5 ТЕХНИЧЕСКАЯ ДОКУМЕНТАЦИЯ

5.1 Руководство для запуска сервера

Приложение полностью написано на языке Java, поэтому запускать его можно без труда на разных операционных системах. Для этого необходимо запустить файл с расширением .jar, который должен храниться вместе с файлами конфигураций. На рисунке 31 изображен пример папки с сервером.

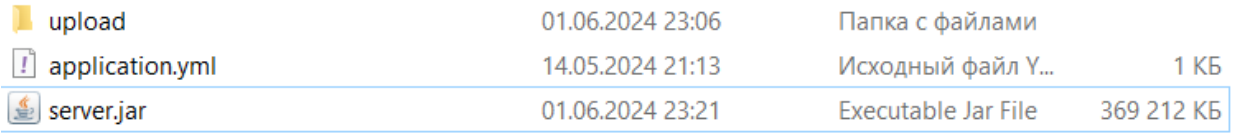

Рисунок 31. – Расположение сервера

При желании скачать исходные файлы можно обратиться к ресурсу https://github.com/shyLooney/emg_server, где они хранятся в открытом формате. На рисунке 32 изображен описываемый репозиторий.

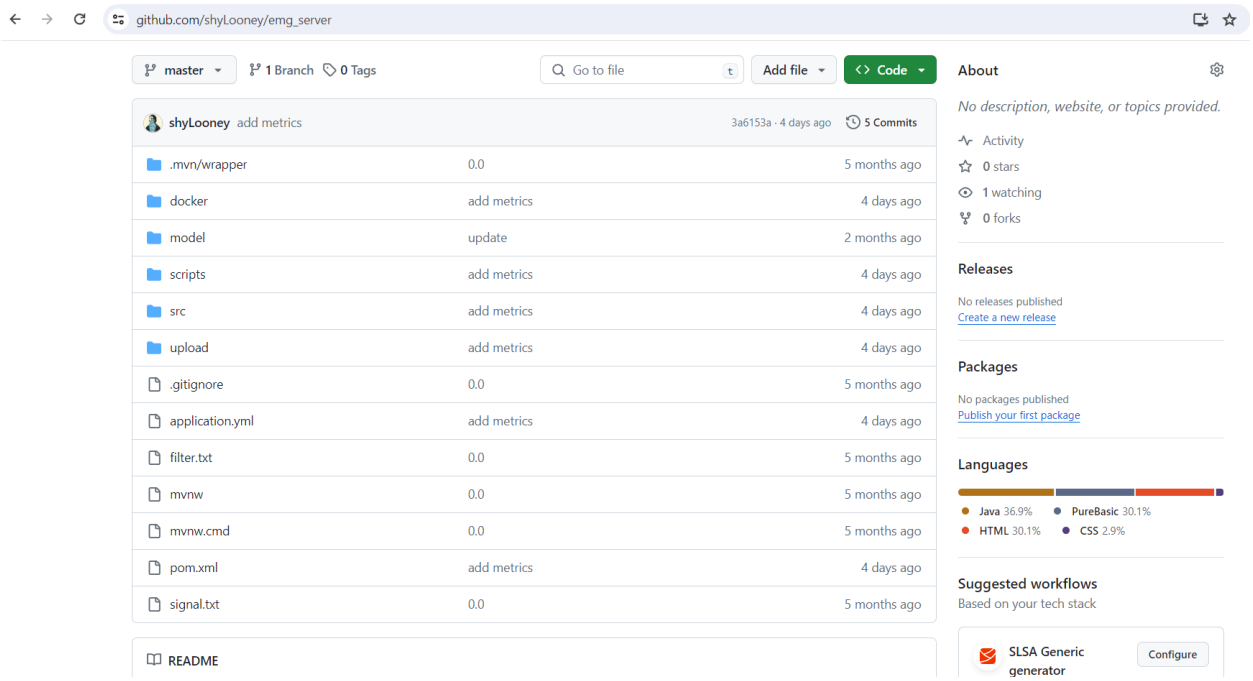

Рисунок 32. – Репозиторий приложения

Для запуска необходимо установить Java SE 21. Затем запустить jar-файл:

1. Нужно установить Java SE Development Kit не менее 21-ой версии. И установить её в переменные среды. Проверить корректность установки можно с помощью команды ‘java --version’.

