

ООО «Научно-аналитические методы»

MYGEOMAP

ИНСТРУКЦИЯ ПО УСТАНОВКЕ И НАСТРОЙКЕ

Количество листов 24

Москва 2025

АННОТАЦИЯ

Данный документ является Инструкцией по установке и настройке MyGeoMap (далее – Платформа).

Платформа представляет собой решение по энергетической тематике с уникальной единой базой знаний.

Настоящий документ выполнен в соответствии с требованиями следующих нормативных документов:

– ГОСТ 2.105-95 «Единая система конструкторской документации. Общие требования к текстовым документам»;

– ГОСТ 19.106-78 «Единая система программной документации. Требования к программным документам, выполненным печатным способом»;

– ГОСТ 34.003–90 «Информационная технология. Комплекс стандартов на автоматизированные системы. Термины и определения»;

– ГОСТ 34.201–89 «Виды, комплектность и обозначение документов при создании автоматизированных систем»;

– ГОСТ 34.601-90 «Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания».

Данный документ входит в состав эксплуатационной документации и состоит из трех разделов, в которых приводятся термины и сокращения, использованные в данном документе, описываются условия установки Платформы, подготовительные действия перед установкой Платформы, а также действия по ее установке и настройке.

По вопросам установки и настройке ПО обращаться с 09:00 по 18:00 (по Московскому времени), за исключением выходных и праздничных дней по телефону +79263153234 или электронной почте: info@mygeomap.ru

Содержание

1 Термины и сокращения	5
2 Условия установки	6
2.1 Условия установки серверной части Платформы	6
2.2 Условия установки клиентской части Платформы	6
3 Последовательность действий по установке и настройке (развертыванию) Платформы	8
3.1 Состав машинного носителя	8
3.2 Подготовительные действия	8
3.3 Установка и настройка Платформы	14

1 ТЕРМИНЫ И СОКРАЩЕНИЯ

АРМ	–	автоматизированное рабочее место
ОЗУ	–	оперативное запоминающее устройство (оперативная память)
ОС	–	операционная система
ПО	–	программное обеспечение
СУБД	–	система управления базами данных
SSL	–	уровень защищенных сокетов (secure sockets layer)
URL	–	унифицированный указатель ресурса (uniform resource locator)

2 УСЛОВИЯ УСТАНОВКИ

2.1 Условия установки серверной части Платформы

Серверное оборудование для установки (развертывания) серверной части Платформы поддерживает размещение ПО в среде виртуализации.

Необходимое ПО и минимальные технические характеристики серверного оборудования для развертывания серверной части платформы указаны в Таблице

.

Таблица 1 – Минимальные технические характеристики серверного оборудования для развертывания серверной части Платформы

№	Назначение сервера и установленное системное ПО	Параметр	Мин. значение
1.1.	server.mygeomap Веб-сервер (в режиме обратного проксирования). Сервер приложений. Требуемое ПО: Linux Debian 12.9 Nginx 1.27 NodeJS 23.9.0 Martin 0.17 Python 3.11+	Количество серверов	1
		Ядер процессора	4
		ОЗУ, Гб	8
		Размер диска, Гб	128
		Производительность диска, IOPS ¹	150
		Сетевой интерфейс, Гбит/с	10
1.2.	db.mygeomap Сервер СУБД Требуемое ПО: Linux Debian 12.9 Docker Engine 27.3.1 PostgreSQL 17	Количество серверов	1
		Ядер процессора	4
		ОЗУ, Гб	8
		Размер диска, Гб	512
		Производительность диска, IOPS	200
		Сетевой интерфейс, Гбит/с	10

2.2 Условия установки клиентской части Платформы

2.2.1 Аппаратные требования к автоматизированному рабочему месту пользователя Платформы

¹ Здесь и далее: IOPS – из условия, что средняя длина очереди диска (Aavgqu-sz) должна быть менее 1 единицы и стремится к нулю

В качестве аппаратного обеспечения АРМ пользователей Платформы рекомендуется использование технических средств с характеристиками, приведенными в Таблице .

