WISETECO

WISETECO IoT

Руководство администратора платформы WISETECO IoT

v3.9

2024-12-27

Содержание

1. Общие сведения . . . 1
1.1. Аннотация . . . 1
1.2. Перечень терминов и сокращений . . . 1
1.3. Введение . . . 2
2. Подготовка к работе . . . 3
3. Конфигурирование Системы . . . 6
4. Tenant и Customer . . . 8
4.1. Tenant менеджер . . . 8
4.2. Customer менеджер . . . 12
5. Потоки обработки данных . . . 19
5.1. Компоненты Потока обработки данных . . . 19
5.2. Системные Потоки обработки данных . . . 20
5.3. Правила обработки (Rules) . . . 21
5.4. Работа с Потоками . . . 22
6. Моделлер конфигурационных элементов (КЭ) . . . 29
6.1. Моделлер КЭ . . . 29
6.2. Браузер КЭ . . . 30
6.3. Триплеты . . . 34
7. Права доступа . . . 36
8. REST интерфейс . . . 49
9. Информационная безопасность . . . 51
10. Дополнительные функции . . . 54
11. Список изменений документа . . . 56

1. Общие сведения

1.1. Аннотация

Данный документ предназначен для *системных администраторов*, участвующих в процессе инсталляции программной платформы **WISETECO** (далее – Система, Платформа).

> В данном документе приведено описание действий по настройке основных параметров и функций Платформы.

Действие данного документа распространяется только на веб-интерфейс **Платформы WISETECO**.

1.2. Перечень терминов и сокращений

Перечень терминов

Вещь

Логическая сущность различного типа, используемая в Системе для описания оборудования, систем, датчиков, сенсоров.

Зона

Логическая сущность различного типа, используемая в Системе для описания местоположения оборудования, систем, датчиков, сенсоров внутри объекта (например, здания, сооружения и т.п.). В том числе используется при создании в выбранной зоне комплексных показателей, позволяющих отслеживать состояние зоны в целом на основе включенных в ее состав вещей.

Объект (объект контроля)

Объект, подлежащий постановке на мониторинг и/или управление при помощи платформы **WISETECO**.

Тенант (Tenant)

Логический компонент, предназначенный для разделения данных внутри общего пространства хранения данных.

Кастомер (Customer)

Логический компонент, предназначенный для логического разделения объектов в рамках одного Тенанта.

Поток (поток обработки данных)

Универсальный инструментарий Платформы, ориентированный на обработку входящих данных.

КП (комплексный показатель)

Функция автоматизации процесса наблюдения за различными показателями внутри потока данных. Комплексные показатели формируются в соответствующем потоке на основе заданных алгоритмов.

Перечень сокращений

Система, Платформа	Платформа интернета вещей и больших данных **WISETECO IoT**.
GUI	Graphical user interface – графический интерфейс пользователя.

1.3. Введение

Наименование Системы

Полное наименование Системы: **Платформа интернета вещей и больших данных WISETECO IoT**.

Краткое наименование Системы: **Платформа WISETECO**.

Сокращённое наименование системы в тексте документа: *Система*, *Платформа*.

Область применения

Платформа **WISETECO** обеспечивает сбор, обработку, визуализацию, анализ данных, получаемых от объектов мониторинга, на которых располагается *IoT* и *ИТ-оборудование*, а также из других источников, включая веб-сервисы, базы данных, файлы с данными и прочее.

Подключение и сбор данных с источников выполняется с использованием набора *программируемых адаптеров*, обеспечивающих унификацию в представлении данных различными информационными ИТ-системами.

> **Платформа** предназначена для разработки на её базе различных отраслевых решений (приложений) и может использоваться для организации и управления отраслевыми экосистемами.

Также на **Платформе** возможно построение прикладных решений, использующих функций платформы в контексте реализации механизмов единого доступа к данным множественных разнородных информационных источников и реализации сопутствующих сервисов.

2. Подготовка к работе

Запуск и авторизация в Системе

Пользовательский интерфейс Системы построен по технологии WEB-клиента. Для работы с Системой Пользователю необходим персональный компьютер с установленным интернет-браузером.

Для запуска WEB-интерфейса в адресной строке браузера следует указать:

```
https://`<hostname>:port'/web/  ①
```

① <hostname>:port» – имя (IP адрес) сервера и порт с установленным приложением.

Откроется экран авторизации (см. рисунок: Окно авторизации).

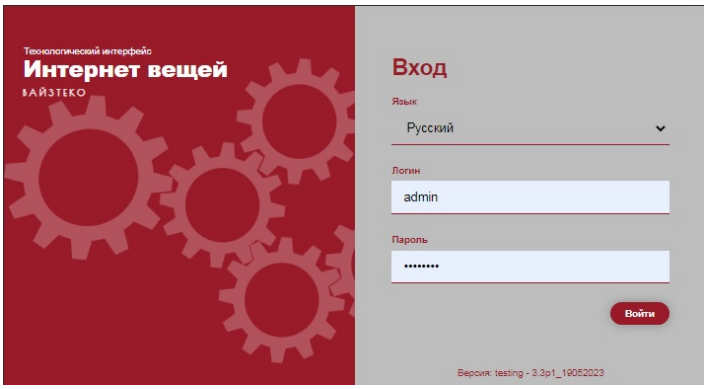

Рисунок 1. Окно авторизации

Для авторизации следует указать Логин и Пароль и нажать на кнопку **[Войти]**.

После завершения процедуры авторизации откроется основной интерфейс платформы (см. рисунок: Главное окно с пиктограммами).

Рисунок 2. Главное окно с пиктограммами

Панель навигации может быть расширена с помощью кнопки в левом нижнем углу так, чтобы основные ссылки были доступны в текстовом виде (см. рисунок: Главное окно с текстовыми ссылками).

Рисунок 3. Главное окно с текстовыми ссылками

Проверка работоспособности интерфейса Системы

Интерфейс Системы *работоспособен*, если в результате действий при авторизации, на экране монитора отобразится основной интерфейс Системы *без выдачи* сообщений о сбое в работе.

Поддержка национальных языков

В настоящее время **Интерфейс Системы** реализован на русском и английском языках.

Выбор языка интерфейса осуществляется в окне логина (см. рисунок: Окно авторизации).

3. Конфигурирование Системы

Настройка конфигурации

Для настройки конфигурации Системы следует перейти в панели навигации в раздел **«Настройки»**, далее **«Конфигурация»**. Будет отображено окно выбора конфигурации (см. рисунок: Окно выбора конфигурации).

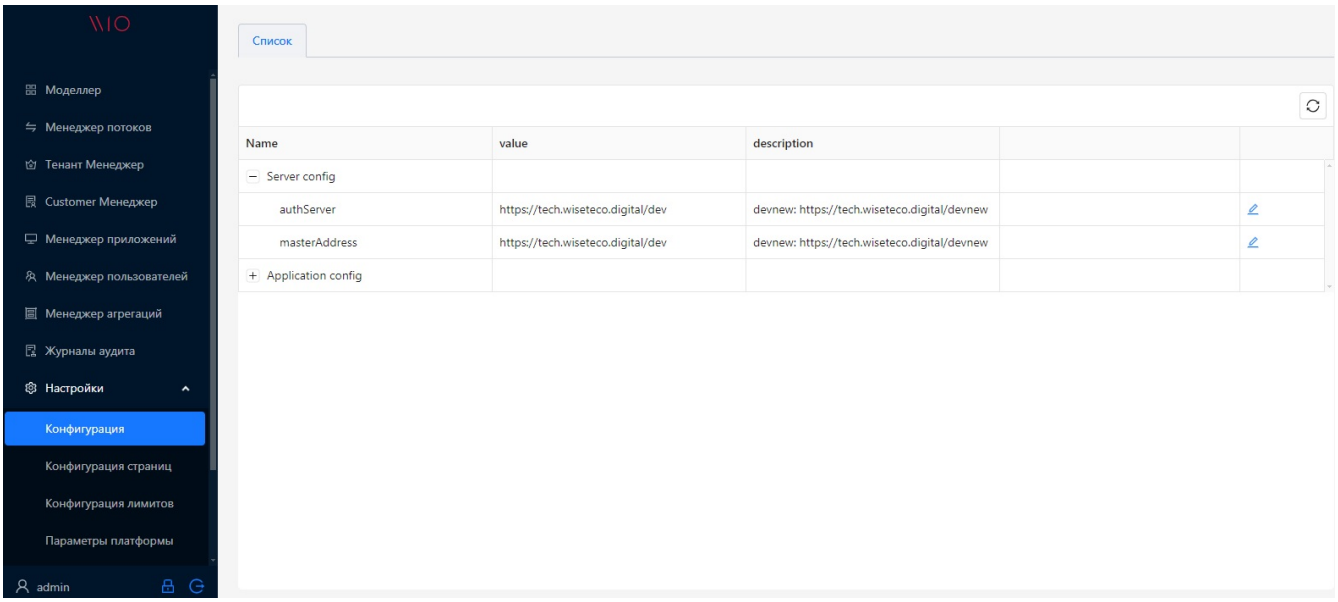

Рисунок 4. Окно выбора конфигурации

Чтобы выполнить настройки URL доступа к серверу запросов необходимо раскрыть элемент "Server config", однократным кликом мыши выбрать нужный объект, и далее нажать на кнопку **[Редактирование]**, после чего откроется окно редактирования объекта конфигурации.

Далее следует задать целевой URL в поле **«Значение»**, ввести при необходимости описание и нажать **[Сохранить]** (см. рисунок: Окно редактирования объекта конфигурации).

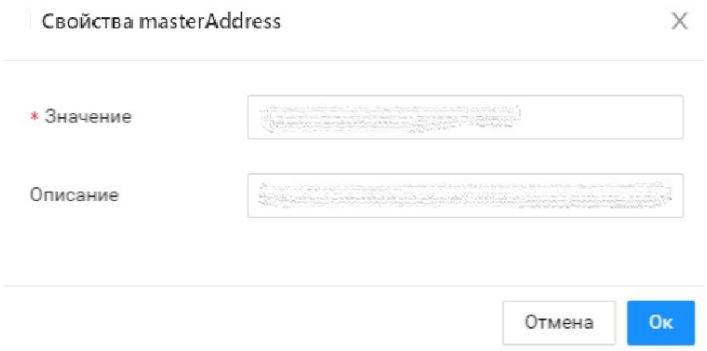

Рисунок 5. Окно редактирования объекта конфигурации

Настройка *конфигурации* является *необходимой настройкой* Системы, которая выполняется Администратором в рамках *установки* Системы. Для получения дополнительной информации по установке и конфигурации Системы необходимо ознакомиться с документом Инструкция по установке.

Менеджер серверов

Функционал позволяет отобразить компоненты, которые подключены к Системе, а также выполнить их регистрацию.

Для просмотра списка компонентов следует перейти в панели навигации в раздел **«Настройки»**, далее **«Менеджер серверов»**. Будет отображено окно со списком компонентов (см. рисунок Список доступных компонентов).

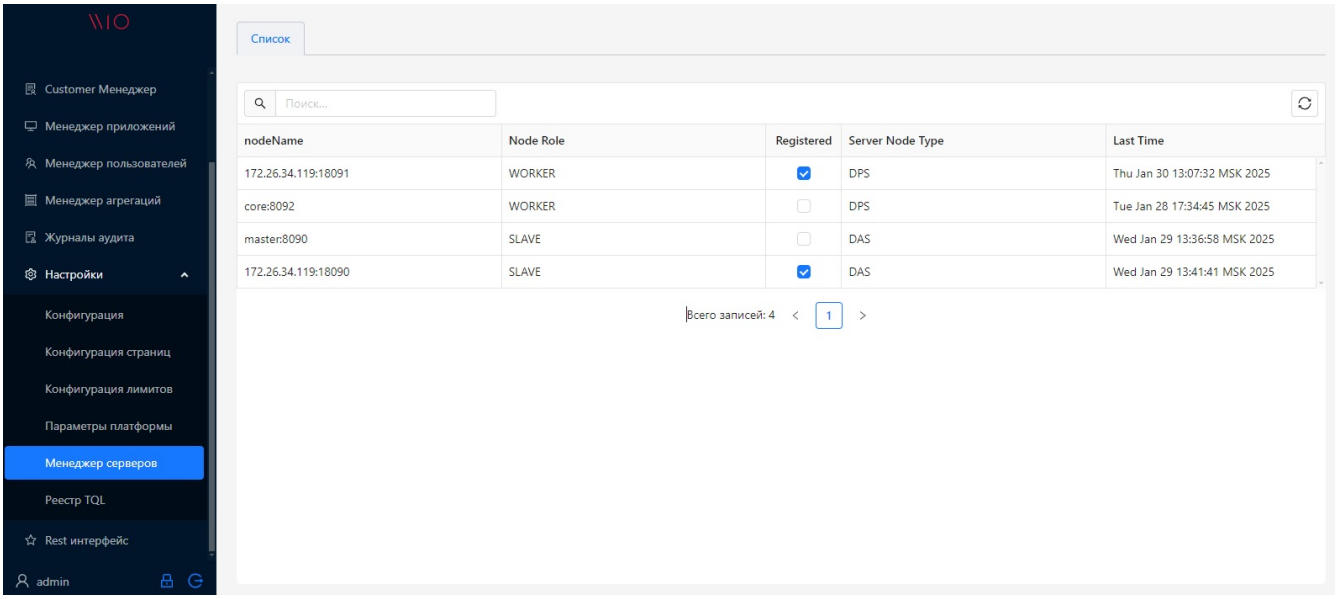

Рисунок 6. Список доступных компонентов

Компоненты с ролью SLAVE (Node Role) являются системными и регистрируются автоматически. Для регистрации компонентов с другими ролями необходимо нажать на кнопку **[Регистрация]**.

4. Tenant и Customer

Платформа поддерживает **мультитенантную** архитектуру, обеспечивая возможность *изолированного* обслуживания Клиентов и их объектов.

4.1. Tenant менеджер

Tenant служит для логического разделения *данных Клиентов*.

Tenant необходимо создавать и конфигурировать в том случае, когда к платформе подключается новый Клиент.

Необходимыми условиями создания Tenant в технологическом интерфейсе являются:

- *Установленное Ядро* – компонент платформы, который будет обслуживать данный Tenant (установка описана в Инструкции по установке);
- *Созданное рабочее пространство клиента*, т.е. схема БД.

При установке Платформы происходит автоматическое создание двух Tenant: `MASTER` и `DEFAULT`.

Tenant `MASTER` является системным и необходим для технологических операций. Удаление системного Tenant приведет к неработоспособности функций Платформы.

Tenant `DEFAULT` используется для подключения Клиента к Платформе. Для этого необходимо его сконфигурировать. Конфигурирование описано в Конфигурирование Tenant.

Просмотр Tenant

Для получения списка Tenant необходимо в основном меню выбрать пункт **«Tenant Менеджер»**, после чего будет открыто окно Tenant Manager (см. рисунок: Окно Tenant Manager):

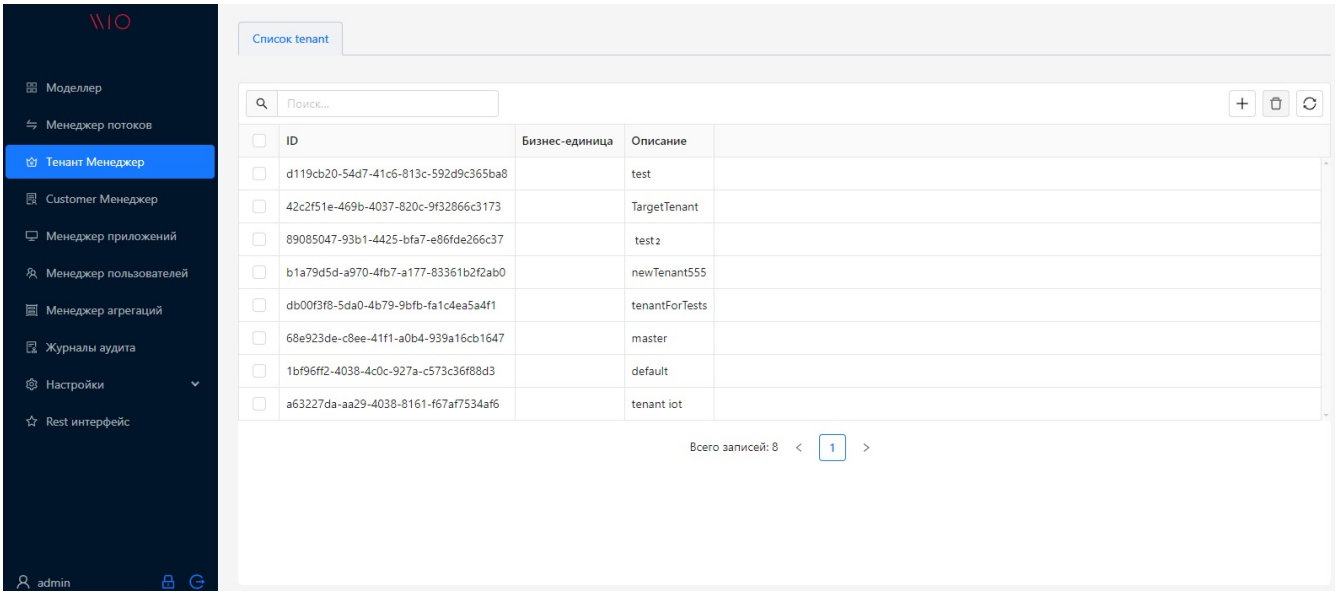

Рисунок 7. Окно Tenant Manager

Для редактирования Tenant следует выбрать запись и нажать дважды левой клавишей мыши, после чего откроется форма (см. рисунок: Форма создания Tenant).

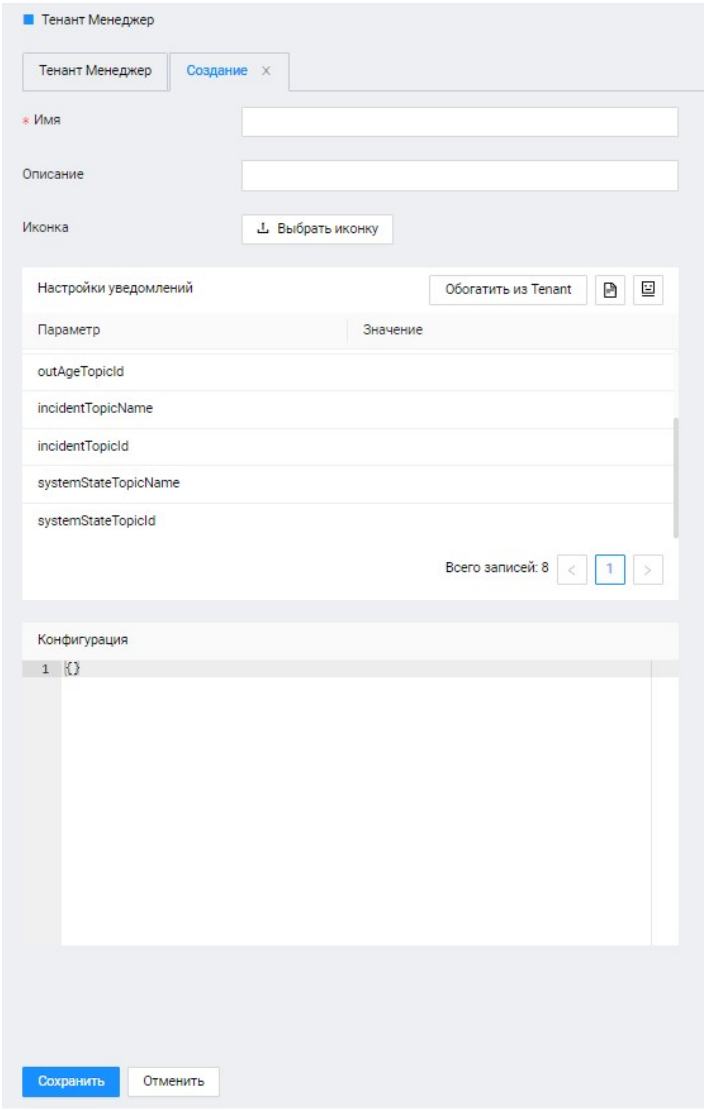

Рисунок 8. Форма создания Tenant

В форме Tenant возможно скорректировать следующие поля:

- **«Имя»** – уникальное имя Tenant;
- **«Описание»** – необязательное поле комментария;
- **«Иконка»** – задать иконку путем выбора графического файла;
- **«Конфигурация»** – задать конфигурацию в формате JSON;

После завершения заполнения всех полей нажать кнопку **[Сохранить]**.

Конфигурирование Tenant

В списке Tenant выбрать запись и нажать дважды левой клавишей мыши. Откроется окно с расширенными параметрами Tenant (см. рисунок: Расширенные параметры Tenant)

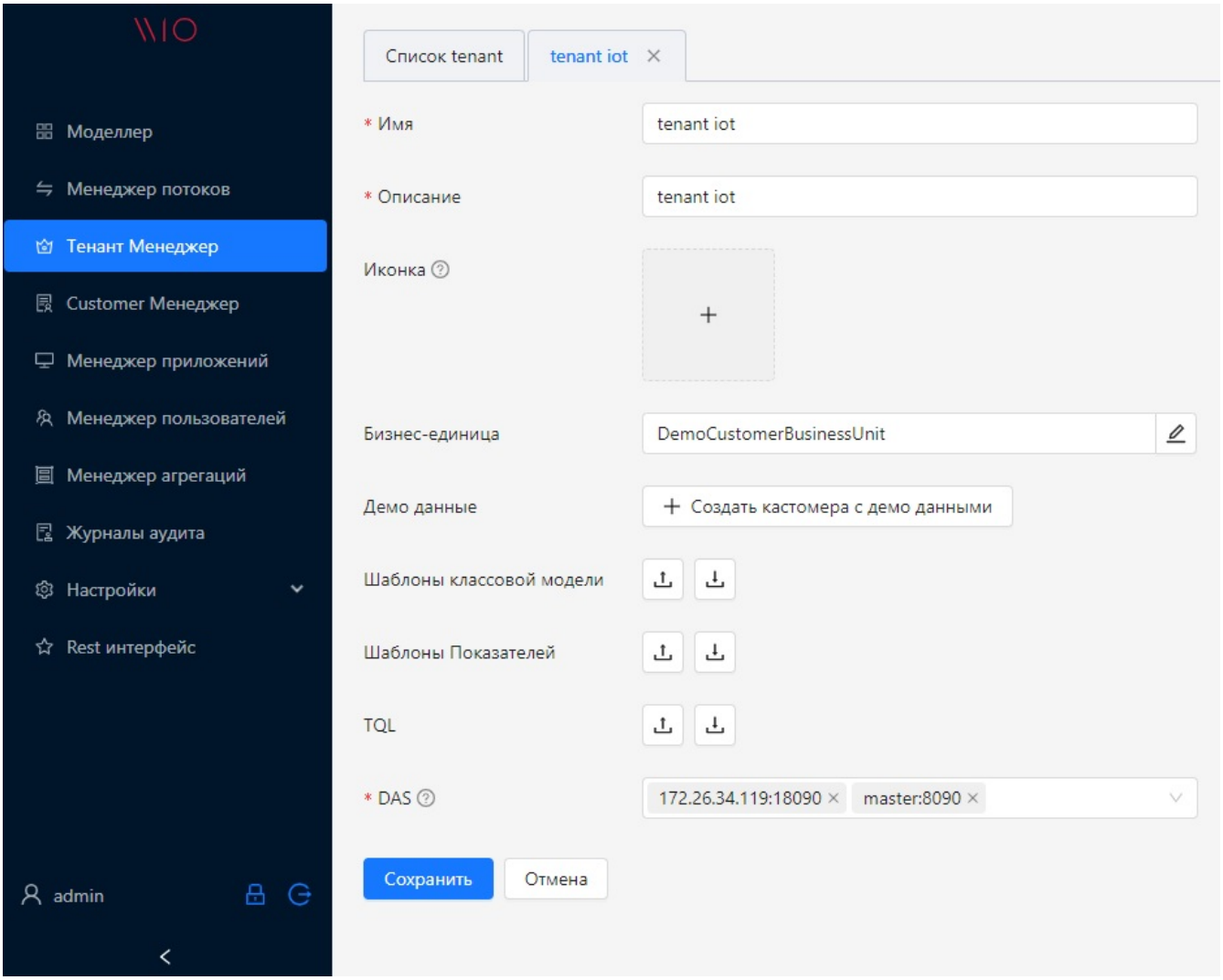

Рисунок 9. Расширенные параметры Tenant

Далее необходимо заполнить атрибуты, описание которых представлено ниже:

- **«Имя»** - Идентификация Tenant. В атрибуте ввести наименование Tenant.
- **«Описание»** - Идентификация Tenant. В атрибуте ввести описание Tenant.
- **«Иконка»** - Логотип КЛИЕНТА. Выбирается файл с логотипом Клиента.
- **«Бизнес-единица** (Наименование) - Создание Клиента или Заказчика. Это наименование будет отображаться в ЛК - Клиента. Ввести наименование, нажать на

кнопку **[Создать]** и заполнить атрибуты в появившемся окне.