2. Далее запустить командную строку. Перейти в расположение сервера. Обычно это делается с помощью команды «cd».
3. После чего необходимо ввести команду «java -jar server.jar». Преимущество такого запуска в том, что командная строка будет отображать логи отдаваемые приложением, соответственно заметно проще следить за его состоянием.

После запуска приложения должна появиться иконка, в Windows 10 — это правый нижний угол. Однако на некоторых системах, не поддерживающих TrayIcon, иконка может не появиться. Пример иконки в Windows 10 изображен на рисунке 33.

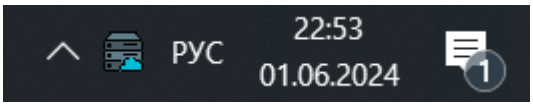

Рисунок 33. – TrayIcon приложения

Приложение содержит в себе веб-сервер, который запускается по умолчанию на порту 8080. Это можно изменить в файле application.yml. Однако, в любом случае, указанный порт не должен быть занят другими процессами, иначе приложение не будет работать.

По итогу требования для полноценного запуска и использования приложения заключаются в следующем:

- JDK 21
- Свободный указанный порт (по умолчанию 8080)
- Современный браузер

5.2 Руководство к пользовательскому веб-интерфейсу

5.2.1 Создание устройства

Взаимодействия с сервером происходит через браузер. Поэтому необходимо его запустить и ввести в адресную строку "localhost:2236" (в случае изменения порта, нужно указать новый порт вместо 2236). После чего откроется основная страница, в котором можно зарегистрировать устройство и перейти в один из графиков. Демонстрация описанного интерфейса изображена на рисунке 34.

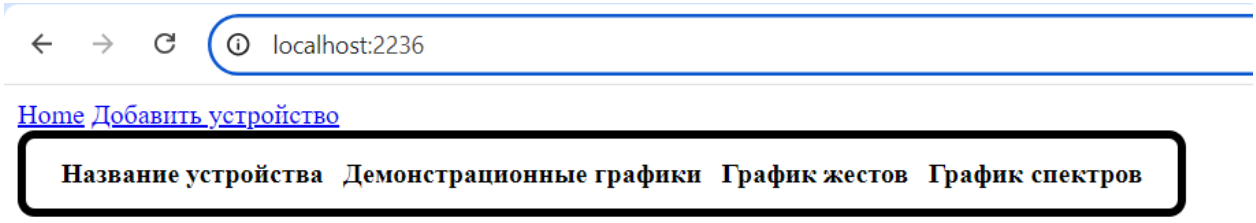

Рисунок 34. – Главная страница

Для добавления нового устройства необходимо нажать на ссылку "Добавить устройство". После чего откроется страница регистрации устройства. Пример изображен на рисунке 35.

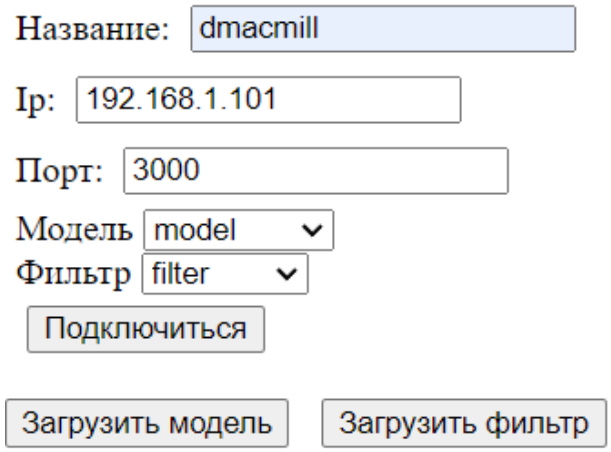

Рисунок 35. – Регистрация устройства

В поле "Название" нужно ввести название устройство, которое затем будет использовать для адресации. **Название должно быть уникальным**, то есть не должно существовать устройства с таким же именем. Затем указывается ip устройства и далее его порт. В двух выпадающих списках нужно указать название фильтра и модели. Тут же имеется возможность загрузить модель и фильтр с помощью соответствующих кнопок. Загрузка более подробно будет рассмотрена далее.