Таблица 2 – Технические характеристики АРМ пользователей

№ п.п.	Технические характеристики	Значение по ТТТ, не менее
1	Системный блок	
1.1	частота процессора, ГГц (не менее)	2,6
1.2	объем оперативной памяти, ГБ (не менее)	4
1.3	объем запоминающего устройства, ГБ (не менее)	128
1.4	сетевой контроллер 10 Мбит (не менее)	1
2	Монитор	
2.1	разрешение экрана, точек на дюйм (не менее)	1920 x 1080

2.2.2 Программные требования к автоматизированному рабочему месту пользователя

Состав и функции частей программного обеспечения Платформы для серверного оборудования и АРМ представлены в Таблице .

Таблица 3 – Состав и функции программного обеспечения Платформы для АРМ пользователей

Наименование	Описание выполняемой функции
Microsoft Windows Apple MacOS Linux	ОС для АРМ
Chrome, Firefox	Браузер на выбор пользователя, предпочтительнее Chrome

3 ПОСЛЕДОВАТЕЛЬНОСТЬ ДЕЙСТВИЙ ПО УСТАНОВКЕ И НАСТРОЙКЕ (РАЗВЕРТЫВАНИЮ) ПЛАТФОРМЫ

3.1 Состав машинного носителя

Платформа поставляется на машинном носителе типа компакт-диск. Корневой каталог диска содержит:

- *server* – каталог с приложениями Платформы;
- *services* – каталог с сервисами;
- *configs* – каталог с примерами файлов конфигураций;
- *version.txt* – файл с номером версии.

Содержимое компакт диска с установочными файлами Платформы необходимо скопировать на серверное оборудование (в каталог *dist* домашнего каталога пользователя).

3.2 Подготовительные действия

Для корректного функционирования Платформы перед ее развертыванием на серверном оборудовании, указанном в Таблице , должна быть предварительно установлена ОС. Для установки ОС не требуются дополнительное ПО и выбор настроек, отсутствуют требования к методу разметки диска.

ОС должна быть сконфигурирована для работы со штатным репозиторием пакетов Debian для установки вспомогательных библиотек и пакетов.

Подготовительные действия должны производиться системным администратором.

3.2.1 Обновление системных пакетов

Перед началом установки пакетов Платформы необходимо обновить системные пакеты. Для этого требуется выполнить следующую команду на всем серверном оборудовании.

```
sudo apt update & apt upgrade
```

На серверах 1.1 и 1.2 из Таблицы необходимо установить вспомогательные библиотеки. Для установки библиотек, требуемых для корректного функционирования Платформы, в консоли нужно выполнить следующие команды:

```
sudo apt-get install curl
```

3.2.2 Установка пакета Docker Engine и docker-compose

Средство контейнеризации docker используется на Платформе для запуска СУБД Postgres и для запуска сервиса Martin. Для его установки выполните следующие шаги. Все команды выполняются на сервере 1.2.

1) Убедитесь, что в системе нет никаких пакетов, связанных с Docker Engine предыдущих версий. Для этого выполните в консоли команду, которая удалит все существующие пакеты, способные вызвать конфликты

```
for pkg in docker.io docker-doc docker-compose podman-docker  
containerd runc; do sudo apt-get remove $pkg; done
```

2) Настройка репозитория с пакетами Docker. Выполните в консоли следующие команды:

```
sudo apt-get update  
sudo apt-get install ca-certificates curl  
sudo install -m 0755 -d /etc/apt/keyrings  
sudo curl -fsSL https://download.docker.com/linux/debian/gpg  
-o /etc/apt/keyrings/docker.asc  
sudo chmod a+r /etc/apt/keyrings/docker.asc  
echo \  
"deb [arch=$(dpkg --print-architecture) signed-  
by=/etc/apt/keyrings/docker.asc]  
https://download.docker.com/linux/debian \  
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" |  
\
```

```
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null  
sudo apt-get update
```

3) Установка пакетов Docker. Выполните в консоли следующую команду:

```
sudo apt-get install docker-ce docker-ce-cli containerd.io  
docker-buildx-plugin docker-compose-plugin
```

4) Для проверки корректности установки Docker Engine выполните следующую команду:

```
sudo docker run hello-world
```

В результате выполнения этой команды должен быть скачан и запущен тестовый контейнер, который выведет приветственное сообщение.