- **«Демо данные** - кнопка **[Создать Кастомера с демо данными]**
- **«Шаблоны классовой модели»** - Загрузка и выгрузка классовой модели в формате JSON файла. Выбрать файл для загрузки классовой модели или ее выгрузки.
- **«Шаблоны Показателей»** - Загрузка и выгрузка шаблонов показателей в формате JSON файла. Выбрать файл для загрузки шаблонов или выгрузки шаблонов.
- **«TQL»** - Загрузка и выгрузка шаблонов TQL в формате JSON файла. Выбрать файл для загрузки шаблонов или выгрузки шаблонов.
- **«DAS»** - IP адреса\порты к серверам доступа к данным. Указывается IP адрес\порт (выбор из доступных серверов).
- **«DPS»** - IP адреса\порты к серверам обработки данных. Указывается IP адрес\порт (выбор из доступных серверов).
- **«Настройка уведомлений»** - Группа атрибутов для конфигурирования уведомлений в мессенджер Telegram. Значения параметров, которые нужны для конфигурирования уведомлений, формируются при помощи бота Telegram. Для перехода к боту Telegram необходимо нажать на кнопку **[Бот]**. Кнопка **[Конфигурация]** предназначена для просмотра конфигурации сервиса уведомлений.
  - **«botName»** - Название бота. Вписать параметр.
  - **«botToken»** - Токен бота. Вписать параметр.
  - **«incidentTopicName»** - Наименование группы для Инцидентов. Вписать параметр.
  - **«incidentTopicId»** - ID группы для Инцидентов. Вписать параметр.
  - **«outAgeTopicName»** - Наименование группы для Аномалий. Вписать параметр.
  - **«outAgeTopicId»** - ID группы для Аномалий. Вписать параметр.
  - **«systemStateTopicName»** - Наименование группы для событий мониторинга данных. Вписать параметр.
  - **«systemStateTopicId»** - ID группы для событий мониторинга данных. Вписать параметр.
- **«Конфигурация»** - Конфигурация формируется автоматически при заполнении атрибутов. Коррекция вручную данного атрибута запрещена.

При установке Платформы есть возможность создать *демонстрационные данные*, которые помогут разобраться с функциями Системы (Платформы и Приложения).

Если данная возможность не была использована при установке, а необходимость в демонстрационных данных есть, то запустить процесс установки демонстрационных данных можно из данного интерфейса.

В карточке Tenant `DEFAULT` будет доступна кнопка **[Создать Кастомера с демо данными]**, если атрибут **«Бизнес-единица** пустой.

После нажатия на кнопку откроется форма с параметрами, которые

необходимо заполнить.

Более детальное описание параметров описано в документе Инструкция по установке.

После загрузки демонстрационных данных *будет создан* весь набор необходимых для работы Системы *сущностей* (Бизнес-единица, Базовая классовая модель, TQL, Настройки страниц Customer, Объект, Потоки, Показатели, Инфопанели).

4.2. Customer менеджер

Customer служит для логического разделения *данных* между *объектами* Клиента.

Customer необходимо *создавать и конфигурировать* в том случае, когда к платформе *подключается новый Объект* Клиента.

Необходимым условием создания Customer является, созданное рабочее пространство клиента – Tenant (описано в Tenant менеджер).

Создание Customer

Для получения списка Customer нужно в основном меню выбрать пункт **«Customer Менеджер»**, после чего будет открыто окно Customer Manager, далее необходимо выбрать Tenant в поле «Tenant», для которого необходимо отобразить всех его Customer (см. рисунок: Список Customer):

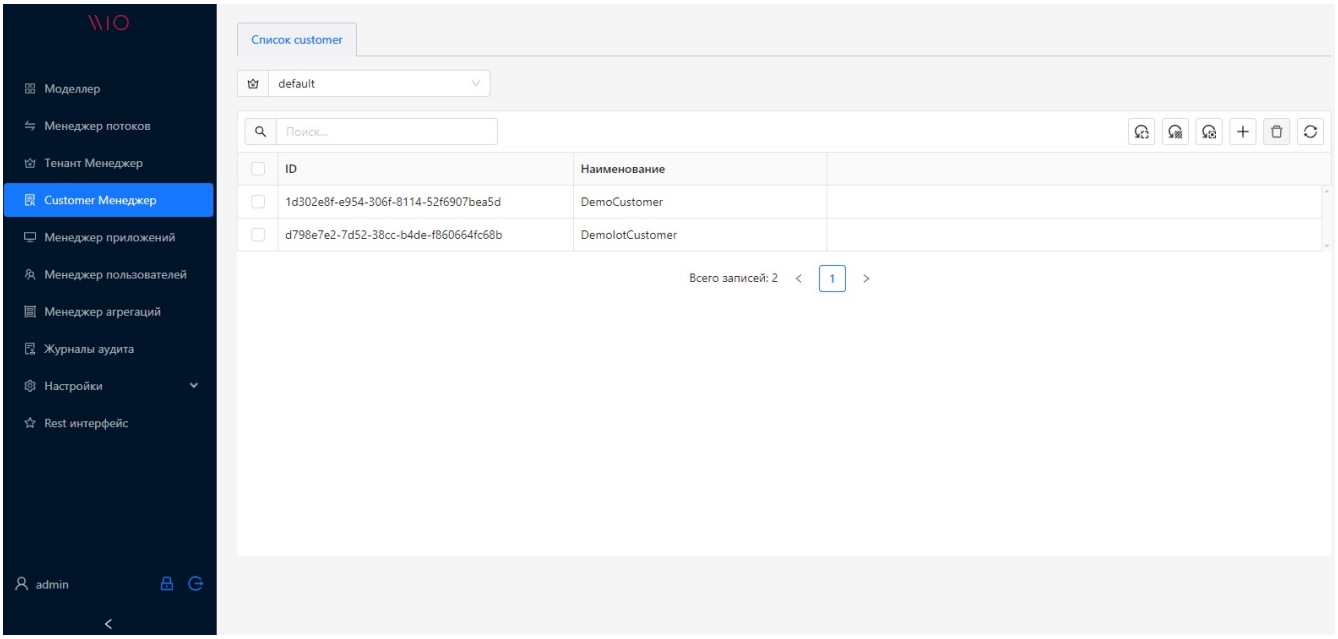

Рисунок 10. Список Customer

Для создания нового Customer следует нажать кнопку **[Создать]**, после чего откроется форма «Создание» (см. рисунок: Окно Customer).

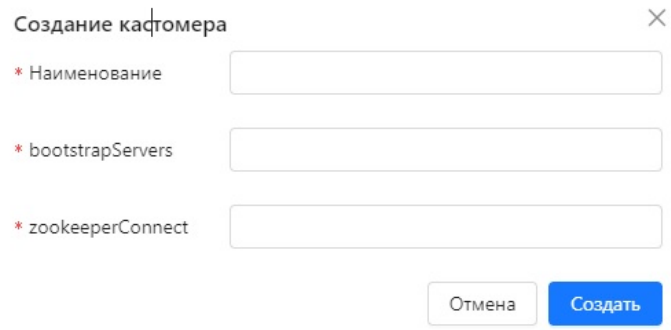

Рисунок 11. Окно Customer

Далее в модальном диалоговом окне следует ввести значения полей:

- **«Наименование»** - Идентификация Customer. В атрибуте ввести наименование Customer.
- **«bootstrapServers»** - IP-адрес\порт доступа к серверу kafka; при наличии кластера kafka IP-адреса\порты всех узлов перечисляются через запятую.
- **«zookeeperConnect»** - IP-адрес\порт доступа к серверу zookeeper.

После этого требуется нажать кнопку **[Создать]**.

Для удаления Customer в списке (см. рисунок: Окно Customer) следует нажать кнопку **[Удалить]**.

Удаление Customer приведет к потере данных соответствующего Объекта, с которым он ассоциирован.

Конфигурирование Customer

В списке Customer выбрать запись и нажать дважды левой клавишей мыши. Откроется окно с расширенными параметрами Customer (см. рисунок: Расширенные параметры Customer):

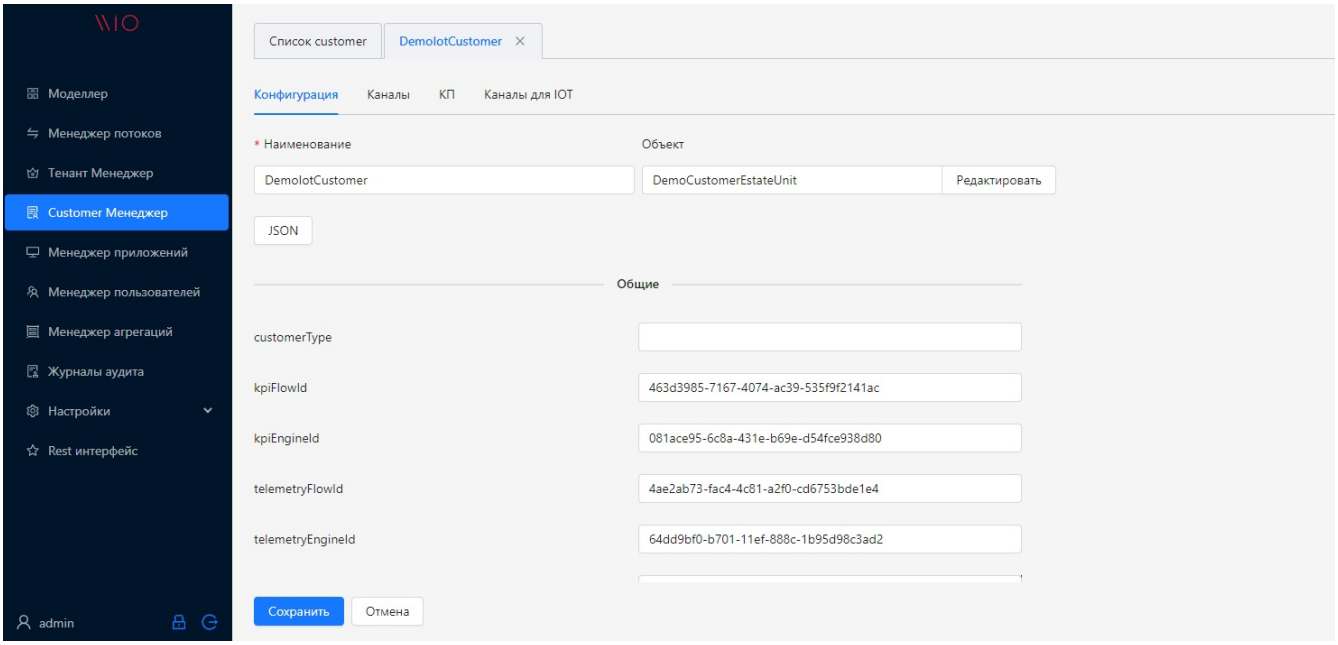

Рисунок 12. Расширенные параметры Customer

Далее необходимо заполнить наборы атрибутов на вкладках:

- **«Конфигурация»** (см.: Атрибуты на вкладке «Конфигурация»);
- **«КПы»** (см. Вкладка «КП Engine»);
- **«Каналы для IOT»** (см. Вкладка «Каналы для IOT»).

Атрибуты на вкладке «Конфигурация»

- **«Наименование»** - Идентификация Customer. В атрибуте ввести наименование Customer.
- **«Объект»** (Наименование). Создание Объекта Заказчика. Это наименование будет отображаться в ЛК – Паспорт объекта. Ввести наименование, нажать на кнопку «Создать» и заполнить атрибуты в появившемся окне.
- **«Атрибуты для создания постоянных параметров конфигурации объекта»**. Группа атрибутов, отвечающая за формирование постоянных параметров конфигурации. Часть параметров доступны только после установки компонентов платформы, которые обслуживают данный объект Заказчика и системы в целом.
  - **«kafkaBrokerUri»** - IP-адрес\порт доступа к серверу kafka; при наличии кластера kafka IP-адреса\порты всех узлов перечисляются через запятую.
  - **«bootstrapServers»** - IP-адрес\порт доступа к серверу kafka; при наличии кластера kafka IP-адреса\порты всех узлов перечисляются через запятую.
  - **«zookeeperConnect»** - IP-адрес\порт доступа к серверу zookeeper.
  - **«metadataBrokerList»** - IP-адрес\порт доступа к серверу kafka; при наличии кластера kafka IP-адреса\порты всех узлов перечисляются через запятую.
  - **«systemTopic»** - Идентификация системного топика системной шины для системных сообщений. Указывается наименование топика для данного объекта.
  - **«topologyTopic»** - Идентификация топика системной шины для формирования

сообщений физической топологии. Указывается наименование топика для данного объекта.

- **«telemetryTopic»** - Идентификация топика системной шины для передачи сообщений телеметрии. Указывается наименование топика для данного объекта.
- **«actionTopic»** - Идентификация топика системной шины для передачи сообщений управляющего воздействия. Указывается наименование топика для данного объекта.
- **«scenarioTopic»** - Идентификация топика системной шины для передачи сценариев управления и их конфигураций. Указывается наименование топика для данного объекта.
- **«dmmAlarmsTopic»** - Идентификация топика системной шины для передачи исключительных событий. Указывается наименование топика системной шины для передачи исключительных событий (формируется автоматически при успешной регистрации Агента в Платформе).
- **«mqttAddress»** - Адрес проксиброкера для получения информации из топиков контроллера. Указывается адрес с учетом ShortSerial и префикса контроллера.
- **«controllerToBrokerConnectionStateMqttAddress»** - Топик для определения состояния подключения контроллера к проксиброкеру. Указывается топик с учетом ShortSerial контроллера.
- **«kpiFlowId»** - ID потока (*модуль расчета показателей*). Указывается ID (формируется автоматически при создании соответствующего потока обработки).
- **«kpiEngineId»** - ID компонента потока (*модуль расчета показателей*). Указывается ID (формируется автоматически при создании соответствующего потока обработки).
- **«telemetryFlowId»** - ID потока (*модуль обработки телеметрии*). Указывается ID (формируется автоматически при создании соответствующего потока обработки).
- **«telemetryEngineId»** - ID компонента потока (*модуль обработки телеметрии*). Указывается ID (формируется автоматически при создании соответствующего потока обработки).
- **«topologyFlowId»** - ID потока (*модуль обработки топологии*). Указывается ID (формируется автоматически при создании соответствующего потока обработки).
- **«topologyEngineId»** - ID компонента потока (*модуль обработки топологии*). Указывается ID (формируется автоматически при создании соответствующего потока обработки).
- **«notifyFlowId»** - ID потока (*модуль обработки уведомлений*). Указывается ID (формируется автоматически).
- **«notifyEngineId»** - ID компонента потока (*модуль обработки уведомлений*). Указывается ID (формируется автоматически).
- **«connectorId»** - ID компонента потоков. Указывается ID (формируется автоматически)
- **«estateUnitId»** - ID Объекта (конфигурационного элемента логической модели верхнего уровня). Указывается ID (формируется автоматически при создании конфигурационного элемента).
- **«DAS»** - IP адреса\порты к серверам доступа к данным. Указывается IP адрес (выбор из

доступных серверов).

- **«DPS»** - IP адреса\порты к серверам обработки данных. Указывается IP адрес (выбор из доступных серверов).

- **«Настройка уведомлений»** - Группа атрибутов для конфигурирования уведомлений в мессенджер Telegram. Значения параметров, которые нужны для конфигурирования уведомлений.
  - **«incidentTopicName»** - Наименование группы мессенджера для Инцидентов. Вписать параметр (параметр определяется правилами формирования групп мессенджера вне работы с системой).
  - **«incidentTopicId»** - ID группы мессенджера для Инцидентов. Вписать параметр (параметр определяется правилами формирования групп мессенджера вне работы с системой).
  - **«anomalyTopicName»** Наименование группы мессенджера для Аномалий. Вписать параметр (параметр определяется правилами формирования групп мессенджера вне работы с системой).
  - **«anomalyTopicId»** - ID группы мессенджера для Аномалий. Вписать параметр (параметр определяется правилами формирования групп мессенджера вне работы с системой).
  - **«outAgeTopicName»** - Наименование группы мессенджера для мониторинга расчетов комплексных показателей (мониторинг данных). Вписать параметр (параметр определяется правилами формирования групп мессенджера вне работы с системой).
  - **«outAgeTopicId»** - ID группы мессенджера для мониторинга расчетов комплексных показателей (мониторинг данных). Вписать параметр (параметр определяется правилами формирования групп мессенджера вне работы с системой).
  - **«systemStateTopicName»** - Наименование группы мессенджера для системных событий. Вписать параметр (параметр определяется правилами формирования групп мессенджера вне работы с системой).
  - **«systemStateTopicId»** - ID группы мессенджера для системных событий. Вписать параметр (параметр определяется правилами формирования групп мессенджера вне работы с системой).
- **«Конфигурация»** - Конфигурация формируется автоматически. Она формируется автоматически при заполнении атрибутов. Коррекция вручную данного атрибута запрещена.

При конфигурировании уведомлений в мессенджер Telegram для получения необходимых параметров следует обратиться к методам их получения из руководства к Telegram.

Вкладка «КП Engine»

Отображается информация о текущем состоянии каждого созданного экземпляра комплексного показателя (см. рисунок КП Engine).

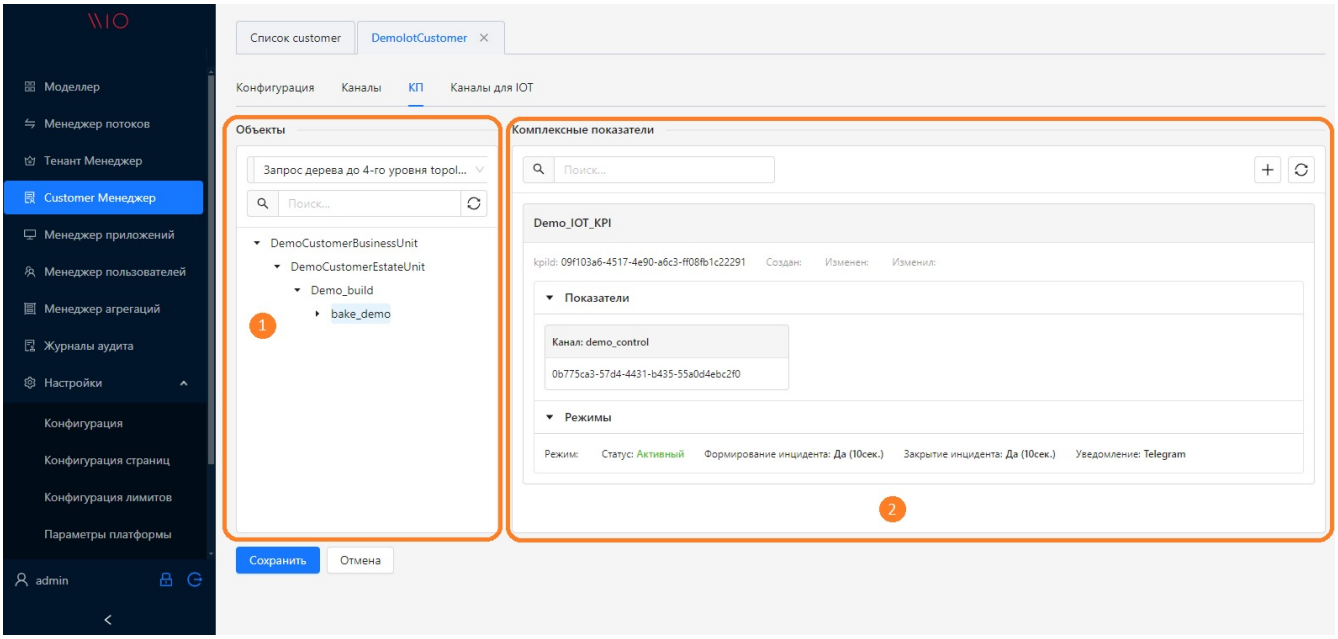

Рисунок 13. КП Engine

① Отображаются логические элементы, с которыми могут быть связаны комплексные показатели:

② Отображаются комплексные показатели, которые связаны с выбранным логическим элементом, и их состояния в механизме расчета комплексных показателей.

Вкладка «Каналы для IOT»

Вкладка предназначена для отображения «физической топологии» и ее параметризации (см. рисунок: Вкладка Каналы).

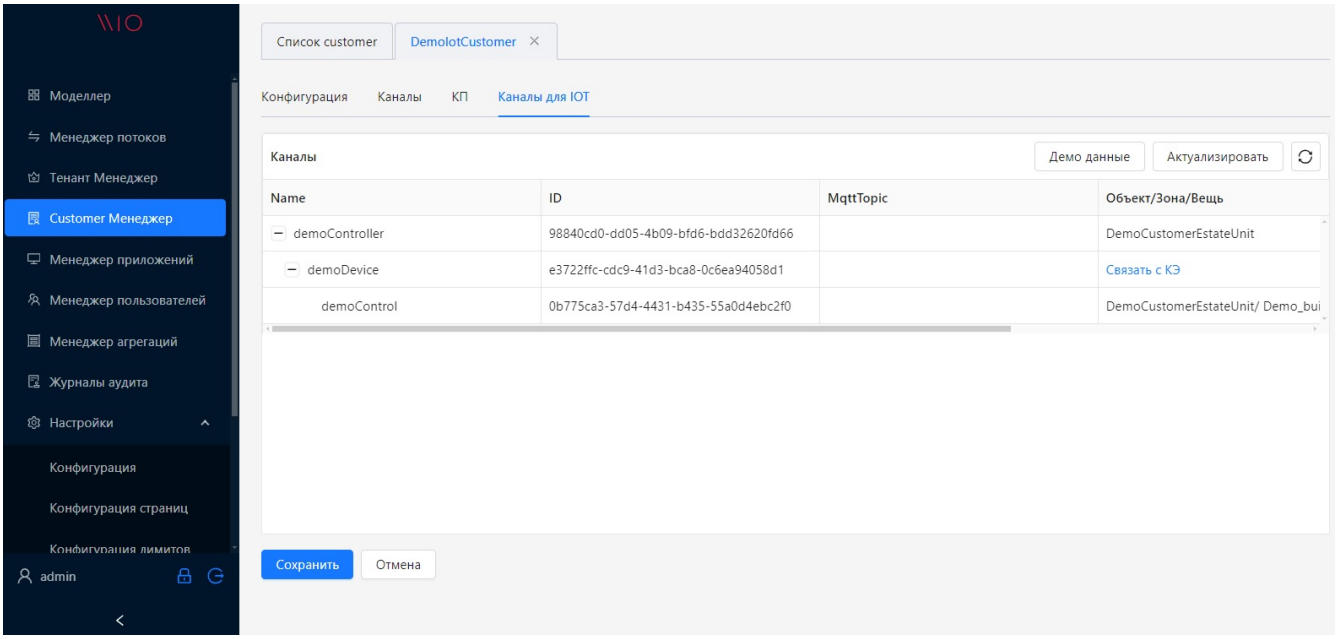

Рисунок 14. Вкладка Каналы

«Физическая топология» - это древовидная структура в виде `Контроллер\Устройство\Канал`, т.е. совокупность всех устройств на данном Объекте.

«Логическая топология» - это древовидная структура в виде `Объект\Зона\Вещь`, т.е.

совокупность всех сущностей объекта в контексте мониторинга.

Форма содержит следующие столбцы:

- Столбец **Name** - наименование элемента в древовидной структуре (Контроллер\Устройство\Канал).
- Столбец **ID** - уникальный идентификатор канала.
- Столбец **System** - используется для определения каналов, которые используются для системного мониторинга устройств (в частности контроллера). Телеметрия данных каналов записывается в БД, но в агрегированном состоянии.
- Столбец **Monitoring** - используется для определения Каналов, телеметрия которых будет записываться в БД. Набор таких каналов образует *«Белый список»*. «Белый список» формируется на основании привязок «физической» и «логической» топологии.
- Столбец **RawData** - используется для определения Каналов, телеметрия которых будет записываться в БД как есть (*«сырец»*).

Доступны следующие действия:

- **Актуализировать** - Для активации внесенных изменений.

Кнопка **[Демо данные]** используется для формирования демонстрационной топологии и запуска генератора по телеметрии для соответствующей топологии. Детали описаны в Инструкции по установке.

5. Потоки обработки данных

Характерной особенностью Платформы является функционал *Потоков обработки данных (Потоков)*.

Поток - это сущность, которая обеспечивает совокупность взаимосвязанных функций, обеспечивающих подключение к источнику данных, обработку и сохранение данных, а также управление этими процессами.

Таким образом, Платформа реализует универсальный функционал работы с разнородными источниками данных и адаптацию самой Платформы и построенных на ее основе приложений под решение разного круга задач.

