**WebSocket сервер**

Приложение служит для обмена сообщениями между клиентами. Клиенты подразделяются на Пользователей и Устройства.

**Структура сообщений**

Для передачи сообщений серверу и клиентам используется JSON.

Обязательным параметром для каждого сообщения является параметр messageId, в этом параметре передается тип команды передаваемой серверу или тип ответа, передаваемый клиенту. Каждая конкретная команда messageId имеет свою определенную структуру. Структура сообщения, содержащее messageId представлена в листинге 1.

{

    "messageId": "Iam"

}

Листинг 1. Структура сообщения содержащее параметр messageId.

**Обработка ошибок**

Сервер производит обработку ошибок по следующим правилам. Если при взаимодействии сервера и клиента произошла ошибка, то в ответ клиенту возвращается ответ в виде JSON сообщения с messageId=”Error” и полем error, в котором описываются причины которые привели к ошибке. Структура сообщения представлена в листинге 2.

{

    "error": "Сообщение об ошибке",

    "messageId": "Error"

}

Листинг 2. Структура сообщения Error.

Если произошла ошибка связанная с некорректной структурой сообщения, неподдерживаемым типом обрабатываемого сообщения, указанного в messageId или неправильным порядком взаимодействия, то клиенту вернутся сообщение описывающее причины приведшие к ошибке. Если произошла ошибка связанная с некорректным поведением сервера, то возвращается сообщение уведомляющее об этом, и содержащее uuid произошедшего исключения, после чего информацию об этой ошибке можно найти в логах приложения.

**Поддерживаемые команды**

Сервер поддерживает определенный набор команд, которые будут перечислены далее. Если указать неподдерживаемый тип команды в messageId и отправить его серверу, то будет возвращена ошибка, уведомляющая клиента об этом.

1. **Команда «Iam»:**

Команда Iam служит для регистрации устройства на сервере. Структура сообщения для пользователя и устройства отличается. Определение типа клиента используется поле id: если регистрируется пользователь - указывается – 1, если регистрируется устройство –указывается 0. Структура сообщения для регистрации пользователя представлена в листинге 3.

{

    "messageId": "Iam",

    "id": 1,

    "panel": ["mac-адрес устройства 1", "mac-адрес устройства 1"]

}

Листинг 3. Структура сообщения Iam для регистрации пользователей.

Для пользователя в поле panel передается список mac адресов, на которые он хочет осуществить подписку и начать получать данные.

Структура сообщения для регистрации устройства представлена в листинге 4.

{

    "messageId": "Iam",

    "id": 0,

    "panel": "mac-адрес устройства"

}

Листинг 4. Структура сообщения Iam для регистрации устройств.

Для устройства в поле panel указывается mac-адрес подключаемого устройства.

!!! Важно отметить, что клиент в пределах одной сессии не может быть зарегистрирован как пользователь и как устройство одновременно.

1. **Команда «Data»:**

Команда Data служит для передачи информации от устройства пользователям. Структура сообщений для передачи от устройства серверу и от сервера пользователю отличается. Все поля, передаваемые от устройства не могут быть null. Структура сообщения, передаваемая устройством представлена в листинге 5. Структура сообщения, передаваемая пользователю представлена в листинге 6.

{

    "messageId": "Data",

    "temp": 0,

    "backlight": 1,

    "volume": 2,

    "workingHours": "12"

}

Листинг 5. Структура сообщения Data, передаваемая устройством.

{

    "messageId": "Data",

    "mac": "mac-адрес устройства, которое отправило данные",

    "temp": 0,

    "backlight": 1,

    "volume": 2,

    "workingHours": "3"

}

Листинг 6. Структура сообщения Data, передаваемая устройством.

**Команды Set**

Все команды set используются для передачи от пользователя к устройству котором указывается mac-адрес устройства, которому необходимо отправить устройству.

1. **Команда «setSource»:**

Структура команды setSource представлена в листинге 7.

{

    "messageId": "setSource",

    "panel": "mac-адрес устройства",

    "source": 0

}

Листинг 7. Структура сообщения setSource.

Поле source не может быть пустым и должно быть более 0.