После добавления устройства произойдет переадресация на главную страницу, где пользователь может выбрать интересующий его график. Пример таблицы графиков устройства изображена на рисунке 36.

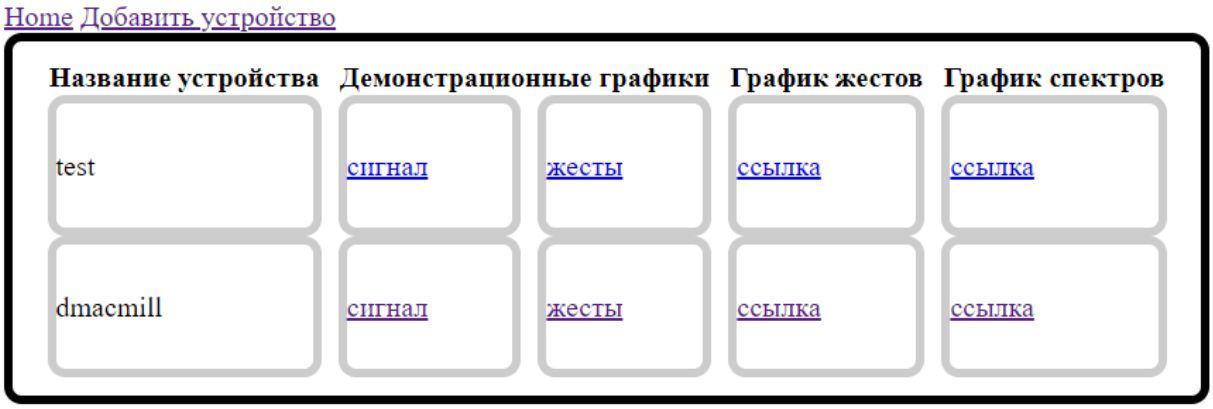

Рисунок 36. – Список устройств

5.2.2 Загрузка файлов на сервер

Загрузка файлов на сервер несколько усложнена из-за разнообразия подаваемых данных, а также необходимостью подгружать загруженные ранее файлы при запуске сервера. Для упрощения логики и пользовательского интерфейса созданы отдельные кнопки для загрузки фильтра и загрузки нейросети. При нажатии кнопок открывается одинаковое меню, изображенное на рисунке 37.

