

**ЦИФРОВАЯ ПЛАТФОРМА УПРАВЛЕНИЯ ЖИЗНЕННЫМ ЦИКЛОМ
ЗДАНИЯ
РУКОВОДСТВО ПО УСТАНОВКЕ**

Листов 10

2022

АННОТАЦИЯ

«Цифровая платформа управления жизненным циклом здания» (далее по тексту – Система) является программной системой для управления жизненным циклом здания.

В документе приведены основные сведения, необходимые для выполнения установки Система, назначение программного обеспечения, конфигурация программно-технического комплекса.

СОДЕРЖАНИЕ

1. Введение.....	4
2. Область применения.....	5
3. Обозначения и сокращения	6
4. Порядок установки	7
4.1. Установка и настройка postgresql	7
5.2. Установка docker и docker-compose	8
5.3. Настройка компонентов Система	9
5.4. Требования к аппаратному обеспечению.....	10

1. ВВЕДЕНИЕ

Полное наименование системы: Цифровая платформа управления жизненным циклом здания.

Целью системы является повышение рентабельности за счет сокращения временных затрат, потери ресурсов и оптимизации процесса управления строительными проектами.

Система предназначена для управления жизненным циклом здания.

2. ОБЛАСТЬ ПРИМЕНЕНИЯ

Система предназначен для использования на объектах, участниках процесса строительства.

Пользователи Система должны владеть начальными навыками использования персонального компьютера и изучить настоящее Руководство.

Система разворачивается на серверах и рабочих станциях заказчика.

3. ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

Система	– Цифровая платформа управления жизненным циклом здания
АРМ	– программно-технический комплекс АС, предназначенный для автоматизации деятельности определенного вида.
Ubuntu	– Дистрибутив GNU/Linux, основанный на Debian GNU/Linux.
PostgreSQL	– Свободная объектно-реляционная система управления базами данных.
Docker	– программное обеспечение для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации, контейнеризатор приложений.
Docker-compose	– пакетный менеджер, позволяющий описывать и запускать многоконтейнерные приложения; конфигурационные файлы для него описываются на языке YAML.
Контейнеризация	– метод виртуализации, при котором ядро операционной системы поддерживает несколько изолированных экземпляров пространства пользователя вместо одного.

4. ПОРЯДОК УСТАНОВКИ

В данной инструкции расписан процесс установки и настройки системы на базе операционной системы семейства Ubuntu. Установка и настройка на других linux системах происходит аналогично.

4.1. Установка и настройка postgresql

Полная инструкция расположена на сайте PostgreSQL по адресу: <https://www.postgresql.org/download/linux/ubuntu/>.

Для установки и настройки СУБД PostgreSQL необходимо выполнить следующие действия (после ввода каждой строки нажимать Enter):

5.1.1. Установить СУБД. Запускаем терминал и вводим команды:

5.1.1.1. `sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'`

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

5.1.1.3. `sudo apt-get update`

5.1.1.4. `sudo apt-get -y install postgresql`

если установка проходит с ошибкой, то имеются проблемы с ключами репозитория PostgreSQL, для исправления выполняем команду:

5.1.1.5. `gpg --keyserver www.postgresql.org --recv-keys ACCC4CF8.asc && gpg --export -a ACCC4CF8.asc | sudo apt-key add -`

после чего повторяем п.5.1.1.3 и п.5.1.1.4

5.1.2. Создать базу данных и пользователя для доступа к ней.

5.1.2.1. `sudo -u postgres psql`

результат: postgres=#

5.1.2.2. `create database mydb;`

результат: CREATE DATABASE

Вместо «mydb» указываем как будет называться база данных. Запоминаем название базы данных, потребуется в дальнейшем.

5.1.2.3. `create user myuser with encrypted password 'mypass';`

результат: CREATE ROLE

Вместо «myuser» указываем имя создаваемого пользователя для работы с вышеуказанной базой данных.

Вместо «mypass» указываем пароль для создаваемого пользователя.

5.1.2.4. `grant all privileges on database mydb to myuser;`

результат: GRANT

Вместо «mydb» указываем имя базы данных из п.5.1.2.2. Вместо «myuser»

указываем имя пользователя из п.5.1.2.3.

5.1.2.5. `alter database mydb set timezone to 'Etc/UTC';`

результат: ALTER DATABASE

Вместо «mydb» указываем имя базы данных из п.5.1.2.2.

5.1.2.6. `select pg_reload_conf();`

результат: pg_reload_conf

t

(1 row)

5.1.2.7. `SHOW config_file;`

результат: /etc/postgresql/13/main/

Запомним данный путь.

5.1.2.8. `\q`

5.1.3. Настроить доступа к базе данных.

