Взаимодействие с бэкенд-сервером мониторинга

- Взаимодействие при помощи Websocket API
  - Подключение к бэкенд-серверу по STOMP через Websocket
  - Отправка запросов и получение ответов
- Взаимодействие при помощи REST API
- Настройка подключения
  - Настройка порта
  - Настройка таймаута Websocket соединения
  - Настройка CORS

Клиент может подключиться к бэкенд-серверу для получения данных и управления мониторингом двумя способами:

- при помощи Websocket API
- при помощи REST API

Взаимодействие при помощи Websocket API

Для получения информации о потоках на WCS в реальном времени, а также для управления мониторингом, используется Websocket API на базе протоколаSTOMP

Подключение к бэкенд-серверу по STOMP через Websocket

Для подключения к бэкенд-серверу необходимо:

1.Установить Secure Websocket соединение по адресу https://hostname:8090/ws, где hostname – имя бэкенд-сервера

2. Установить соединение по протоколу STOMP, отправив сообщение

```
CONNECT
accept-version:1.2
host:hostname

^@
```

3. Подписаться на очередь /alarms для получения тревог, отправив сообщение

```
SUBSCRIBE
id:sub-0
destination:/alarm

^@
```

4.Подписаться на очередь для получения ответов на отправленные запросы /user/service, отправив сообщение

```
SUBSCRIBE
id:sub-1
destination:/user/service

^@
```

Отправка запросов и получение ответов

Запросы отправляются в виде STOMP-сообщений, например

```
SEND
destination:/app/api/metric/list
content-length:100

{"requestId":"eb2c2807-8c2f-4418-aebe-03622404e4bb","realm":"/api/metric/list","payload":{"id":"3"}}^@
```

Здесь:

- destination - URI запроса

- content-length - длина тела запроса

Тело запроса представляет собой JSON объект со следующими параметрами:

- requestId - уникальный идентификатор запроса
- realm - используемый метод
- payload - данные запроса, состав зависит от используемого метода

Ответ на запрос будет получен асинхронно, в очереди /user/service в виде STOMP-сообщения, например

```
MESSAGE
destination:/user/service
content-type:application/json;charset=UTF-8
subscription:sub-1
message-id:3-8
content-length:159

{"requestId":"eb2c2807-8c2f-4418-aebe-03622404e4bb","status":200,"reason":"SUCCESS","payload":[{"id":3,"name":"
Video rate","note":"","enumName":"VIDEO_RATE"}]}^@
```

Здесь:

- destination - URI очереди
- content-type - тип содержимого
- subscription: - идентификатор подписки
- message-id - идентификатор сообщения
- content-length - длина тела ответа

Тело запроса представляет собой JSON объект со следующими параметрами:

- requestId - уникальный идентификатор запроса
- status - стандартный HTTP статус ответа
- reason - текстовое представление статуса
- payload - данные ответа, состав зависит от запроса и статуса ответа

Взаимодействие при помощи REST API

Для получения информации о потоках на WCS, а также для управления мониторингом, может быть использовано REST API

REST-запрос должен быть HTTPS POST запросом в таком виде:

- HTTPS:https://wcsoam.flashphoner.com:8090/api/stream/history

Здесь:

- wcsoam.flashphoner.com - адрес бэкенд-сервера
- 8090 - HTTPS порт
- /api/stream/history- используемый REST-метод

Настройка подключения

Настройка порта

HTTPS порт для обработки REST запросов и WSS подключений настраивается при помощи параметра

```
server.port = 8090
```

Настройка таймаута Websocket соединения

Таймаут соединения настраивается при помощи параметрав файле wcsoam.properties

```
stomp_max_timeout=1000
```

Настройка CORS

Для установки соединения из браузера необходимо разрешить поддержку кроссдоменных запросов при помощи параметра

```
access_control_allow_origin=*
```