Руководство пользователя платформы по автоматизации VERSiUM

Раздел 1. Интерфейс и навигация

1. Общее описание платформы

VERSiUM — это платформа автоматизации реагирования на инциденты информационной безопасности (SOAR), разработанная с учётом требований к отечественному программному обеспечению. Система предназначена для:

- Управления инцидентами информационной безопасности;
- Автоматизации действий по стандарту NIST;
- Интеграции с внешними системами (SIEM, EDR, AD, брандмауэры и др.);
- Администрирования пользователей, ролей и настроек.

Интерфейс платформы полностью локализован на русский и английский языки. Архитектура системы соответствует современным требованиям к юзабилити, безопасности и импортозамещению.

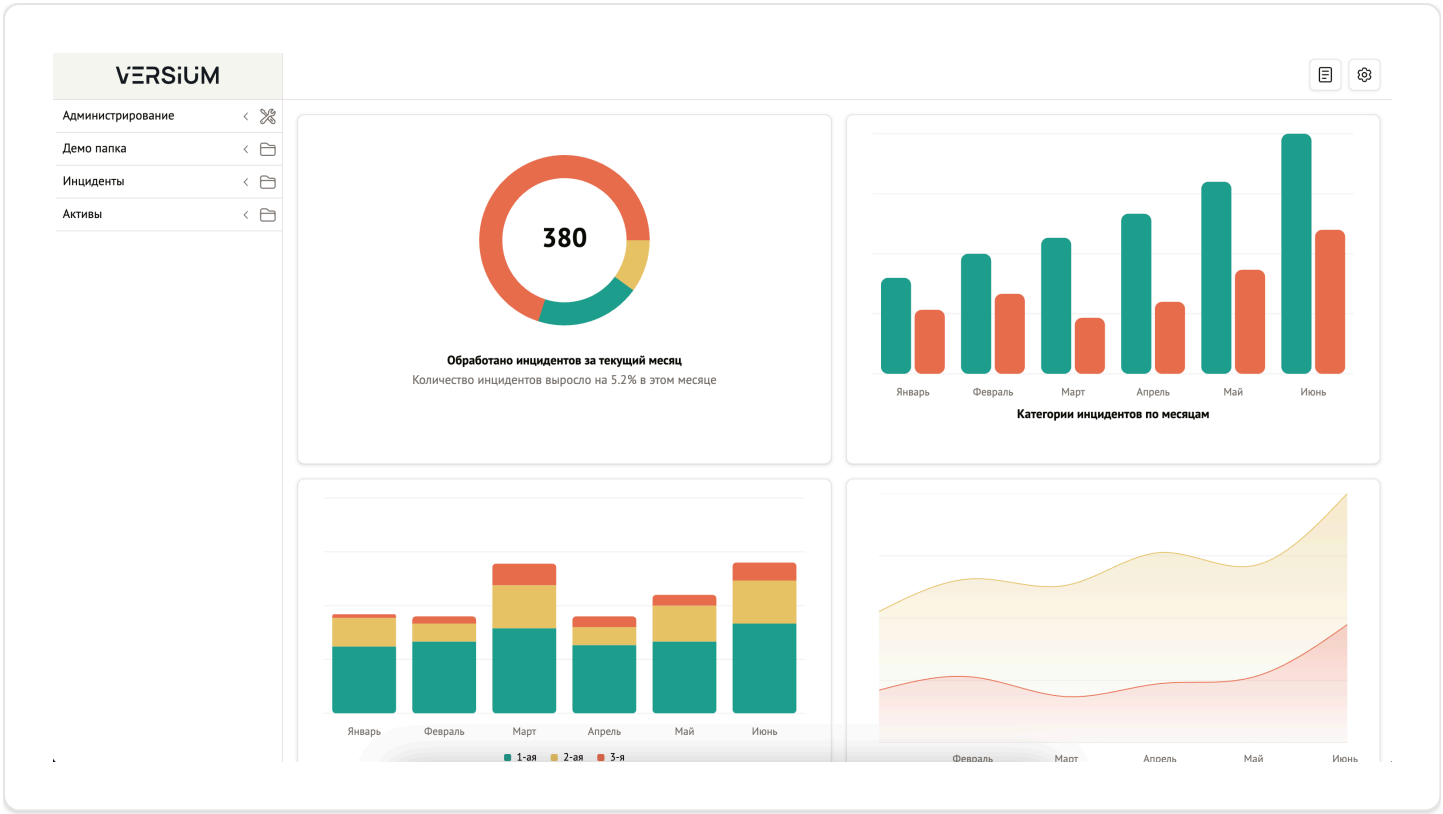
2. Стартовый экран и основные элементы интерфейса

После успешной аутентификации пользователь попадает на главную страницу, которая служит центральной точкой доступа к функциональным возможностям платформы.

2.1. Область дашбордов

Центральная часть интерфейса предназначена для размещения аналитических дашбордов. Это гибкое пространство, в котором администраторы и аналитики могут:

- Отображать сводные метрики по инцидентам, активам и действиям;
- Располагать любое количество виджетов (графиков, таблиц, счётчиков);
- Настраивать дашборды через раздел «Виджеты» в администрировании.



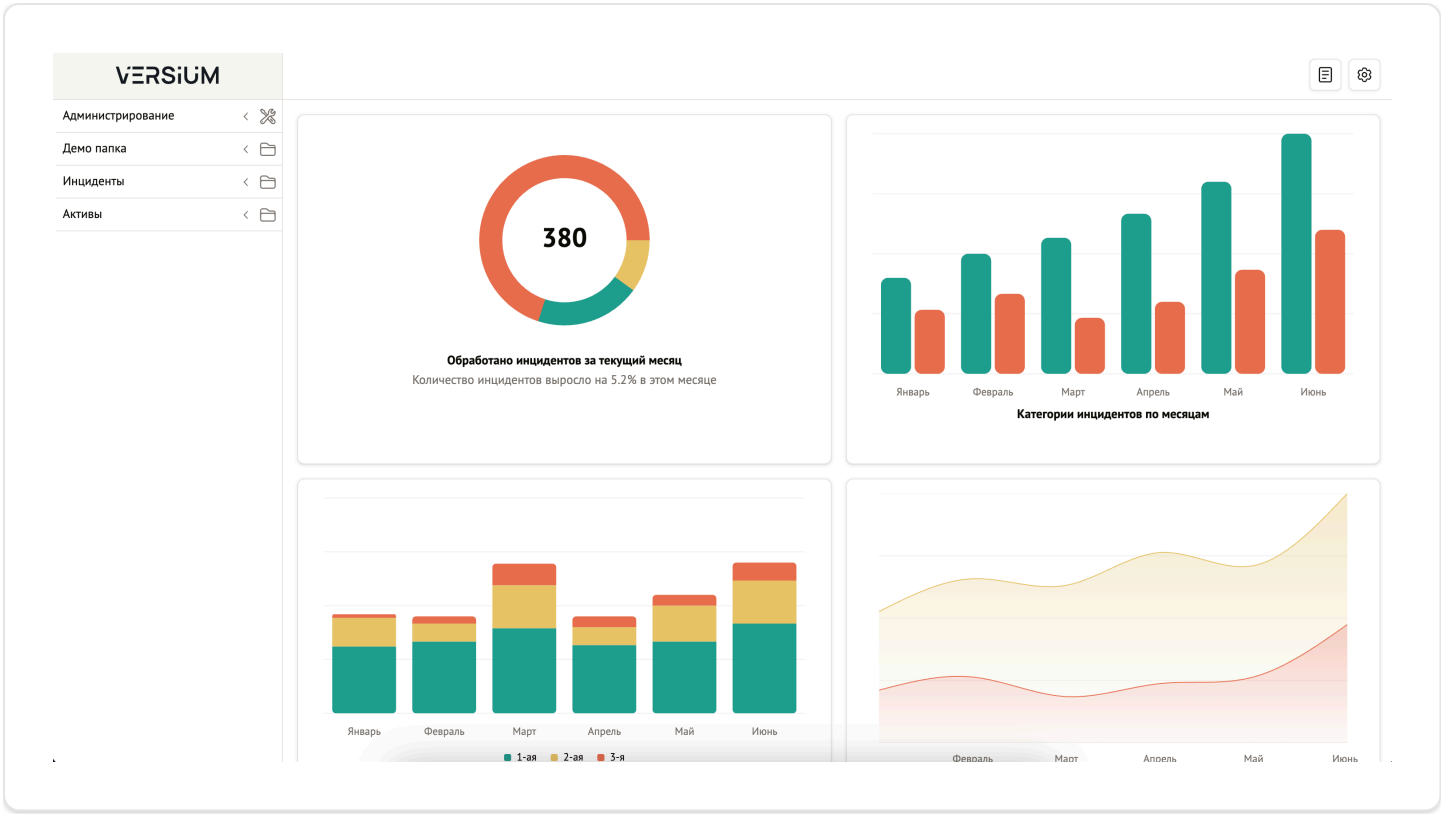
Все дашборды создаются и настраиваются в разделе **Администрирование → Виджеты**.

2.2. Боковая панель навигации

Слева расположена боковая панель навигации, обеспечивающая быстрый доступ к ключевым разделам платформы.

Типичные пункты меню:

- **Инциденты** — список всех зарегистрированных инцидентов;
- **Активы** — база ИТ-активов;
- **Администрирование** — доступ к настройкам системы (только для администраторов).

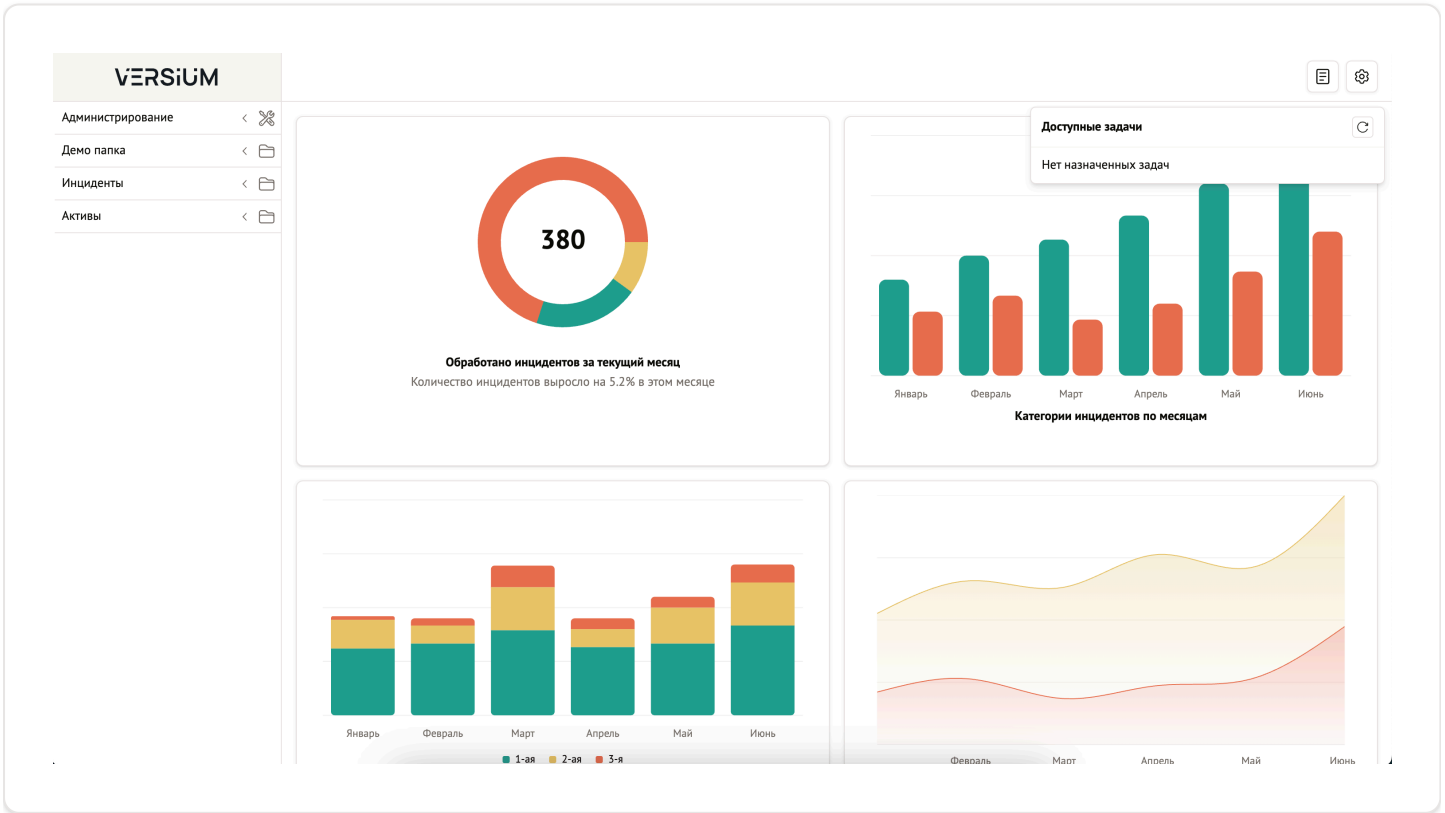


2.3. Панель быстрого доступа к задачам

В правом верхнем углу интерфейса расположена иконка в виде списка. При нажатии открывается выпадающее окно «Доступные задачи».