1. **Команда «setReboot»:**

Структура команды setReboot представлена в листинге 8.

{

    "messageId": "setReboot",

    "panel": "mac-адрес устройства"

}

Листинг 8. Структура сообщения setReboot.

1. **Команда «setShutdown»:**

Структура команды setShutdown представлена в листинге 9.

{

    "messageId": "setShutdown",

    "panel": "mac-адрес устройства"

}

Листинг 9. Структура сообщения setShutdown.

1. **Команда «setSleep»:**

Структура команды setSleep представлена в листинге 10.

{

    "messageId": "setSleep",

    "panel": "mac-адрес устройства"

}

Листинг 10. Структура сообщения setSleep.

1. **Команда «setWakeUp»:**

Структура команды setWakeUp представлена в листинге 11.

{

    "messageId": "setWakeUp",

    "panel": "mac-адрес устройства"

}

Листинг 11. Структура сообщения setWakeUp.

1. **Команда «setTimeAlarm»:**

Структура команды setTimeAlarm представлена в листинге 12.

{

    "messageId": "setTimeAlarm",

    "panel": "mac-адрес устройства",

    "time": "2000:12:01T09:15:02"

}

Листинг 12. Структура сообщения setTimeAlarm.

Поле time указывается как строка в формате YYYY:MM:DDThh:mm:ss.

1. **Команда «setAutoSourceSwitch»:**

Структура команды setAutoSourceSwitch представлена в листинге 13.

{

    "messageId": "setAutoSourceSwitch",

    "panel": "mac-адрес устройства"

}

Листинг 13. Структура сообщения setAutoSourceSwitch.

1. **Команда «setBacklight»:**

Структура команды setBacklight представлена в листинге 14.

{

    "messageId": "setBacklight",

    "panel": "mac-адрес устройства"

}

Листинг 14. Структура сообщения setBacklight.

1. **Команда «setBrightness»:**

Структура команды setBrightness представлена в листинге 15.

{

    "messageId": "setBrightness",

    "panel": "mac-адрес устройства"

}

Листинг 15. Структура сообщения setBrightness.

1. **Команда «setPictureMode»:**

Структура команды setPictureMode представлена в листинге 16.

{

    "messageId": "setPictureMode",

    "panel": "mac-адрес устройства",

    "value": 5

}

Листинг 16. Структура сообщения setPictureMode.

Поле value может принимать значения от 0 до 10.

1. **Команда «setSaturation»:**

Структура команды setSaturation представлена в листинге 17.

{

    "messageId": "setSaturation",

    "panel": "mac-адрес устройства",

    "value": 50

}

Листинг 17. Структура сообщения setSaturation.

Поле value может принимать значения от 0 до 100.

1. **Команда «setSharpness»:**

Структура команды setSharpness представлена в листинге 18.

{

    "messageId": "setSharpness",

    "panel": "mac-адрес устройства",

    "value": 50

}

Листинг 18. Структура сообщения setSharpness.

Поле value может принимать значения от 0 до 100.

1. **Команда «setVolumeUp»:**

Структура команды setVolumeUp представлена в листинге 19.

{

    "messageId": "setVolumeUp",

    "panel": "mac-адрес устройства"

}

Листинг 19. Структура сообщения setVolumeUp.

1. **Команда «setVolumeDown»:**

Структура команды setVolumeDown представлена в листинге 20.

{

    "messageId": "setVolumeDown",

    "panel": "1"

}

Листинг 20. Структура сообщения setVolumeDown.

1. **Команда «setVolume»:**

Структура команды setVolume представлена в листинге 21.

{

    "messageId": "setVolume",

    "panel": "mac-адрес устройства",

    "value": 50

}

Листинг 21. Структура сообщения setVolume.

Поле value может принимать значения от 0 до 100.

1. **Команда «setTurningDevice»: (enayble desayble power)**

Структура команды setShutdown представлена в листинге 21a. Команда выполняет отслеживание включение/выключение и перезагрузки панели

{

"messageId": "setTurningDevice",

"process": "enable",

"dateTime": "2023-03-10T15:31:07.260813"

}

Листинг 21a. Структура сообщения setTurningDevice.