> Поток обработки данных можно представить в виде конвейера, по компонентам которого происходит движение данных.

Элементы выстраиваются в условную цепочку обработки и каждый элемент цепочки выполняет заданную роль.

Механизм Потоков, как и архитектура платформы, опирается на ***акторную модель***.

Поток формируется из набора типизированных элементов, каждый из которых может быть изменен в контексте решаемой задачи. Описание компонентов, их функционал, особенности применения представлено в Компоненты Потока обработки.

5.1. Компоненты Потока обработки данных

В общем случае, каждый Поток характеризуется *стандартными компонентами* и *набором правил обработки (RULES)*.

Каждый поток создает объекты `CONNECTOR` и `SOURCE`.

Начальным элементов потока является `CONNECTOR`. Именно он определяет *источник подключения и способы подключения* к источнику данных.

Для управления и координации компонентов Потока используется другой компонент Потока - `SOURCE`.

Source можно рассматривать как *менеджер*, который управляет процессами получения данных через CONNECTOR, запускает необходимые для функционирования Потока процессы, контролирует прохождение данных через поток и передачу данных на конце Потока в еще один компонент Потока – `PLUGIN`.

`PLUGIN` это – специализированный компонент потоковой обработки данных, который определяет размещение (запись) обработанных данных. PLUGIN это компонент, которым заканчивается Поток.

> Поток обработки может заканчиваться сохранением данных,

> например, в базу данных. Также данные могут быть переданы в другой сервис обработки или Поток обработки. При этом Поток не обязательно заканчивается одним PLUGIN. Данные могут быть направлены в различные PLUGIN-ы.

Непосредственно процедуры обработки данных в потоке формируются из последовательности т.н. RULES. Rules бывают нескольких типов. Типы Rules определяют правила их функционирования в Потоке.

5.2. Системные Потоки обработки данных

Администратор Платформы может самостоятельно создавать и модифицировать Потоки в контексте решаемых задач. Кроме пользовательских Потоков, существуют ***Системные потоки*** обработки данных.

В текущей реализации представлены следующие Системные потоки:

- `telemetry` - поток для получения и сохранения поступающей телеметрии.
- `KPI` - поток, обеспечивающий расчет Комплексных показателей (КП, см. ниже) и результатов этих расчетов на основе поступающей телеметрии.
- `topology` - поток, который обеспечивает прием топологической информации и сохранение в базе данных конфигурационных элементов.

Системные потоки формируются в Платформе автоматически и не должны подвергаться изменениям со стороны оператора Платформы.

В контексте описания Системных потоков необходимо сказать несколько слов о функционале `Комплексных показателей (КП)`.

> КП может использоваться в платформе как функция автоматизации процесса наблюдения за ключевыми показателями. Комплексные показатели формируются в соответствующем потоке.

КП «наблюдает» за одним или несколькими единицами `телеметрической информации` (каналами) или за дочерними КП.

В качестве критерия расчета состояния КП, используются `Пороги` (пороговые значения), которые устанавливаются Пользователями Приложения на основе бизнес-требований.

Определяется набор каналов, которые будут служить базой для расчета КП и различные функции, которые определяют каким именно образом надо рассчитывать показатели. При расчете КП используется системный компонент ENGINE. Конфигурированию и модификации этот компонент не подлежит и является частью Системного потока (КП).

После старта Системного потока с этим компонентом запускается Актор `ENGINE`. Он получает на вход список КП, который есть в данном Customer. На одного Customer, достаточно одного

потока, в котором содержится один ENGINE (т.е. если запустить два ENGINE, то он получит тот же список КП и будет дублем первого, поэтому это делать не имеет смысла).

Для каждого КП ENGINE создает дочерний Актор КП (в контексте каждого из КП). После старта Потока данные телеметрии поступают в ENGINE от SOURCE. Внутри ENGINE хранится карта соответствия между каналами и КП.

> Таким образом, ENGINE выступает в роли коммутатора: на какой Актор посылать какую телеметрию.

Актор обрабатывает телеметрию (рассчитывает КП) и возвращает результат расчета КП в нужный специализированный PLUGIN, который сохраняет результаты расчета КП в базу.

5.3. Правила обработки (Rules)

Рассмотрим следующие правила обработки:

`FILTER` - компонент потоковой обработки данных, который перенаправляет полученные данные в один из двух путей: true/false согласно настроенным правилам.

`SWICH`- компонент потоковой обработки данных, который имеет возможность использования нескольких портов для перенаправления данных.

`OPERATOR` - компонент потоковой обработки данных, который выполняет вычислительные операции с данными и их передачу.

`CONVERTER` - специализированный компонент потоковой обработки данных, подготавливает обработанные данные для передачи их в Плагин для дальнейшей записи.

> Для загрузки данных из источников необходимо настроить механизмы сбора, обработки и передачи данных путем создания соответствующих объектов, обеспечивающих все необходимые стадии процесса (`Поток`, `Подключение`, `Сущность`, `Правило`, `Плагин`), а также задать `расписание загрузки`.

Настройка, запуск и контроль механизма сбора/обработки/передачи данных производится по следующему сценарию:

- Настройка Потока
  - Настройка Подключения;
  - Настройка Сущности;
  - Настройка Плагина;
  - Настройка Правила;
  - Настройка Расписания;
- Запуск Потока;

- Контроль за состоянием Потока;
- Управление Потоком;
- Управление компонентами Потока.

После выполнения описанных действий процесс загрузки может функционировать как на регулярной основе, так и в режиме «по требованию Администратора».

> При необходимости, пользователь Системы может отключить автоматическое выполнение процесса и инициировать в ручном режиме внеплановую загрузку данных из любого источника.

5.4. Работа с Потоками

Создание Потока

Для создания Потока следует перейти в панели навигации в раздел **«Менеджер потоков»**, в результате чего будет открыто окно работы с Потоками.

В открывшемся окне выбрать **Tenant** и **Customer** и нажать кнопку **[Создать]** (см. рисунок: Форма работы с потоками).

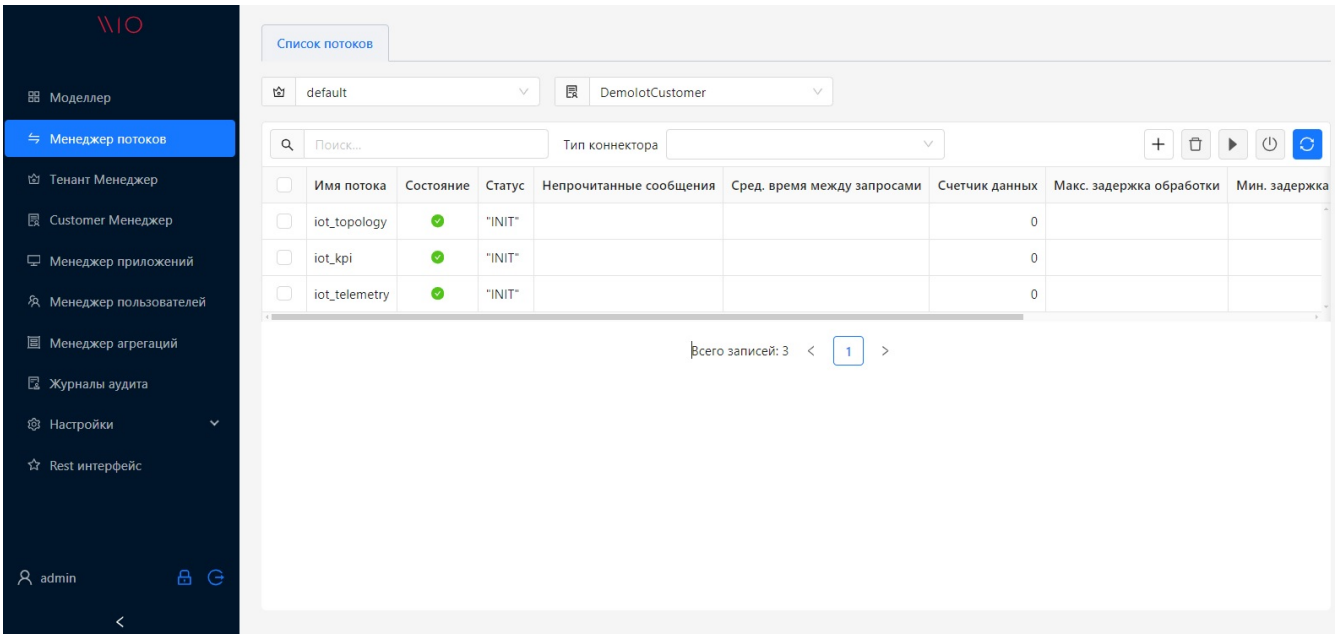

Рисунок 15. Форма работы с потоками

Откроется вкладка **«Создание»**, в которой следует приступить к конфигурированию потока.

Конфигурирование Потока

В списке потоков выбрать запись и нажать дважды левой клавишей мыши. Откроется вкладка с параметрами потока (см. рисунок: Атрибуты потока):

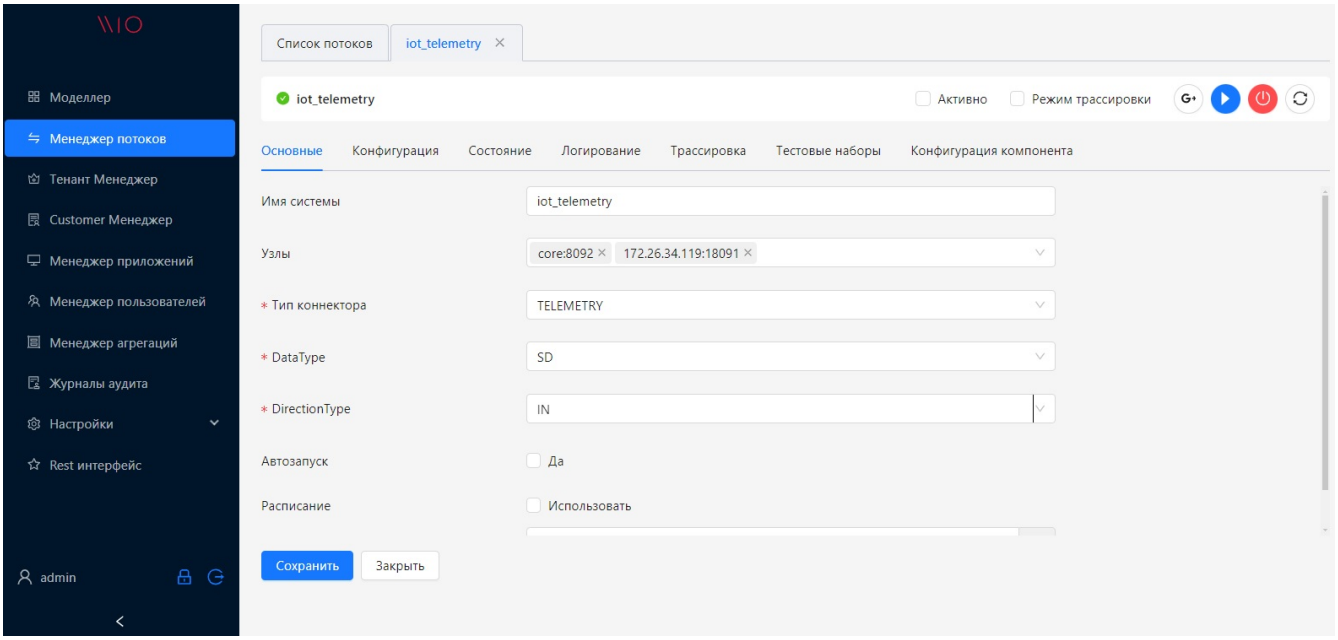

Рисунок 16. Атрибуты потока

Назначение полей атрибутов Потока на вкладке **«Основное»**:

- **«Имя системы»** - определяет наименование Потока, может содержать только латинские символы, необходимо выбрать из списка доступных.
- **«Узлы»** - выбрать из списка доступные узлы кластера, на которых будет выполняться Поток.
- **«Тип коннектора»** - определяет тип создаваемого потока. Для выбора из списка доступны: KPI, TELEMETRY, TOPOLOGY, CUSTOM, UNTYPED.
- **«DataType»** - определяет тип данных потока, должно быть установлено в значение `RT`.
- **«DirectionType»** - определяет тип обработки данных потока, должно быть установлено в значение `IN`.
- **«Автозапуск»** - параметр определяет возможность запуска потока автоматически после перезапуска системы.
- **«Расписание»** - параметр позволяет настроить расписание работы потока. Расписание указывается в формате `CRON` строки.

Далее следует перейти на вкладку **«Конфигурация»**.

Настройка Конфигурации

WEB-интерфейс конфигурации Потока представляет собой инструмент для создания и управления *потоковой передачей данных* в виде *направленного графа*, в узлах которого происходит *перенаправление, валидация, трансформация или выполнение какой-либо логики* над поступившими данными согласно настроенным Правилам.

Для настройки правил обработки Потока следует перейти в панели навигации на вкладку **«Конфигурация»**, в результате чего будет открыто окно конфигурации Потока (см. рисунок: Окно Конфигурации Потока):

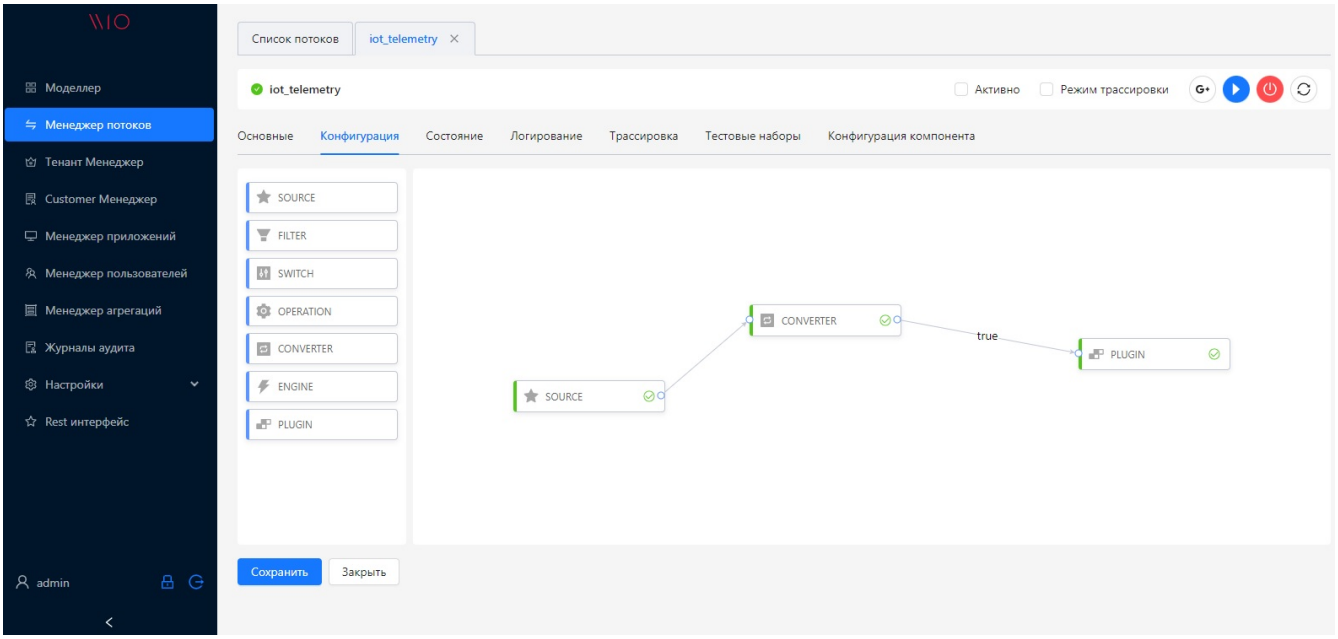

Рисунок 17. Окно Конфигурации Потока

В левой части окна присутствует палитра, из которой следует выбирать элементы в качестве узлов графа (возможные типы правил: `SOURCE`, `FILTER`, `SWITCH`, `OPERATION`, `CONVERTER`, `ENGINE`, `PLUGIN`), в правой части - холст, на котором непосредственно собирается граф обработки. Таким образом, настройка потока включает размещение на холсте компонентов нужных типов в качестве узлов графа, и настройку их атрибутов.

Настройка Правил (RULES)

Все Правила Системы абстрактно можно разделить на:

- Правила, для которых логика вычисления задается ***скриптом*** (`Java Script`), при настройке свойств таких Правил необходимо выбрать доступный метод постобработки и описать скрипт (вкладка **«Скрипт»**).
- Правила, для которых логика и метод постобработки запрограммированы ***java-кодом***, при настройке свойств таких Правил необходимо описать только параметры в конфигурации метода.

Для настройки Правил необходимо выбрать узел графа, и далее открыть через контекстное меню **«Свойства»** форму для редактирования атрибутов (**«Компонент потока»**) (см. рисунок: Контекстное меню для вызова свойств Правила).

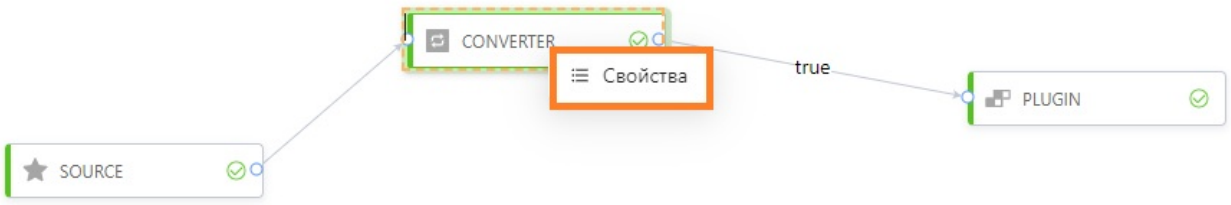

Рисунок 18. Контекстное меню для вызова свойств Правила

Структура окна для типов Правил: `FILTER`, `SWICH`,`OPERATOR` аналогична и включает набор необходимых свойств (см. рисунок: Окно Свойства для настройки Правил).

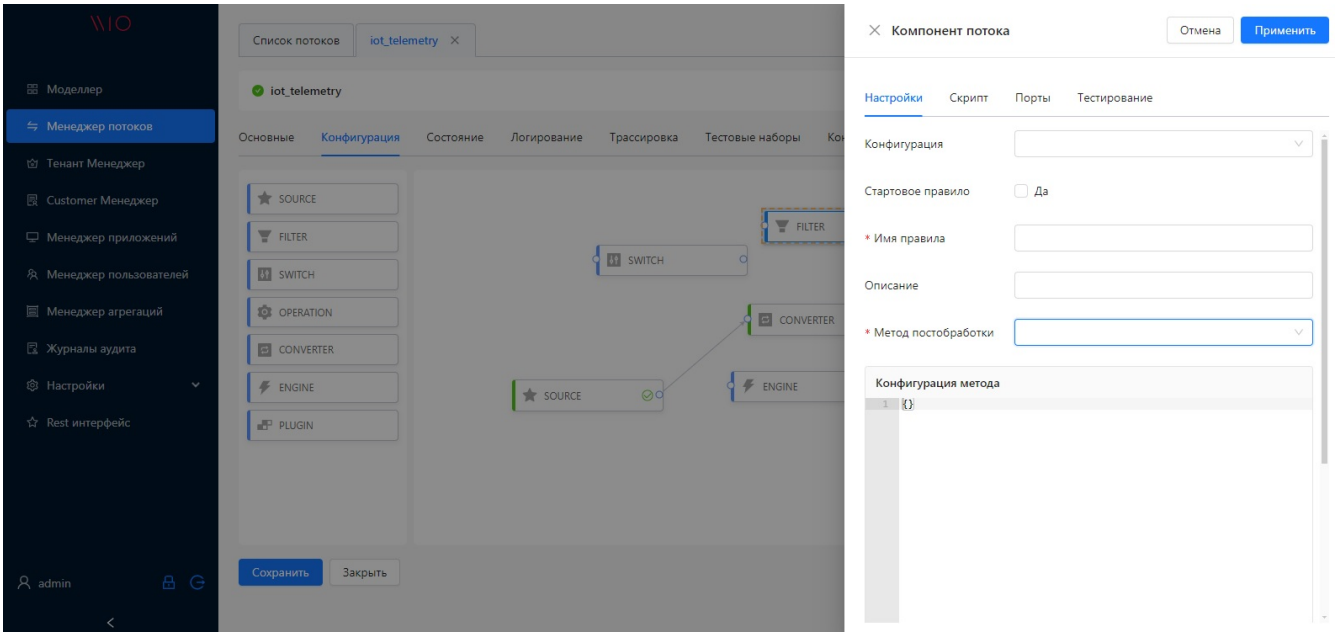

Рисунок 19. Окно Свойства для настройки Правил

Назначение полей атрибутов **«Правил»**:

- **«Имя правила»** - определяет наименование Правила, заполняется латиницей.
- **«Метод постобработки»** - является ключевой характеристикой Правила, определяет метод обработки данных, необходимо выбрать значение из списка доступных в Системе согласно выбранному типу Правила.
- **«Стартовое правило»** - признак обработки Правила первым в цепочке, заполняется автоматически.
- **«Описание»** - справочная информация для пользователя, комментарий. Конфигурация метода - строка в формате `JSON` определяет, конфигурацию для обработки данных согласно выбранному методу постобработки.
- **«Скрипт»** - опционная вкладка, необходима для описания и тестирования скрипта обработки данных.
- **«Порты»** - опционная вкладка, необходима для указания портов.
- **«Конфигурация метода»** - строка в формате JSON, задающая конфигурацию выбранного типа обработки.

Далее следует перейти к созданию и настройке **«Плагина» (PLUGIN)**.

Настройка Плагина (PLUGIN)

Для создания и настройки Плагина следует добавить его на холст Потока и выбрать **«Свойства»** через контекстное окно (см. рисунки: Контекстное меню для создания Плагина, Окно создания Плагина).

Рисунок 20. Контекстное меню для создания Плагина

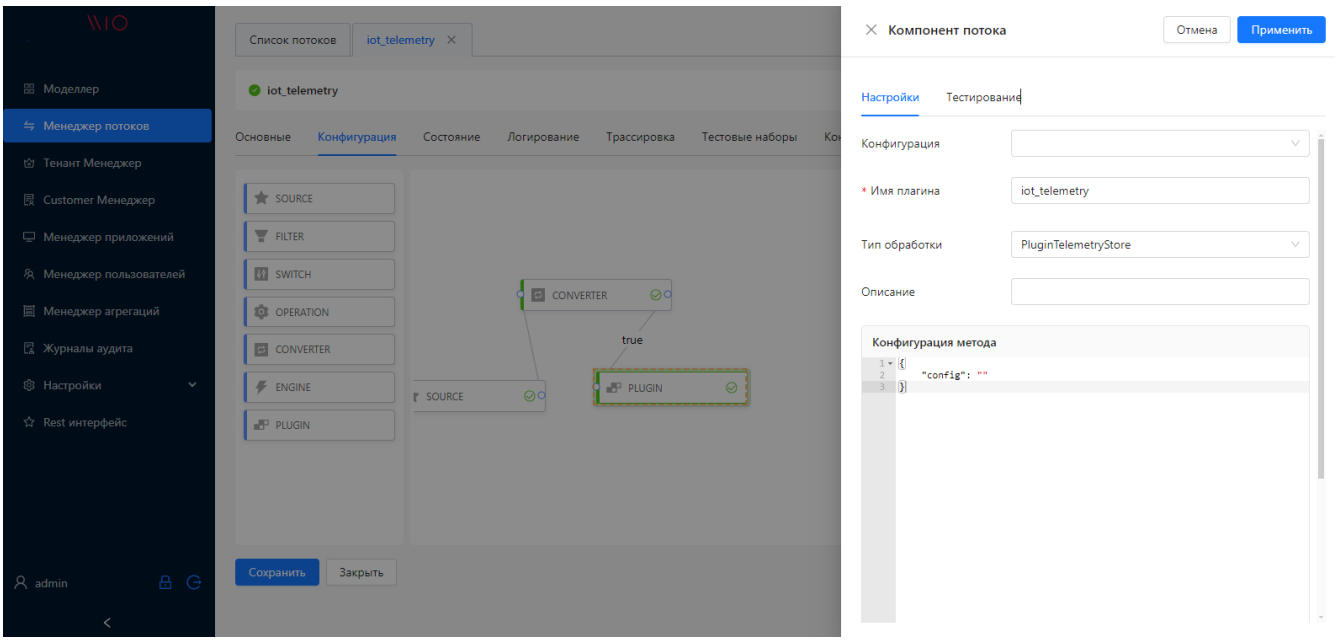

Рисунок 21. Окно создания Плагина

Назначение полей атрибутов **«Плагина»**:

- **«Имя плагина»** - определяет наименование Плагина для регистрации его в Системе, используется при фильтрации и поиске в Системе.
- **«Описание»** - справочная информация для пользователя, комментарий.
- **«Тип обработки»** - определяет тип обработки данных, необходимо выбрать значение из списка доступных согласно типу целевой среды.
- **«Конфигурация метода»** - строка в формате JSON, задающая конфигурацию выбранного типа обработки.

