

Общество с ограниченной ответственностью Меотех



**Инструкция по установке экземпляра программного
обеспечения
«MeoDesk»**

2026 г.

Оглавление

1 Функциональное описание и средства разработки.....	3
2 Технические требования.....	3
3 Установка и настройка серверной части	4
3.1 Требования к системе	4
3.2 Предварительная настройка системы.....	4
3.3 Установка и настройка PostgreSQL.....	4
3.4 Установка сервера «MeoDesk»	5
3.5 Настройка базы данных в PostgreSQL.....	6
3.6 Настройка автоматического запуска ПО через systemd.....	7
3.7 Установка NGINX.....	8
3.8 Настройка NGINX	8
3.9 Настройка ИИ интеграции	10
3.10 Ограничения в процессе эксплуатации	11

1 Функциональное описание и средства разработки

Back-end	– Язык программирования C# – Платформа Microsoft .NET 10.0 (открытые исходные коды) – Платформа ASP.NET Core (открытые исходные коды)
Front-end	– Язык программирования JavaScript – Платформа для создания rich internet applications Razor Pages (открытые исходные коды).

Система построена на основе сервисной архитектуры. Основные блоки системы реализованы как веб-сервисы и взаимодействуют друг с другом по протоколам, используемым в Интернет- и Интранет- сетях.

Базовые компоненты взаимодействуют по REST API. В качестве формата сериализации данных при вызове REST API используется JSON.

Публикация общего REST API осуществляется с использованием технологии reverse проху. Прокси-сервер – инфраструктурный компонент, который используется для распределения нагрузки при горизонтальном масштабировании системы, сведения API-вызовов к одному доменному адресу, осуществления кеширования, авторизации и аутентификации запросов. В качестве прокси-сервера поддерживаются NGINX, HAProxy или YARP. В инструкции описан рекомендуемый метод по настройке прокси-сервера NGINX.

REST API и сервера очередей также могут использоваться для интеграции с внешними системами клиента.

Хранение данных сервисов осуществляется с использованием СУБД с открытым исходным кодом PostgreSQL.

2 Технические требования

Сервер «MeoDesk»:

- ПО – Ubuntu Server 24.04;
- 4 Гб ОЗУ или более;
- Процессор 2,6 ГГц, 4 ядра или более;

- Жесткий диск (SSD) 250Гб свободного места и более;
- Наличие Интернета;
- Внешний IP-адрес для подключения мобильного приложения;
- Доступ к портам сервера 80 и 443 из Интернета по внешнему IP;
- Зарегистрированное и привязанное к внешнему IP доменное имя;
- SSL-сертификат для организации HTTPS-подключения по зарегистрированному имени.

Примечание:

На сервере может быть развернута исключительно внедряемая система, всё необходимое для нее ПО. Наличие и функционирование на сервере иного ПО не допускается.

3 Установка и настройка серверной части

3.1 Требования к системе

Для установки серверной части системы «MeoDesk» необходим физический или виртуальный сервер с установленной ОС Ubuntu Server 24.04.

Для работы сервера должны быть установлены следующие компоненты:

- [PostgreSQL 16](#)
- [NGINX](#)

Все дальнейшие действия производятся от пользователя root (администратор) в консоли сервера.

3.2 Предварительная настройка системы

Для корректной установки пакетов в системе необходимо обновить репозитории следующей командой:

```
sudo apt update
```

3.3 Установка и настройка PostgreSQL

Для установки PostgreSQL версии 16 необходимо ввести следующую команду:

```
sudo apt install -y postgresql
```

Далее, необходимо убедиться, что была установлена необходимая версия ПО при помощи команды:

```
psql --version
```

В консоли должен отобразиться текст подобного содержания:

```
psql (PostgreSQL) 16.13 (Ubuntu 16.13-0ubuntu0.24.04.1)
```

Версия после «(PostgreSQL)» должна отобразиться начиная с «16...»

После установки PostgreSQL, необходимо провести её конфигурацию. Нужно ввести следующую команду для открытия консоли PostgreSQL от имени пользователя «postgres»:

```
sudo -u postgres psql
```

В консоли должно отобразиться сообщения подобного содержания:

```
psql (16.13 (Ubuntu 16.13-0ubuntu0.24.04.1))
```

```
Type "help" for help.
```

```
postgres=#
```

Далее, нужно поменять пароль пользователю «postgres» следующей командой:

```
ALTER USER postgres with encrypted password 'НОВЫЙ_ПАРОЛЬ';
```

Где «НОВЫЙ_ПАРОЛЬ» - пароль для пользователя postgres. Его необходимо запомнить – далее он будет необходим в конфигурации ПО.