Поле process может принимать значения:

- enable– включение панели;

- desable -выключение панели;

- reset – перезагрузка панели.

**Команды Get и Rpl**

Все команды get используются для передачи от пользователя к серверу, каждое сообщение содержит обязательный параметр «panel», в котором указывается mac-адрес устройства, которому необходимо отправить устройству. Далее команда передается устройству, и к команде добавляется поле userSession, чтобы указать какой пользователь запросил данные. Пользователю в ответ возвращаются данные от устройства командой rpl, устройство в структуре сообщения должно указать обязательный параметр userSession. Пользователю возвращается сообщения, содержащее значение запрошенного параметра, а также mac адрес устройства в поле panel, которое его отправило.

1. **Команды «getSource» и «rplSource»:**

Структура команды getSource, отправляемой на сервер пользователем, представлена в листинге 22.

{

    "messageId": "getSource",

    "panel": "mac-адрес устройства"

}

Листинг 22. Структура команды getSource, отправляемой на сервер пользователем

Структура команды getSource, отправляемой сервером устройству, представлена в листинге 23.

{

    "messageId": "getSource",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 23. Структура команды getSource, отправляемой сервером устройству

Структура сообщения rplSource, отправляемой на сервер устройством, представлена в листинге 24.

{

    "messageId": "rplSource",

    "userSession": "значение сессии пользователя",

    "value": 50

}

Листинг 24. Структура сообщения rplSource, отправляемой на сервер устройством

Значение поля value не может быть меньше 0.

Структура сообщения rplSource, отправляемой от сервера пользователю, представлена в листинге 25.

{

    "messageId": "rplSource",

    "panel": "mac-адрес устройства",

    "value": 50

}

Листинг 25. Структура сообщения rplSource, отправляемой от сервера пользователю

1. **Команды «getSleep» и «rplSleep»:**

Структура команды getSleep, отправляемой на сервер пользователем, представлена в листинге 26.

{

    "messageId": "getSleep",

    "panel": "mac-адрес устройства"

}

Листинг 26. Структура команды getSleep, отправляемой на сервер пользователем

Структура команды getSleep, отправляемой сервером устройству, представлена в листинге 27.

{

    "messageId": "getSleep",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 27. Структура команды getSleep, отправляемой сервером устройству

Структура сообщения rplSleep, отправляемой на сервер устройством, представлена в листинге 28.

{

    "messageId": "rplSleep",

    "userSession": "значение сессии пользователя",

    "value": 1

}

Листинг 28. Структура сообщения rplSleep, отправляемой на сервер устройством

Значение поля value должно быть 1 или 0.

Структура сообщения rplSleep, отправляемой от сервера пользователю, представлена в листинге 29.

{

    "messageId": "rplSleep",

    "panel": "mac-адрес устройства",

    "value": 50

}

Листинг 29. Структура сообщения rplSleep, отправляемой от сервера пользователю

1. **Команды «getTimeAlarm» и «rplTimeAlarm»:**

Структура команды getTimeAlarm, отправляемой на сервер пользователем, представлена в листинге 30.

{

    "messageId": "getTimeAlarm",

    "panel": "mac-адрес устройства"

}

Листинг 30. Структура команды getTimeAlarm, отправляемой на сервер пользователем

Структура команды getTimeAlarm, отправляемой сервером устройству, представлена в листинге 31.

{

    "messageId": "getTimeAlarm",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 31. Структура команды getTimeAlarm, отправляемой сервером устройству

Структура сообщения rplTimeAlarm, отправляемой на сервер устройством, представлена в листинге 32.

{

    "messageId": " rplTimeAlarm ",

    "userSession": "значение сессии пользователя",

    "time": "2000:12:01T09:15:02"

}

Листинг 32. Структура сообщения rplTimeAlarm, отправляемой на сервер устройством

Поле time указывается как строка в формате YYYY:MM:DDThh:mm:ss.

Структура сообщения rplTimeAlarm, отправляемой от сервера пользователю, представлена в листинге 33.

