunicorn; openpyxl 2.6.0; docxtpl 0.10.0; pandas.

3. Установка и настройка серверного программного обеспечения

3.1. Установка сопутствующих программных компонентов

```
sudo apt install openssh-server, net-tools, mc, git, nginx
```

3.2. Установка и настройка PostgreSQL

```
sudo wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc |  
sudo apt-key add -
```

```
echo "deb http://apt.postgresql.org/pub/repos/apt/ `lsb_release -cs`-pgdg main" |  
sudo tee /etc/apt/sources.list.d/pgdg.list
```

```
sudo apt update
```

```
sudo apt -y install postgresql-12 postgresql-client-12
```

```
sudo su - postgres
```

```
psql -c "alter user postgres with password 'postgres'"
```

```
psql -c "create user user;"
```

```
psql -c "create database db;"
```

```
psql -c "grant all privileges on database db to user;"
```

```
psql -c "alter user user with encrypted password 'password';"
```

3.3. Клонирование проекта из репозитория

```
cd /home
```

```
sudo git clone <project_repo>
```

```
sudo git branch --set-upstream-to=origin/software_release
```

```
sudo git pull
```

3.4. Настройка виртуального окружения

```
sudo apt-get install python3-venv
```

```
sudo python3 -m venv venv
```

```
cd venv/bin/
```

```
source activate
```

```
pip install --upgrade pip
```

```
pip install -r requirements.txt
```

```
python manage.py migrate
```

3.5. Установка и настройка RabbitMQ

```
sudo apt-get install curl gnupg apt-transport-https -y
```

```
curl -1sLf "https://keys.openpgp.org/vks/v1/by-fingerprint/0A9AF2115F4687BD29803A206B73A36E6026DFCA" | sudo gpg --dearmor | sudo tee /usr/share/keyrings/com.rabbitmq.team.gpg > /dev/null
```

```
curl -1sLf "https://keyserver.ubuntu.com/pks/lookup?op=get&search=0xf77f1eda57ebb1cc" | sudo gpg --dearmor | sudo tee /usr/share/keyrings/net.launchpad.ppa.rabbitmq.erlang.gpg > /dev/null
```

```
curl -1sLf "https://packagecloud.io/rabbitmq/rabbitmq-server/gpgkey" | sudo gpg --dearmor | sudo tee /usr/share/keyrings/io.packagecloud.rabbitmq.gpg > /dev/null
```

```
sudo tee /etc/apt/sources.list.d/rabbitmq.list <<EOF
```

```
deb [signed-by=/usr/share/keyrings/net.launchpad.ppa.rabbitmq.erlang.gpg] http://ppa.launchpad.net/rabbitmq/rabbitmq-erlang/ubuntu bionic main
```

```
deb-src [signed-by=/usr/share/keyrings/net.launchpad.ppa.rabbitmq.erlang.gpg] http://ppa.launchpad.net/rabbitmq/rabbitmq-erlang/ubuntu bionic main
```

```
deb [signed-by=/usr/share/keyrings/io.packagecloud.rabbitmq.gpg] https://packagecloud.io/rabbitmq/rabbitmq-server/ubuntu/ bionic main
```

```
deb-src [signed-by=/usr/share/keyrings/io.packagecloud.rabbitmq.gpg] https://packagecloud.io/rabbitmq/rabbitmq-server/ubuntu/ bionic main
```

```
EOF
```

```
sudo apt-get update -y
```

```
sudo apt-get install -y erlang-base \
```

```
    erlang-asn1 erlang-crypto erlang-eldap erlang-ftp erlang-inets \
```

```
    erlang-mnesia erlang-os-mon erlang-parsetools erlang-public-key \
```

```
    erlang-runtime-tools erlang-snmp erlang-ssl \
```

```
    erlang-syntax-tools erlang-tftp erlang-tools erlang-xmerl
```

```
## Install rabbitmq-server and its dependencies
```

```
sudo apt-get install rabbitmq-server -y --fix-missing
```

Configure RabbitMQ:

```
sudo systemctl enable rabbitmq-server
sudo rabbitmq-plugins enable rabbitmq_management
sudo rabbitmqctl add_user admin admin
sudo rabbitmqctl set_user_tags admin administrator
sudo rabbitmqctl set_permissions -p / admin ".*" ".*" ".*"
```

3.6. Установка и настройка компонентов пользовательского интерфейса

```
curl -fsSL https://deb.nodesource.com/setup_current.x | sudo -E bash -
sudo apt-get install -y nodejs
sudo npm install -g n
sudo n stable
```

3.7. Создание файлов сокета и служебных файлов systemd для Gunicorn

```
sudo nano /etc/systemd/system/gunicorn.socket
```

[Unit]

Description=gunicorn socket

[Socket]

ListenStream=/run/gunicorn.sock

[Install]

WantedBy=sockets.target

```
sudo nano /etc/systemd/system/gunicorn.service
```

[Unit]

Description=gunicorn daemon

Requires=gunicorn.socket

After=network.target

```
[Service]
User=user
Group=www-data
WorkingDirectory=/home/project/<path_to_project>
ExecStart=/home/user/<project_path>/venv/bin/gunicorn \
    --access-logfile - \
    --workers 3 \
    --bind unix:/run/gunicorn.sock \
    <project_name>.wsgi:application
```

```
[Install]
```

```
WantedBy=multi-user.target
```

3.8. Проверка файла сокета Gunicorn

```
sudo systemctl status gunicorn.socket
file /run/gunicorn.sock
```

3.9. Тестирование активации сокета

```
sudo systemctl status gunicorn
curl --unix-socket /run/gunicorn.sock localhost
sudo systemctl daemon-reload
sudo systemctl restart gunicorn
```

3.10. Настройка Nginx как прокси для Gunicorn

```
sudo nano /etc/nginx/sites-available/default
```

```
server {
    listen 80;
    server_name _;

    location /static {
        alias /home/user/<project>/<django_project_name>/static/;
    }
}
```

```
location / {  
    include proxy_params;  
    proxy_pass http://unix:/run/gunicorn.sock;  
}  
}
```

```
sudo nginx -t  
sudo systemctl restart nginx
```