Функциональность:

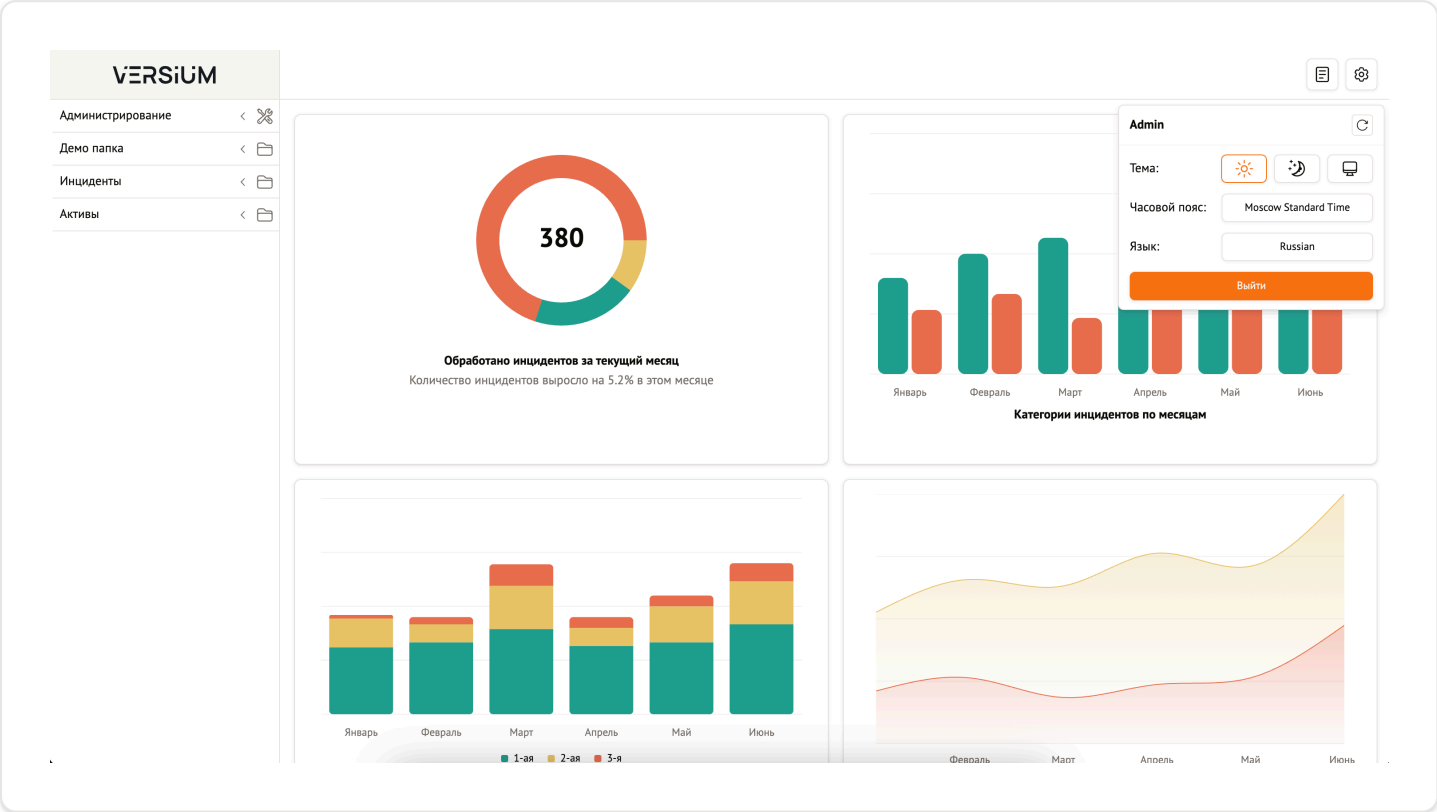
- Отображает задачи, назначенные текущему пользователю;
- Позволяет быстро перейти к выполнению действия;
- Поддерживает обновление списка вручную.



2.4. Профиль пользователя

Рядом с панелью задач расположена иконка настроек. При клике открывается меню профиля, где пользователь может:

- Выбрать тему интерфейса: светлая, тёмная или системная;
- Установить часовой пояс (по умолчанию — MSK);
- Изменить язык интерфейса (поддерживается русский и английский);
- Выйти из системы.

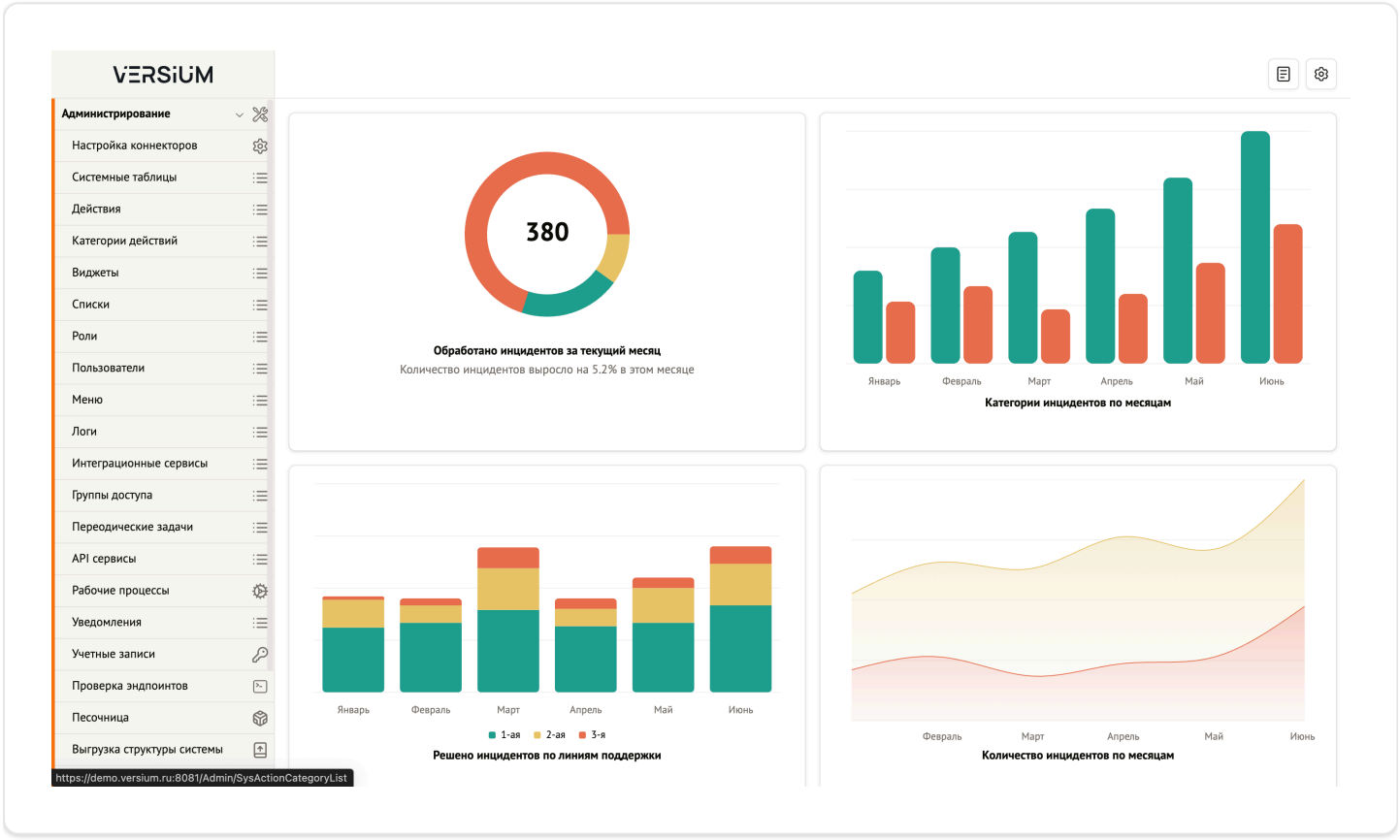


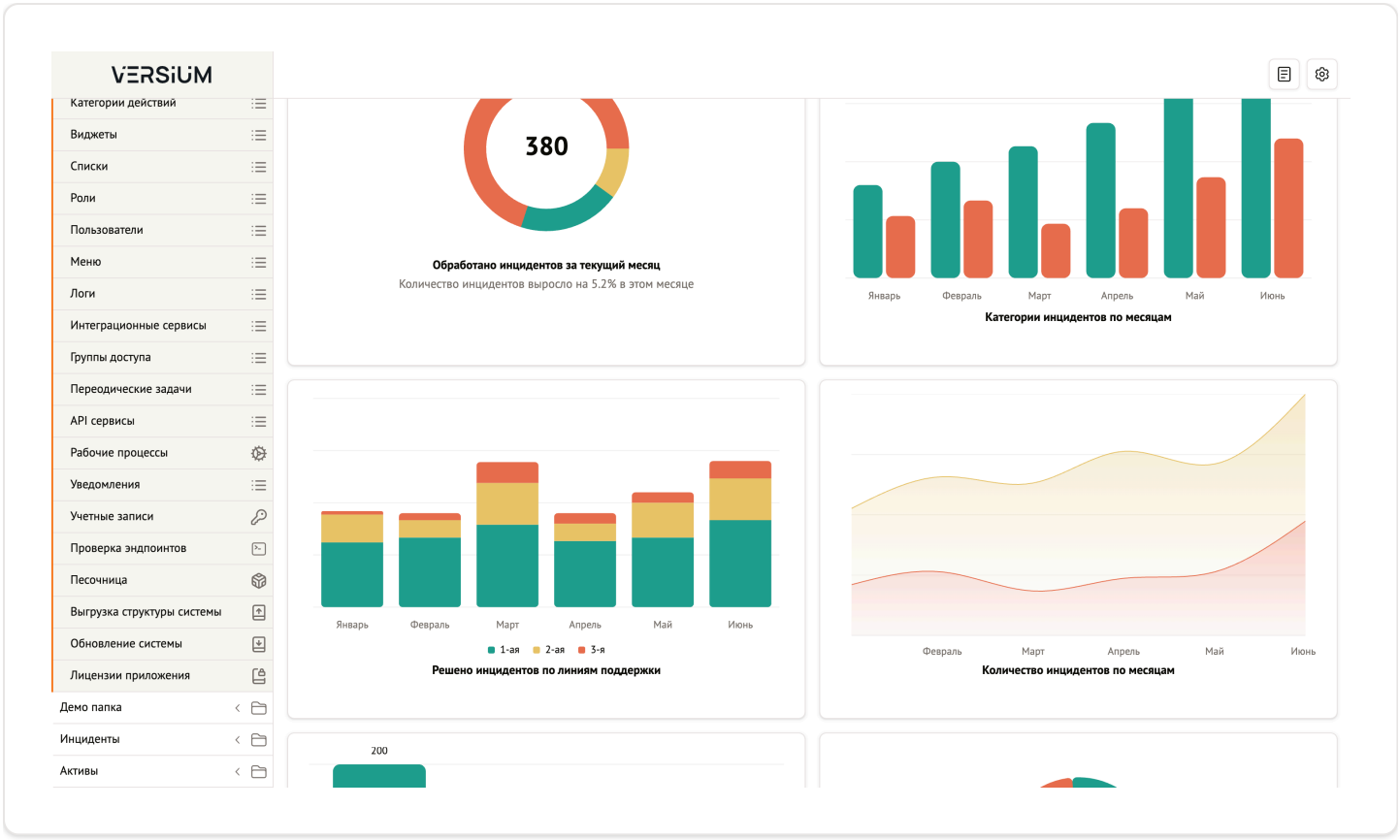
2.5. Раздел администрирования

Пункт «Администрирование» в боковом меню предоставляет доступ к полному набору настроек платформы. Доступен только пользователям с ролью администратора.
Функциональные блоки администрирования:

- Настройка коннекторов — управление внешними источниками данных (API, базы, файлы);
- Системные таблицы — работа с базовыми справочниками и структурой данных;
- Действия / Категории действий — определение бизнес-операций и их классификации;
- Виджеты — создание, редактирование и управление шаблонами виджетов;
- Списки — управление списками значений (например, статусы, приоритеты);
- Роли — настройка ролей пользователей и прав доступа;
- Пользователи — управление учётными записями, паролями, активностью;
- Меню — конфигурация пользовательского интерфейса — настройка доступных пунктов меню;
- Логи — просмотр событий системы, ошибок, действий пользователей;

- Интеграционные сервисы — настройка и мониторинг внешних интеграций (Webhooks, API, очереди);
- Группы доступа — создание групп пользователей для удобного управления правами;
- Периодические задачи — настройка фоновых задач (например, резервное копирование, обновление данных);
- API сервисы — управление API-интерфейсами платформы (ключ, ограничения, документация);
- Рабочие процессы — настройка автоматизированных рабочих потоков (workflow);
- Уведомления — настройка каналов оповещений;
- Учетные записи — управление системными аккаунтами (например, для интеграций);
- Проверка эндпоинтов — тестирование доступности внутренних и внешних URL;
- Песочница — модуль для безопасного тестирования функций без влияния на основную систему;
- Выгрузка структуры системы — экспорт всей конфигурации в формате JSON/XML;
- Обновление системы — проверка и установка обновлений платформы;
- Лицензии приложения — просмотр и управление лицензиями (ключами, сроками действия).