После ввода команды должно отобразиться сообщение «ALTER ROLE», после чего можно выйти из программы «psql» комбинацией клавиш «Ctrl+D» или вводом команды «quit».

3.4 Установка сервера «MeoDesk»

Для установки сервера «MeoDesk» необходимо разместить файлы сервера «MeoDesk» на машине в любом месте. После размещения сервера необходимо сделать исполняемый файл «MeoDesk» доступным к выполнению. Для этого, находясь в папке с исполняемым файлом, необходимо ввести следующую команду:

```
sudo chmod +x MeoDesk
```

3.5 Настройка базы данных в PostgreSQL

Для настройки базы данных в PostgreSQL, необходимо отредактировать файл «appsettings.json» и запустить сервер «MeoDesk» в режиме миграции. В файле «appsettings.json» нужно изменить данные для подключения к PostgreSQL – ввести пароль под ключом «Password=...» от пользователя «postgres», который был указан во время конфигурации PostgreSQL. Указать пароль необходимо в поле «DefaultConnection» внутри объекта «ConnectionStrings»:

```
"ConnectionStrings": {  
    "DefaultConnection": "User  
ID=postgres;Password=ПАРОЛЬ;Host=localhost;Port=5432;Database=meodesk"  
},
```

Где «ПАРОЛЬ» - пароль, указанный при настройке PostgreSQL ранее.

После указания пароля БД необходимо запустить сервер в режиме миграции. ПО автоматически настроит таблицы в БД PostgreSQL. Для запуска сервера в режиме миграции, в папке с исполняемым файлом, необходимо выполнить следующую команду:

```
./MeoDesk --apply-migrations
```

В консоли должна отобразиться следующая строка:

```
Applying migrations to the DB...
```

После этой строки, в консоли должны отобразиться успешные операции миграции Entity Framework Core. Пример отображаемой строки:

```
info: Microsoft.EntityFrameworkCore.Database.Command[20101]  
      Executed DbCommand (105ms) [Parameters=[], CommandType='Text',  
CommandTimeout='30']  
      CREATE DATABASE meodesk;
```

При успешном запуске сервера и настройке БД, в консоль должно вывестись следующее сообщение в конце вывода, без ошибок:

```
info: Microsoft.Hosting.Lifetime[14]
```

```
Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
    Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
    Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
    Content root path: /home/user/MeoDesk
```

3.6 Настройка автоматического запуска ПО через systemd

Для автоматического запуска сервера «MeoDesk» необходимо создать systemd-сервис. Для этого нужно создать файл по пути «/etc/systemd/system/meodesk.service» со следующим содержанием:

```
[Unit]
Description=MeoDesk
After=network.target

[Service]
WorkingDirectory=ПУТЬ К ИСПОЛНЯЕМОМУ ФАЙЛУ MeoDesk
ExecStart=ПАПКА С ИСПОЛНЯЕМЫМ ФАЙЛОМ MeoDesk
Restart=always
Type=notify

[Install]
WantedBy=multi-user.target
```

Где «ПУТЬ К ИСПОЛНЯЕМОМУ ФАЙЛУ MeoDesk» - полный путь к исполняемому файлу «MeoDesk», «ПАПКА С ИСПОЛНЯЕМЫМ ФАЙЛОМ MeoDesk» - полный путь к папке, в котором находится исполняемый файл «MeoDesk».

Далее, необходимо перезагрузить конфигурацию systemd. Для этого необходимо ввести следующую команду:

```
sudo systemctl daemon-reload
```

После этого необходимо активировать и включить созданный сервис. Делается это следующей командой:

```
sudo systemctl enable --now meodesk
```

При успешной активации сервиса, в консоль должна вывестись строка с примерно следующим содержанием:

```
Created symlink /etc/systemd/system/multi-user.target.wants/meodesk.service → /etc/systemd/system/meodesk.service.
```

После этой строки не должны отображаться ошибки. В этом случае сервис успешно активирован и сервер начал работу.

