

ТРЕБОВАНИЯ К ТЕХНИЧЕСКОМУ ОБЕСПЕЧЕНИЮ

Техническое обеспечение системы должно включать в себя сервер баз данных и сервер приложений.

В качестве СУБД должна использоваться СУБД Postgres Pro, версии не ниже 11. Операционная система сервера приложений должна быть на базе ядра Linux, версии не ниже 3.10.0.

ТРЕБОВАНИЯ К АППАРАТНОМУ ОБЕСПЕЧЕНИЮ

Аппаратное обеспечение системы должно удовлетворять следующим требованиям:

Табл. 6. Требования к серверу приложения

Процессор	Архитектура x86_64
Оперативная память	64 Гб
ОС	Linux

Табл. 7. Требования к серверу СУБД

Процессор	Архитектура x86_64
Оперативная память	64 Гб
ОС	Linux

ТРЕБОВАНИЯ К АДМИНИСТРАТОРУ СИСТЕМЫ

К квалификации администраторов Системы предъявляются следующие требования:

- Знание и навыки администрирования СУБД Postgres Pro;
- Знание языка запросов SQL и PL SQL;
- Понимание общих принципов построения системы и её иерархии;
- Опыт работы с Unix системами;
- Опыт работы с системой контейнеризации Docker.

АРХИТЕКТУРА СИСТЕМЫ

Функциональная архитектура системы представлена на Рис. 1.

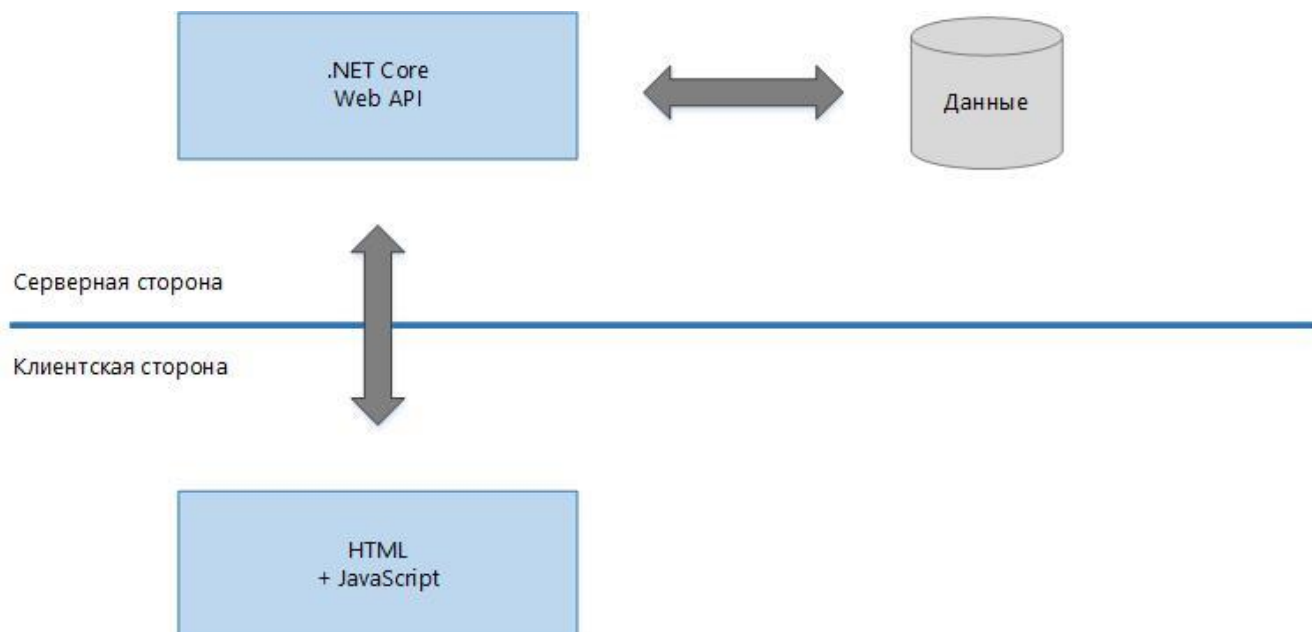


Рис. 1 – Общий вид архитектуры

Архитектура системы включает в себя:

- Веб-сервер клиентского приложения. Представляет собой SPA приложение, разработанное с помощью фреймворка Angular 11+. SPA - это веб-приложение или веб-сайт, использующий единственный HTML-документ как оболочку для всех веб-страниц и организующий взаимодействие с пользователем через динамически подгружаемые HTML, CSS, JavaScript.
- Веб-сервер API. Представляет собой веб-службу, к которой могут обращаться другие приложения по протоколу HTTP/HTTPS. Данные приложения могут представлять любую технологию и платформу: веб-приложения, мобильные или десктопные клиенты. Данная служба разработана на платформе NET CORE 2.1 WEB API.
- БД Postgres Pro.