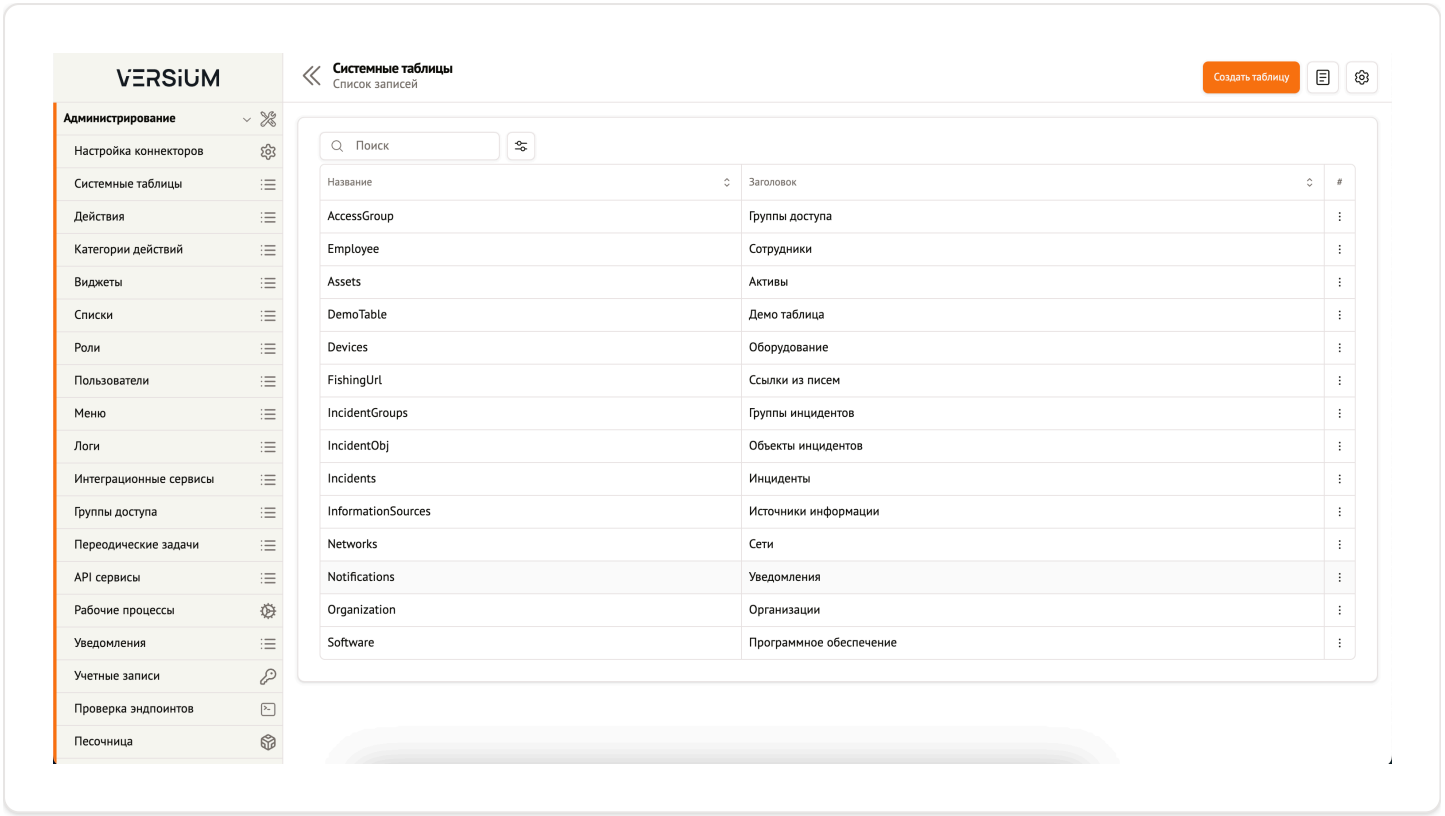
Все данные разделы доступны только авторизованным пользователям с соответствующими правами.

Раздел 2. Настройка коннекторов и системных таблиц

2.1. Системные таблицы

Раздел «Системные таблицы» представляет собой центральный модуль управления структурой данных платформы. Он позволяет администраторам:

- Создавать и редактировать базовые справочники;
- Модифицировать состав полей объектов;
- Настраивать пользовательские интерфейсы (экраны списка и формы);
- Расширять функционал системы без программирования.

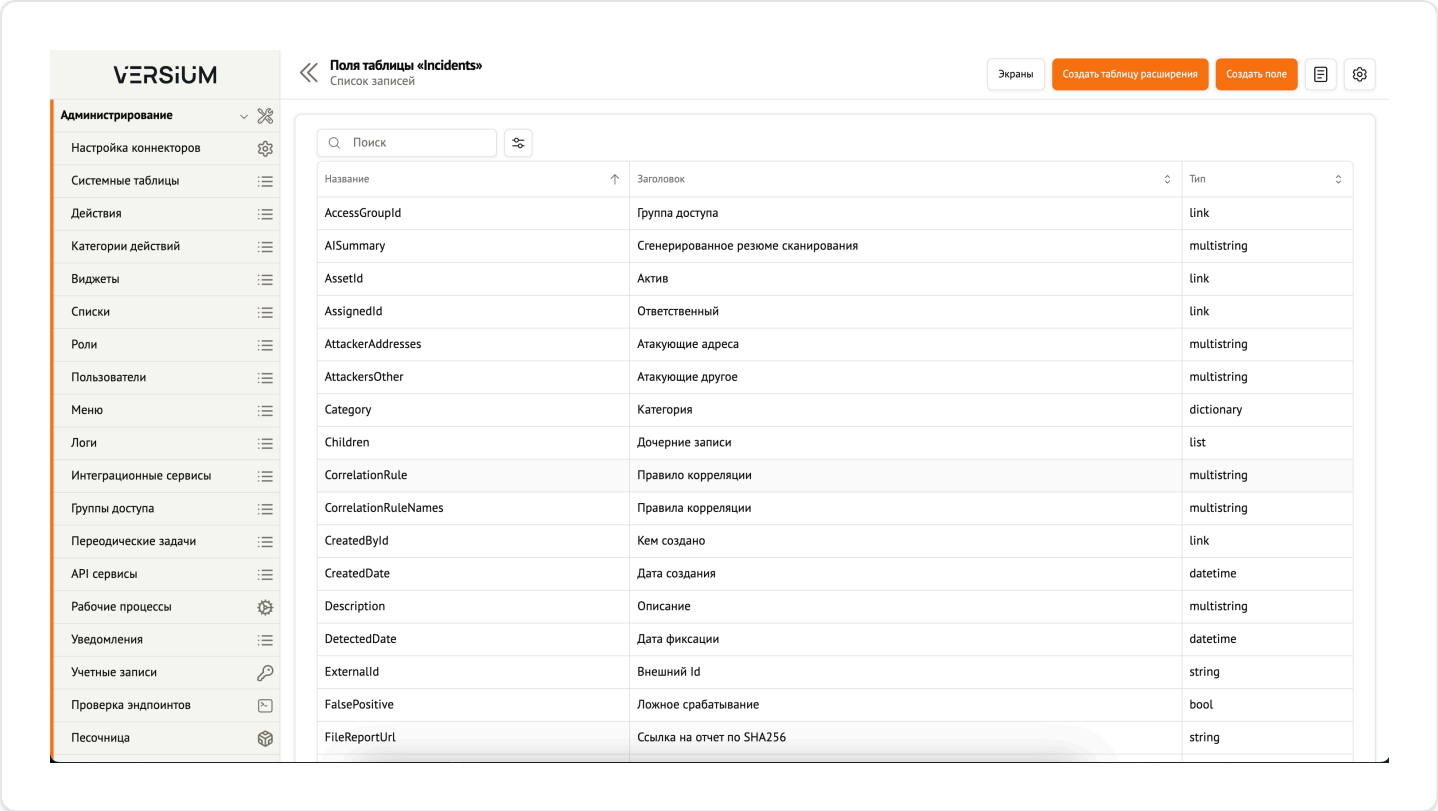


Все изменения в системных таблицах влияют на поведение всей платформы и требуют внимательного подхода.
Кнопка **«Создать таблицу»** позволяет добавить новую пользовательскую таблицу с произвольной структурой.

2.2. Поля таблицы «Incidents»

После перехода в таблицу «Incidents» открывается список её полей — это внутренняя структура данных, определяющая, какие параметры будет содержать инцидент.
Поля таблицы не перечисляются в документации, так как их состав может изменяться в зависимости от конфигурации организации. Однако типовые категории включают:

- Идентификаторы и номера;
- Временные метки (создание, изменение, события);
- Ссылки на активы, пользователей, группы;
- Классификационные поля (категория, тип, приоритет, статус);
- Описательные и аналитические поля (описание, решение, вердикт);
- Интеграционные поля (внешние ID, хэши, ссылки на отчёты);
- Поля для автоматизации (фаза реагирования, плановые сроки).



В верхней части экрана доступны следующие действия:

- **«Экраны»** — переход к настройке отображения данных в интерфейсе;
- **«Создать таблицу расширения»** — добавление новой связанной таблицы;
- **«Создать поле»** — добавление нового поля в текущую таблицу.

Кнопка «Создать таблицу расширения»

Назначение: создание новой таблицы, связанной с текущей по принципу «один ко многим» или «многие ко многим».

Процесс создания:

1. Открывается диалоговое окно «Создание таблицы расширения»;
2. Указываются:
   - Название (внутреннее имя);
   - Заголовок (отображаемое имя в интерфейсе).
3. После сохранения таблица автоматически связывается с исходной.

Примеры использования:

- Создание таблицы «Вложения» для хранения файлов, прикреплённых к инциденту;
- Создание таблицы «История изменений» для активов.

Кнопка «Создать поле»

Назначение: добавление нового поля в структуру таблицы.

Процесс создания:

1. Открывается форма создания поля;
2. Заполняются параметры:

- Заголовок — отображаемое название;
- Тип данных — строка, число, дата, логическое значение, ссылка, справочник и др.;
- Флаг «Сохранять историю» — включение аудита изменений;
- Максимальная длина (для строковых полей).

Дополнительные возможности:

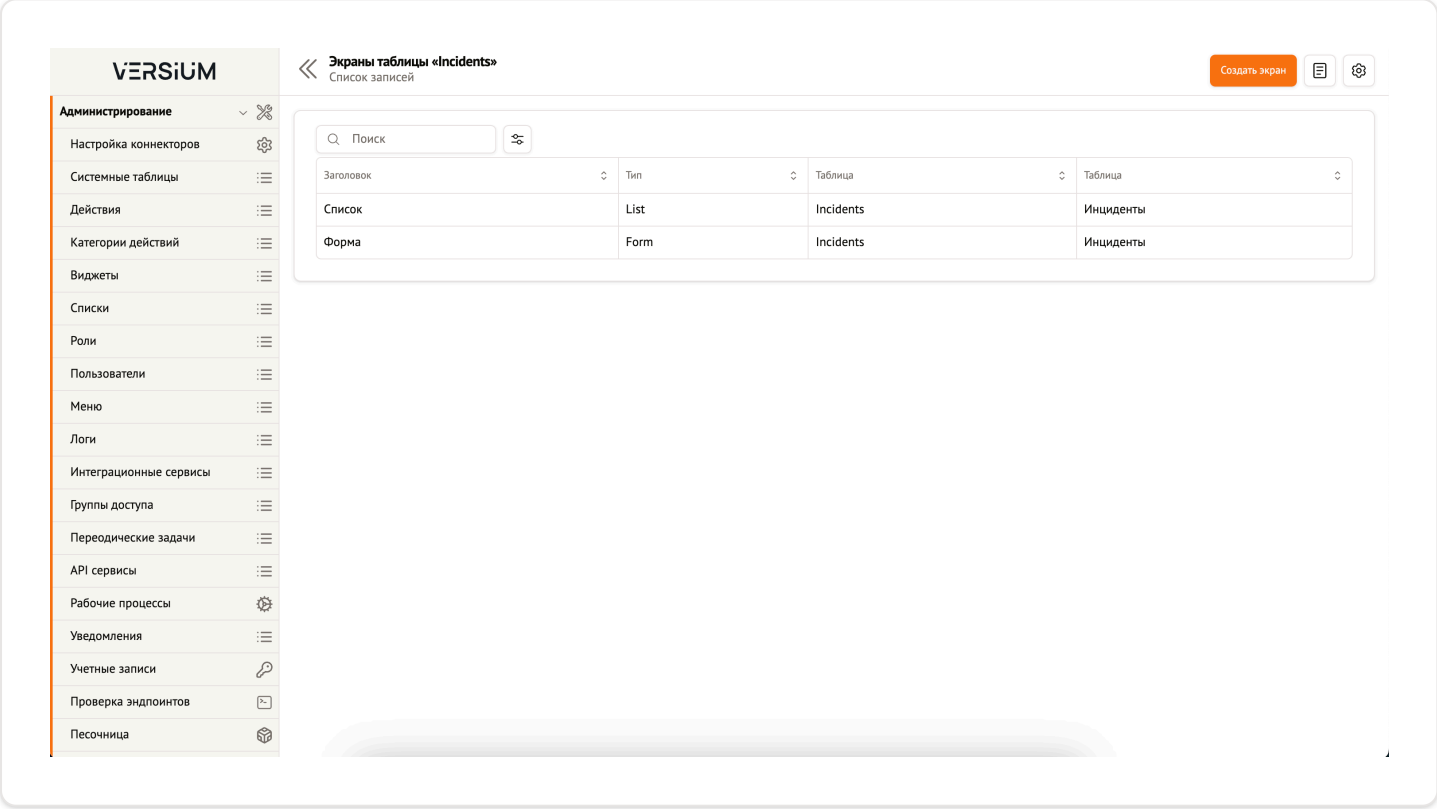
- Создание связи «Много-к-Одному»;
- Создание связи «Много-ко-Многим».

Создание новых полей и таблиц расширения позволяет адаптировать платформу под специфические требования организации без необходимости программирования.

2.3. Экраны таблицы «Incidents»

Раздел «Экраны» позволяет настраивать, как данные из таблицы будут отображаться в пользовательском интерфейсе. Это ключевой механизм кастомизации UX.
На экране отображаются типы представлений:

- **Список (List)** — отображение записей в табличном виде;
- **Форма (Form)** — отображение одной записи в виде карточки.



Кнопка **«Создать экран»** позволяет добавить новые представления (например, для отчётов или специальных режимов).