{

    "messageId": "rplTimeAlarm",

    "panel": "mac-адрес устройства",

    "time": "2000:12:01T09:15:02"

}

Листинг 33. Структура сообщения rplTimeAlarm, отправляемой от сервера пользователю

1. **Команды «getAutoSourceSwitch» и «rplAutoSourceSwitch»:**

Структура команды getAutoSourceSwitch, отправляемой на сервер пользователем, представлена в листинге 34.

{

    "messageId": " getAutoSourceSwitch ",

    "panel": "mac-адрес устройства"

}

Листинг 34. Структура команды getAutoSourceSwitch, отправляемой на сервер пользователем

Структура команды getAutoSourceSwitch, отправляемой сервером устройству, представлена в листинге 35.

{

    "messageId": "getAutoSourceSwitch",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 35. Структура команды getAutoSourceSwitch, отправляемой сервером устройству

Структура сообщения rplAutoSourceSwitch, отправляемой на сервер устройством, представлена в листинге 36.

{

    "messageId": "rplAutoSourceSwitch",

    "userSession": "значение сессии пользователя",

    "value": 1

}

Листинг 36. Структура сообщения rplAutoSourceSwitch, отправляемой на сервер устройством

Значение поля value должно быть 1 или 0.

Структура сообщения rplAutoSourceSwitch, отправляемой от сервера пользователю, представлена в листинге 37.

{

    "messageId": "rplAutoSourceSwitch",

    "panel": "mac-адрес устройства",

    "value": 1

}

Листинг 37. Структура сообщения rplAutoSourceSwitch, отправляемой от сервера пользователю

1. **Команды «getBacklight» и «rplBacklight»:**

Структура команды getBacklight, отправляемой на сервер пользователем, представлена в листинге 38.

{

    "messageId": "getBacklight",

    "panel": "mac-адрес устройства"

}

Листинг 38. Структура команды getBacklight, отправляемой на сервер пользователем

Структура команды getBacklight, отправляемой сервером устройству, представлена в листинге 39.

{

    "messageId": "getBacklight",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 39. Структура команды getBacklight, отправляемой сервером устройству

Структура сообщения rplBacklight, отправляемой на сервер устройством, представлена в листинге 40.

{

    "messageId": "rplBacklight",

    "userSession": "значение сессии пользователя",

    "value": 1

}

Листинг 40. Структура сообщения rplBacklight, отправляемой на сервер устройством

Значение поля value должно быть 1 или 0.

Структура сообщения rplBacklight, отправляемой от сервера пользователю, представлена в листинге 41.

{

    "messageId": "rplBacklight",

    "panel": "mac-адрес устройства",

    "value": 1

}

Листинг 41. Структура сообщения rplBacklight, отправляемой от сервера пользователю

1. **Команды «getPictureMode» и «rplPictureMode»:**

Структура команды getPictureMode, отправляемой на сервер пользователем, представлена в листинге 42.

{

    "messageId": "getPictureMode",

    "panel": "mac-адрес устройства"

}

Листинг 42. Структура команды getPictureMode, отправляемой на сервер пользователем

Структура команды getPictureMode, отправляемой сервером устройству, представлена в листинге 43.

{

    "messageId": " getPictureMode ",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 43. Структура команды getPictureMode, отправляемой сервером устройству

Структура сообщения rplPictureMode, отправляемой на сервер устройством, представлена в листинге 44.

{

    "messageId": "rplPictureMode",

    "userSession": "значение сессии пользователя",

    "value": 10

}

Листинг 44. Структура сообщения rplPictureMode, отправляемой на сервер устройством

Значение поля value должно быть 0 или 10.

Структура сообщения rplPictureMode, отправляемой от сервера пользователю, представлена в листинге 45.

{

    "messageId": "rplPictureMode",

    "panel": "mac-адрес устройства",

    "value": 10

}

Листинг 45. Структура сообщения rplPictureMode, отправляемой от сервера пользователю

1. **Команды «getSaturation» и «rplSaturation»:**

Структура команды getSaturation, отправляемой на сервер пользователем, представлена в листинге 46.