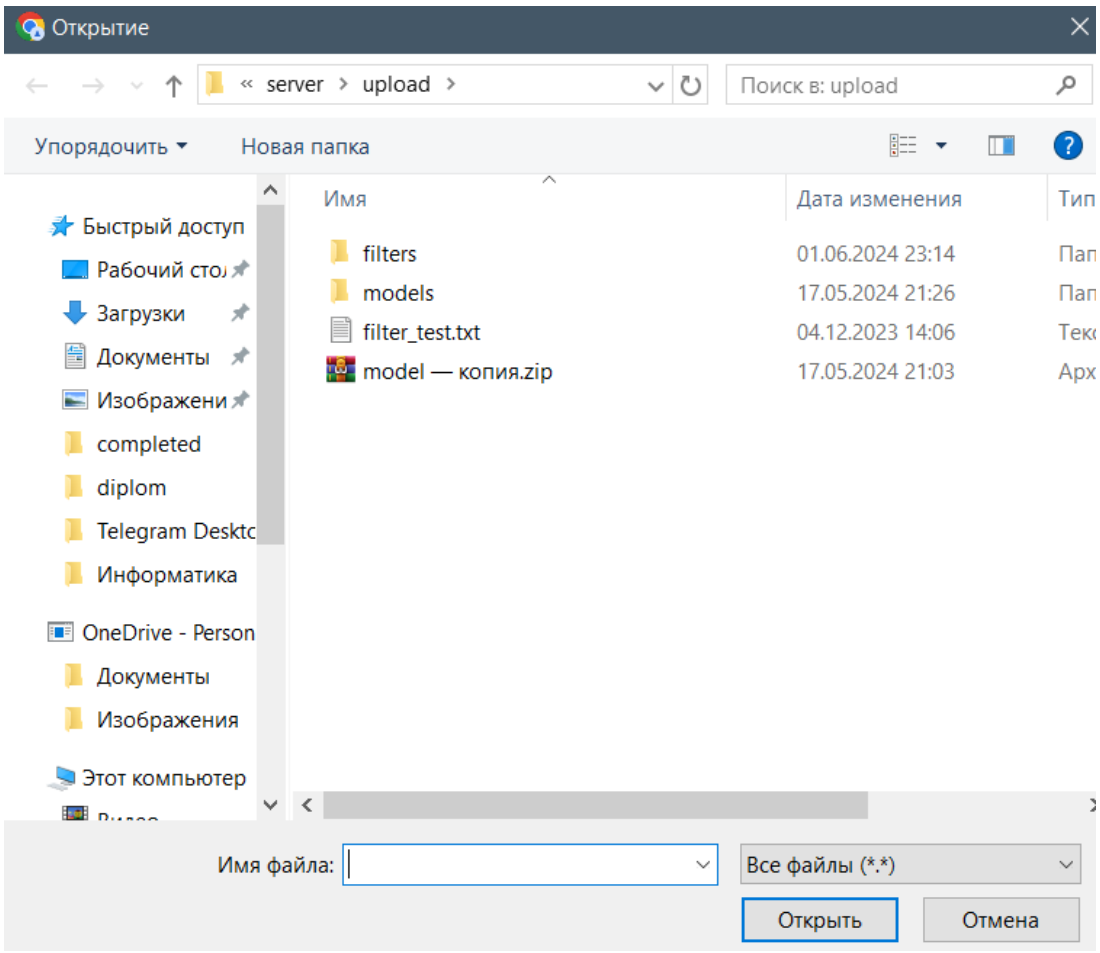

Рисунок 37. – Окно отправки файлов

Все файлы загружаются в папку, указанную в application.yml под тегами storage.location, по умолчанию папка называется upload. Если изменить название папки, то сохраненные раннее файлы в предыдущую папку не будут загружены при запуске сервера. Этот функционал можно использовать для создания профилей.

В случае загрузки фильтра, сервер ожидает файл с расширением .txt, содержащий коэффициенты, разделенные переносом строки. Если расширение будет отличаться от указанного, то **файл не сохранится**. Название текстового файла служит названием фильтра в системе. Пример файла изображен на рисунке 38.