У каждого типа обработки предусмотрен свой набор полей конфигурации.

Далее следует перейти к настройке **Расписания Потока**.

Настройка расписания

Настройка Расписания Потока предусматривает указание времени и его продолжительности в CRON-формате, где:

- *1й* символ секунды **0-59** (`, - * /`);
- *2й* символ минуты **0-59** (`, - * /`);
- *3й* символ часы от **0-23** (`, - * /`);

- *4й* символ день месяца от **1-31** (`, - * /`);
- *5й* символ месяц от **1-12|JAN-DEC** (`, - *`);
- *6й* символ день недели **1-7|MON-SUN** (`? , - *`);

Данная настройка выполняется в атрибуте Расписание на вкладке **«Основные»** Потока.

Управление потоком

При создании нового Потока его активация задается автоматически, при необходимости автоматическую активацию можно отключить (галочка **«Активно»**), отследить этот признак можно на панели управления (см. рисунок: Панель управления Потока).

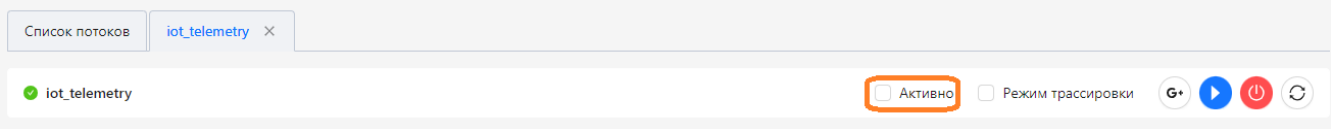

Рисунок 22. Панель управления Потока

Панель управления Потока позволяет пользователю использовать функции:

- *Разовый запуск* - используется в случае необходимости разового запуска Потока обработки данных в ручном режиме независимо от настроенного расписания.
- *Остановить текущее выполнение* - используется для принудительной остановки Потока обработки данных независимо от настроенного расписания.
- *Обновить статус* - используется для получения обновленной информации о состоянии Потока.

Контроль за состоянием потоков

С целью мониторинга и контроля механизма обработки данных Система позволяет пользователю отслеживать состояние Потока посредством цветового индикатора.

Общее состояние Потока доступно через окно работы с Потоками (см. рисунок: Окно работы с Потоками - Состояние Потока).

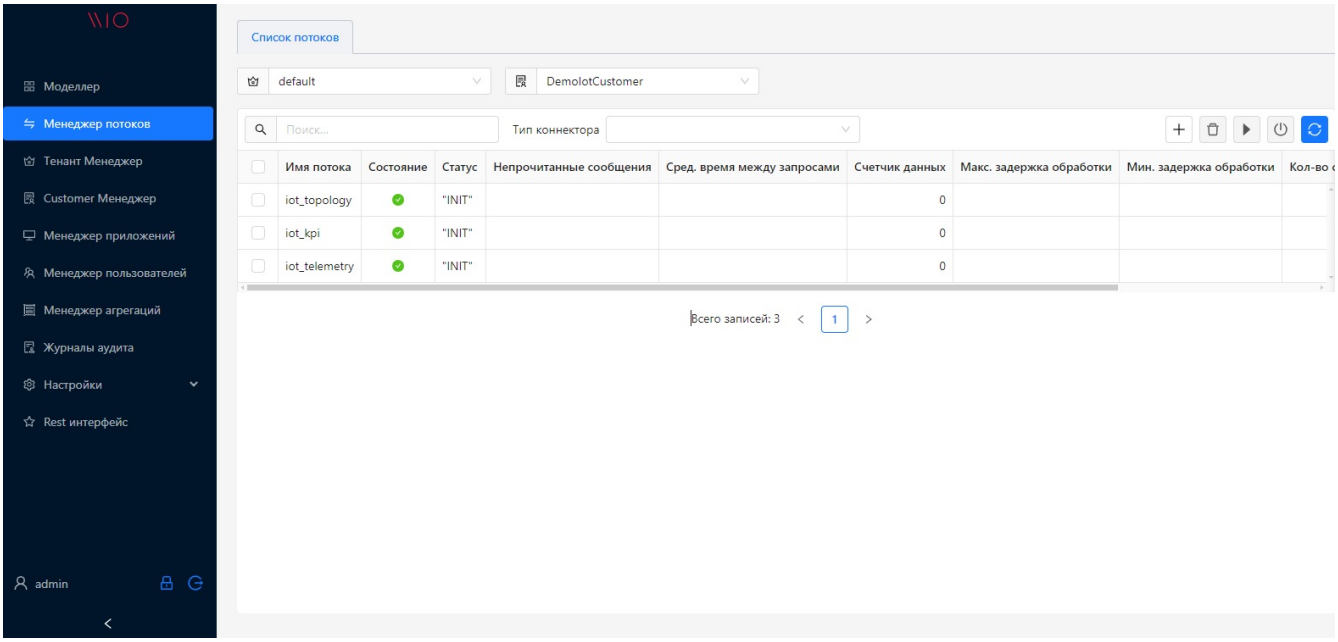

Рисунок 23. Окно работы с Потоками - Состояние Потока

Чтобы отследить состояние Потока отдельно по узлам пользователю требуется перейти на вкладку «Состояние» и выбрать конкретный узел (см. рисунок: Окно работы с Потоками - Состояние узла Потока).

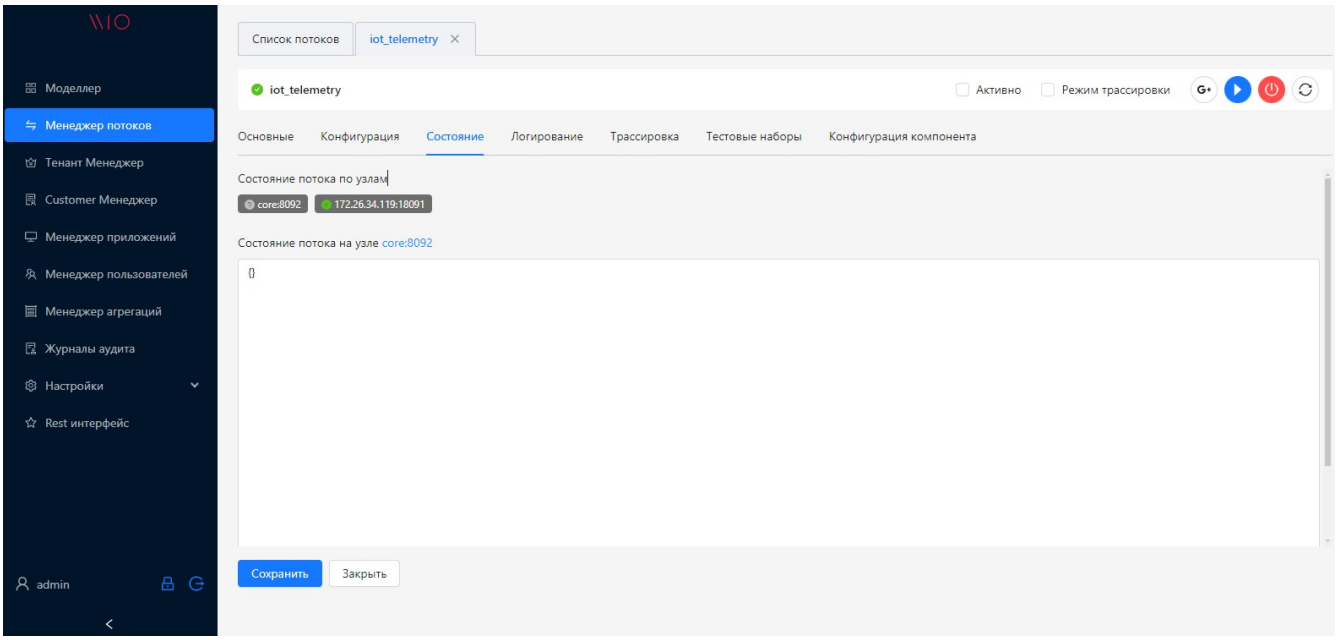

Рисунок 24. Окно работы с Потоками - Состояние узла Потока

Индикатор состояния Потока в соответствии с цветом означает:

- Зеленый – поток работоспособен;
- Желтый – ошибка соединения;
- Красный – поток не работоспособен.

6. Моделлер конфигурационных элементов (КЭ)

Настройка модели данных в Системе предполагает формирование *логической иерархической структуры* данных посредством *описания* объектов реального мира и их отношений.

6.1. Моделлер КЭ

Данный интерфейс позволяет пользователю описать *структуру данных* Системы, оперируя конфигурационными элементами `Сущность` и `Связь`.

> Структура данных представлена в виде иерархического дерева типов, где объектами всегда являются:
>
> - **«CI»** - для Типа *сущности*;
> - **«Relation»** - для Типа *связи*.

Для открытия моделлера КЭ необходимо выбрать в меню **Моделлер** и далее перейти во вкладку [Моделлер типов КЭ]. В открывшемся окне выбрать **Tenant**, для которого необходимо сформировать типы КЭ (см. рисунок: Моделлер типов КЭ).

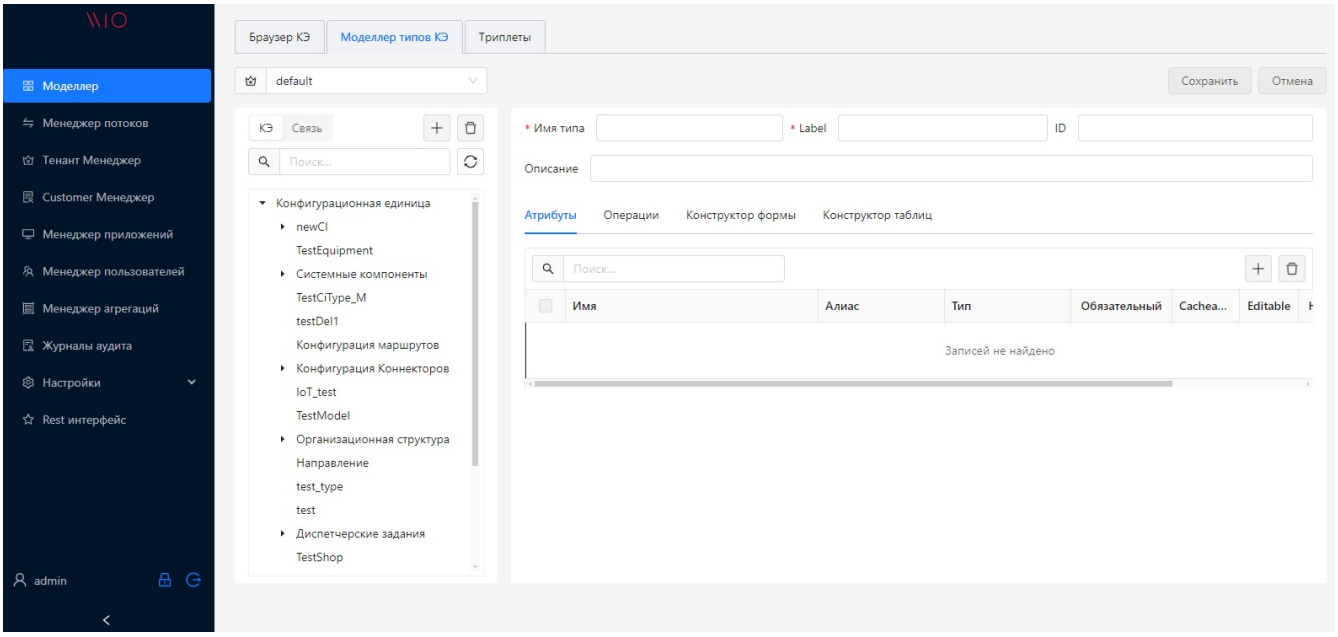

Рисунок 25. Моделлер типов КЭ

Для создания нового Типа сущности требуется выбрать элемент из дерева типа для назначения его в качестве родительского элемента и нажать кнопку **[Создать]**.

Далее, необходимо описать свойства создаваемого Типа сущности и определить атрибутный состав (см. рисунок: Окно создания Типа сущности)

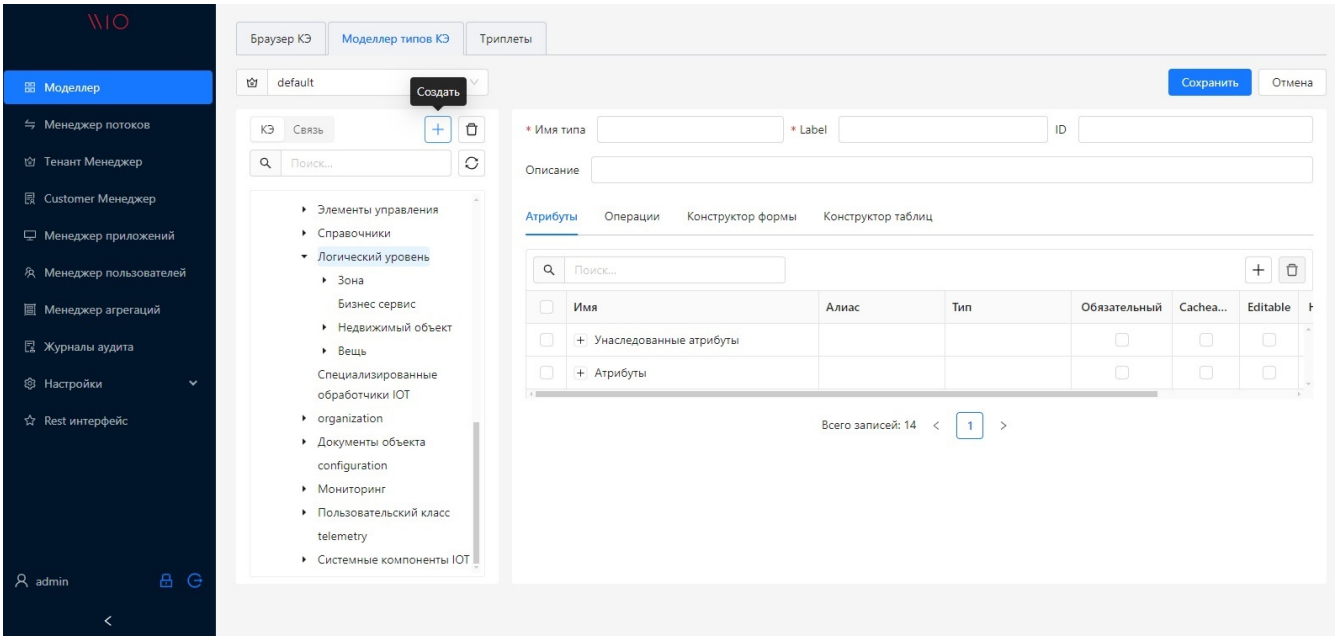

Рисунок 26. Окно создания Типа сущности

Назначение полей атрибутов **формы создания Типа сущности**:

- **«Имя типа»** - определяет наименование Типа сущности, может содержать только латинские символы, идентифицирует Тип сущности при использовании фильтра и поиска.
- **«Label»** - описание типа в произвольной форме, используется для отображения на экранных формах.
- **«Описание»** - описание создаваемого типа.
- **«Атрибуты типа»** - отображает состав наследуемых атрибутов объекта, при необходимости для создаваемого нового типа состав можно дополнить новыми атрибутами.
- **«ID»** - идентификатор, формируется автоматически.

После заполнения всех свойств и атрибутов необходимо сохранить создаваемый Тип сущности, нажав копку **[Сохранить]**.

> Структура окна и операции создания Типа связи аналогичны описанным выше.

6.2. Браузер КЭ

Управление объектами реализуется через функции просмотра, создания, изменения и удаления экземпляров объектов реального мира и их связей.

Управление объектами осуществляется через интерфейс **«Браузер КЭ»**. Для его открытия необходимо в главном меню выбрать **Моделлер** и далее перейти во вкладку [Браузер КЭ] (см. рисунок: Браузер КЭ).

Перед тем как выполнить какую-либо функцию над КЭ, необходимо выбрать **Tenant** и

Customer в соответствующих полях.

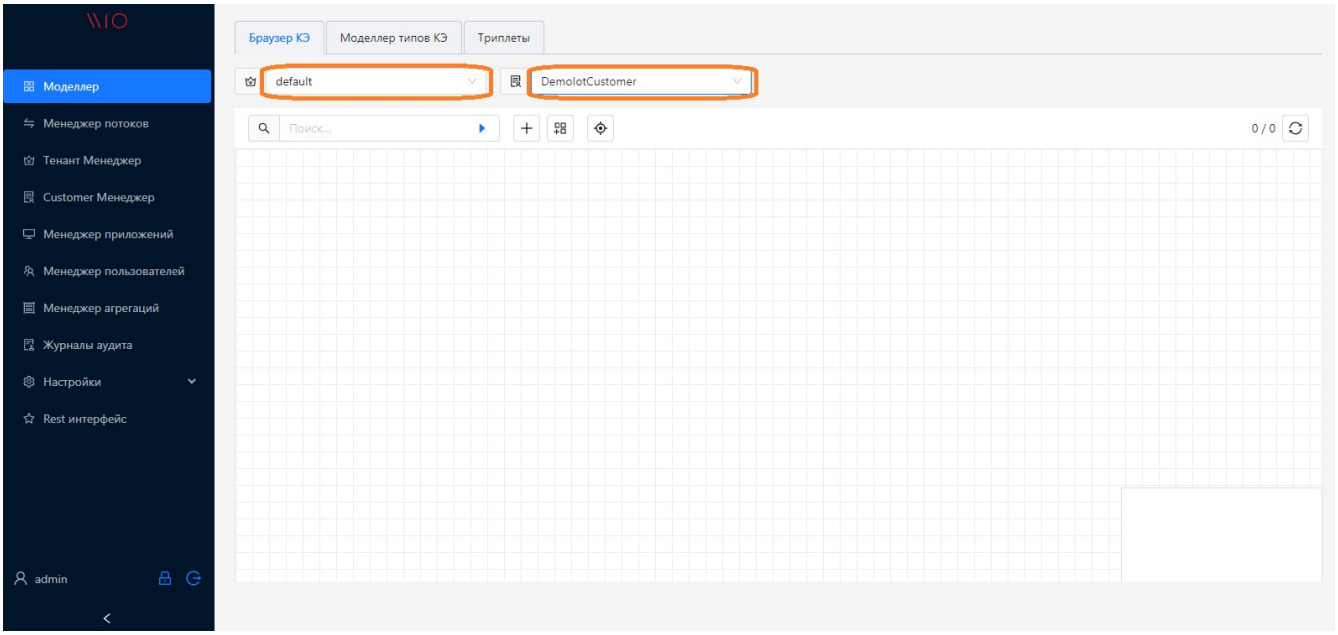

Рисунок 27. Браузер КЭ

Данный интерфейс позволяет пользователю создать *новый экземпляр*, осуществить *поиск* необходимого экземпляра по его типу, внести *изменения* в свойства экземпляра, назначить ему *связь* с другими экземплярами, а также *удалить* экземпляр.

Просмотр КЭ и связей

Для просмотра экземпляров и его связей необходимо в браузере КЭ выбрать тип КЭ, далее через контекстное меню отобразить все связанные с ним КЭ (см. рисунки: Выбор типа и экземпляра КЭ, Просмотр КЭ и его взаимосвязей).

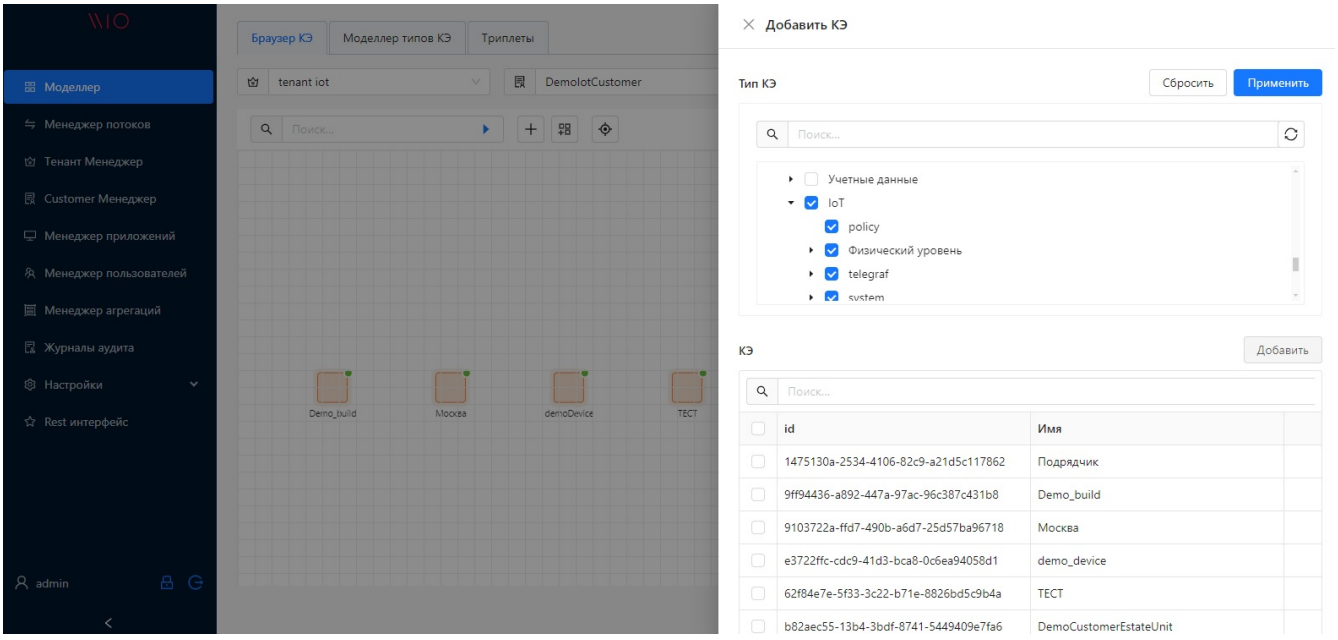

Рисунок 28. Выбор типа и экземпляра КЭ

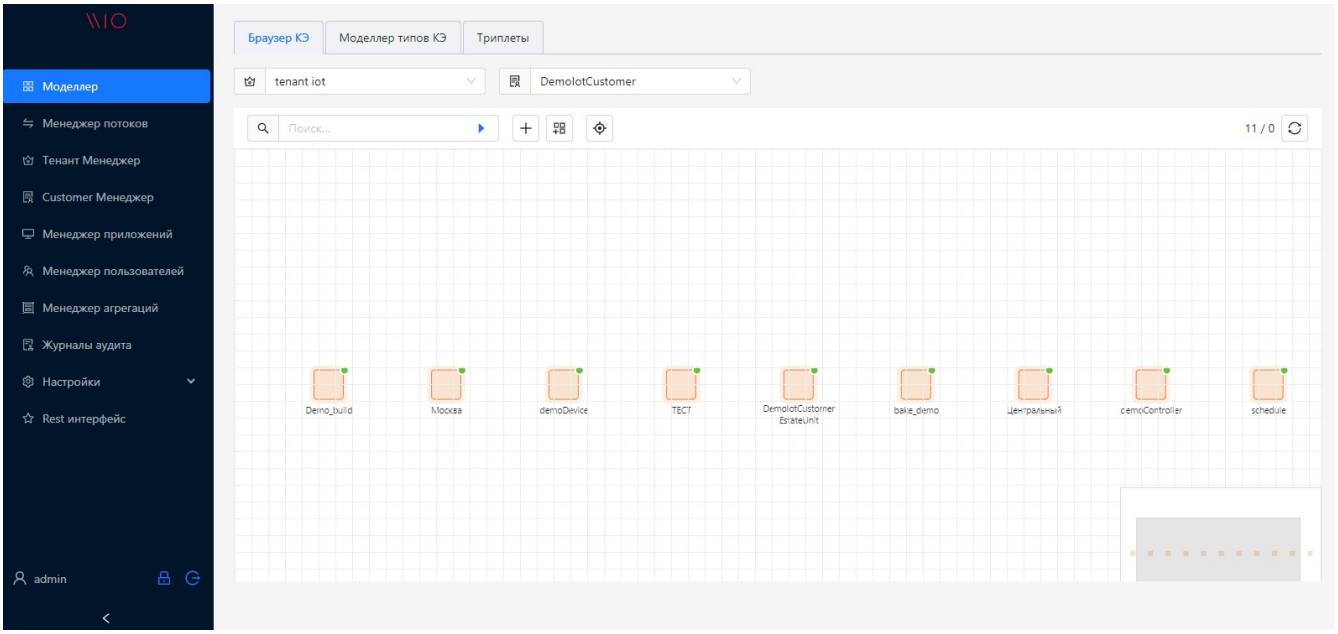

Рисунок 29. Просмотр КЭ и его взаимосвязей

> Панель, где отображаются выбранные экземпляры КЭ представляет собой графическую среду для управления экземплярами объектов и построения отношений между ними.