3.7 Установка NGINX

Для установки NGINX необходимо ввести следующую команду:

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

3.8 Настройка NGINX

Для настройки сайта в NGINX, необходимо редактировать стандартную конфигурацию сайта. Предварительно на сервер нужно загрузить HTTPS сертификаты сайта в виде двух файлов – файла сертификата и файла приватного ключа. Затем, необходимо редактировать файл, находящийся по пути «/etc/nginx/sites-available/default», заменив содержимое файла на следующий код:

```
server {  
    listen 80;  
    server_name ДОМЕН;  
    return 301 https://$host$request_uri;  
}  
  
server {  
    listen 443 ssl;  
    server_name ДОМЕН;  
  
    ssl_certificate ПУТЬ К СЕРТИФИКАТУ;
```

```
ssl_certificate_key ПУТЬ К ПРИВАТНОМУ КЛЮЧУ;  
ssl_protocols TLSv1.2 TLSv1.3;
```

```
location / {  
    proxy_pass http://127.0.0.1:5000;  
    proxy_set_header Host $host;  
    proxy_set_header X-Real-IP $remote_addr;  
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
}  
}
```

Где «ДОМЕН» - домен, используемый для входа в приложение, «ПУТЬ К СЕРТИФИКАТУ» - полный путь к файлу сертификата, «ПУТЬ К ПРИВАТНОМУ КЛЮЧУ» - полный путь к приватному ключу.

После сохранения файла необходимо убедиться, что сайт включён – для этого в папке «/etc/nginx/sites-enabled» должна находиться ссылка на сайт «default». Это можно проверить следующей командой:

```
ls /etc/nginx/sites-enabled -al
```

В консоль должна отобразиться информация схожего содержания:

```
drwxr-xr-x 2 root root 4096 Jan  1 00:00 .  
drwxr-xr-x 8 root root 4096 Jan  1 00:00 ..  
lrwxrwxrwx 1 root root  34 Jan  1 00:00 default -> /etc/nginx/sites-available/default
```

После этого, необходимо убедиться в отсутствии ошибок в файле, который был отредактирован. Это можно сделать следующей командой:

```
nginx -t
```

Должна отобразиться информация схожего содержания:

```
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok  
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

При отсутствии ошибок необходимо перезапустить NGINX для активации новой конфигурации. Делается это следующей командой:

nginx -s reload

В консоли должно либо ничего не отобразиться, либо должна отобразиться строка схожего содержания:

2025/01/01 00:00:00 [notice] 8284#8284: signal process started

После запуска сервиса и настройки PostgreSQL и NGINX, ПО должно быть доступно по адресу, введённому при конфигурации NGINX, а также должна быть доступна локальная версия по адресу «http://127.0.0.1:5000».

В системе пользователь-администратор создаётся посредством регистрации первого пользователя. Для регистрации пользователя в системе необходимо обратиться к «Руководству пользователя».

3.9 Настройка ИИ интеграции

Для подключения интеграции с ИИ через OpenAI-совместимый клиент, необходимо отредактировать ключи в конфигурационном файле внутри раздела «OpenAICompatibleClient»:

```
"OpenAICompatibleClient": {  
    "EnableAi": false,  
    "InstanceUrl": "http://localhost:5001/",  
    "ApiKey": "PasteYourApiKey",  
    "ModelName": "SomeModelName",  
    "EnableThinking": false,  
    "RemoveThinkSections": true  
}
```

- «EnableAi» - включает/отключает обращение к ИИ клиенту. Если значение выставлено в false, то все запросы к ИИ клиенту будут выдавать статическое значение;
- «InstanceUrl» - ссылка на инстанцию OpenAI-совместимого клиента (без дополнительных путей);
- «ApiKey» - API-ключ. Если клиент не требует API-ключа, значение может быть пустым;

- «ModelName» - название используемой модели. Если клиент не требует название модели, значение может быть пустым;
- «EnableThinking» - использовать ли размышление в ИИ при помощи тегов <think>. Если значение выставлено в false, в конце промптов добавляется «/nothink», если в true — в промпт ничего не добавляется;
- «RemoveThinkSections» - удалять ли текст, обёрнутый в теги <think>.

3.10 Ограничения в процессе эксплуатации

В процессе эксплуатации категорически не рекомендуется:

- Производить какие-либо изменения параметров в файле конфигурации **appsettings.json**. Это может привести к потере связей между компонентами системы.
- Производить апгрейд или даунгрейд компонентов системы, указанных в пункте «Технические требования».
- Переносить рабочую систему на другое оборудование или виртуальные машины. Это приведёт к потере лицензий, привязанных к учетным записям выездных сотрудников.
- Производить замену комплектующих частей компьютера (или компонентов виртуальной машины), на котором установлен сервер «MeoDesk» – материнских плат, процессоров, памяти, жестких дисков и т. п. Это может привести к потере лицензий, привязанных к учетным записям выездных сотрудников.
- Для развертывания других копий сервера потребуется приобретение дополнительных персональных лицензий.