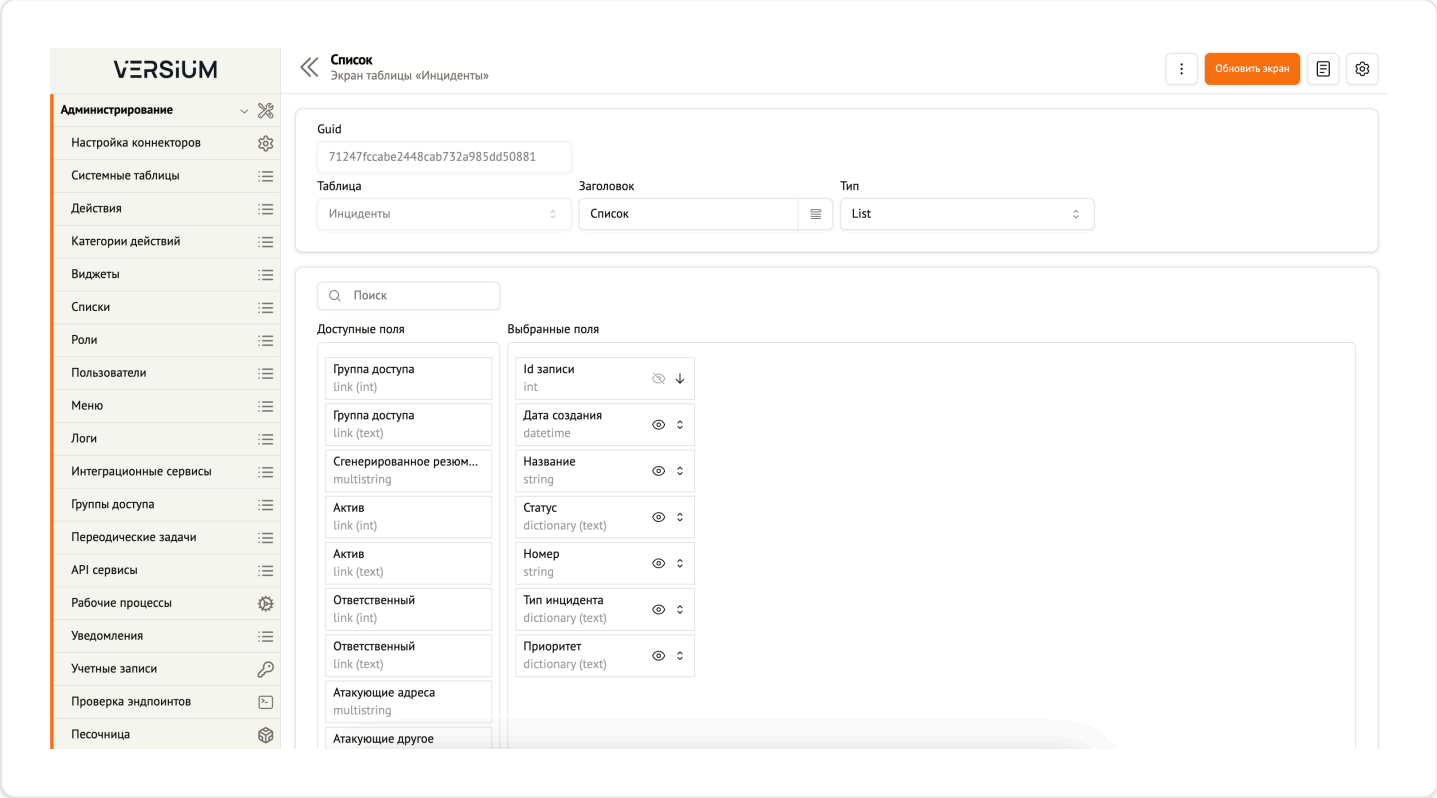
2.4. Настройка экрана «Список» для таблицы «Инциденты»

На этом экране происходит конфигурация пользовательского интерфейса для списка инцидентов. Интерфейс содержит две колонки:

- **Доступные поля** — все поля из таблицы, которые можно добавить;
- **Выбранные поля** — поля, отображаемые в списке.

Пользователь может:

- Перетаскивать поля между колонками;
- Изменять порядок отображения;
- Управлять видимостью полей;
- Настроить ширину столбцов.

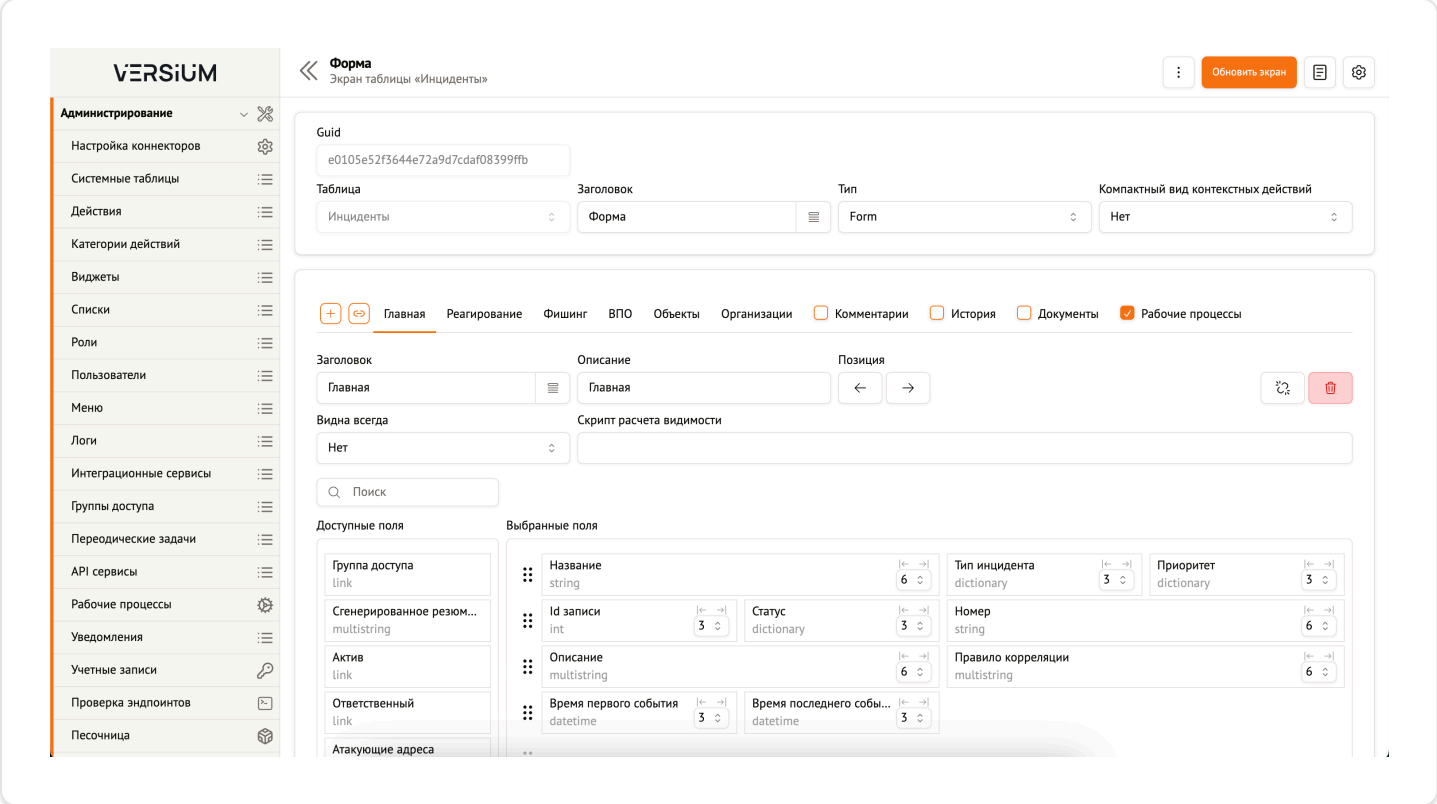


После настройки нажимается кнопка **«Обновить экран»**, что сохраняет изменения и применяет их ко всем пользователям.

2.5. Настройка экрана «Форма» для таблицы «Инциденты»

Раздел «Форма» позволяет настраивать интерфейс редактирования и просмотра отдельной записи. Возможности:

- Выбор отображаемых полей;
- Группировка полей по вкладкам (например, «Главная», «Реагирование», «Фишинг»);
- Настройка порядка и ширины полей;
- Управление видимостью и логикой отображения.



Изменения, внесённые в экран «Форма», напрямую влияют на структуру карточки объекта при работе с инцидентом.

2.6. Обобщение процесса настройки пользовательского интерфейса

Принцип работы механизма конфигурации:

1. Администратор выбирает системную таблицу;
2. Переходит в раздел «Экраны»;
3. Выбирает тип экрана: «Список» или «Форма»;
4. Настраивает состав и порядок полей, их группировку по вкладкам и параметры отображения;
5. Сохраняет изменения с помощью кнопки «Обновить экран».

Результат:

- Экран «Список» определяет, какие поля отображаются в таблице записей;
- Экран «Форма» определяет структуру карточки объекта при просмотре и редактировании.

Механизм обеспечивает полную конфигурацию интерфейса без необходимости внесения изменений в код.

2.7. Применимость к другим системным таблицам

Описанный процесс настройки экранов является универсальным и применяется ко всем системным таблицам платформы:

- `Assets` — настройка формы и списка активов;
- `Employees` — конфигурация карточки сотрудника;
- `Notifications` — управление отображением уведомлений;

- `AccessGroup` — настройка групп доступа и их атрибутов.

Каждая таблица поддерживает независимую конфигурацию экранов, что позволяет гибко адаптировать систему под требования различных подразделений и бизнес-процессов.

Раздел 3. Действия

3.1. Раздел «Действия»

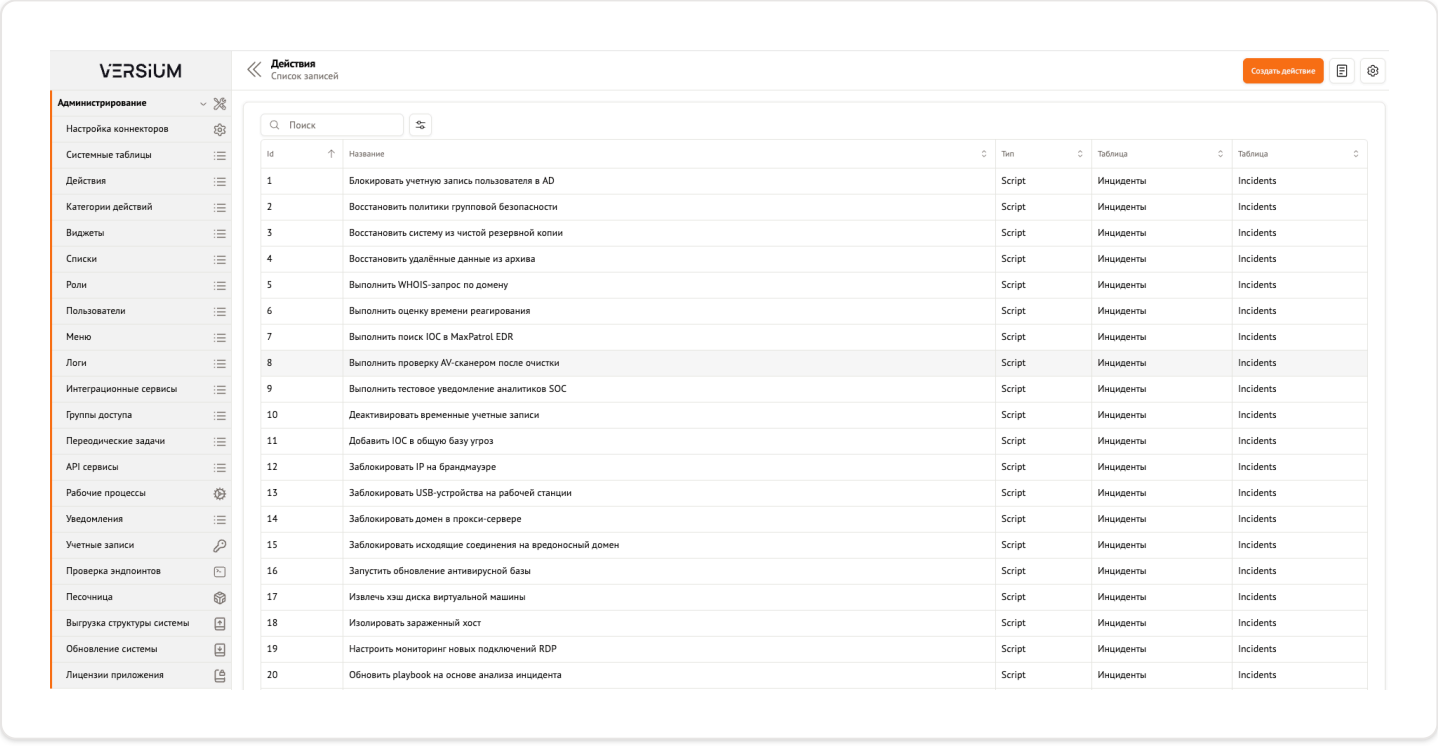
Раздел **«Действия»** в административном меню предназначен для управления набором операций, которые могут быть выполнены в рамках обработки инцидентов, активов или других объектов платформы.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять действия;
- Настраивать их поведение в зависимости от контекста;
- Использовать действия в рабочих процессах и автоматизации.

Каждое действие представляет собой функциональную единицу, которая может быть вызвана пользователем вручную или запущена автоматически через workflow.