Связь между экземплярами отображается линией.

При управлении экземплярами объектов Системы пользователю доступны функции (см. рисунок: Контекстное меню КЭ):

- **Свойства КЭ** - используется для просмотра и изменения характеристик экземпляра.
- **Связанные КЭ** - используется для отображения всех связанных с экземпляром объектов.
- **Удалить из представления** - используется для удаления экземпляра (без физического удаления из базы данных, только из графической среды).
- **Удалить из базы** - используется для удаления экземпляра из базы данных.

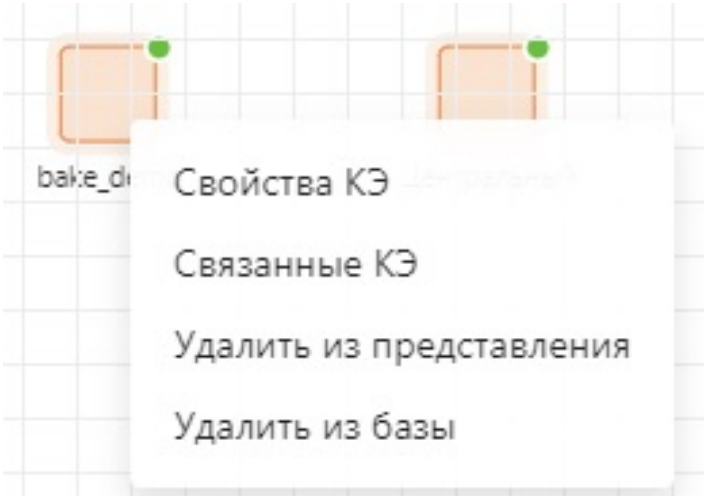

Рисунок 30. Контекстное меню КЭ

Создание КЭ и связей

Для создания нового экземпляра нажать на кнопку **[Создать КЭ]**, после следует в дереве Тип КЭ выбрать необходимый тип объекта и Customer и нажать **[Далее]** (см. рисунок: Создание КЭ).

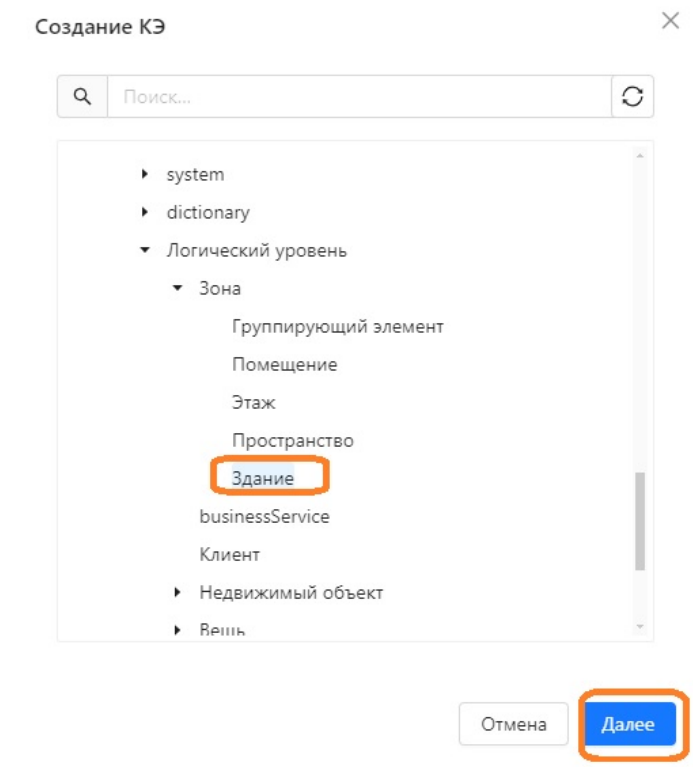

Рисунок 31. Создание КЭ

Далее заполнить все необходимые атрибуты и нажать **[Сохранить]** (см. рисунок: Атрибуты КЭ определенного типа).

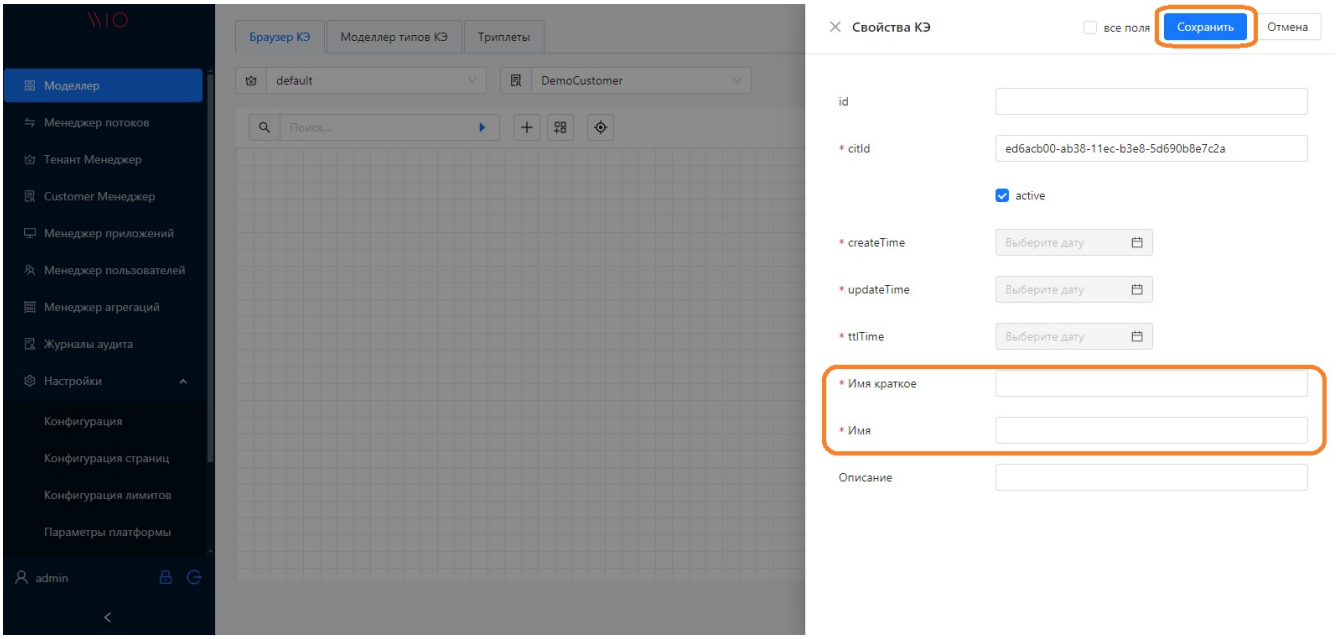

Рисунок 32. Атрибуты КЭ определенного типа

> Для создания связи между КЭ необходимо от родительского КЭ

> протянуть связь до дочернего, выбрав при этом тип связи.

6.3. Триплеты

Данный функционал позволяет *задать правила* для создания экземпляров КЭ, если таковые необходимы.

> Правила определяются набором триплетов, которые описывают возможные взаимосвязи между типами КЭ. Данные правила могут быть применены для организации работы с сущностями в Приложениях, построенных на базе Платформы.

Управление триплетами осуществляется через интерфейс **«Триплеты»**. Для его открытия необходимо в главном меню выбрать **Моделлер** и далее перейти во вкладку Триплеты (см. рисунок: Форма триплетов).

Перед тем как выполнить какую-либо функцию над триплетом, необходимо выбрать **Tenant** в соответствующем поле.

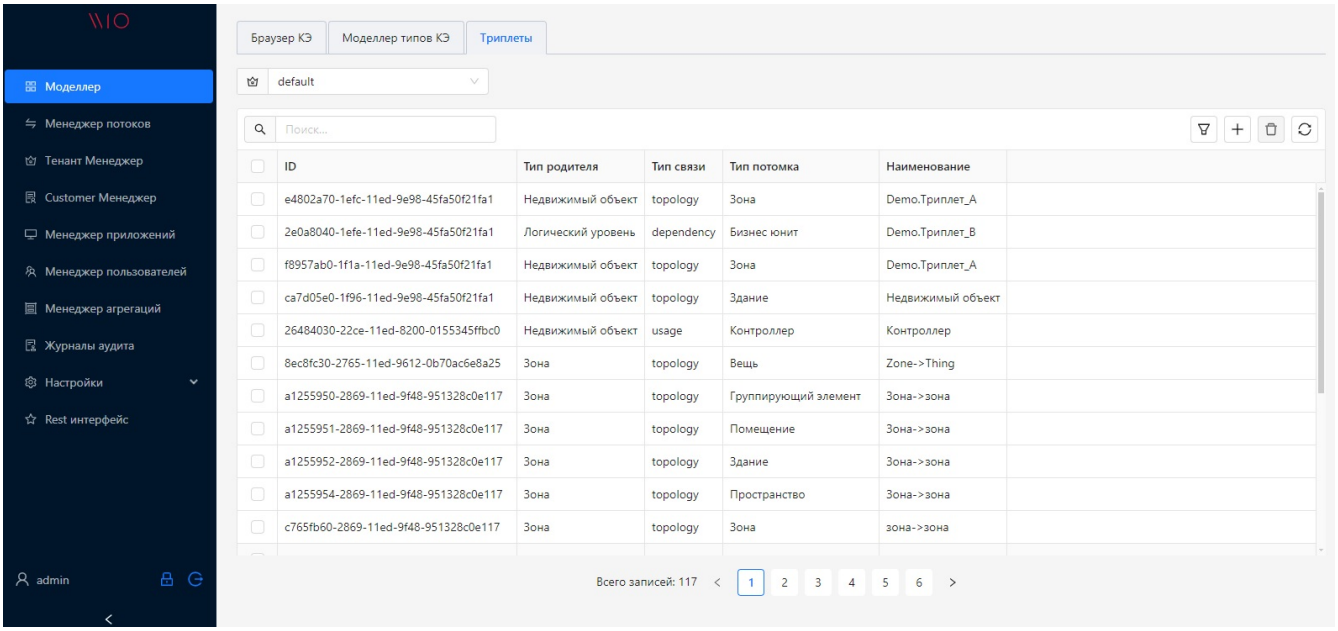

Рисунок 33. Форма триплетов

Чтобы выполнить настройки триплета необходимо нажать левой клавишей мыши на строке или, если происходит создание триплета, на кнопку **[Создать]**, после чего откроется окно редактирования триплета (см. рисунок: Атрибуты триплета).

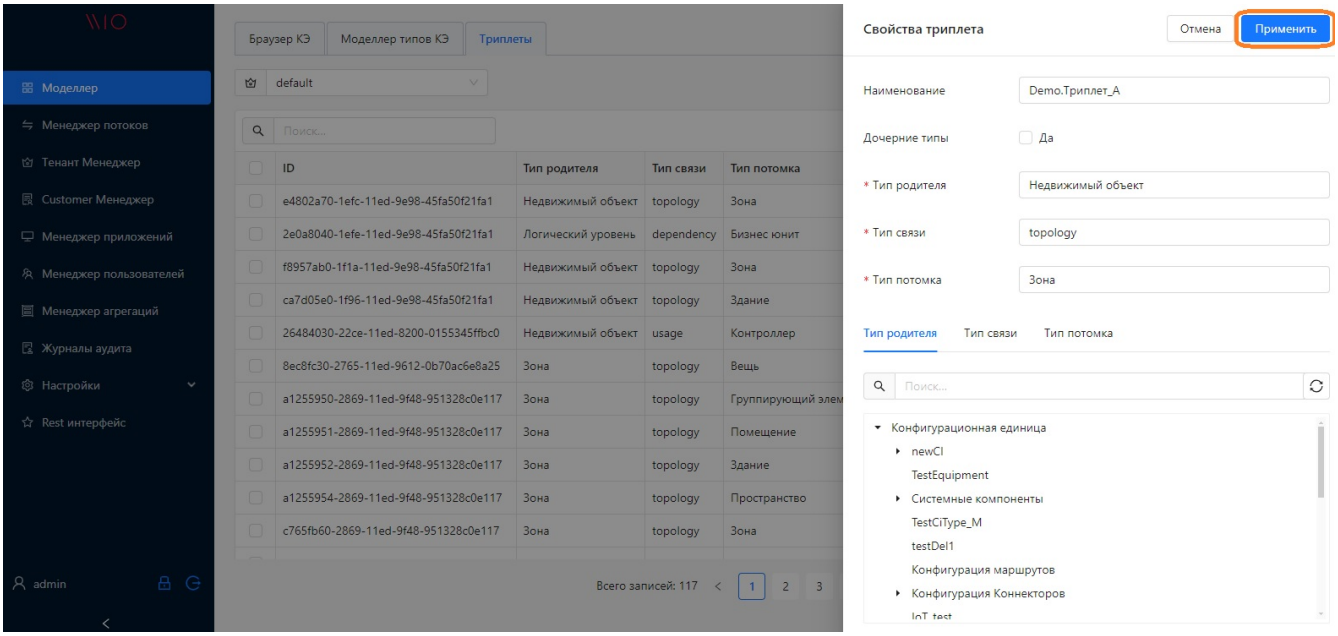

Рисунок 34. Атрибуты триплета

Обязательно к указанию типы родителя и потомка, а также типа связи.

Далее необходимо заполнить атрибуты на форме и нажать **[Применить]**.

7. Права доступа

Платформа обеспечивает централизованное управление доступом. Управление доступом включает в себя поддержку:

- единого для всех приложений реестра пользователей;
- управление реестром ролей для каждого приложения;
- назначение каждой роли списка доступных ресурсов;
- назначение пользователю прав доступа к конкретному приложению с заданной ролью;

Управление приложениями и ролями доступа

Общее описание функционала

Функционал позволяет управлять реестром приложений и настраивать роли доступа к ним.

Описание интерфейса

Для доступа к компоненту **«Менеджер приложений»** необходимо в основном меню веб-интерфейса пользователя выбрать пункт меню **Менеджер приложений**.

Создание\Редактирование учётной записи приложения

Для создания учетной записи приложения следует:

1. Перейти в меню **Менеджер приложений**.
2. В выпадающем списке выбрать тенант (см. рисунок: Реестр учётных записей приложений)
3. Нажать кнопку **[Создать]**

Редактирование учётной записи приложения осуществляется при двойном «клике» по записи в списке учётных записей.

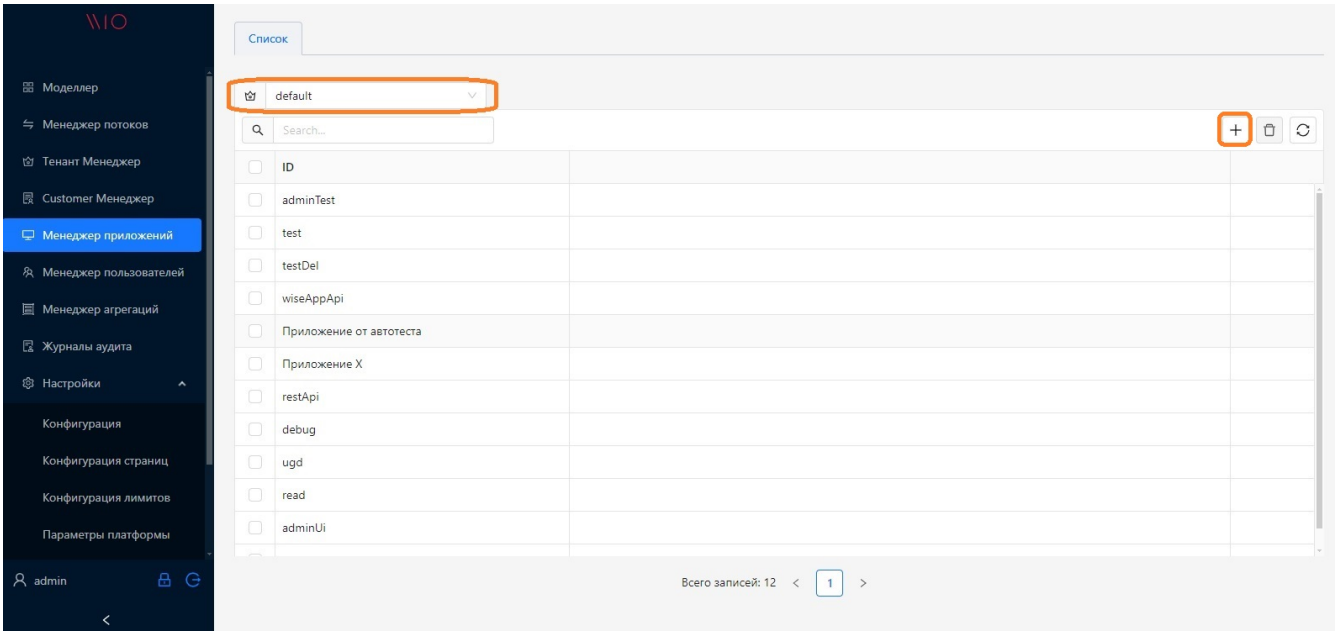

Рисунок 35. Реестр учётных записей приложений

В новой вкладке откроется карточка учетной записи приложения. Необходимо заполнить требуемые атрибуты "Наименование", "Секрет" и нажать кнопку **[Сохранить]** (см. рисунок: Карточка учётной записи приложения).

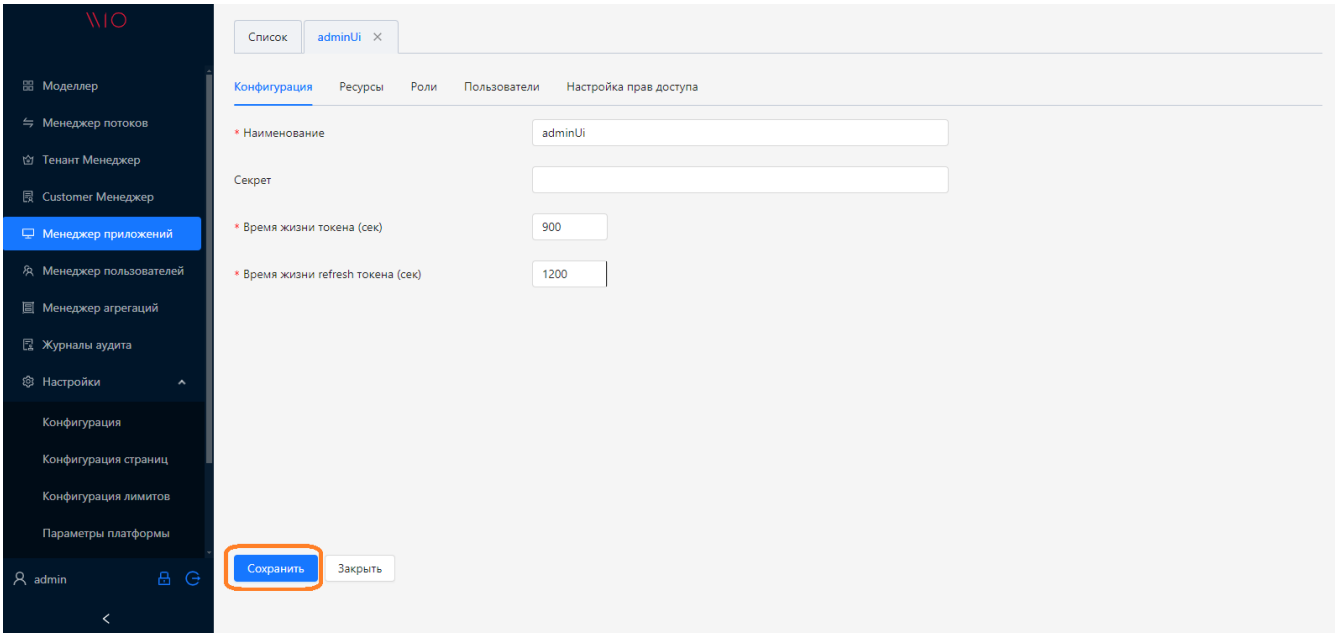

Рисунок 36. Карточка учётной записи приложения

Настройка доступа к приложению

> При настройке доступа к приложениям используется понятие Роли, которая содержит набор ресурсов, доступных Пользователю, имеющему данную Роль.

При настройке ролевого доступа к приложению доступны следующие функции:

▼ *Назначение ресурса приложению*

Для назначения ресурсов приложению следует:

1. Перейти во вкладку Ресурсы
2. Нажать на кнопку **[Добавить]** (см. рисунок: Переход к добавлению ресурса приложению)
3. В модальном окне "Добавление ресурса" (см. рисунок: Окно добавления ресурса приложению) выбрать необходимые ресурсы и нажать кнопку **[Добавить]**,
4. Нажать кнопку **[Сохранить]**

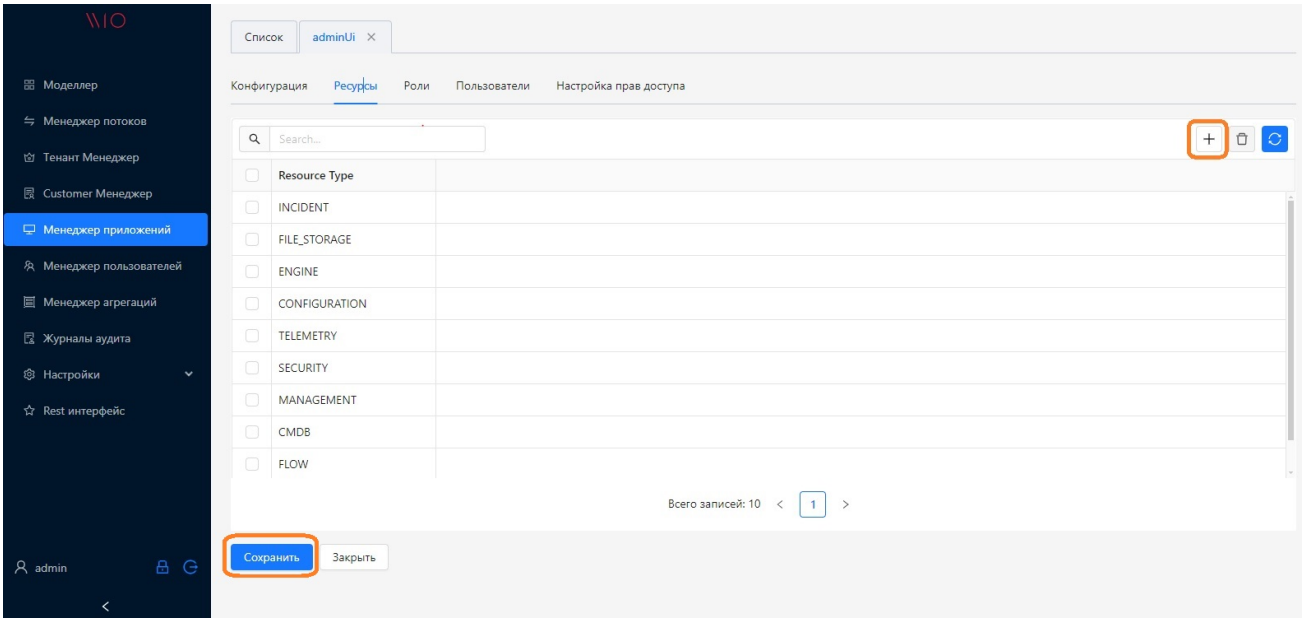

Рисунок 37. Переход к добавлению ресурса приложению

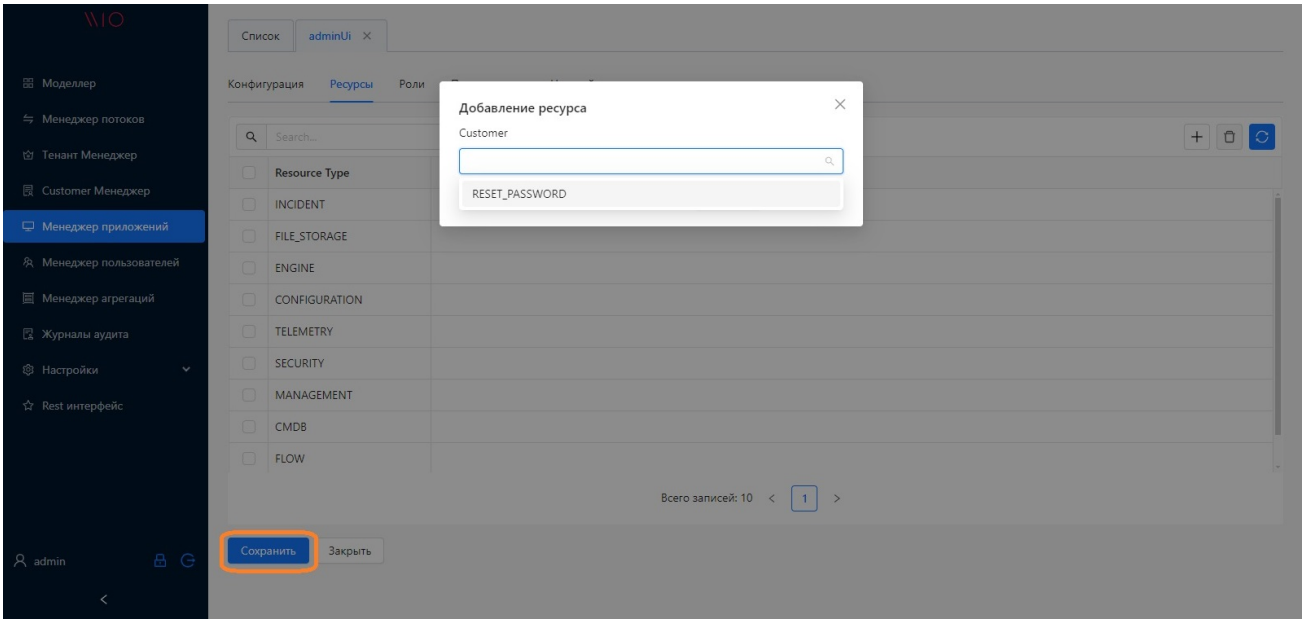

Рисунок 38. Окно добавления ресурса приложению

▼ *Создание|Редактирование роли доступа к приложению*

Для создания учетной записи роли следует перейти во вкладку "Роли" и выполнить следующие действия:

1. Нажать кнопку **[Добавить]** (см. рисунок: Переход к добавлению роли доступа к

приложению), после чего в новой вкладке откроется карточка учетной записи роли приложения

2. Во вкладке Конфигурация Заполнить требуемый атрибут "Наименование" и опционально установить чекбокс Администратор (см. рисунок: Карточка роли доступа к приложению), что даст создаваемой роли административные права, и далее нажать кнопку Сохранить
3. Во вкладке Ресурсы следует установить чекбоксы: в столбце "Доступен" для нужных ресурсов, в столбце "Ведение журнала" для ресурсов, факты доступа к которым сохранять в журнал.
4. Нажать кнопку **[Сохранить]** (см. рисунок: Сопоставление ресурсов роли доступа к приложению).