Утилита `docker-compose` используется для оркестрации контейнеров. Для её установки на сервере 1.2 выполните следующие команды.

```
sudo apt-get install docker-compose-plugin
```

Для проверки корректности установки выполните в консоли сервера 1.2 следующую команду, которая должна отобразить номер версии утилиты.

```
docker compose version
```

3.2.3 Установка и настройка сервера базы данных PostgreSQL с расширением PostGIS

Платформа использует для своей работы СУБД PostgreSQL с расширением PostGIS, которое обеспечивает хранение и обработку геоинформационных данных. Для установки удобнее всего воспользоваться официальным образом `docker`. В каталоге *configs* машинного носителя расположен файл `docker-compose-pg.yaml`, в котором описана конфигурация для запуска контейнера СУБД на базе образа `postgis/postgis:17-3.5`. Предварительно нужно создать в домашнем каталоге пользователя каталог *postgres* и скопировать туда этот файл. Дополнительно в этом каталоге нужно создать файл *.env*, в котором указать желаемый пароль администратора базы данных (пользователь

postgres) и дополнительное расширение pg_trgm, которое служит для обработки полнотекстовой информации. Далее представлен пример файла *.env*, также пример файла находится в каталоге configs машинного носителя (env.postgres)

```
POSTGRES_PASSWORD: пароль
POSTGRES_EXTENSIONS: pg_trgm
```

Запуск контейнера осуществляется командой в консоли:

```
docker compose -f docker-compose-pg.yaml up
```

Для внесения изменений в системные настройки экземпляра СУБД PostgreSQL можно использовать файл с именем egeo-prod-postgres-17.conf При запуске контейнера этот файл будет отображен на стандартный конфигурационный файл /etc/postgresql.conf

Для работы Платформы предпочтительнее создать новую базу данных. В дальнейшем в качестве имени базы данных будет использоваться *egeo_prod* Также настоятельно рекомендуем создать дополнительного пользователя базы данных, в дальнейшем будем использовать для этого имя *egeo_prod*

3.2.4 Установка пакета Martin

Для формирования векторных изображений карты на основе сведений из базы данных используется пакет Martin версии 0.17 Установка этого пакета выполняется на сервере 1.1 из Таблицы . Для установки нужно выполнить в консоли следующие команды.

```
mkdir martin
cd martin
curl -O
https://github.com/maplibre/martin/releases/latest/download/martin-Debian-x86\_64.deb
sudo dpkg -i /martin-Debian-x86_64.deb
```

Для проверки корректности установки можно выполнить в консоли следующую команду.

```
martin --help
```

3.2.5 Установка пакета NodeJS

Для сборки и запуска сервисов Платформы необходимо установить пакет NodeJS. Установка этого пакета выполняется на сервере 1.1 из Таблицы . Для установки нужно выполнить следующую команду.

```
curl -fsSL https://deb.nodesource.com/setup_23.x -o  
nodesource_setup.sh  
  
sudo apt-get install -y nodejs
```

Для проверки корректности установки пакета NodeJS можно использовать следующую команду.

```
node -v
```

В результате выполнения этой команды будет отображен номер версии пакета NodeJS, он не должен быть ниже **22.15.0**.

3.2.6 Установка Python

Для работы сервисов Платформы необходимо наличие интерпретатора python версии 3.11 Проверить наличие установленного интерпретатора и его версии можно с помощью следующей команды в консоли

```
python --version
```

Если интерпретатор python не установлен, тогда нужно выполнить следующие команды.

```
sudo apt install python3
```

Дополнительно нужно установить утилиту pip для работы с пакетами python. Для этого выполните в консоли следующую команду

```
sudo apt install python3-pip
```

Для установки модуля работы с виртуальным окружением `venv` нужно выполнить в консоли следующую команду.

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

3.2.7 Сборка и установка Nginx

Nginx необходимо установить на сервере 1.1 из Таблицы . Для установки Nginx следует выполнить следующую команду:

```
sudo apt-get install -y nginx
```

После установки Nginx необходимо перезапустить сервис, выполнив команды консоли:

```
sudo systemctl enable nginx
```

```
sudo systemctl restart nginx
```

и убедиться в его работоспособности путем проверки статуса, который будет отображен командой:

```
sudo systemctl status nginx
```

3.2.8 Установка пакета supervisor — подсистема управления процессами

Для запуска отдельных компонент платформы используется подсистема управления процессами `supervisor`. Установка осуществляется с помощью следующей команды.

```
sudo apt-get install supervisor
```

Для запуск сервисов Платформы можно использовать файл конфигурации `supervisord.conf`. Пример этого файла приведен в каталоге **configs** машинного носителя, файл `supervisord.conf`. В следующей таблице приведено описание основных секций этого файла.