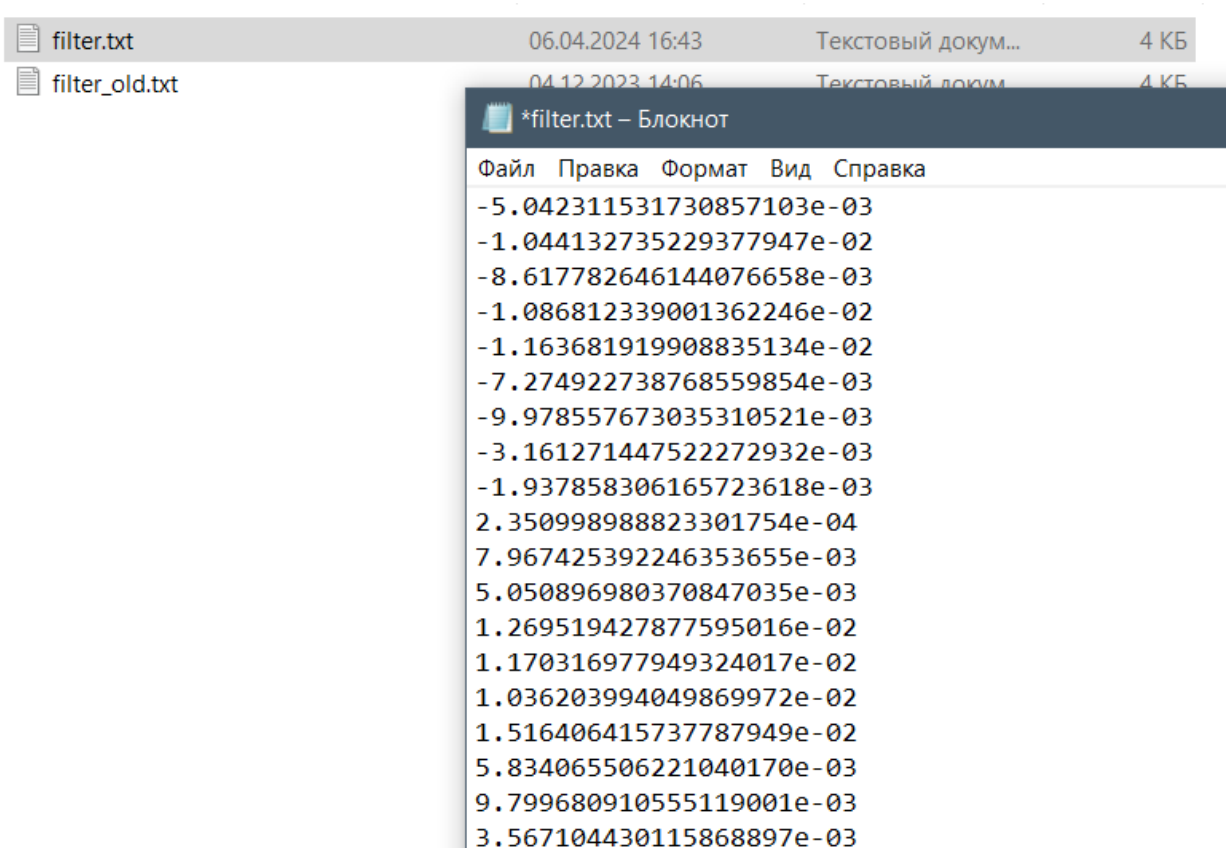

Рисунок 38. – Сохраненный фильтр в файле

В случае загрузки фильтра, сервер ожидает архив с расширением .zip, содержащий папку с сохраненной моделью tensorflow. Если расширение будет отличаться от указанного, то **файлы из архива не сохранятся**. Название папки внутри архива служит названием нейросети в системе. В папке обязательно должен присутствовать **файл конфигурации нейросети**, обозначающий операции на вхождение данных и выход. Пример такого файла изображен на рисунке 39.

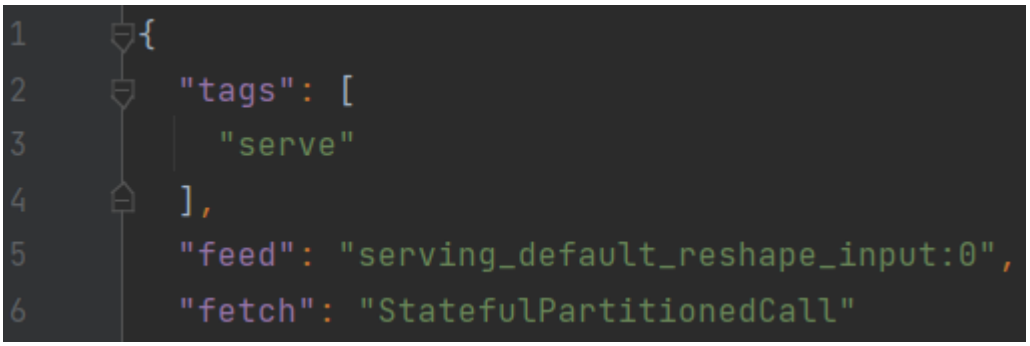

Рисунок 39. – Конфигурация нейросети

Пример извлеченного архива с нейросетью и конфигурацией в проводнике изображен на рисунке 40.