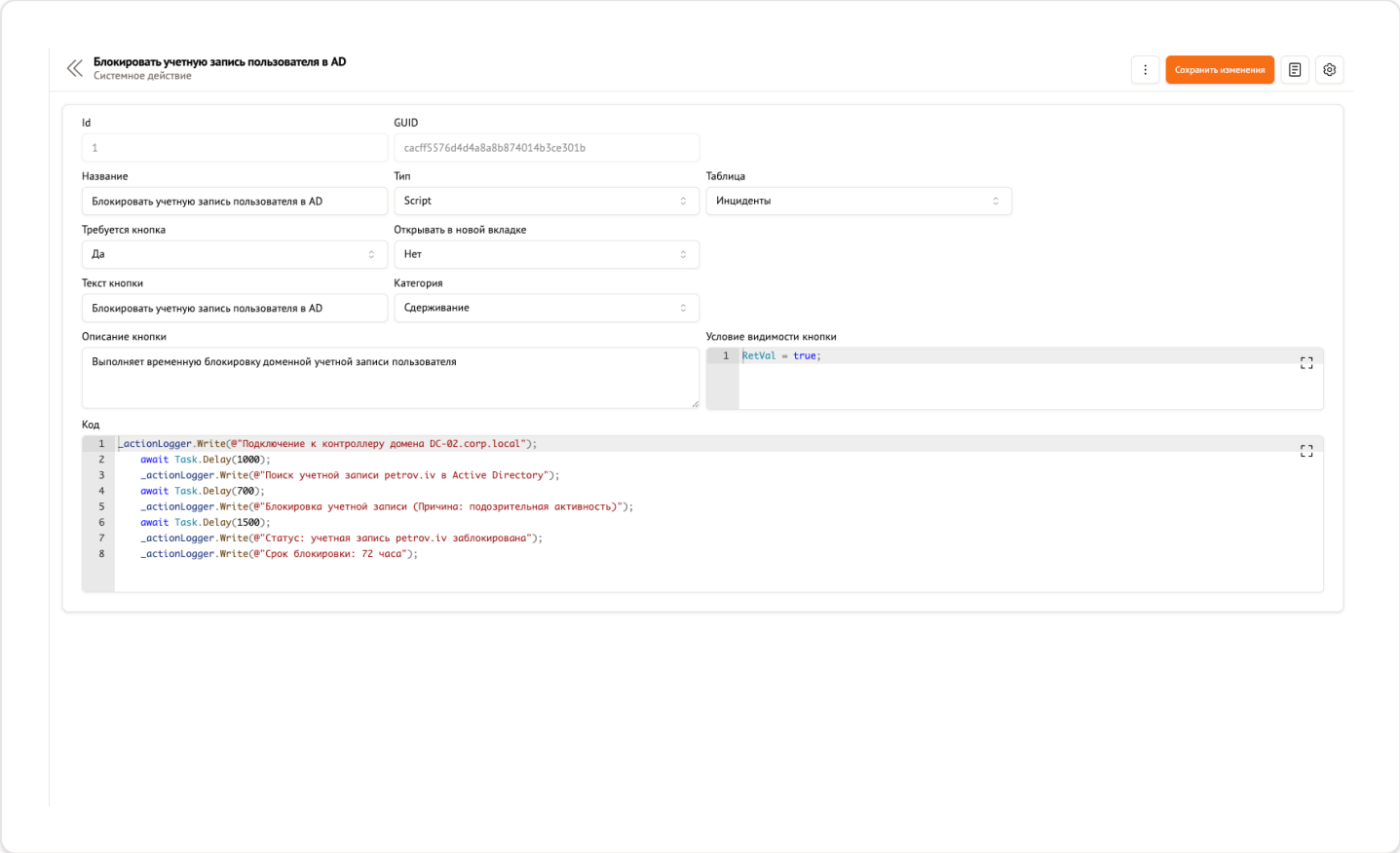
На экране отображается список всех зарегистрированных действий с возможностью фильтрации и поиска. Каждая запись содержит:

- Id — уникальный идентификатор действия;
- Название — отображаемое имя;
- Тип — тип действия (например, `Script` , `Wizard`);
- Таблица — объект, к которому применимо действие (например, `Инциденты`).

Кнопка **«Создать действие»** позволяет добавить новое действие в систему.

3.2. Форма редактирования действия

При переходе к редактированию существующего действия открывается форма настройки его параметров.



На форме доступны следующие поля:

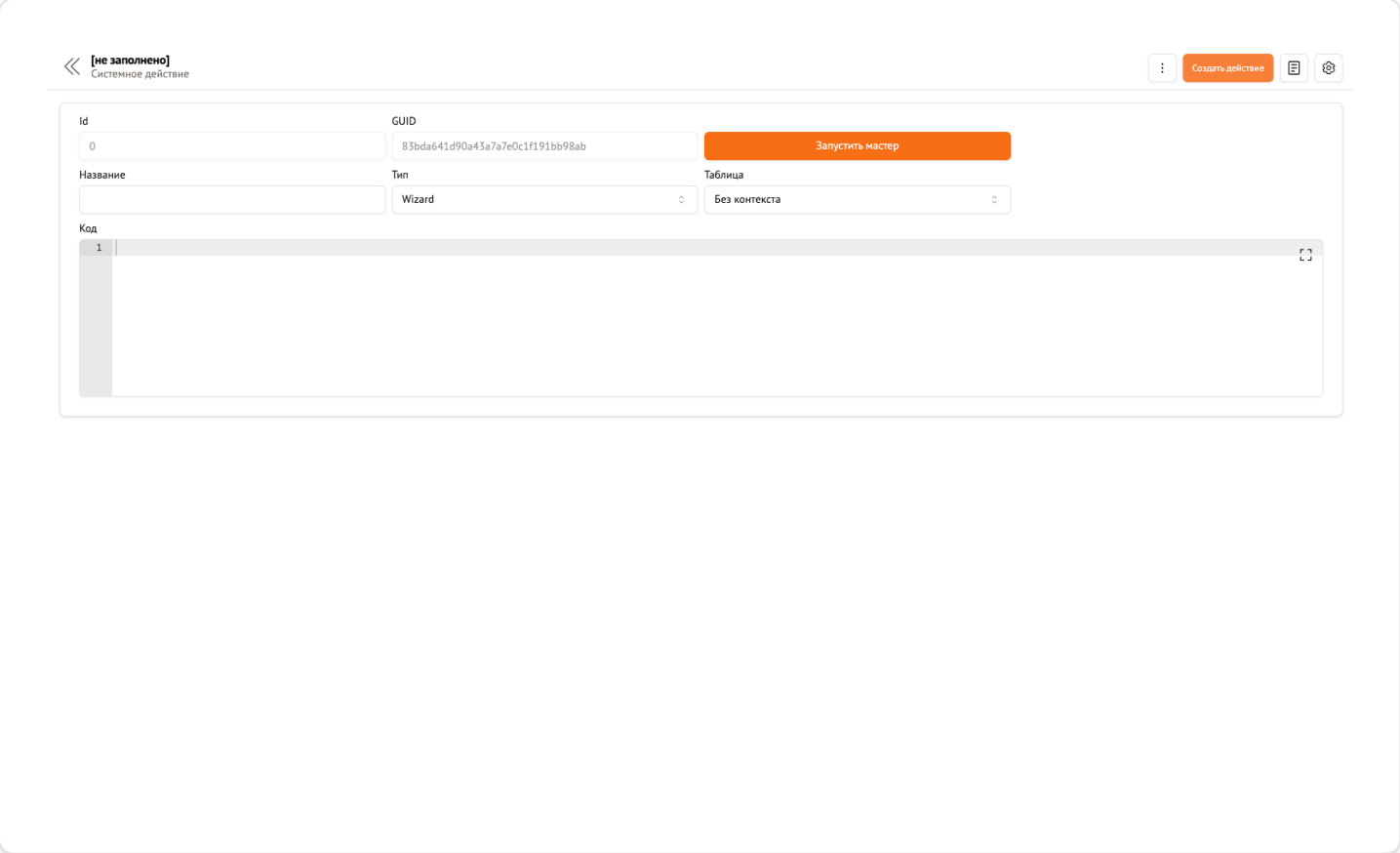
- Id — уникальный идентификатор действия (автоматически генерируется при создании);
- GUID — глобальный идентификатор действия;
- Название — отображаемое название действия;
- Тип — тип действия:
  - `Script` — выполнение пользовательского кода (например, C#);
  - `Wizard` — запуск мастера с шагами;
- Таблица — выбор объекта, к которому применяется действие (например, `Инциденты`);
- Требуется кнопка — флаг, определяющий, будет ли отображаться кнопка в интерфейсе;
- Открывать в новой вкладке — флаг, определяющий, будет ли действие выполняться в новом окне;
- Текст кнопки — текст, отображаемый на кнопке;
- Категория — группа, к которой относится действие (например, «Сдерживание», «Анализ», «Уведомление»);
- Описание кнопки — подробное описание действия для пользователя;

- Условие видимости кнопки — логическое выражение, определяющее, когда кнопка будет отображаться;
- Код — поле для ввода исходного кода (доступно только для типа `Script`).

После заполнения формы нажимается кнопка **«Сохранить изменения»**, что применяет настройки к действию.

3.3. Создание нового действия

При нажатии кнопки **«Создать действие»** открывается форма создания нового действия.



На форме доступны следующие параметры:

- Id — автоматически генерируется;
- GUID — автоматически генерируется;
- Название — обязательное поле;
- Тип — выбор из доступных типов: `Script`, `Wizard`;
- Таблица — выбор объекта, к которому относится действие (по умолчанию — «Без контекста»);
- Код — поле для ввода кода (доступно только для типа `Script`).

Кнопка **«Запустить мастер»** позволяет начать создание действия через визуальный конструктор, если выбран тип `Wizard`.
Детали реализации кода не описываются, так как они являются частной настройкой и зависят от конкретного сценария.

3.4. Обобщение функциональности

Принцип работы действий как функциональной единицы

1. Администратор создаёт новое действие через форму;
2. Выбирает тип (Script или Wizard);
3. Настраивает параметры отображения и поведения;
4. Вводит код или настраивает шаги мастера;
5. Сохраняет действие.

Применимость

- Действия используются в рабочих процессах;
- Могут быть вызваны вручную из карточки объекта;
- Поддерживают динамическую логику (условия видимости, параметры);
- Позволяют интегрировать платформу с внешними системами (AD, EDR, SIEM и др.).

Все действия можно редактировать, копировать и удалять в любое время.

4. Раздел «Категории действий»

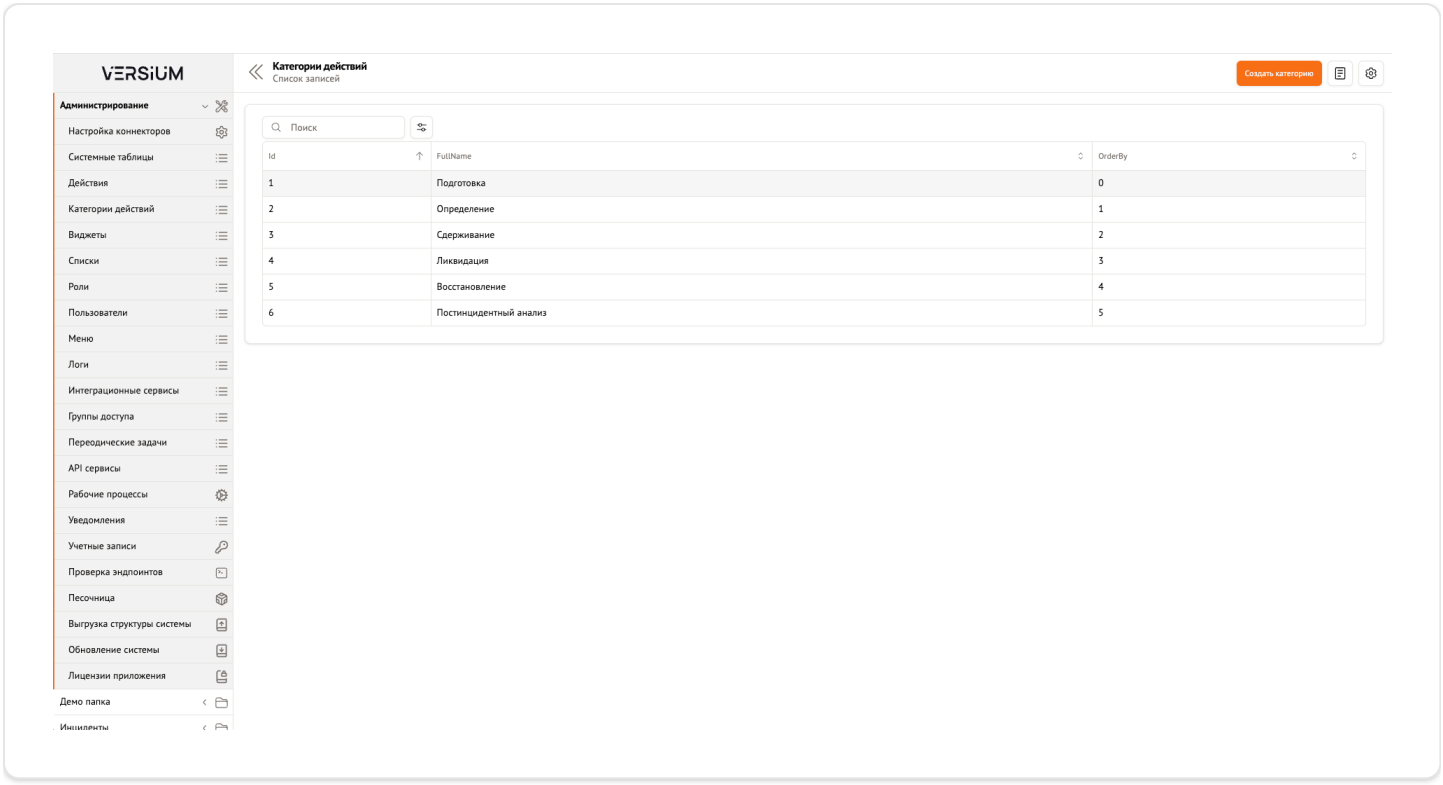
Раздел **«Категории действий»** предназначен для управления классификацией операций, выполняемых в рамках обработки инцидентов и других объектов платформы.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять категории;
- Организовывать действия по смысловым группам;
- Управлять порядком отображения категорий в интерфейсе.