Название секции	Описание
[supervisord]	Общие настройки демона <code>supervisord</code>
[supervisorctl]	Общие настройки для клиента (консольного приложения) <code>supervisorctl</code>

[program:back]	Запуск сервера Платформы
[fcgi-program:admin]	Запуск административной части платформы, WEB-интерфейс и API аутентификации и регистрации пользователей.
[fcgi-program:api]	Запуск сервиса трансформации координатных систем
[fcgi-program:visual_plotly]	Запуск сервиса 3D визуализации моделей
[program:georef]	Запуск сервиса геопривязки изображений
[program:reserves]	Запуск сервиса подсчета запасов
[program:styles_api]	Запуск сервиса генерации стилей
[program:wiki]	Запуск сервиса Wiki - помощи по системе

Запуск демона supervisor осуществляется с помощью команды в консоли:

```
supervisord -c supervisord.conf
```

Предварительно нужно перейти в каталог, где находится файл `supervisord.conf`. Рекомендуем создать для этого специальный каталог **configs** в домашнем каталоге пользователя.

Для управления запуском сервисов можно использовать утилиту `supervisorctl`. Запуск этой утилиты осуществляется из каталога, где находится файл `supervisord.conf` следующей командой.

```
supervisorctl -c supervisord.conf
```

Для просмотра статуса всех сервисов Платформы введите команду **status**. Для запуска всех сервисов введите команду **start all**. Для остановки всех сервисов введите команду **stop all**. Для выхода из утилиты введите команду **exit**.

3.3 Установка и настройка Платформы

3.3.1 Настройка сервера геоизображений Martin

Сервер геоизображений Martin запускается с использованием среды контейнеризации `docker`. Создайте в домашнем каталоге каталог `martin` и скопируйте туда файлы `martin-config.yaml` и `martin-docker-compose.yaml` из каталога **configs** машинного носителя. Далее переименуйте эти файлы в `config.yaml` и `docker-compose.yaml` соответственно.

По умолчанию сервер Martin работает на порту 9000. При необходимости, можно внести соответствующие изменения в секции ports файла docker-compose.yaml

Перед запуском сервера нужно внести изменения в конфигурационный файл config.yaml и указать строку подключения к серверу базы данных PostgreSQL.

Запуск сервера Martin осуществляется с помощью команды в консоли:

```
docker compose up
```

Эту команду нужно выполнять, находясь в каталоге ~/martin

Для проверки работоспособности сервера Martin можно сделать HTTP запрос GET на локальный сервер <http://localhost:9000/catalog> Ответ на запрос должен иметь статус 200.

3.3.2 Установка и настройка сервера Платформы

Для установки сервера (backend) Платформы нужно распаковать архив backend.tgz в каталог ~/eGeo/Backend. Для этого нужно скопировать файл backend.tgz из каталога server машинного носителя в каталог ~/eGeo/Backend и выполнить команду в консоли

```
tar -xvf backend.tgz
```

После этого нужно установить необходимые для работы JS-пакеты следующей командой.

```
npm install
```

После установки пакетов сервер готов к запуску, однако предварительно нужно задать ряд конфигурационных параметров. Эти параметры расположены в файле config.js каталога Backend. Описание параметров приведено в следующей таблице. В качестве SERVER_NAME нужно указать доменное имя (включая протокол) сервера, на котором будет развернута платформа (или отдельный сервис).