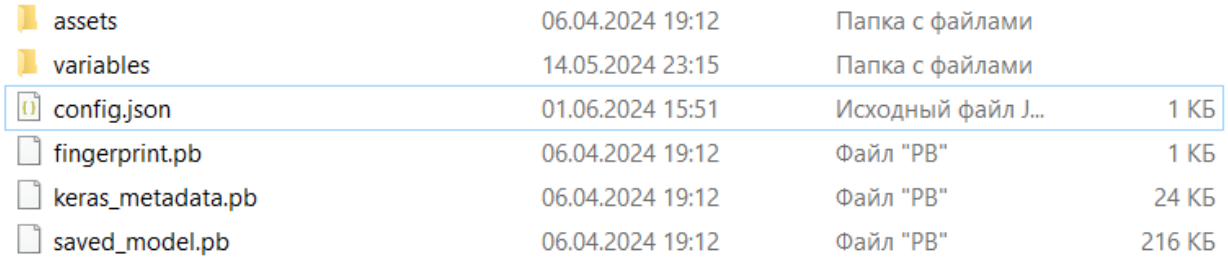

Рисунок 40. – Извлеченная модель

5.2.3 Просмотр графиков

Для перехода к графикам необходимо выбрать соответствующую ссылку из таблицы на главной странице. Демонстрационные графики не предоставляют никакого функционала, кроме изменения масштаба графика и возможность переподключиться к устройству. Отправка данных для отображения обоих графиков происходит по протоколу WebSocket.

График спектров выводит вычисленные спектры сигналов обнаруженных потенциалов действий. Каждый новый обнаруженный жест отображается случайным цветом и добавляется к уже отрисованным ранее. Для очищения накопленных данных нужно обновить страницу. Отправка данных происходит через WebSocket.

График жестов выводит сигнал обнаруженного потенциала действия и предсказание нейросетью. Предоставляет возможность сохранения отображаемых сигналов в ручном режиме, либо в автоматическом. В поле указывается название папки, поэтому необходимо вводить допустимые знаки для её создания. Сигнал сохраняется в текстовый файл .txt, где каждое значение разделено переносом строки, а название устанавливается в соответствии с названием папки с добавлением даты и времени создания файла. Пример сохраненных файлов изображен на рисунке 41.