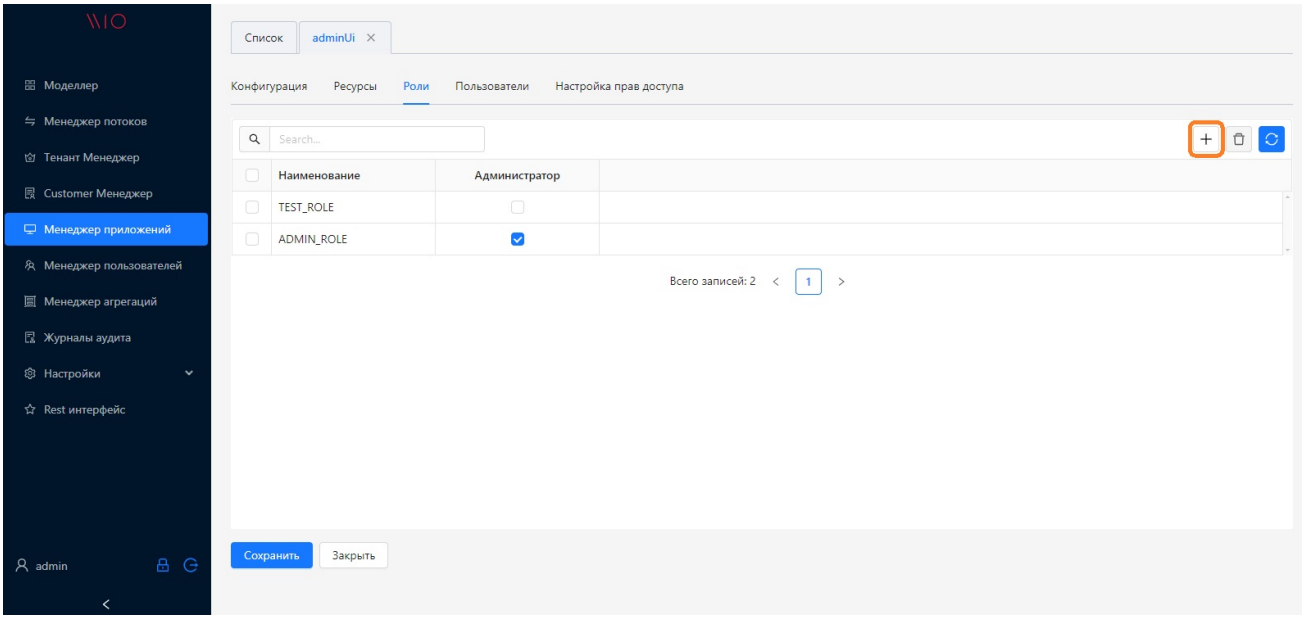

Рисунок 39. Переход к добавлению роли доступа к приложению

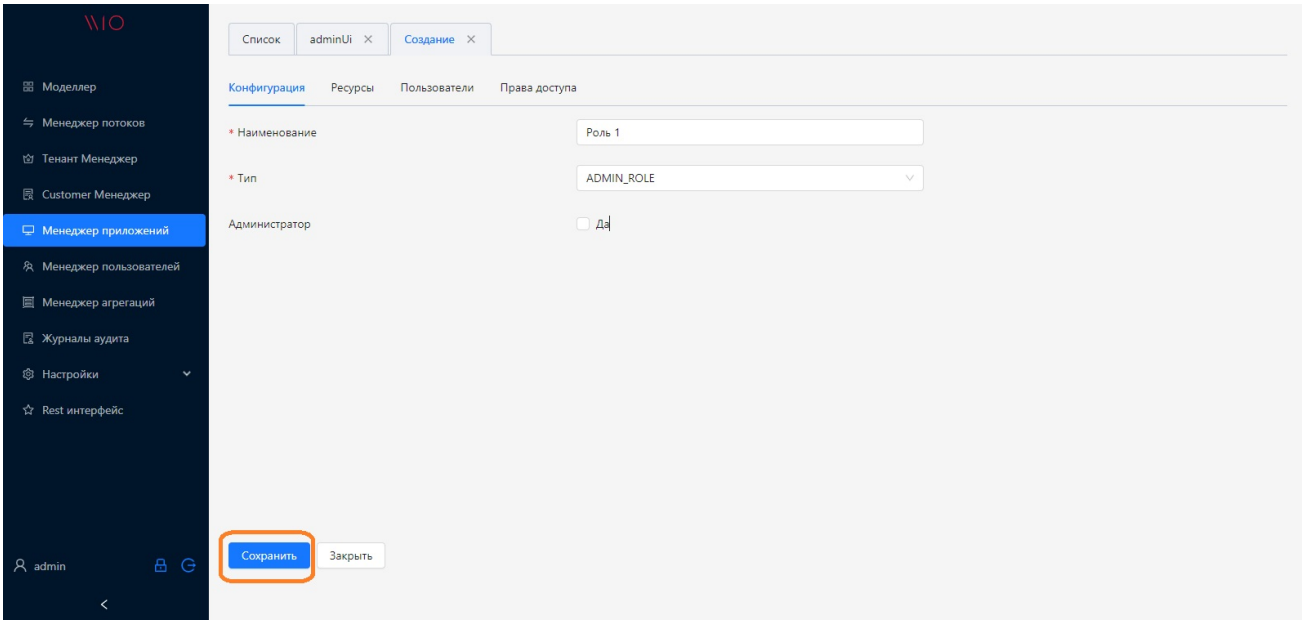

Рисунок 40. Карточка роли доступа к приложению

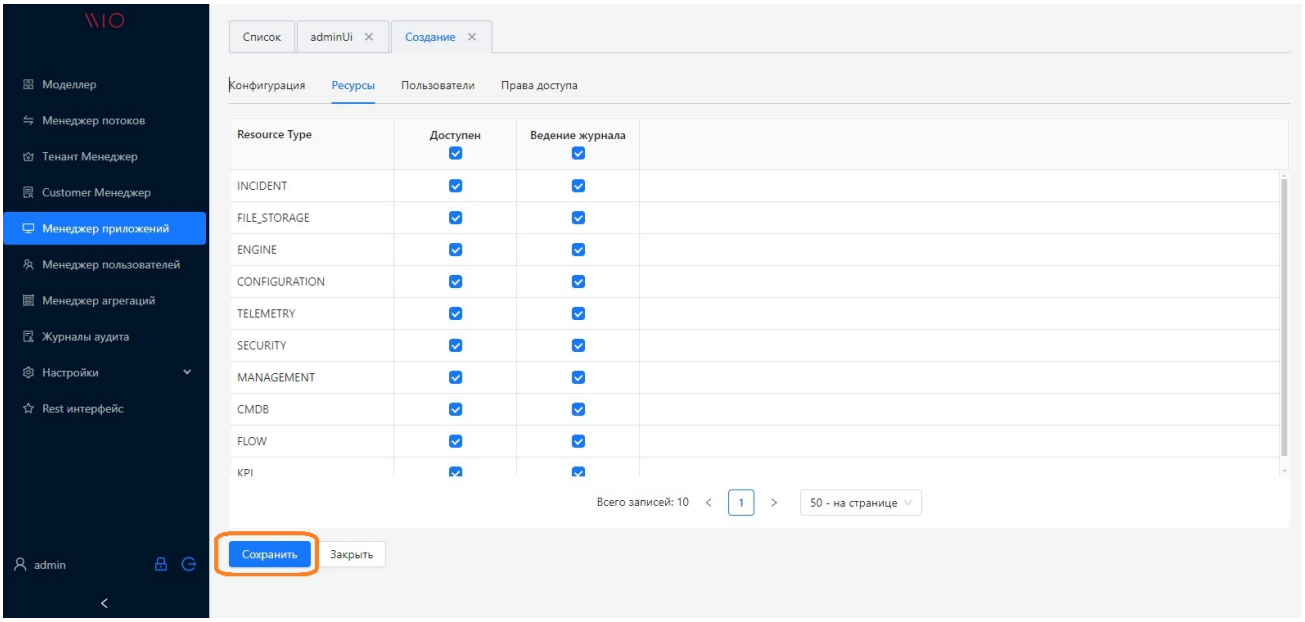

Рисунок 41. Сопоставление ресурсов роли доступа к приложению

Редактирование учётной записи роли осуществляется при двойном «клике» по записи в списке учётных записей.

▼ *Просмотр пользователей с доступом к приложению*

Для просмотра пользователей, имеющих назначенный доступ к приложению, следует в карточке приложения перейти во вкладку "Пользователи", в результате чего будет отображен список пользователей (см. рисунок: Список пользователей с доступом к приложению).

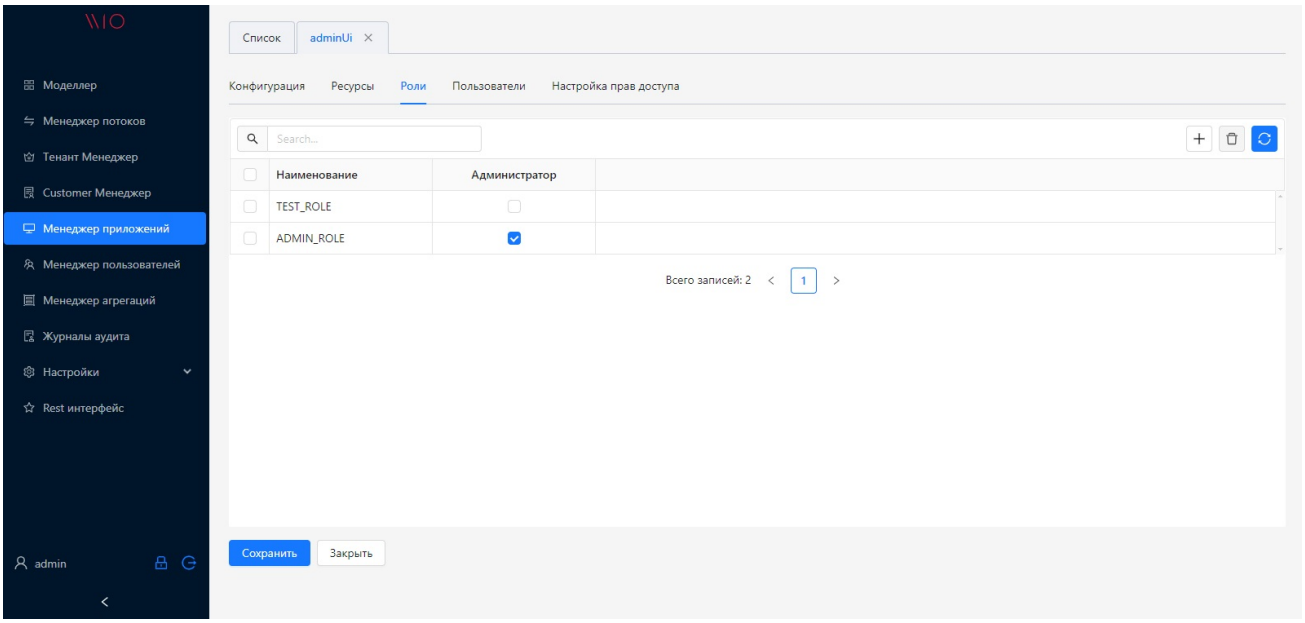

Рисунок 42. Список пользователей с доступом к приложению

▼ *Удаление роли доступа к приложению*

Для удаления роли доступа к приложению следует в карточке приложения перейти во вкладку "Роли" и далее:

1. Выбрать чекбоксом требуемую роль (роли)

2. Нажать кнопку **["Удалить"]**(см. рисунок: Удаление роли доступа к приложению)
3. В модальном окне подтверждения нажать на кнопку **["Удалить"]** (см. рисунок: Подтверждение удаления роли доступа к приложению)

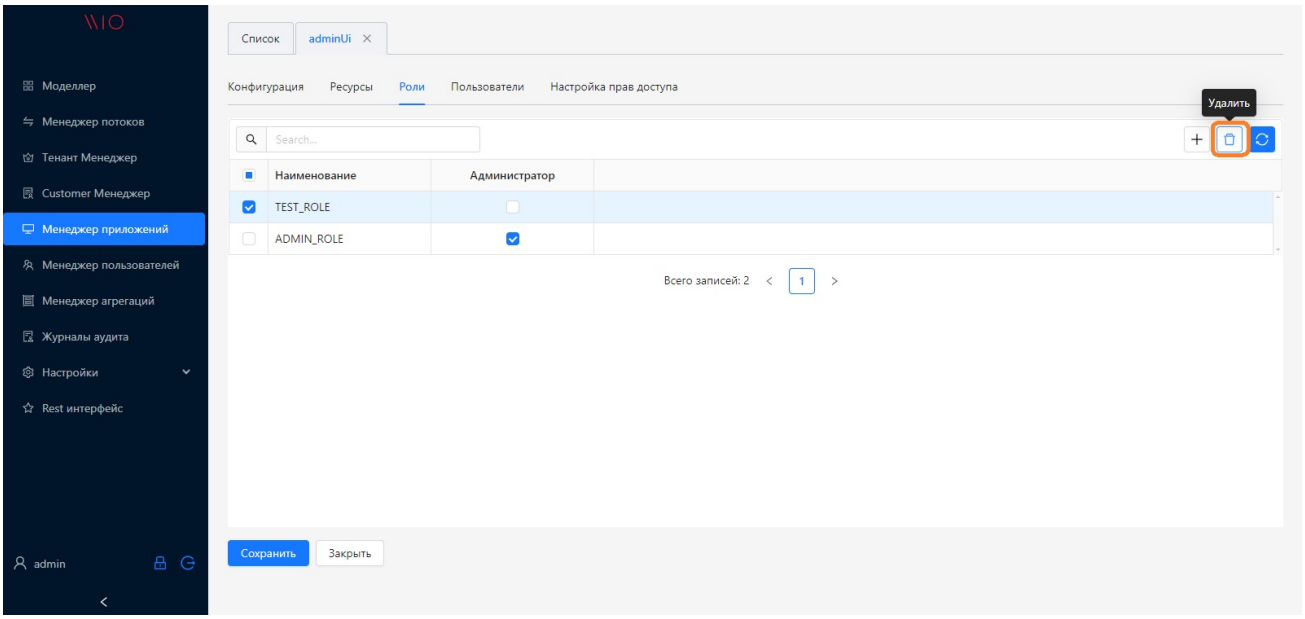

Рисунок 43. Удаление роли доступа к приложению

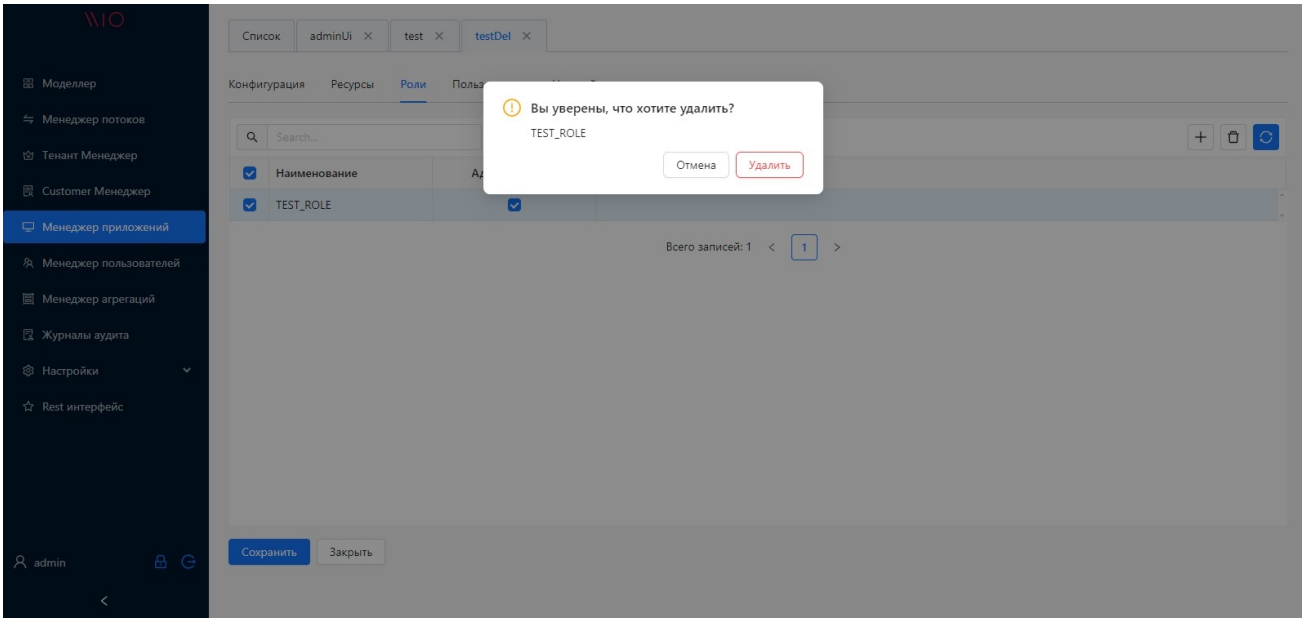

Рисунок 44. Подтверждение удаления роли доступа к приложению

Удаление учётной записи приложения

Для удаления учётной записи приложения необходимо:

1. В списке приложений выбрать строчку с учётной записью и нажать кнопку **[Удалить]** (см. рисунок: Переход к удалению учётной записи приложения).
2. В модальном окне подтверждения нажать на кнопку **["Удалить"]** (см. рисунок: Подтверждение удаления учётной записи приложения)

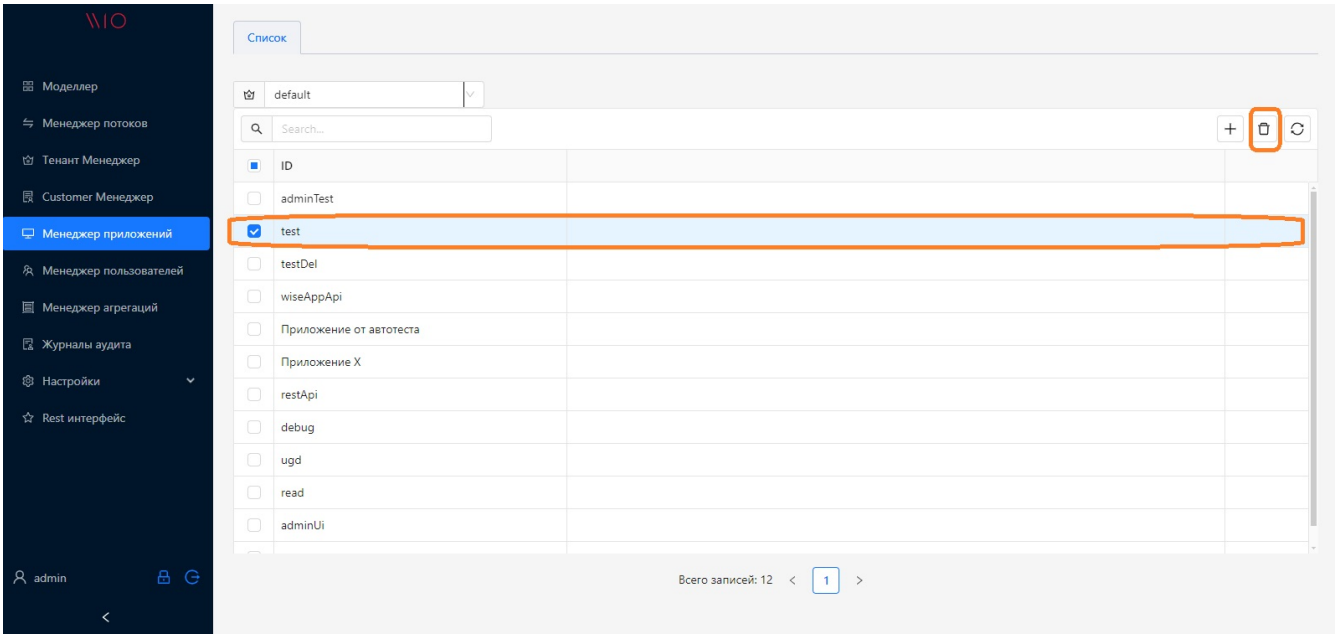

Рисунок 45. Переход к удалению учётной записи приложения

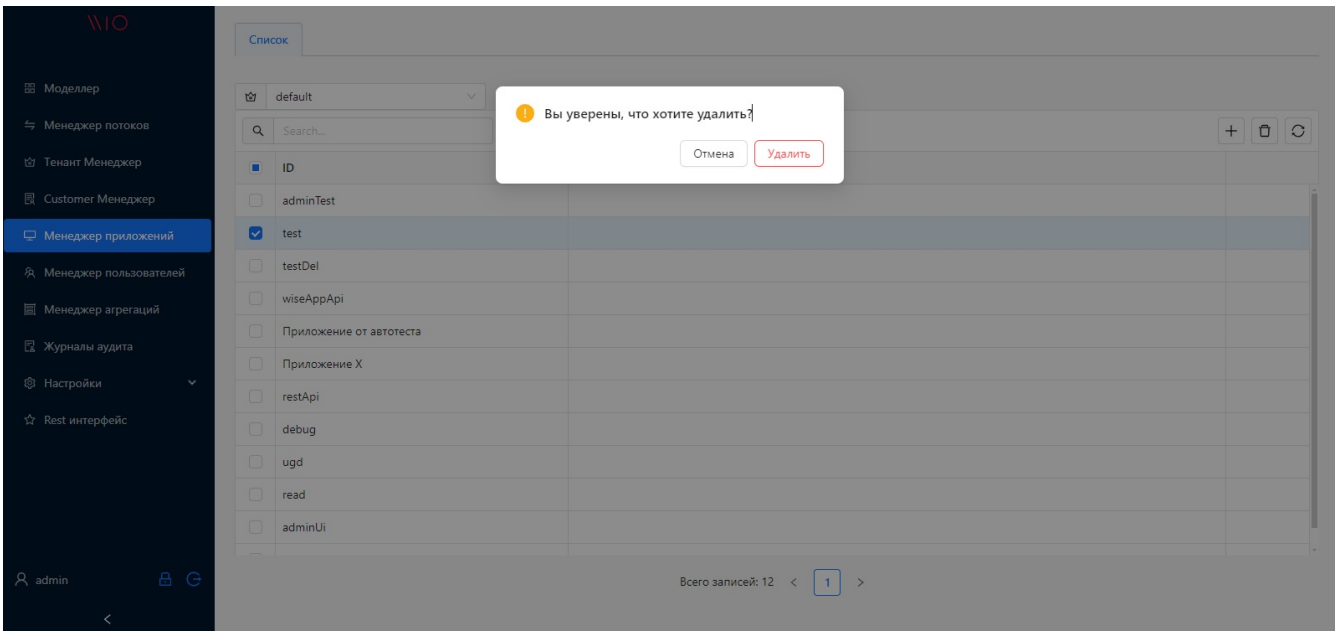

Рисунок 46. Подтверждение удаления учётной записи приложения

Управление пользователями

Общее описание функционала

Функционал позволяет управлять учётными записями пользователей, которым предоставляется доступ к Системе (см. рисунок: Реестр учётных записей пользователей), а также осуществлять назначение прав доступа.