5.1.3.1. В конфигурационный файл `pg_hba.conf` (путь к файлу берем из п.5.1.2.7) добавить в конце файла новую строку:

host mydb myuser 0.0.0.0/0 md5

Вместо «mydb» и «myuser» указываем пользователя и имя базы данных из п.5.1.2.2 и п.5.1.2.3.

5.1.3.2. Отредактировать строку конфигурационного файла `postgresql.conf` (путь к файлу берем из п.5.1.2.7) следующим образом:

listen_addresses = ''*

5.1.4. Открыть внешний доступ к порту 5432

5.2. Установка docker и docker-compose

Установка данного ПО заключается в развертывании на сервере следующих компонентов:

- Контейнеризатор приложений Docker
- Инструмент для запуска и управления мультиконтейнерными приложениями docker-compose

Порядок установки:

5.2.1. `sudo apt-get install docker.io curl`

5.2.2. `sudo curl -L`

`"https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose`

5.2.3. `sudo chmod +x /usr/local/bin/docker-compose`

5.2.4. `sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose`

Проверить, что все установилось

5.2.5. sudo docker version

Результат:

```
Client:
Version:      20.10.7
API version:  1.41
Go version:   go1.13.8
Git commit:   20.10.7-0ubuntu5~21.04.2
Built:        Mon Nov 1 00:21:07 2021
OS/Arch:      linux/amd64
Context:      default
Experimental: true

Server:
Engine:
Version:      20.10.7
API version:  1.41 (minimum version 1.12)
Go version:   go1.13.8
Git commit:   20.10.7-0ubuntu5~21.04.2
Built:        Tue Oct 19 01:00:27 2021
OS/Arch:      linux/amd64
Experimental: false
containerd:
Version:      1.5.5-0ubuntu3~21.04.1
GitCommit:
runc:
Version:      1.0.1-0ubuntu2~21.04.1
GitCommit:
docker-init:
Version:      0.19.0
GitCommit:
```

5.2.6. sudo docker-compose version

Результат:

```
docker-compose version 1.29.2, build 5becea4c
docker-py version: 5.0.0
CPython version: 3.7.10
OpenSSL version: OpenSSL 1.1.0l 10 Sep 2019
```

5.3. Настройка компонентов Система

5.3.1. Распаковать архив с кодом в отдельную папку. Папка должна быть заранее создана. Для этого необходимо выполнить следующие команды:

5.3.1.1. `unzip <archive_name>`

Вместо `<archive_name>` указать имя архива с исходным кодом

5.3.1.2. Переходим в извлеченный каталог:

`cd <dir_name>`

5.3.1.3. В файле `.env` изменить параметры подключения:

`FRONTEND_HOST = server_web`

`DB_CONNECT_IP = server_db`

`DB_CONNECT_NAME = name_db`

`DB_CONNECT_USER = user_db`

`DB_CONNECT_PSWD = password_db`

После чего сохраняем файл.

Указанные в примере значения заменяем на ранее созданные:

server_web = ip-адрес или hostname сервера, где будет доступен web-интерфейс.

name_db = имя базы данных из пункта 5.1.2.2;

user_db = имя пользователя из пункта 5.1.2.3;

password_db = пароль пользователя из пункта 5.1.2.3;

server_db = ip-адрес или hostname сервера с базой данных.

5.3.1.4. Запустить сборку через docker-compose из корня папки созданный в пункте 5.3.1.1.

```
sudo docker-compose -f docker-compose.yml up --build -d
```

Результат:

```
Creating digital_construction_backend      ... done
```

```
Creating digital_construction_frontend     ... done
```

```
Creating digital_construction_postgres_1   ... done
```

5.3.2. Если IP адрес не менялся, то запуск в браузере будет доступен по адресу <http://127.0.0.1:5100>

5.3.3. Для того, чтобы посмотреть список запущенных контейнеров необходимо выполнить команду:

5.3.3.1. `docker ps - a`

Для корректной работы Система должны быть запущены следующие контейнеры:

```
digital_construction_backend
```

```
digital_construction_frontend
```

```
digital_construction_postgres_1
```

Для вывода логина и пароля администратора системы выполните команду:

```
docker exec digital_construction_backend cat files/Readme
```

5.4. Требования к аппаратному обеспечению

Для штатной работы всех модулей Система необходима минимальная конфигурация серверов, которая представлена в таблице 1.

Таблица 1 – Минимальная конфигурация сервера

Характеристика	Единица измерения	Значение
ЦПУ	Вычислительные ядра	Не менее 2 доступных
ОЗУ DDR4	Гбайт	4 или более
Жесткий диск HDD/SSD	Гбайт	50 или более
Операционная система	Версия	Linux/Windows