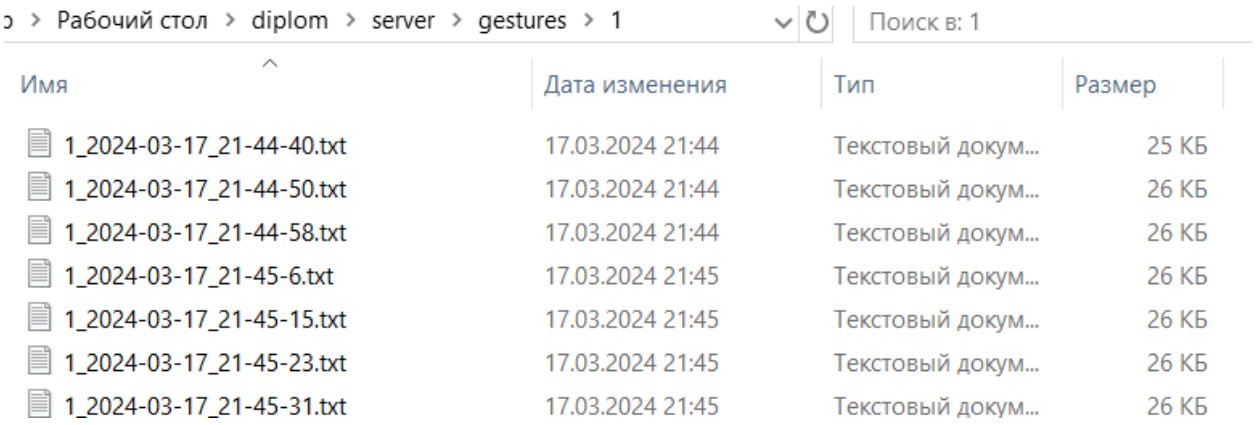

Рисунок 41. – Сохраненные сигналы

5.3 API сервера

Для просмотра API сервера подключен Swagger, который доступен по адресу /swagger-ui/. Там можно будет рассмотреть все существующие конечные точки. Их часть изображена на рисунке 42.

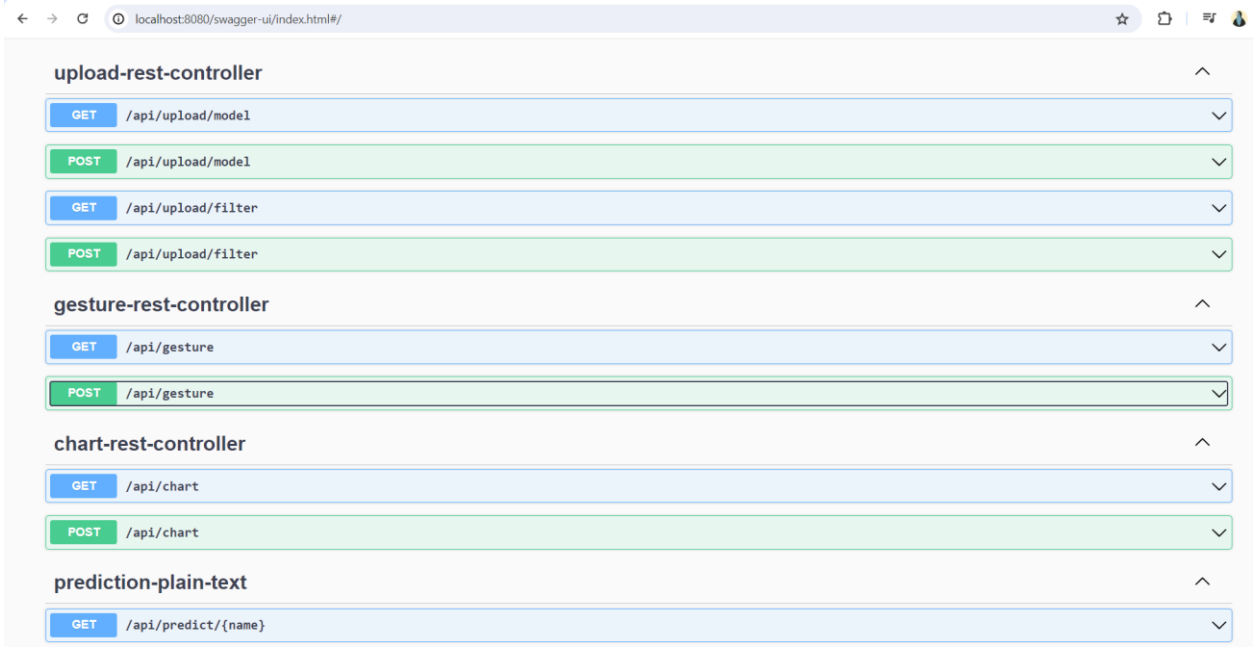

Рисунок 42. – Swagger

Основная конечная точка сервера – это предоставление выходных данных нейросети. Она расположена по адресу **api/predict/{name}**. Где {name} – название устройства. Это API возвращает 5 чисел. Первое – текущее время, записанное в типе long (8 байтовое целочисленное значение), остальные 4 являются предсказанием нейросети. Время указывается для исключения повторного использования одного и того же значения. Подразумевается, что

пользователь будет проверять это значение с временем на предыдущей итерации.

Другие конечные точки используются для обеспечения работы функций, таких как загрузка файлов, получение данных для графиков, сохранение сигналов, переподключение к устройству.