{

    "messageId": "getSaturation",

    "panel": "mac-адрес устройства"

}

Листинг 46. Структура команды getSaturation, отправляемой на сервер пользователем

Структура команды getSaturation, отправляемой сервером устройству, представлена в листинге 47.

{

    "messageId": "getSaturation",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 47. Структура команды getSaturation, отправляемой сервером устройству

Структура сообщения rplSaturation, отправляемой на сервер устройством, представлена в листинге 48.

{

    "messageId": "rplSaturation",

    "userSession": "значение сессии пользователя",

    "value": 100

}

Листинг 48. Структура сообщения rplSaturation, отправляемой на сервер устройством

Значение поля value должно быть 0 до 100.

Структура сообщения rplSaturation, отправляемой от сервера пользователю, представлена в листинге 49.

{

    "messageId": "rplSaturation",

    "panel": "mac-адрес устройства",

    "value": 100

}

Листинг 49. Структура сообщения rplSaturation, отправляемой от сервера пользователю

1. **Команды «getSharpness» и «rplSharpness»:**

Структура команды getSharpness, отправляемой на сервер пользователем, представлена в листинге 50.

{

    "messageId": "getSharpness",

    "panel": "mac-адрес устройства"

}

Листинг 50. Структура команды getSharpness, отправляемой на сервер пользователем

Структура команды getSharpness, отправляемой сервером устройству, представлена в листинге 51.

{

    "messageId": " getSharpness ",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 51. Структура команды getSharpness, отправляемой сервером устройству

Структура сообщения rplSharpness, отправляемой на сервер устройством, представлена в листинге 52.

{

    "messageId": "rplSharpness",

    "userSession": "значение сессии пользователя",

    "value": 100

}

Листинг 52. Структура сообщения rplSharpness, отправляемой на сервер устройством

Значение поля value должно быть 0 до 100.

Структура сообщения rplSharpness, отправляемой от сервера пользователю, представлена в листинге 53.

{

    "messageId": "rplSharpness",

    "panel": "mac-адрес устройства",

    "value": 100

}

Листинг 53. Структура сообщения rplSharpness, отправляемой от сервера пользователю

1. **Команды «getVolume» и «rplVolume»:**

Структура команды getVolume, отправляемой на сервер пользователем, представлена в листинге 54.

{

    "messageId": "getVolume",

    "panel": "mac-адрес устройства"

}

Листинг 54. Структура команды getVolume, отправляемой на сервер пользователем

Структура команды getVolume, отправляемой сервером устройству, представлена в листинге 51.

{

    "messageId": " getVolume",

    "panel": "mac-адрес устройства",

"userSession": "значение сессии пользователя"

}

Листинг 55. Структура команды getVolume, отправляемой сервером устройству

Структура сообщения rplVolume, отправляемой на сервер устройством, представлена в листинге 56.

{

    "messageId": "rplVolume",

    "userSession": "значение сессии пользователя",

    "value": 100

}

Листинг 56. Структура сообщения rplVolume, отправляемой на сервер устройством

Значение поля value должно быть 0 до 100.

Структура сообщения rplVolume, отправляемой от сервера пользователю, представлена в листинге 57.

{

    "messageId": "rplVolume",

    "panel": "mac-адрес устройства",

    "value": 100

}

Листинг 57. Структура сообщения rplVolume, отправляемой от сервера пользователю

**Запуск приложения**

Для запуска приложения необходима установленная на хосте Java 17, и созданная база данных PostgreSQL.

Перед запуском необходимо скомпилировать jar файл, после чего перейти в директорию в которой располагается этот jar и выполнить команду, заполнив необходимые переменные окружения:

*java -jar -DPORT=[порт] -DDB\_HOST=[порт и хост подключения к базе данных]   
-DDB\_NAME=[имя базы данных] -DDB\_USERNAME=[имя пользователя для подключения к базе данных] -DDB\_PASSWORD=[пароль пользователя для подключения к базе данных] web-socket-server.jar*

При локальном запуске к серверу можно подключиться по адресу ws://localhost:8080/, при условии что приложение было запущено на порту 8080.