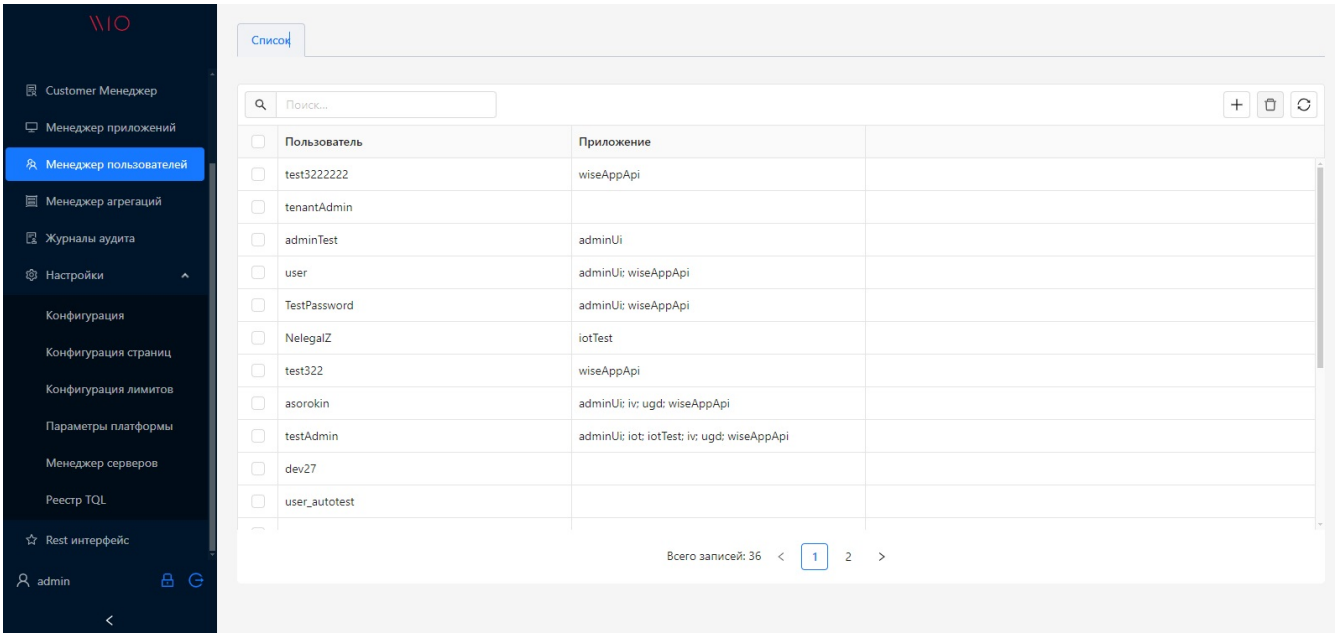

Рисунок 47. Реестр учётных записей пользователей

Описание интерфейса

Для доступа к компоненту **«Пользователи»** необходимо в основном меню веб-интерфейса пользователя выбрать пункт **Менеджер пользователей**.

> Интерфейс компонента «Менеджер пользователей» состоит из списка учётных записей и карточки учётной записи пользователя

Доступные функции:

▼ *Создание|Редактирование учётной записи пользователя*

Для создания новой учётной записи пользователя необходимо:

1. Открыть список учетных записей пользователей **Менеджер пользователей**.
2. В списке учётных записей нажать на кнопку **[Создать]** (см. рисунок: Переход к созданию учетной записи пользователя), после чего в новой вкладке откроется карточка учетной записи пользователя. Редактирование учётной записи осуществляется при двойном «клике» по записи в списке учётных записей.
3. В карточке пользователя (см. рисунок: Карточка учетной записи пользователя) необходимо заполнить требуемый атрибут "Логин".
4. Чекбокс **[Активный]** должен быть установлен в состояние ВКЛЮЧЕНО.
5. Чекбокс **[Изменить пароль при следующем входе]** может быть установлен во время редактирования учетной записи существующего пользователя, в результате чего он должен будет сменить пароль во время следущего входа в систему.
6. В поле **[Срок действия учетной записи]** может быть задан временной интервал действия учетной записи (дата/время начала и дата/время окончания действия).
7. Нажать кнопку **[Пароль]**.
8. В окне установки пароля дважды ввести новый пароль и нажать кнопку

[Применить] (см. рисунок: Установка пароля пользователя); пароль должен удовлетворять требованиям политики безопасности.

9. В карточке пользователя нажать кнопку **[Сохранить]**.

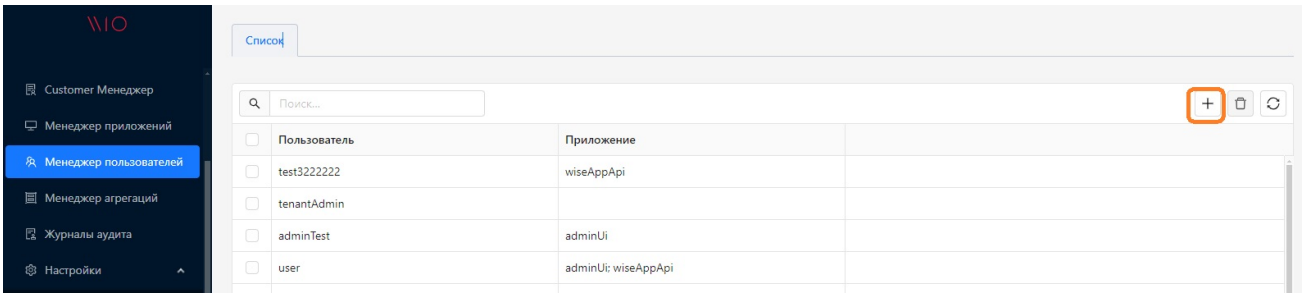

Рисунок 48. Переход к созданию учетной записи пользователя

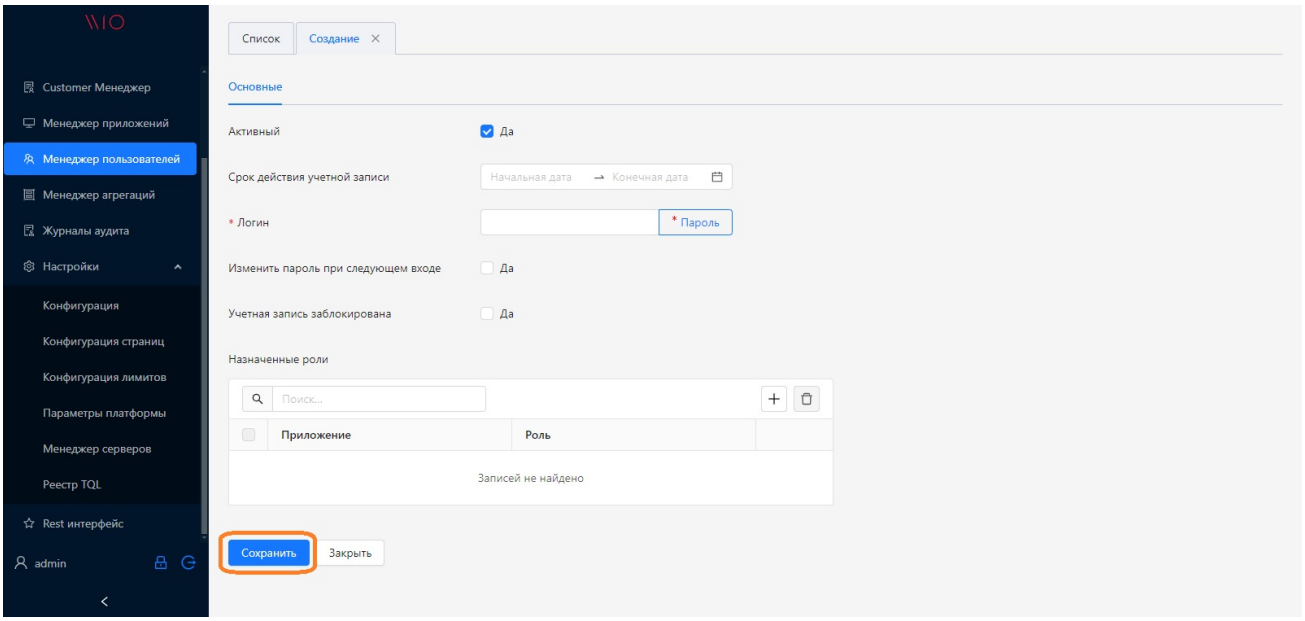

Рисунок 49. Карточка учетной записи пользователя

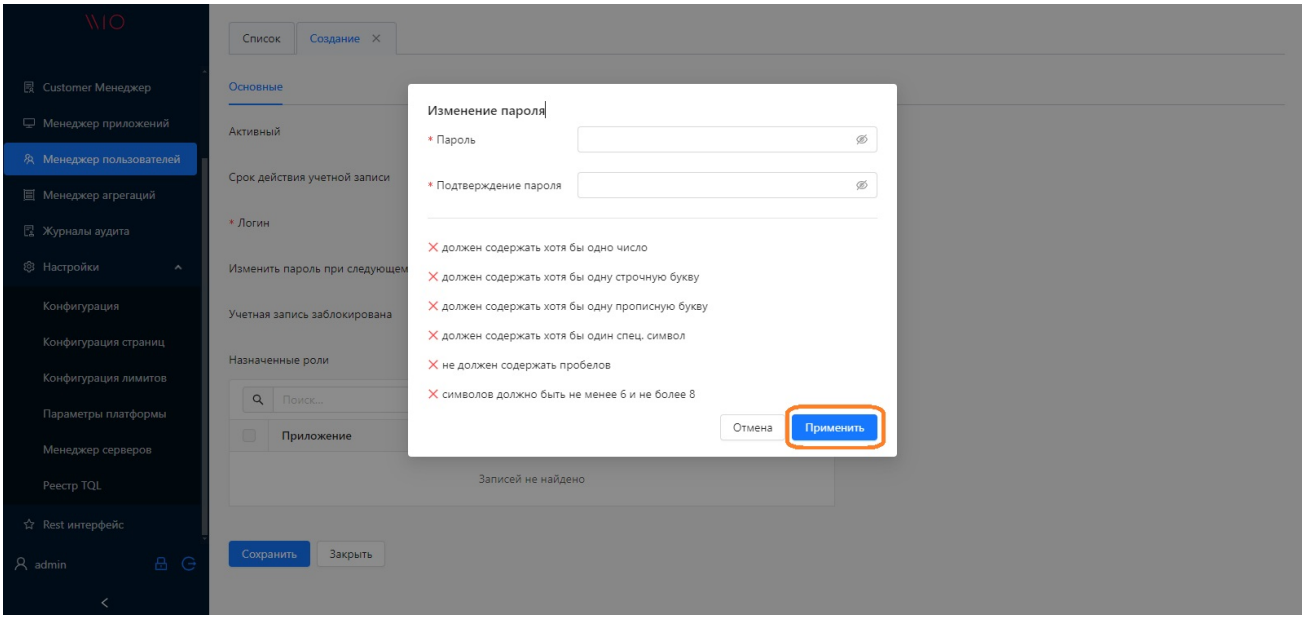

Рисунок 50. Установка пароля пользователя

▼ *Управление блокировкой учетной записи пользователя*

Согласно политике безопасности, пользователь может быть заблокирован - т.е. не иметь

возможности войти в систему (например, когда пользователь ввел неправильный пароль 3 раза). В таком случае чекбокс **[Учетная запись заблокирована]** будет иметь состояние **[ВКЛЮЧЕНО]** (см. рисунок: Карточка учетной записи пользователя). Для разблокировки такой учетной записи администратор должен во время ее редактирования установить данный чекбокс в состояние **[ВЫКЛЮЧЕНО]**, и далее нажать кнопку **[Сохранить]**.

▼ *Назначение пользователю прав доступа к приложению*

Назначение прав доступа пользователю к приложению осуществляется путем привязывания роли доступа к учетной записи пользователя. Пользователю может быть назначена единственная роль доступа в конкретном приложении. Для привязывания роли доступа к учетной записи пользователя следует перейти в **Менеджер пользователей** и далее выполнить действия:

1. В списке учетных записей выбрать двойным "кликом" мыши нужного пользователя, в результате чего будет открыта карточка пользователя
2. Нажать кнопку **[Назначить роль]** (см. рисунок: Переход к назначению роли пользователю),
3. В окне [Назначение роли] выбрать из выпадающего списка требуемые приложение и роль, и далее нажать кнопку **[Добавить]** (см. рисунок: Назначение роли пользователю)
4. В карточе пользователя нажать кнопку **[Сохранить]**

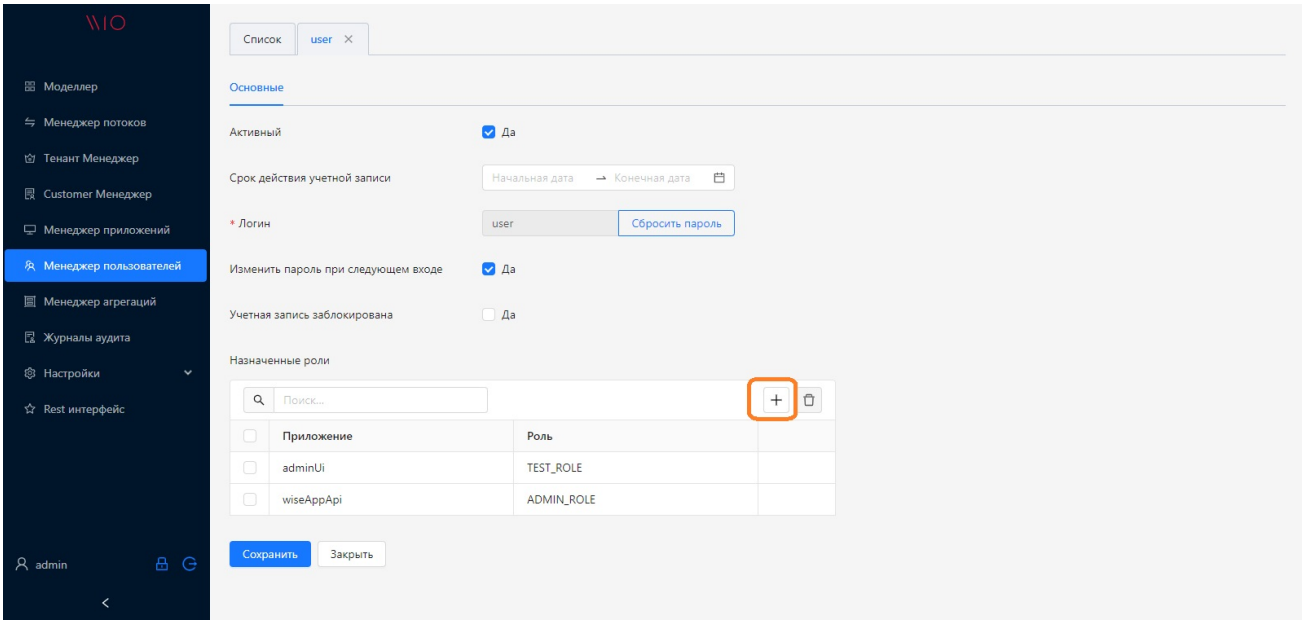

Рисунок 51. Переход к назначению роли пользователю

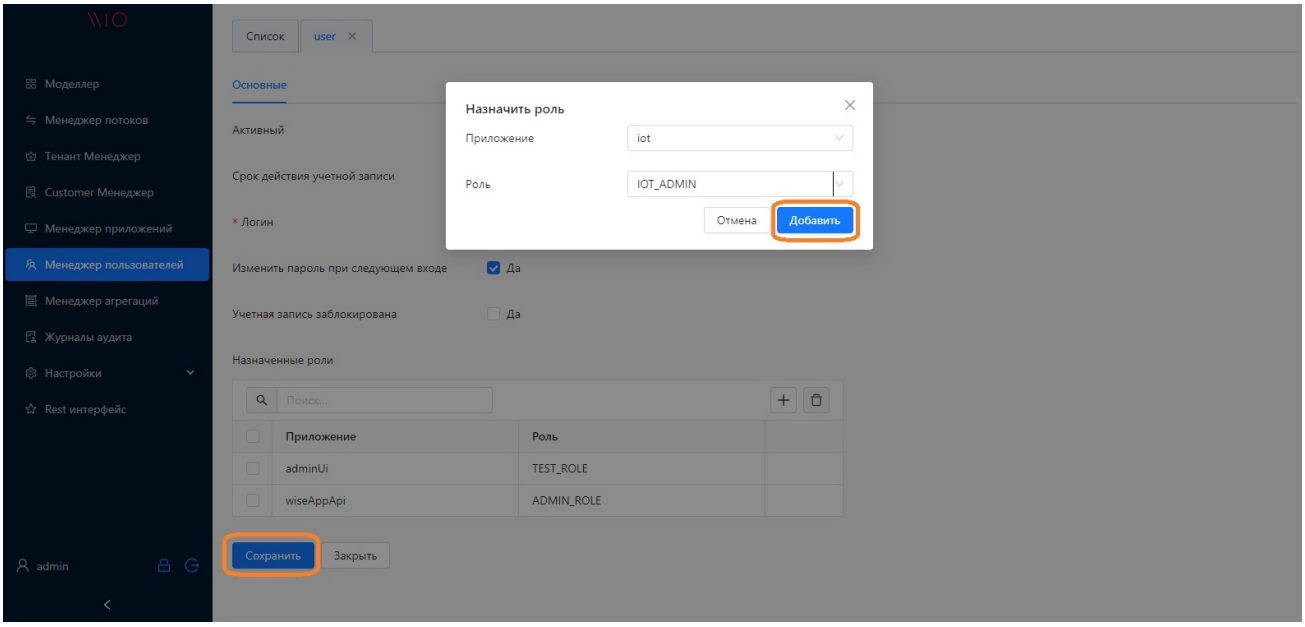

Рисунок 52. Назначение роли пользователю

▼ *Удаление прав доступа пользователя к приложению*

Удаление прав доступа пользователя к приложению осуществляется путем отвязывания роли доступа от учетной записи пользователя. Для отвязывания роли доступа от учетной записи пользователя следует перейти в **Менеджер пользователей** и далее выполнить действия:

1. В списке учетных записей выбрать двойным кликом мыши нужного пользователя, в результате чего будет открыта карточка пользователя
2. Выбрать чекбоксом приложения, доступ к которым необходимо удалить, и нажать кнопку **[Отменить роль]** (см. рисунок: Переход к отвязыванию роли от учетной записи пользователя),
3. В модальном окне подтверждения нажать на кнопку **["Удалить"]**(см. рисунок: Подтверждение отвязывания роли от учетной записи пользователя)

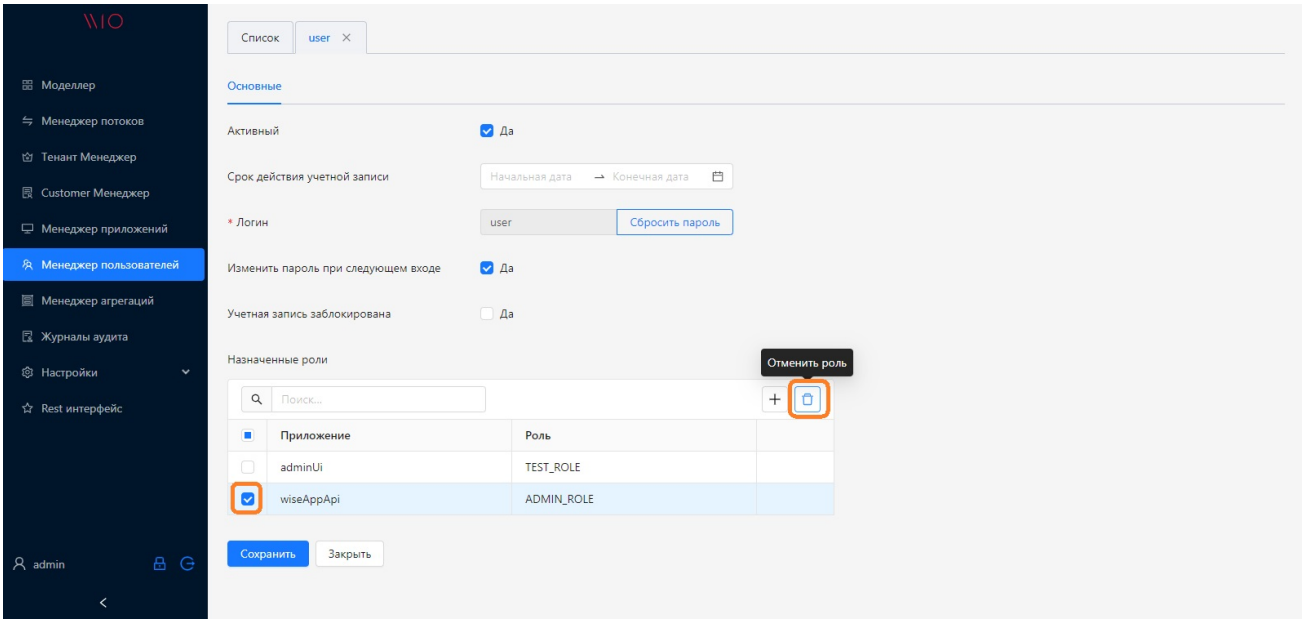

Рисунок 53. Переход к отвязыванию роли от учетной записи пользователя

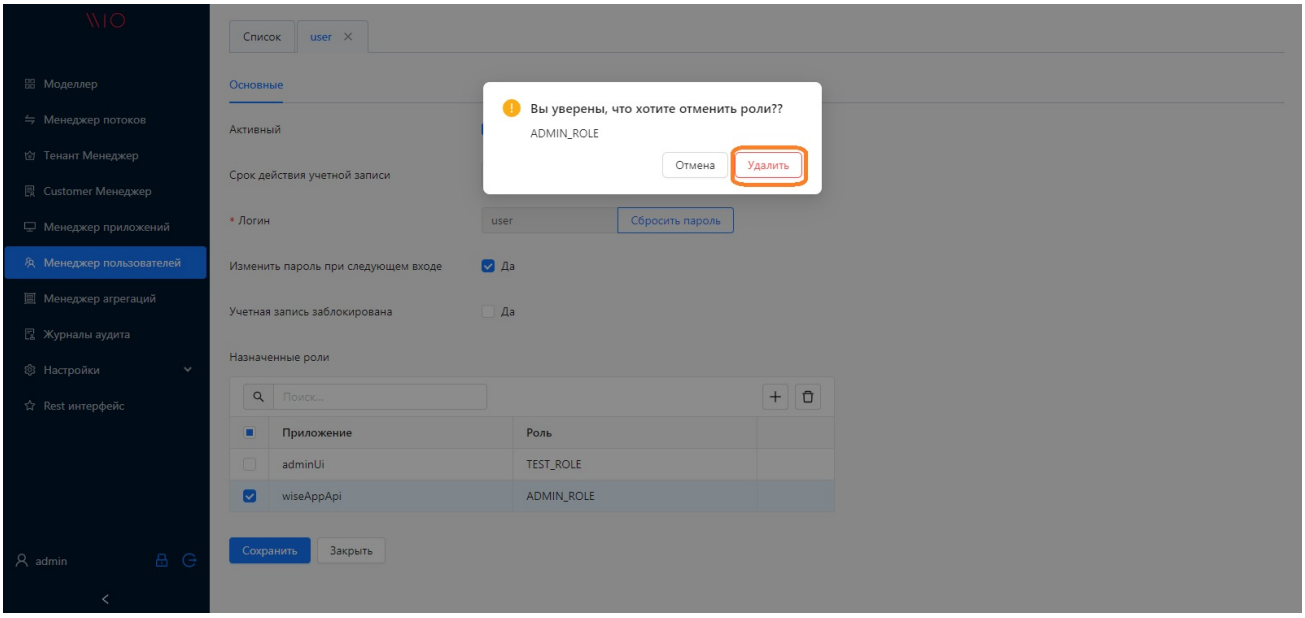

Рисунок 54. Подтверждение отвязывания роли от учетной записи пользователя

▼ *Удаление учётной записи пользователя*

Для удаления учетной записи пользователя следует перейти в **Менеджер пользователей** и далее выполнить действия:

1. В списке учетных записей выбрать чекбоксом нужного пользователя
2. Нажать кнопку **[Удалить]** (см. рисунок: Переход к удалению учетной записи пользователя),
3. В модальном окне подтверждения нажать на кнопку **["Удалить"]** (см. рисунок: Подтверждение удаления учетной записи пользователя)