Параметр	Описание	Рекомендуемое значение
authURL	Адрес сервиса проверки аутентификации пользователя	SERVER_NAME/auth/
checkURL	Адрес сервиса проверки аутентификации пользователя с использованием токена	/check-auth/
loginURL	Адрес сервиса аутентификации пользователя	SERVER_NAME/login/
resetURL	Адрес сервиса для сброса пароля	SERVER_NAME/password_reset/
regURL	Адрес сервиса регистрации нового пользователя	SERVER_NAME/register/
kern	Адрес сервиса работы с базой данных кернов	
kernapi	Адрес сервиса API работы с базой данных кернов	
reprojectApi	Адрес сервиса преобразования геоанных	SERVER_NAME/crs_converter/
ProtocolMartin	Тип протокола для доступа к сервису Martin	https
ProtocolBackend	Тип протокола для доступа к серверу Платформы	https
HOSTMartin	Адрес сервиса Martin	SERVER_NAME
HOSTBackend	Доменный адрес Платформы	SERVER_NAME
PORTBackend	Номер порта адреса Платформы для доступа извне	443
PORTMartin	Номер порта сервера Martin	
StartPortBackend	Номер порта, на котором запускается сервер Платформы	9001
PrefixMartin	Часть адреса URL, используемая для сервиса Martin	/martin/rpc для версии 0.17
PrefixBackend	Часть адреса URL, используемая для доступа к серверу Платформы	/server
FormatMartin	Формат расширения URL, используемый для сервиса Martin	.pbf для версии 0.17
georef	Адрес сервиса геопривязки изображений	SERVER_NAME/georef/georef
georefReproject	Адрес сервиса преобразования проекции изображения	SERVER_NAME/georef/imagerproj
georefApi	Адрес сервиса API геопривязки изображений	SERVER_NAME/georef
reservesApi	Адрес сервиса подсчета запасов	SERVER_NAME/reserves_api
viewer3dApi	Адрес сервиса 3D визуализации	SERVER_NAME/3dvisualizer

Пример конфигурации приведен в файле `back_config.js` в каталоге `configs` машинного носителя.

Для настройки доступа в базе данных в каталоге `~/eGeo/Backend` нужно создать файл `.env` с указанием параметров для доступа к базе. В качестве образца можно использовать файл `env.backend` из каталога **configs** машинного носителя. Далее приведено описание параметров из этого файла.

Название параметра	Описание
<code>pooluser</code>	Имя пользователя базы данных, см. раздел 3.2.3
<code>poolpassword</code>	Пароль пользователя базы данных
<code>poolhost</code>	Адрес сервера, где расположена СУБД, сервер 1.2 из Таблица 1
<code>poolport</code>	Порт, используемый сервером базы данных
<code>pooldatabase</code>	Имя базы данных
<code>poolmax</code>	Максимальное количество подключений
<code>poolmin</code>	Минимальное количество подключений
<code>poolidleTimeoutMillis</code>	Время завершения сессии по простоям

Для запуска сервиса можно использовать следующую команду в секции конфигурационного файла `supervisord.conf`

```
[program:back]
```

```
...
```

```
node Server.js
```

Полное описание секции приведено в примере, файл `supervisord.conf` в каталоге **configs** машинного носителя.

3.3.3 Установка, настройка и сборка клиентской (фронтенд) части Платформы.

Клиентская часть платформы предварительно должна быть транслирована в набор JS-скриптов, HTML-страниц, стилевых отображений и изображений. Непосредственно обработку запросов к фронтенд части Платформы осуществляет сервис `nginx`, установка и настройка которого описана далее.

Для установки, настройки и сборки фронтенд части Платформы выполните следующие действия.

1) В каталоге `~/eGeo/` нужно создать каталог **Frontend**, скопировать туда файл `frontend.tgz` из каталога **server** машинного носителя и распаковать его с использованием следующей команды в консоли

```
tar -xvf frontend.tgz
```

2) Внести изменения в конфигурационный файл `~/eGeo/Frontend/src/config.js` Формат конфигурационного файла и используемые параметры аналогичны серверной части Платформы, см. раздел 3.3.2

3) Провести трансляцию и сборку исходного кода командой

```
npm run-script build
```

В результате работы этой команды будет создан каталог `build`, включающий в себя необходимые файлы для работы фронтенд части Платформы.

4) Скопировать созданные файлы в каталог, обслуживаемый сервисом `nginx`. В Платформе сервер `nginx` настроен на обслуживание каталога `/var/www`, копирование осуществляется следующей командой.

```
sudo cp -r build/ /var/www/
```

В случае успешной установки и настройки `nginx` и фронтенд части Платформы в ответ на запрос `GET` для указанного доменного имени будет возвращаться корректный статус и содержимое файла `index.html`

3.3.4 Настройка сервиса `nginx`

После установки и настройки сервисов Платформы, необходимо настроить службу `nginx`.