Каждая категория представляет собой функциональную единицу, которая используется при настройке действий и рабочих процессов. Категории применяются для группировки действий в пользовательском интерфейсе (например, на карточке инцидента).



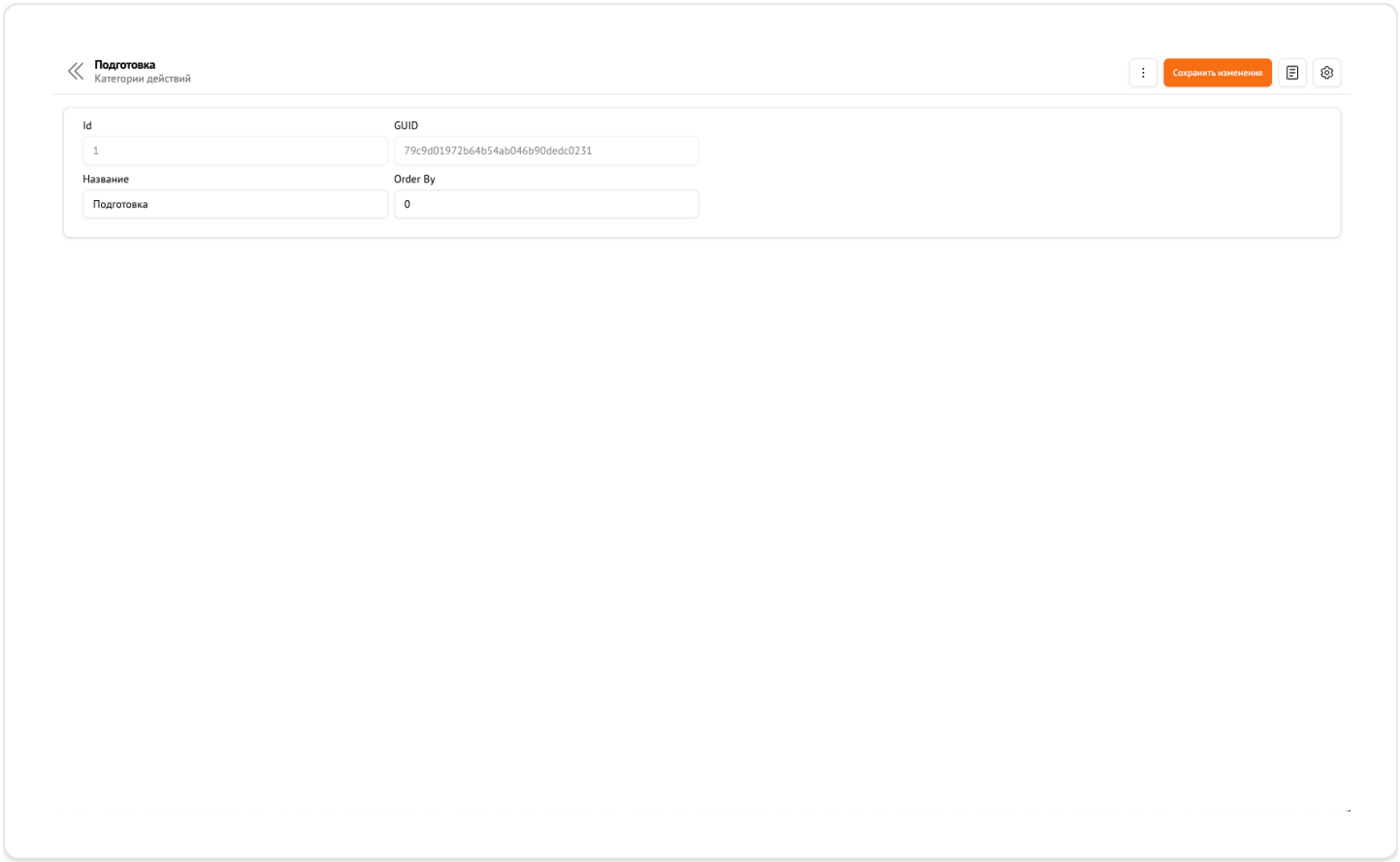
На экране отображается список всех зарегистрированных категорий действий с возможностью поиска и сортировки. Каждая запись содержит:

- Id — уникальный идентификатор категории;
- FullName — отображаемое название категории;
- OrderBy — порядковый номер, определяющий положение категории в списке.

Кнопка **«Создать категорию»** позволяет добавить новую категорию в систему.

4.1. Форма редактирования категории

При переходе к редактированию существующей категории открывается форма настройки её параметров.



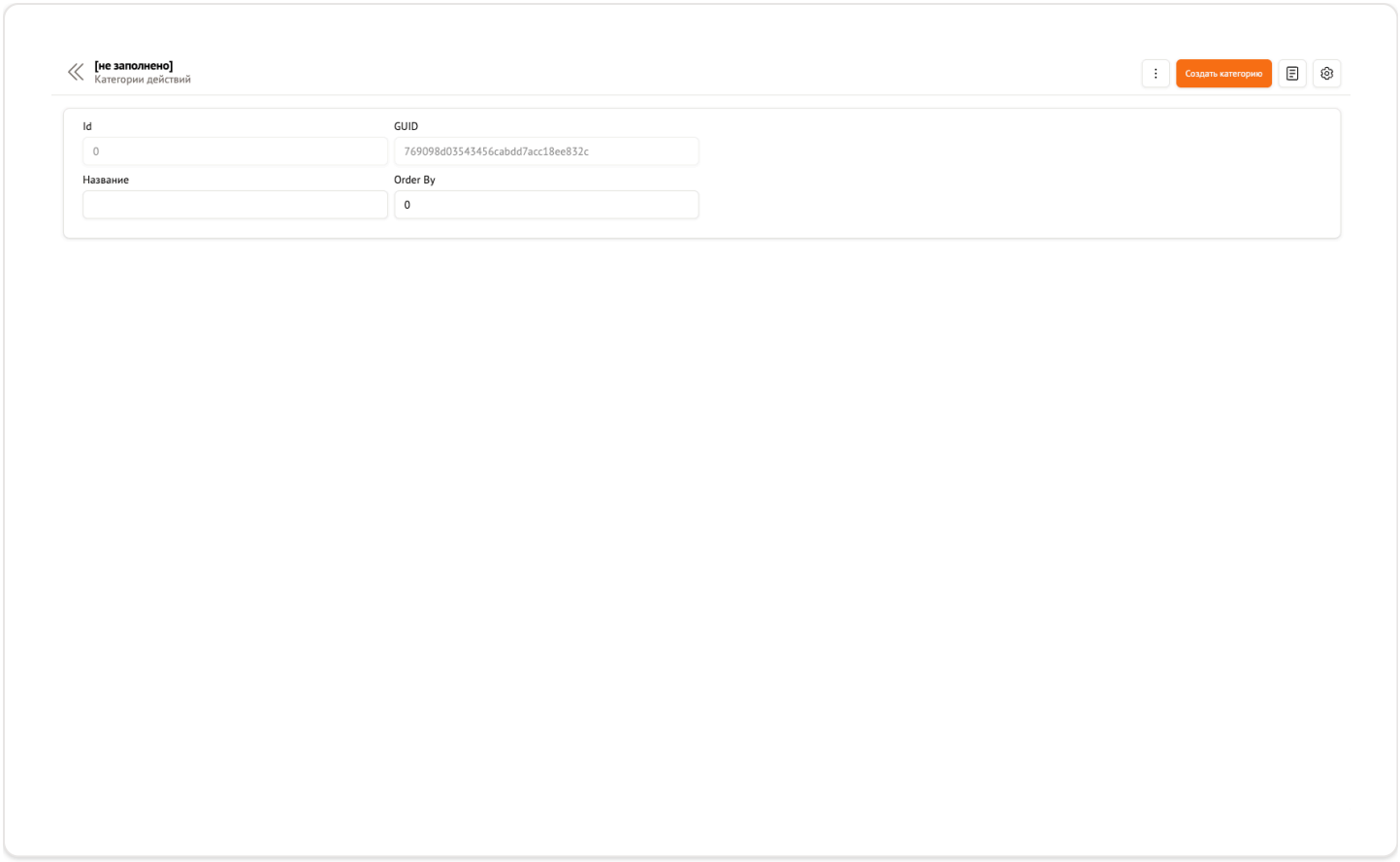
На форме доступны следующие поля:

- Id — уникальный идентификатор категории (автоматически генерируется при создании);
- GUID — глобальный идентификатор категории;
- Название — отображаемое имя категории;
- Order By — порядковый номер, определяющий положение категории в списке (чем меньше значение, тем выше позиция).

После заполнения формы нажимается кнопка **«Сохранить изменения»**, что применяет настройки к категории.

4.2. Создание новой категории

При нажатии кнопки **«Создать категорию»** открывается форма создания новой категории.



На форме доступны следующие параметры:

- Id — автоматически генерируется;
- GUID — автоматически генерируется;
- Название — обязательное поле;
- Order By — порядковый номер (по умолчанию — 0).

После заполнения формы и сохранения категории она становится доступной для использования в настройке действий.

4.3. Обобщение функциональности

Принцип работы категорий действий как функциональной единицы

1. Администратор создаёт новую категорию через форму;
2. Задаёт название и порядок отображения;
3. Сохраняет категорию;
4. Применяет категорию к действиям в разделе «Действия».

Применимость

- Категории используются для группировки действий в интерфейсе;
- Позволяют организовать рабочий процесс по этапам (например, «Подготовка», «Определение», «Ликвидация»);
- Поддерживают динамическую сортировку в списках;
- Используются в конфигурации рабочих процессов и форм.

Все категории можно редактировать, копировать и удалять в любое время.

5. Раздел «Виджеты»

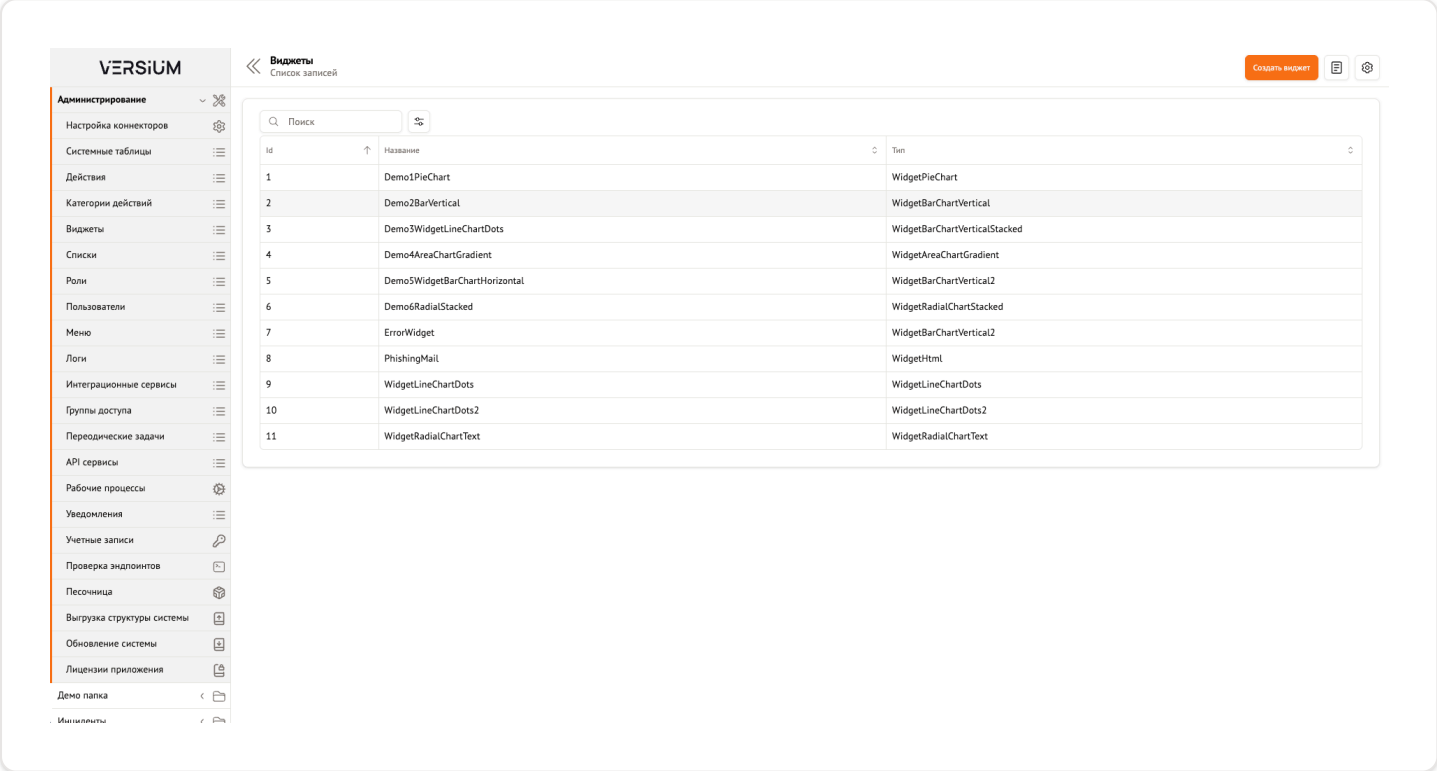
Раздел **«Виджеты»** предназначен для управления набором аналитических и информационных элементов, отображаемых на дашбордах платформы.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять виджеты;
- Настраивать их поведение и внешний вид;
- Использовать в дашбордах для визуализации метрик и данных.

Каждый виджет представляет собой функциональную единицу, которая может быть добавлена на любой дашборд и настроена под конкретные задачи анализа.