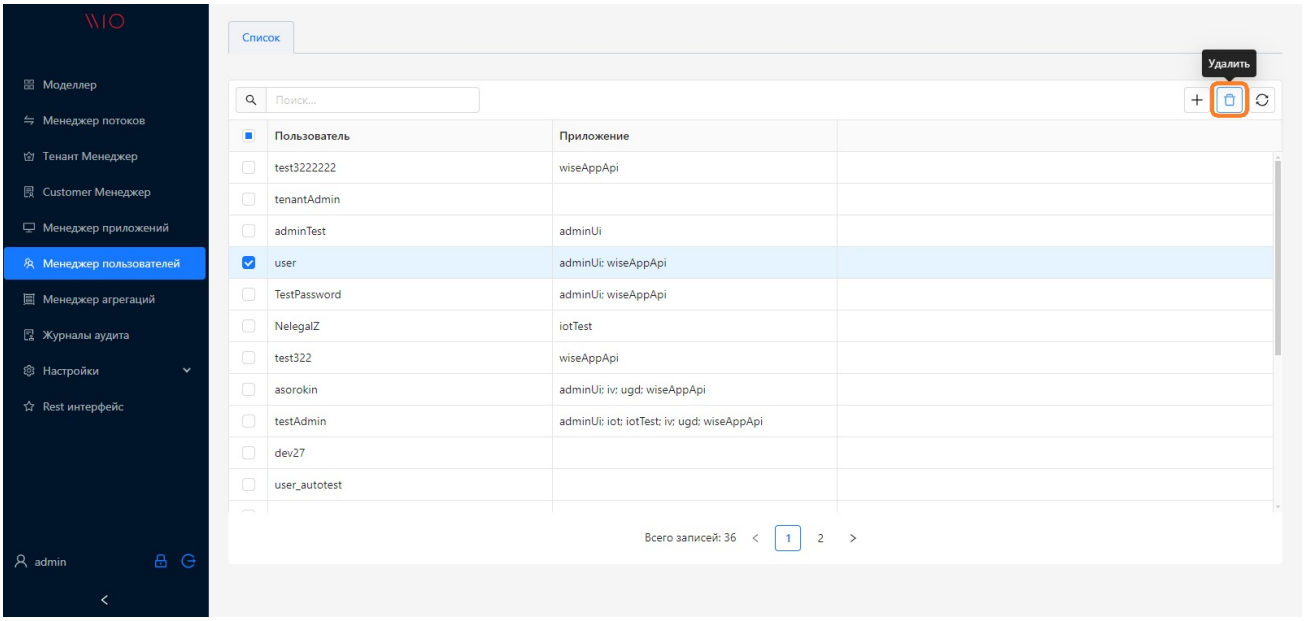

Рисунок 55. Переход к удалению учетной записи пользователя

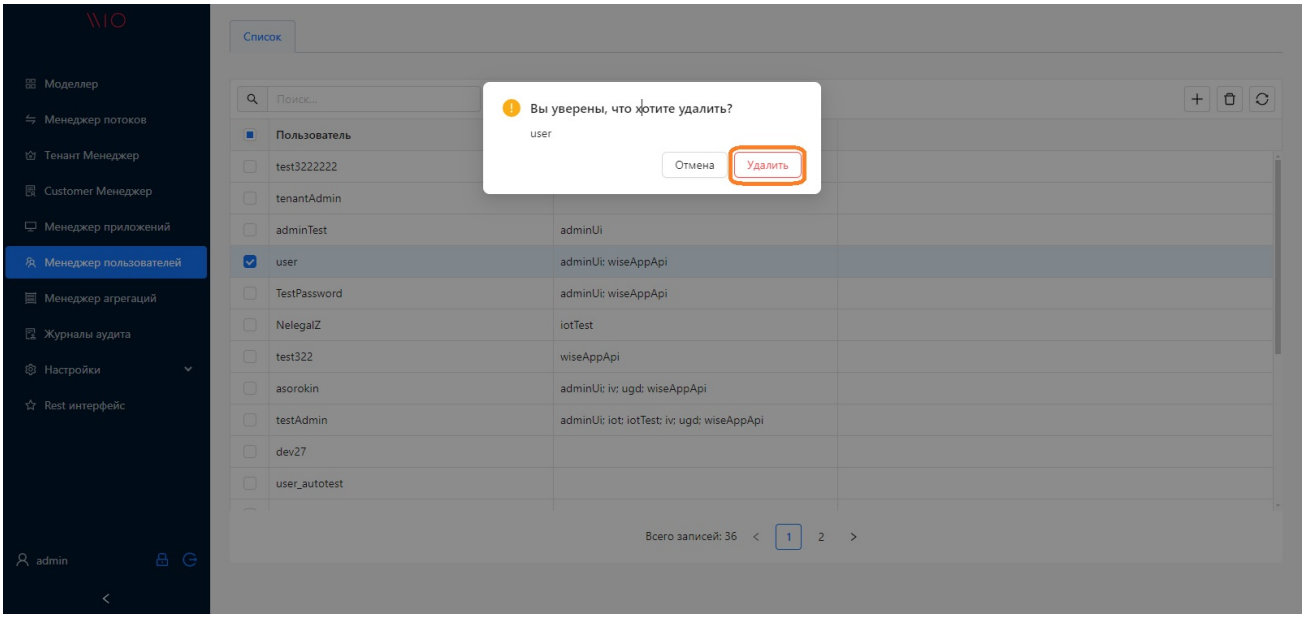

Рисунок 56. Подтверждение удаления учетной записи пользователя

8. REST интерфейс

Платформа позволяет использовать *технологическую надстройку* для вызова и тестирования отдельных компонентов *без использования WEB-интерфейса* посредством технологического REST-интерфейса.

Описание интерфейса

Для запуска REST-интерфейса следует перейти в панели навигации в раздел **«Rest интерфейс»**. Будет отображено окно с Rest-интерфейсом (см. рисунок: Rest-интерфейс)

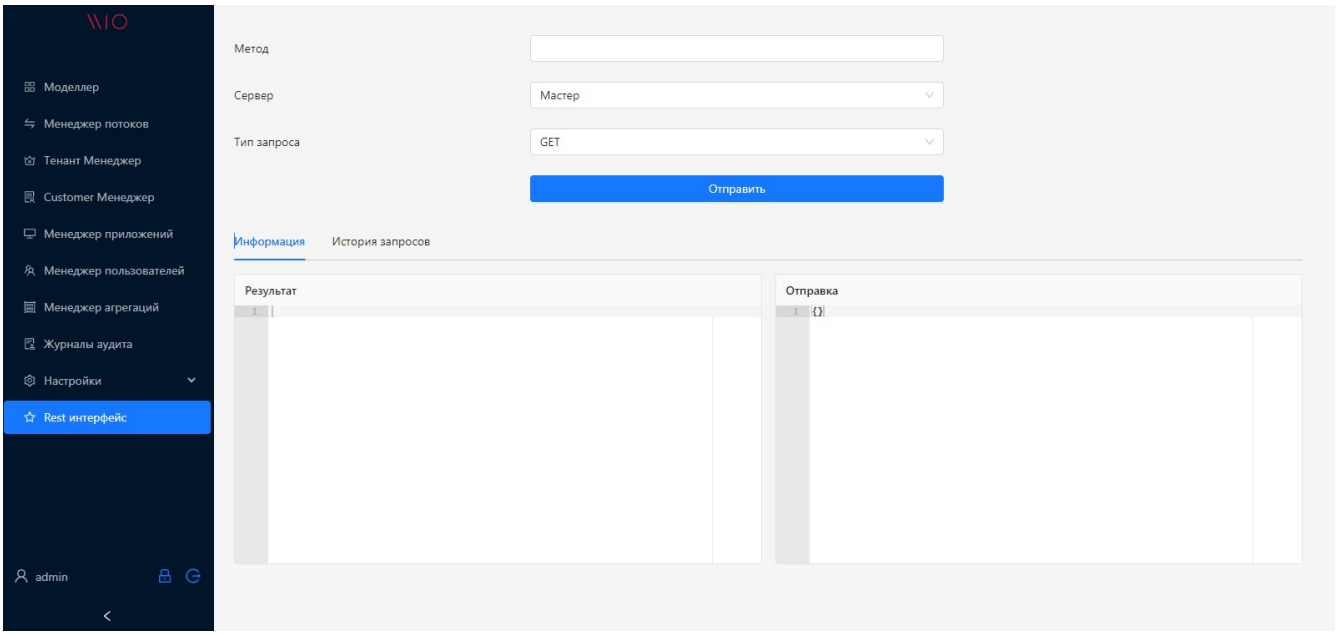

Рисунок 57. Rest-интерфейс

Описание интерфейса:

- **«Метод»** - необходимо указать метод для его исполнения (описание API выходит за рамки данного руководства);
- **«Тип запроса»** - выбирается один из типов запросов для работы с ресурсом метода:
  - `GET` - получение ресурса;
  - `POST` - создание ресурса;
  - `PUT` - обновление ресурса;
  - `DELETE` - удаление ресурса;
- **«Информация»** - отображаются сведения после обработки запроса;
- **«История запросов»** - отображается история запросов.

Пример работы с интерфейсом

Рассмотрим пример получения всех экземпляров Tenant.

Для этого в Rest-интерфейсе необходимо задать следующие параметры:

```
Метод: /rest/tenant/find
Тип запроса: PUT
```

Далее нажать на кнопку **[Отправить]**.

Результат выполнения запроса отобразится в области с наименованием «Информация» (см. рисунок: Пример Rest запроса).

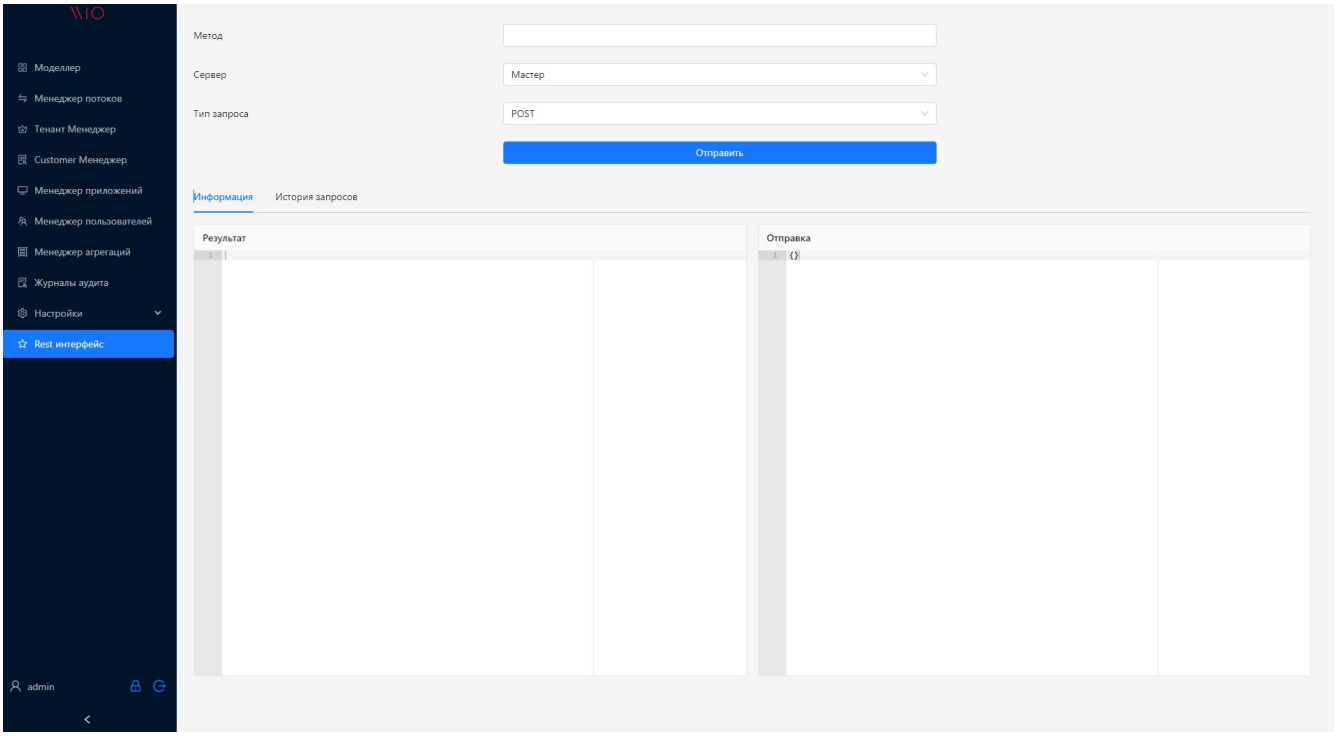

Рисунок 58. Пример Rest запроса

9. Информационная безопасность

Авторизация пользователей

В основе технической реализации **Платформы** в контексте выполнения процессов авторизации и доступа пользователей к различным ресурсам используется схема (протокол) авторизации **OAuth 2.0** (см. рисунок: Схема процесса авторизации)

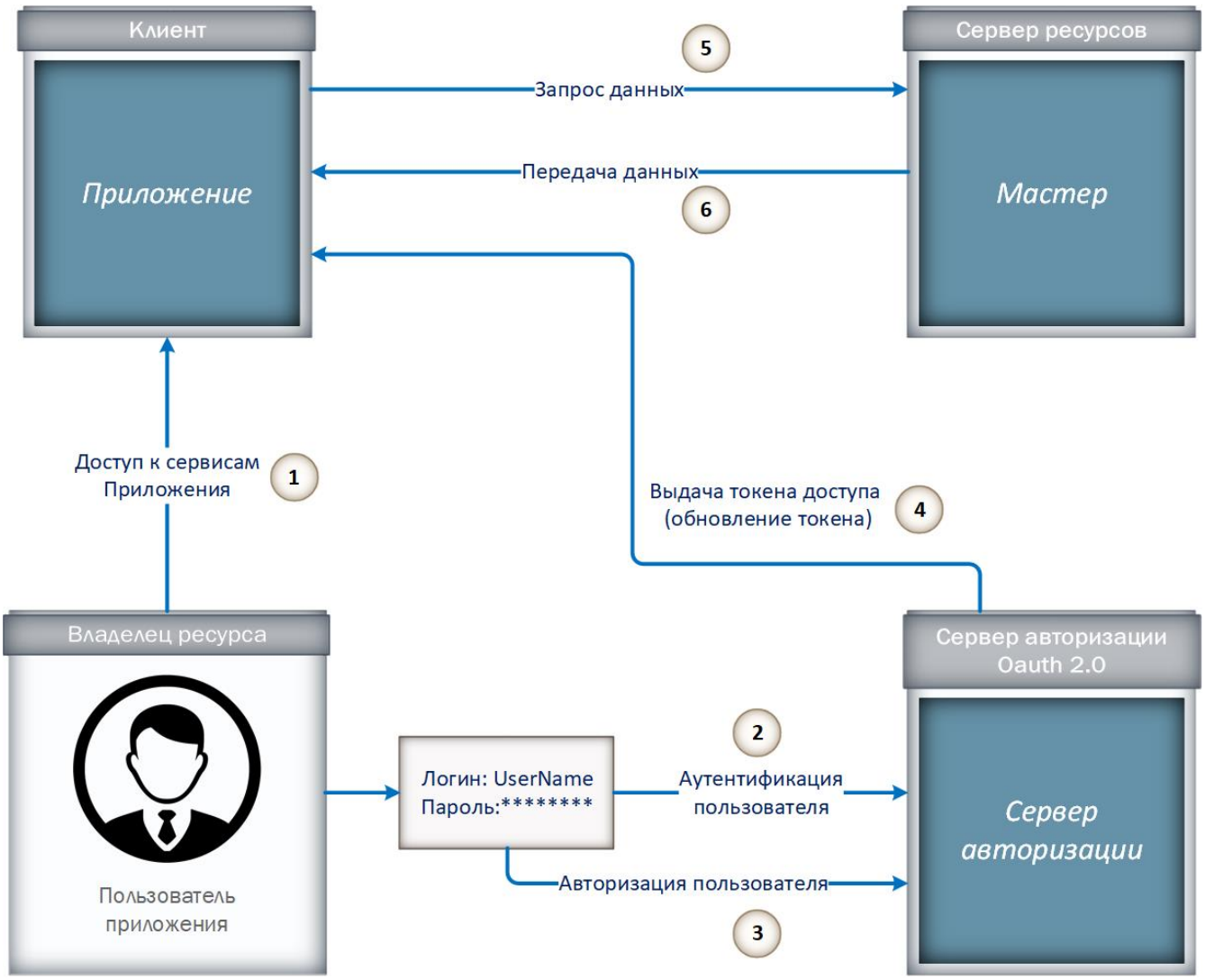

Рисунок 59. Схема процесса авторизации

Для построения указанной схемы в составе **Платформы** существует соответствующий *Сервер авторизации*, который осуществляет выпуск токенов для клиентских приложений после успешной аутентификации и авторизации *Владельца ресурсов*.

В качестве **Клиента** выступает *Приложение* (приложение конечного пользователя), которое осуществляет доступ к защищенным ресурсам от имени *Владельца ресурсов* (в данном случае от имени Пользователя приложения).

Приложение, в свою очередь, обращается к т.н. **Мастеру**, который является в данном случае *Сервером ресурсов*, т.е. может предоставлять для *Приложения* защищаемые ресурсы и обладающий возможностью получения и формирования ответа на запросы к защищаемым ресурсам посредством использования токена доступа.

Для использования Приложения, пользователь вводит *логин* и *пароль* в соответствующей форме. Авторизация пользователя производится на Сервере авторизации.

В результате авторизации Пользователя на Сервере авторизации формируется *уникальный токен* (access token), который является «пропуском» к необходимым ресурсам при работе Пользователя с Приложением.

Используя токен доступа Приложение получает право доступа к соответствующим ресурсам на Сервере ресурсов (на Мастере Платформы).

При этом токен имеет следующие особенности:

- Ограничен по времени действия.
- Существует независимо от пароля и логина пользователя.
- Позволяет предоставлять доступ только к части данных.
- Техническое решение может быть расширено функционалом отзыва токена для немедленного «отключения» пользователя» и прекращения доступа к ресурсам.
- По токену можно фильтровать доступные данные (Клиент запрашивает список разрешений, которые необходимы ему для работы, а Сервер ресурсов решает выдать такие права или нет).

> Описанная выше технология обеспечивает ограничение прав доступа единообразно, на уровне Пользовательского интерфейса Приложения и на уровне REST API. Таким образом злоумышленник не может изменить свои полномочия, например, сформировав соответствующий REST-запрос и обеспечить таким образом право доступа к неавторизованным для него ресурсам.

Обеспечение безопасного хранения паролей

Для безопасного хранения паролей в базе данных в приложении реализована технология хеширования паролей.

Для хеширования используется адаптивная криптографическая хэш-функция `bcrypt` — адаптивная криптографическая хеш-функция формирования ключа.

Правила формирования паролей

Политика безопасности задает следующие правила для паролей учетных записей пользователей:

- Число символов: от 6 до 8
- Пароль должен обязательно содержать символы в верхнем и нижнем регистре
- Пароль должен содержать спецсимволы из числа: @#$%^&+=

- Пароль должен содержать цифры
- Пароль не должен содержать пробелы
- Задаваемый пароль не должен совпадать с одним из трех последних паролей учетной записи

Допускается не более трех подряд ошибочных вводов неверного пароля, если такое происходит - учетная запись блокируется. Разблокировка учетной записи осуществляется адмнистратором платформы.

Каждые 90 дней система требует от пользователя сменить пароль на новый.

10. Дополнительные функции

Изменение пароля учетной записи

После успешной авторизации в интерфейсе есть возможность *изменить пароль* учетной записи.

Для этого в навигационном меню необходимо выбрать кнопку **[Изменение пароля]** (см. рисунок: Изменение пароля).

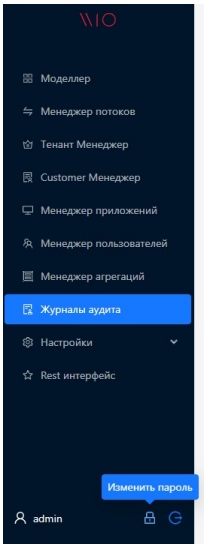

Рисунок 60. Изменение пароля

В появившемся окне указать старый и новый пароли и нажать на кнопку **[Применить]** (см. рисунок: Ввод пароля).

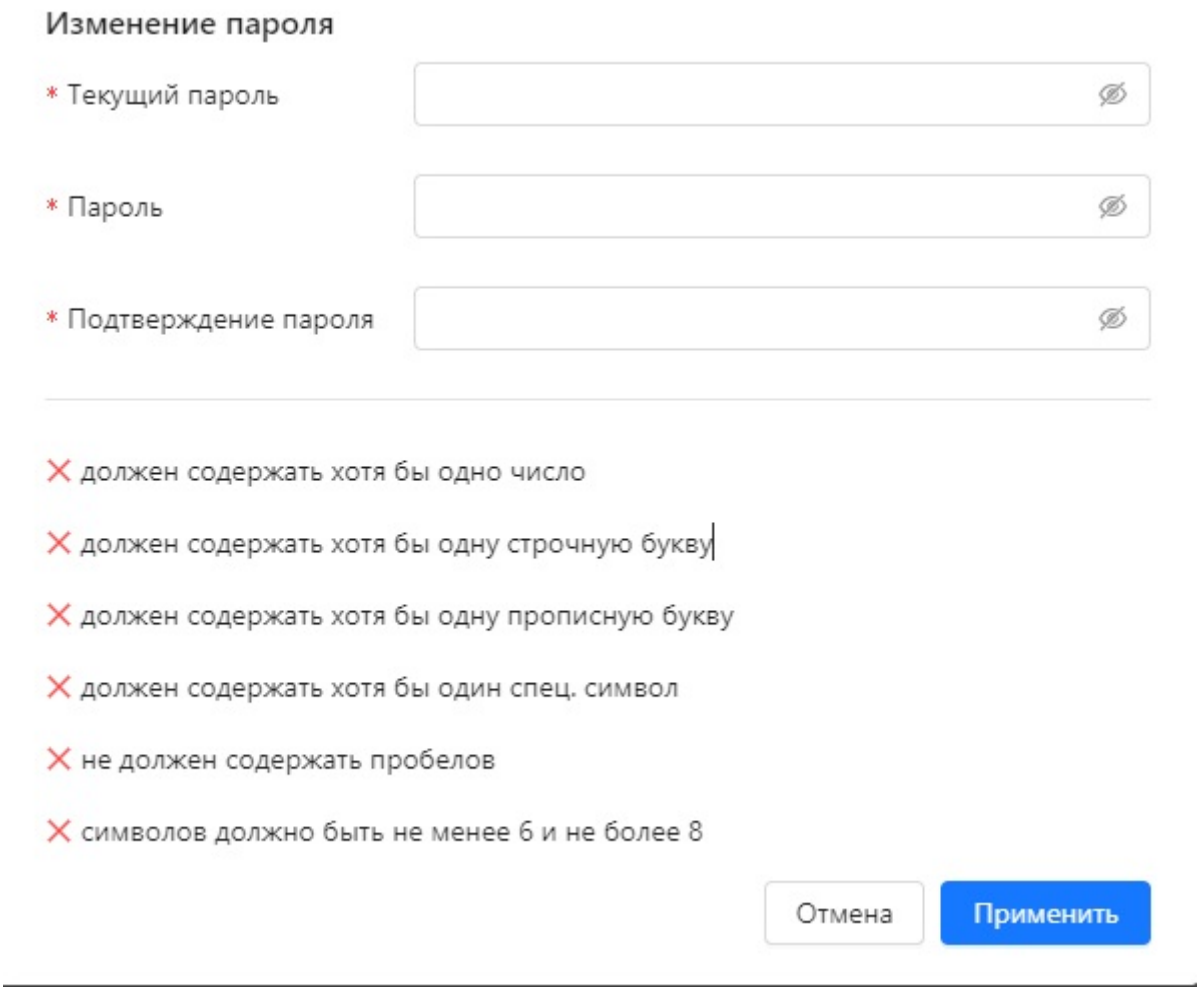

Рисунок 61. Ввод пароля

Выход из интерфейса

Для выхода из интерфейса необходимо в навигационном меню выбрать кнопку **[Выход]** (см. рисунок: Выход из интерфейса).

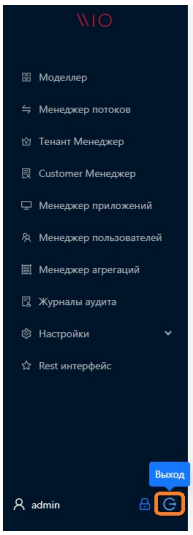

Рисунок 62. Выход из интерфейса

11. Список изменений документа

- Удалены разделы "Менеджер Агрегации", а также "Конфигурация страниц", "Конфигурация лимитов", "Параметры платформы" в "Настройках"
- Актуализированы все разделы, в том числе обновлены скриншоты