УСТАНОВКА ОКРУЖЕНИЯ

DOCKER ENGINE

Для скачивания и установки дистрибутивов **Docker Engine** необходимо перейти по ссылке <https://docs.docker.com/v17.09/engine/installation/>.

POSTGRES PRO

Необходимо установить Postgres Pro версии не ниже 11.

РАЗВЁРТЫВАНИЕ СИСТЕМЫ

Для установки ИС ТЭП на сервере необходимо выполнить следующие шаги:

1. Развернуть базы данных на сервере СУБД из резервных копий из предоставленного дистрибутива, используя команды
 - БД архива первичных значений `psql -h <адрес сервера СУБД> -U <имя пользователя> -W -f <имя БД ARCHIVE>.sql`
 - БД ИС ТЭП `psql -h <адрес сервера СУБД> -U <имя пользователя> -W -f <имя БД АСТЕР>.sql`
2. Загрузить образы сервисы из предоставленных дистрибутивов, используя команды
`docker load < astep_calc.tar`
`docker load < astep_web.tar`
`docker load < astep_adm.tar`
3. Создать конфигурационный файл для сервиса расчета в кодировке UTF-8:
`/etc/opt/astep/<имя БД>/service/Config.xml`

следующего содержания:

```
<access>
<Setting>
  <DataBase><имя БД АСТЕР></DataBase>
  <Server> <адрес сервера СУБД> : <порт сервера СУБД></Server>
  <UserName> <имя пользователя БД> </UserName>
  <Password><пароль> </Password>
  <ProviderName>Postgre</ProviderName>
  <ApplicationName><Имя приложения></ApplicationName>
  <TimeOut>600</TimeOut>
  <CommandTimeOut>300</CommandTimeOut>
</Setting>
<ArchiveDBSetting>
  <DataBase><имя БД ARCHIVE></DataBase>
  <Server><адрес сервера СУБД> : <порт сервера СУБД> </Server>
  <UserName><имя пользователя БД> </UserName>
  <Password><пароль> </Password>
  <ProviderName>Postgre</ProviderName>
  <ApplicationName><Имя приложения></ApplicationName>
  <TimeOut>600</TimeOut>
  <CommandTimeOut>300</CommandTimeOut>
</ArchiveDBSetting>
<AstepImports>
<AllImports>1</AllImports>
  <SSPTIImport>1</SSPTIImport>
</AstepImports>
<WebApi>
  <AllowOrigins>
```

```

    <Uri>://<адрес сервера клиентского приложения (указать порт, если используется отличный
от 80)></Uri>
    <Uri>://<адрес сервера модуля администратора (указать порт, если используется отличный
от 80)>;</Uri>
    </AllowOrigins>
    <port>8000</port>
    <HostName>localhost</HostName>
    <TimeOut>600</TimeOut>
</WebApi>
<CalculateService>
    <Uri>net.tcp://localhost:9001/</Uri>
</CalculateService>
<MessengerService>
    <Uri>net.tcp://localhost:9002/</Uri>
</MessengerService>
<Restart>
    <TimeOut>-1</TimeOut>
    <LogReadValuesFromArchive>>false</LogReadValuesFromArchive>
</ServiceLog>
<SchedulerSettings>
    <!--cron расписание UTC-->
    <TechTimeSchedule>0 20 * * *</TechTimeSchedule>
    <!--Время простоя-->
    <TechTime>60</TechTime>
</SchedulerSettings>
</access>

```

4. Создать конфигурационный файл для клиентского приложения и модуля администратора

```
/etc/opt/astep/<имя БД>/web/config.json
```

```

{
    "apiEndpoint": " http://адрес сервиса расчёта: порт сервиса расчёта"
}

```

5. Запустить контейнеры:

```

sudo docker run -d -p <порт сервиса расчёта>:8000 --restart always -v /etc/opt/astep/<имя
БД>/service:/etc/astep --env useConfig=1 --name astep_calculate_<имя БД>
10.64.241.9:5000/astepcalculateservice_dev:20200709.5 && \
sudo docker run -d -p <порт модуля администратора>:80 -v /etc/opt/astep/<имя
БД>/web:/etc/astep --name astep_adm_<имя БД> 10.64.241.9:5000/astep_adm_inter:20200708.3 && \
sudo docker run -d -p <порт клиентского приложения>:80 -v /etc/opt/astep/<имя
БД>/web:/etc/astep --name astep_web_<имя БД> 10.64.241.9:5000/astep_front_inter:20200708.3

```

Чтобы приступить к работе в системе, необходимо перейти по ссылке <http://адрес сервера клиентского приложения>: *порт сервиса клиентского приложения (если используется отличный от 80)* и выполнить вход под следующей учётной записью: логин - **user_inter**, пароль - **user_inter**.