В качестве образца файла настройки можно использовать файл ***egeo-nginx.conf***, который находится в каталоге ***configs*** машинного носителя. Отредактированный файл нужно разместить в каталоге ***/etc/nginx/sites-enabled/***. Конфигурационные параметры службы `nginx` указаны в таблице .

Таблица 4 – Конфигурационные параметры службы nginx

Название параметра	Описание
server_name	Доменное имя используемого сервера, должно совпадать с конфигурационным параметром HOST сервиса web (см. 3.2.2)
location/set	В качестве значения для переменной нужно указать адрес сервера и порт, совпадающие с конфигурационными параметрами HOST и PORT сервиса web (см.3.2.2)
ssl_certificate	Путь к файлу SSL-сертификата, созданного для доменного имени сервера
ssl_certificate_key	Путь к файлу ключа SSL-сертификата, созданного для доменного имени сервера

В данном файле, кроме общей настройки сервера Платформы, приведены секции для обратной пересылки запросов (reverse-forwarding), которые используются для запуска отдельных сервисов системы. Далее в таблице представлено описание этих секций

Обрабатываемый адрес	Описание	Рекомендуемый порт для запуска сервиса
/server/	Основной сервер Платформы (backend)	9001
/docs/	Каталог статических документов	
/martin/	Сервер геоизображений Martin	9000
/register/	Сервис регистрации пользователей, реализуется в сервисе административной части, см. 3.3.6	8098
/login/	Сервис логина, реализуется в сервисе административной части, см. 3.3.6	8098
/3dvisualizer/	Сервис 3D-визуализации	9005
/check-auth/	Сервис проверки аутентификации по токену, реализуется в сервисе административной части, см. 3.3.6	8098
/auth/	Сервис аутентификации, реализуется в сервисе административной части, см. 3.3.6	8098

После настройки службы nginx, ее необходимо перезапустить, выполнив команду консоли:

```
sudo systemctl restart nginx
```

3.3.5 Настройка сервиса supervisord

3.3.6 Установка и настройка административной части Платформы

Для установки и настройки административной части Платформы нужно выполнить следующие действия.

1) Создать каталоги **services/common** в домашнем каталоге. Скопировать туда файл **admin.tgz** из каталога **services** машинного носителя. Распаковать этот файл командой:

```
tar -xvf admin.tgz
```

2) После распаковки архива будет создан каталог **egeo_admin_user**. Нужно перейти в этот каталог и создать виртуальное окружение **python**, выполнив в консоли команду

```
python -m venv .venv
```

3) Установить требуемые для работы сервиса пакеты, выполнив в консоли команду

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

4) Создать файл **.env** и указать в нем следующие конфигурационные параметры:

- **DATABASE_URL**: строка соединения с базой данных
- **SECRET_KEY**: секретный ключ для криптографических целей.
- **EMAIL_HOST_PASSWORD**: пароль для доступа к учетной записи электронной почты.

5) Внести изменения в файл **admin_user/settings/production.py** Обязательно нужно добавить запись в список **ALLOWED_HOSTS** с указанием имени домена, где развернут сайт. Пример представлен в каталоге **configs**, файл **egeo_admin_production.py**

6) Для запуска сервиса используется секция [fcgi-program:admin] в файле `supervisord.conf`, см. раздел 3.2.8

3.3.7 Установка и настройка сервиса конвертации трансформации координатных систем

Сервис предназначен для перепроецирования (трансформации) геопространственных векторных данных между различными системами координат (CRS — Coordinate Reference System).

Для установки и настройки сервиса нужно выполнить следующие действия.

1) Создать в каталоге **services** каталог **eGeo_python**. Скопировать туда файл `crs_converter.tgz` из каталога **services** машинного носителя. Распаковать этот файл командой:

```
tar -xvf crs_converter.tgz
```

2) После распаковки архива будет создан каталог **egeo_api**. Нужно перейти в этот каталог и создать виртуальное окружение `python`, выполнив в консоли команду

```
python -m venv .venv
```

3) Установить требуемые для работы сервиса пакеты, выполнив в консоли команду

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

4) Внести изменения в файл `egeo_api/settings/production.py` Обязательно нужно добавить запись в список `ALLOWED_HOSTS` с указанием имени домена, где развернут сайт. Пример представлен в каталоге **configs**, файл `crs_converter_production.py`

5) Для запуска сервиса используется секция [fcgi-program:api] в файле `supervisord.conf`, см. раздел 3.2.8

3.3.8 Установка и настройка сервиса 3D визуализации моделей

Для установки и настройки сервиса нужно выполнить следующие действия.