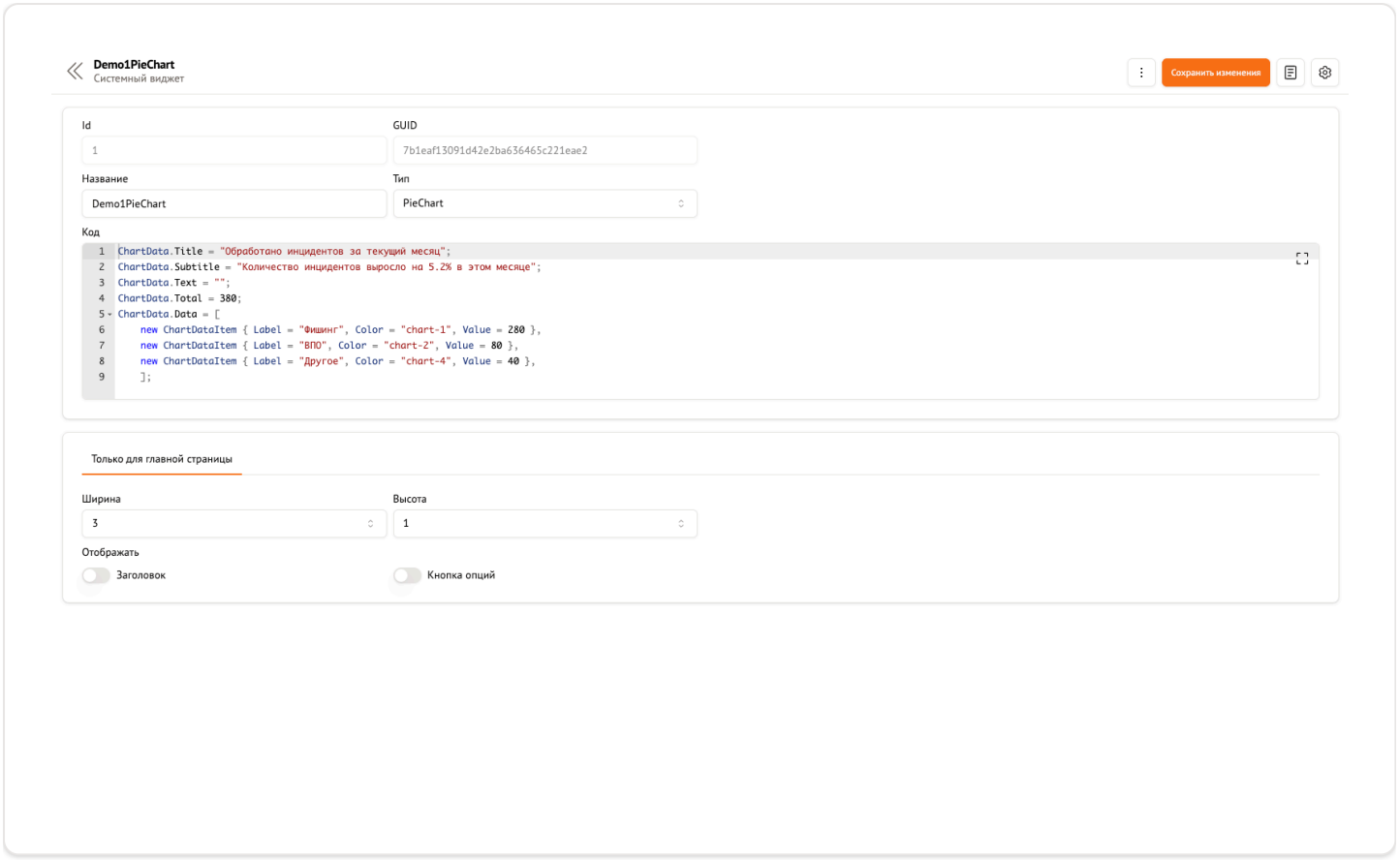
На экране отображается список всех зарегистрированных виджетов с возможностью поиска и сортировки. Каждая запись содержит:

- Id — уникальный идентификатор виджета;
- Название — отображаемое имя;
- Тип — тип виджета (например, `WidgetPieChart` , `WidgetBarChartVertical` , `WidgetHtml`).

Кнопка **«Создать виджет»** позволяет добавить новый виджет в систему.

5.1. Форма редактирования виджета

При переходе к редактированию существующего виджета открывается форма настройки его параметров.



На форме доступны следующие поля:

- Id — уникальный идентификатор виджета (автоматически генерируется при создании);
- GUID — глобальный идентификатор виджета;
- Название — отображаемое имя виджета;
- Тип — тип виджета:
  - `PieChart` — круговая диаграмма;
  - `BarChartVertical` — вертикальная столбчатая диаграмма;
  - `LineChartDots` — линейный график с точками;
  - `Html` — произвольный HTML-контент;
  - и другие.
- Код — поле для ввода исходного кода, определяющего данные и поведение виджета.

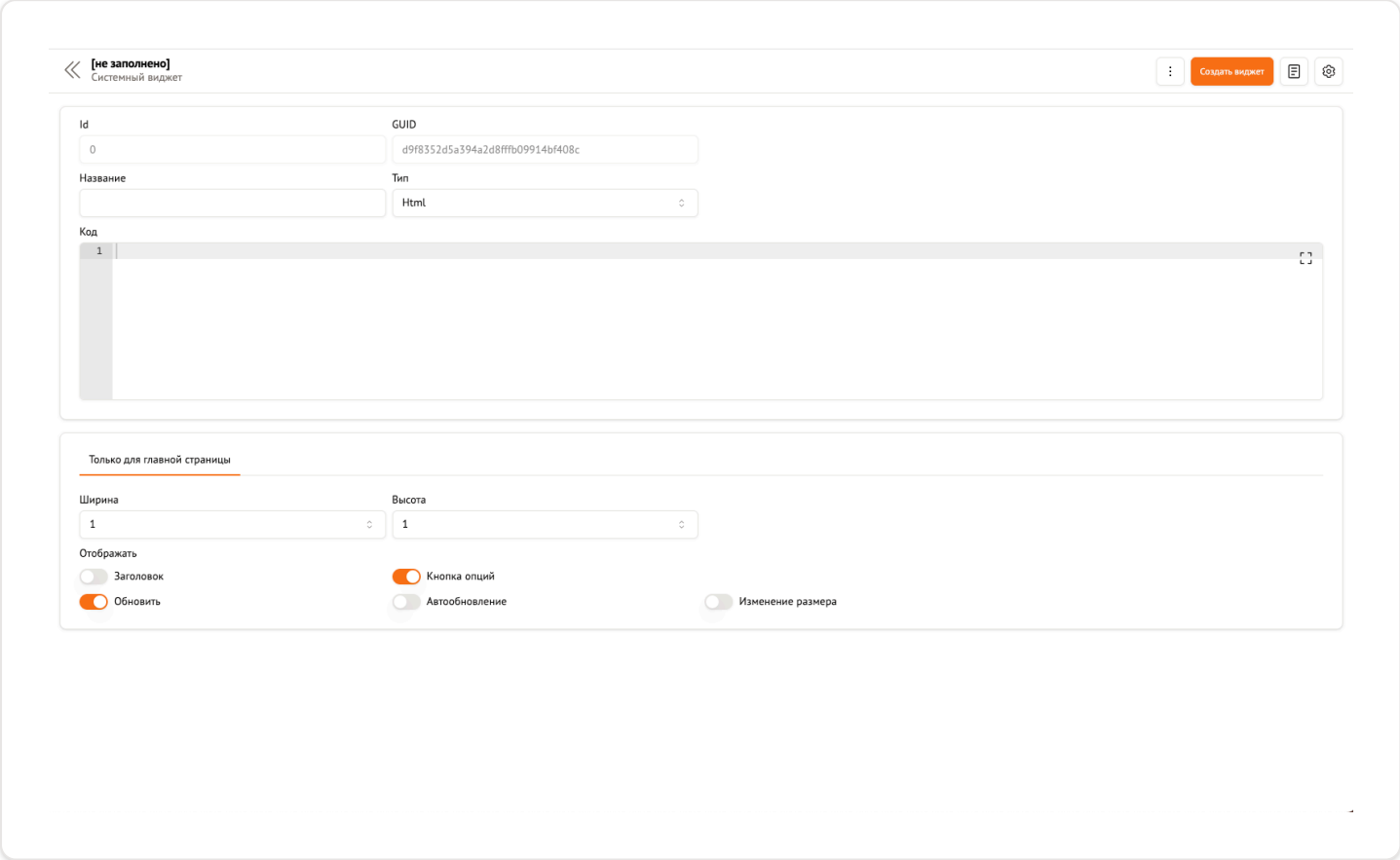
Дополнительные параметры:

- Ширина — количество колонок, занимаемых виджетом на дашборде;
- Высота — количество строк, занимаемых виджетом;
- Отображать — флаги, определяющие, будут ли отображаться:
  - Заголовок;
  - Кнопка опций;
  - Обновить;
  - Автообновление;
  - Изменение размера.

После заполнения формы нажимается кнопка **«Сохранить изменения»**, что применяет настройки к виджету.

5.2. Создание нового виджета

При нажатии кнопки **«Создать виджет»** открывается форма создания нового виджета.



На форме доступны следующие параметры:

- Id — автоматически генерируется;
- GUID — автоматически генерируется;
- Название — обязательное поле;
- Тип — выбор из доступных типов (например, `Html` , `PieChart`);
- Код — поле для ввода исходного кода;
- Параметры отображения: ширина, высота, флаги отображения элементов.

После заполнения формы и сохранения виджет становится доступным для использования на дашбордах.

5.3. Обобщение функциональности

Принцип работы виджетов как функциональной единицы

1. Администратор создаёт новый виджет через форму;
2. Выбирает тип и настраивает параметры;
3. Вводит код для определения данных и поведения;

4. Сохраняет виджет;
5. Добавляет виджет на дашборд через раздел «Виджеты».

Применимость

- Виджеты используются для визуализации метрик на дашбордах;
- Поддерживают различные типы графиков и текстового контента;
- Могут быть настроены под любые бизнес-задачи (статистика инцидентов, анализ угроз, мониторинг активов);
- Интегрируются с данными из системных таблиц и API.

Все виджеты можно редактировать, копировать и удалять в любое время.

6. Раздел «Списки» или «Системные словари»

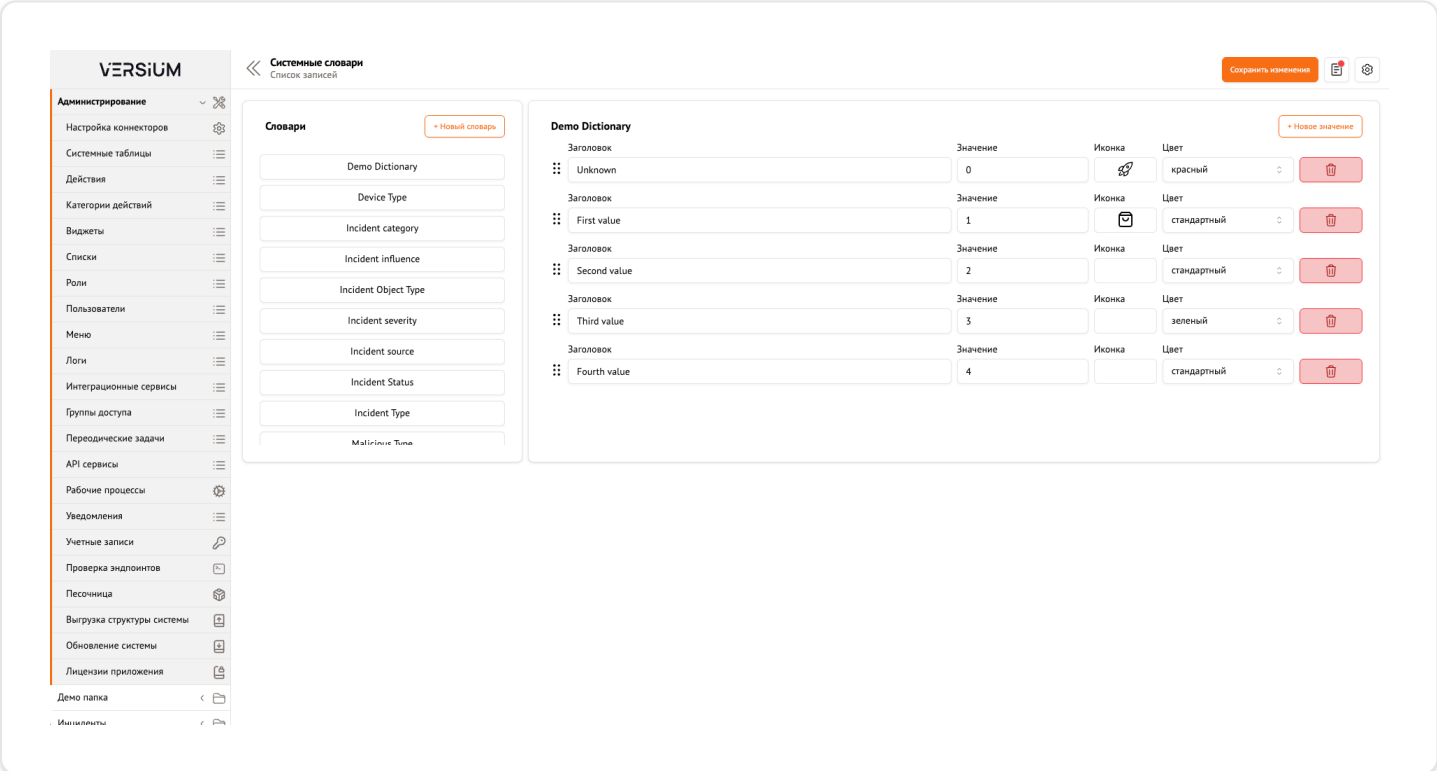
Раздел **«Системные словари»** предназначен для управления справочниками значений, используемых в платформе для классификации данных.