1) Создать в каталоге **services** каталог **petro_utils**. Скопировать туда файл **3dvisualizer.tgz** из каталога **services** машинного носителя. Распаковать этот файл командой:

```
tar -xvf 3dvisualizer.tgz
```

2) После распаковки архива будет создан каталог **visualizer_plotlyjs**. Нужно перейти в этот каталог и создать виртуальное окружение python, выполнив в консоли команду

```
python -m venv .venv
```

3) Установить требуемые для работы сервиса пакеты, выполнив в консоли команду

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

4) Внести изменения в файл **mgmvis/settings/production.py** Обязательно нужно добавить запись в список **ALLOWED_HOSTS** с указанием имени домена, где развернут сайт. Пример представлен в каталоге **configs**, файл **3dvisualizer_production.py**

5) Для запуска сервиса используется секция **[fcgi-program:visual_plotly]** в файле **supervisord.conf**, см. раздел 3.2.8

3.3.9 Установка и настройка сервиса геопривязки изображений

Сервис предназначен для геопривязки растровых изображений (геореференцирования) путём сопоставления пиксельных координат изображения с географическими координатами на карте.

Для установки и настройки сервиса нужно выполнить следующие действия.

1) Создать в каталоге **services** каталог **eGeo_python**, если ранее этот каталог не был создан. Скопировать туда файл **georef.tgz** из каталога **services** машинного носителя. Распаковать этот файл командой:

```
tar -xvf georef.tgz
```

2) После распаковки архива будет создан каталог **georef**. Нужно перейти в этот каталог и создать виртуальное окружение **python**, выполнив в консоли команду

```
python -m venv .venv
```

3) Установить требуемые для работы сервиса пакеты, выполнив в консоли команду

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

4) Для запуска сервиса используется секция **[program:georef]** в файле **supervisord.conf**, см. раздел 3.2.8

3.3.10 Установка и настройка сервиса подсчета запасов ^[обв]

Для установки и настройки сервиса подсчета запасов нужно выполнить следующие действия.

1) В каталоге **services** создать каталог **eGeo_python**, если он ранее не был создан. Скопировать туда файл **reserves.tgz** из каталога **services** машинного носителя. Распаковать этот файл командой:

```
tar -xvf reserves.tgz
```

2) После распаковки архива будет создан каталог **ReservesApi**. Нужно перейти в этот каталог и создать виртуальное окружение **python**, выполнив в консоли команду

```
python -m venv .venv
```

3) Установить требуемые для работы сервиса пакеты, выполнив в консоли команду

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

4) Для запуска сервиса используется секция [program:reserves] в файле `supervisord.conf`, см. раздел 3.2.8

3.3.11 Установка и настройка сервиса генерации стилей

Сервис генерации json стилей карты для `maplibre`. Применяется в мобильном приложении для отображения всех слоёв карт, а также для офлайн скачивания. Содержит в себе настройки слоёв, шрифтов, иконок, стилей геометрий, ссылки для получения тайлов (частей карты на разных зумах и координатах). Растровые изображения для спутникового слоя и векторные в Protocolbuffer Binary Format.

Для установки и настройки сервиса нужно выполнить следующие действия.

1) В каталоге **services** создать каталог **eGeo_python**, если он ранее не был создан. Скопировать туда файл `styles_api.tgz` из каталога **services** машинного носителя. Распаковать этот файл командой:

```
tar -xvf styles_api.tgz
```

2) После распаковки архива будет создан каталог **styles_api**. Нужно перейти в этот каталог и создать виртуальное окружение python, выполнив в консоли команду

```
python -m venv .venv
```

3) Установить требуемые для работы сервиса пакеты, выполнив в консоли команду

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

4) Для запуска сервиса используется секция [program:styles_api] в файле `supervisord.conf`, см. раздел 3.2.8