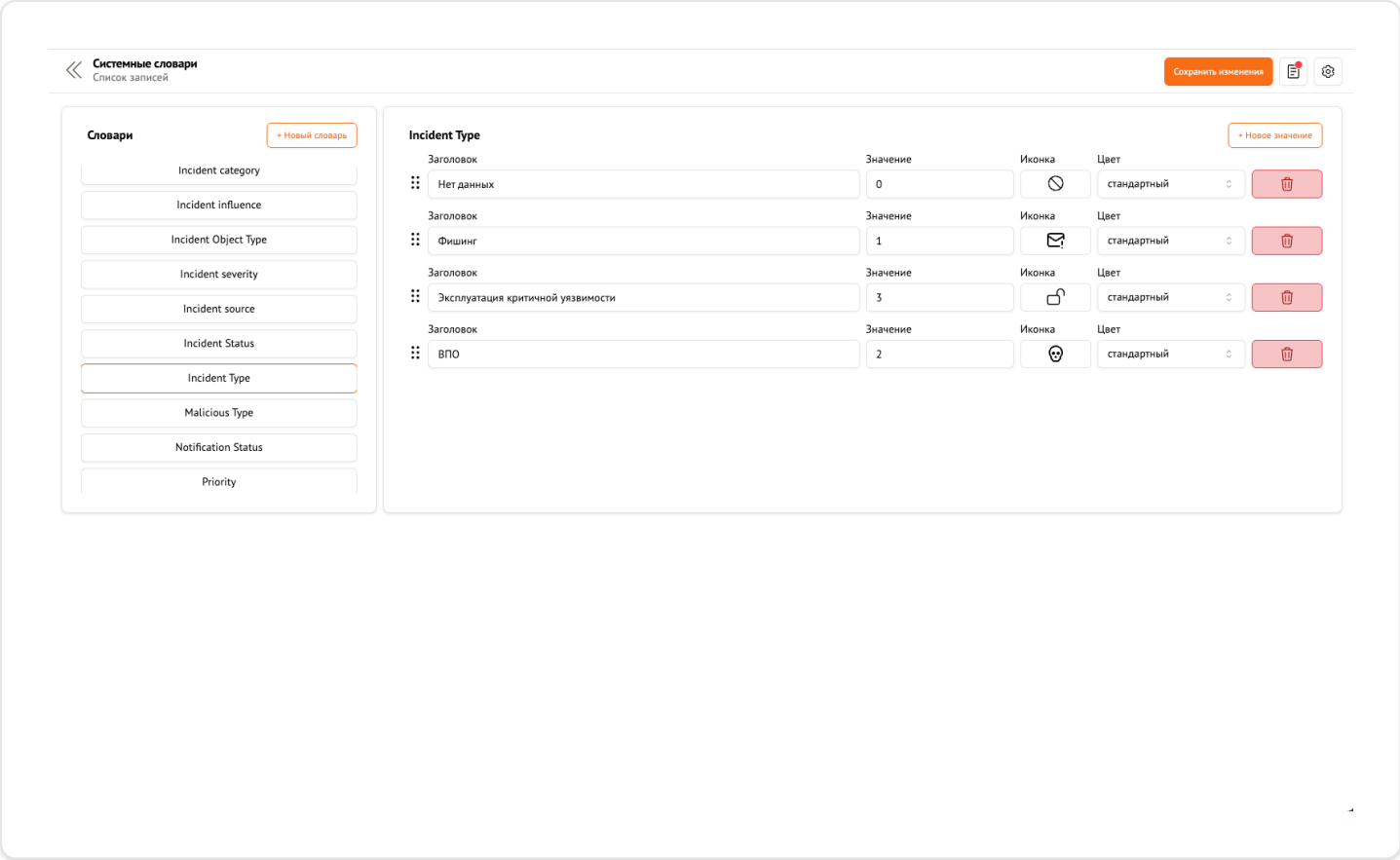
Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять системные словари;
- Управлять набором значений, отображаемых в полях типа `dictionary` ;
- Настраивать визуальное представление значений (цвет, иконка);
- Обеспечивать единое семантическое пространство для всех пользователей системы.

Каждый словарь представляет собой функциональную единицу, которая используется в полях таблиц, формах, дашбордах и рабочих процессах.



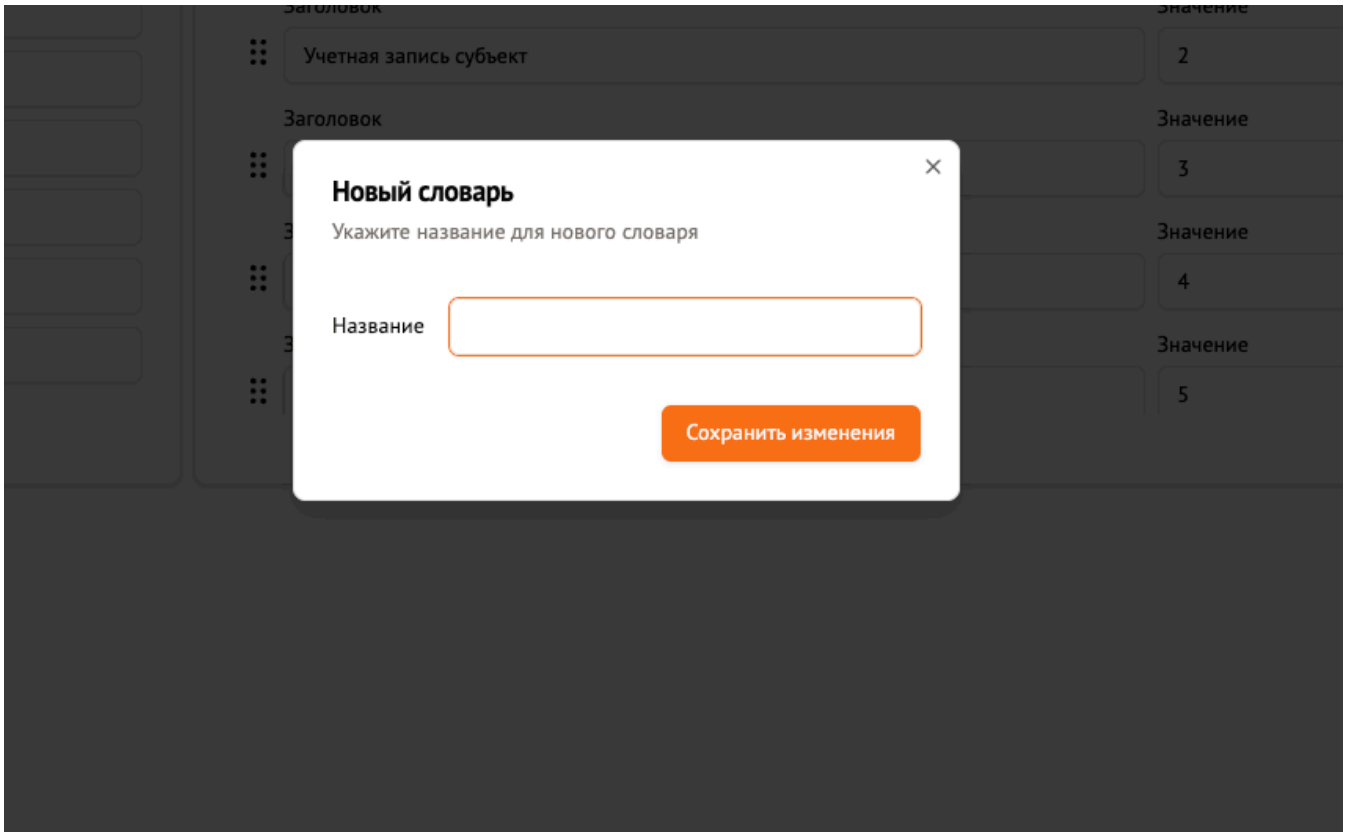


На экране отображается список всех зарегистрированных словарей. Каждый словарь представлен в виде карточки с его названием. Слева расположен список доступных словарей, а справа — содержимое выбранного словаря.

Кнопка **«+ Новый словарь»** позволяет создать новый справочник.

6.1. Форма создания нового словаря

При нажатии кнопки **«+ Новый словарь»** открывается диалоговое окно для задания имени нового словаря.



В окне необходимо указать:

- **Название** — отображаемое имя словаря (например, «Тип инцидента», «Статус»).

После сохранения создается новый словарь, который становится доступным для редактирования.

6.2. Редактирование словаря

При выборе словаря из списка отображается его содержимое: список значений с параметрами.
Для каждого значения доступны следующие поля:

- **Заголовок** — отображаемое название значения;
- **Значение** — внутреннее числовое значение (используется в API и логике);
- **Иконка** — возможность назначить иконку для визуального отличия;
- **Цвет** — цветовая метка (например, красный, зелёный, стандартный);
- **Удаление** — кнопка удаления значения (с подтверждением).

Кнопка **«+ Новое значение»** позволяет добавить новую запись в словарь.
Все изменения применяются после нажатия кнопки **«Сохранить изменения»**.

6.3. Обобщение функциональности

Принцип работы системных словарей как функциональной единицы

1. Администратор создаёт новый словарь через форму;
2. Добавляет значения с параметрами (заголовок, значение, цвет, иконка);
3. Сохраняет изменения;

4. Использует словарь в полях типа `dictionary` в системных таблицах, формах и виджетах.

Применимость

- Словари используются для унификации терминологии в системе;
- Позволяют контролировать выбор значений при заполнении полей;
- Поддерживают визуальную идентификацию значений (цвет, иконка);
- Применяются в настройке статусов, приоритетов, категорий, типов инцидентов и других классификационных полей.

Все словари можно редактировать, копировать и удалять в любое время.

7. Раздел «Роли»

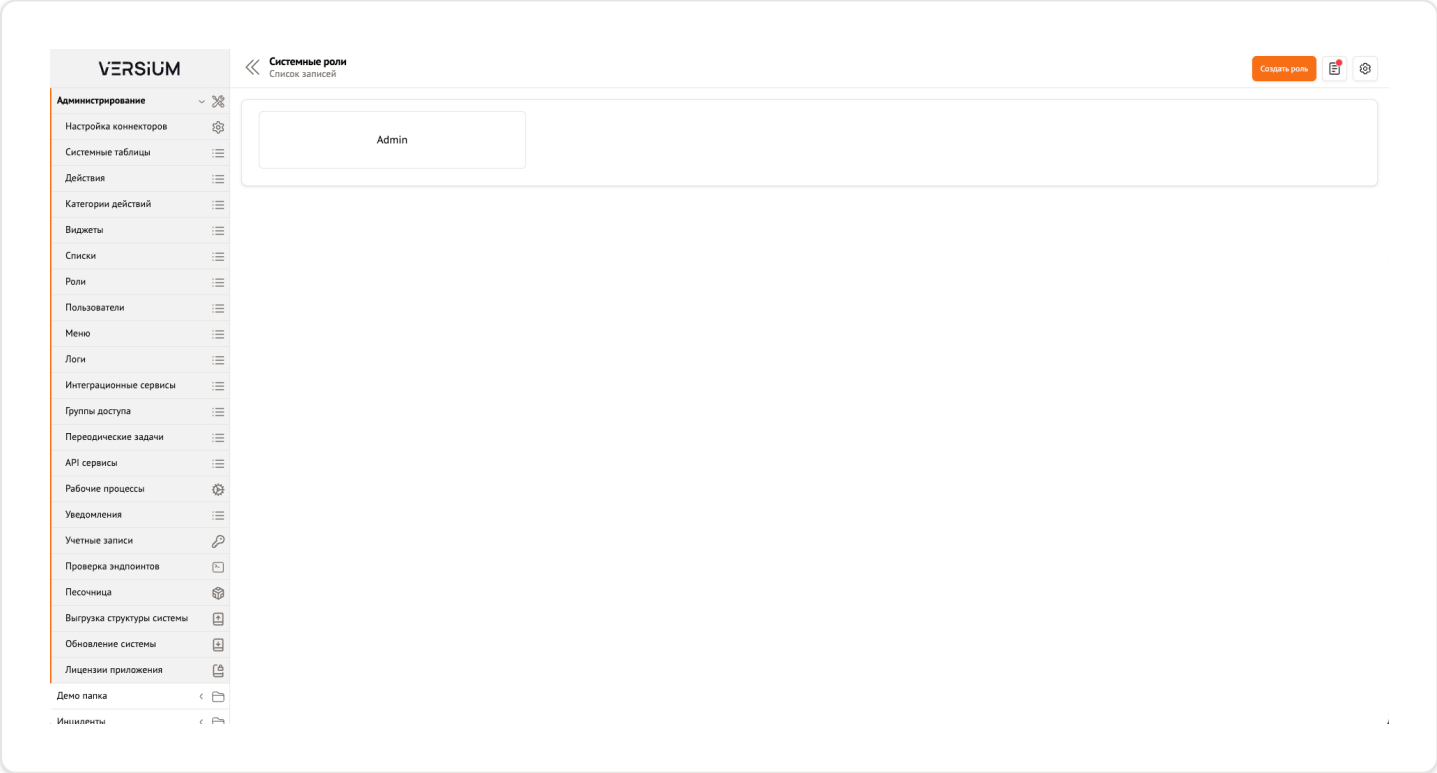
Раздел **«Роли»** предназначен для управления системными правами доступа пользователей к функциональным возможностям платформы.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять роли;
- Настраивать права доступа по таблицам, меню, действиям и виджетам;
- Организовывать иерархию прав в соответствии с требованиями безопасности и бизнес-процессами.

Каждая роль представляет собой функциональную единицу, которая определяет набор разрешений для пользователя или группы пользователей.

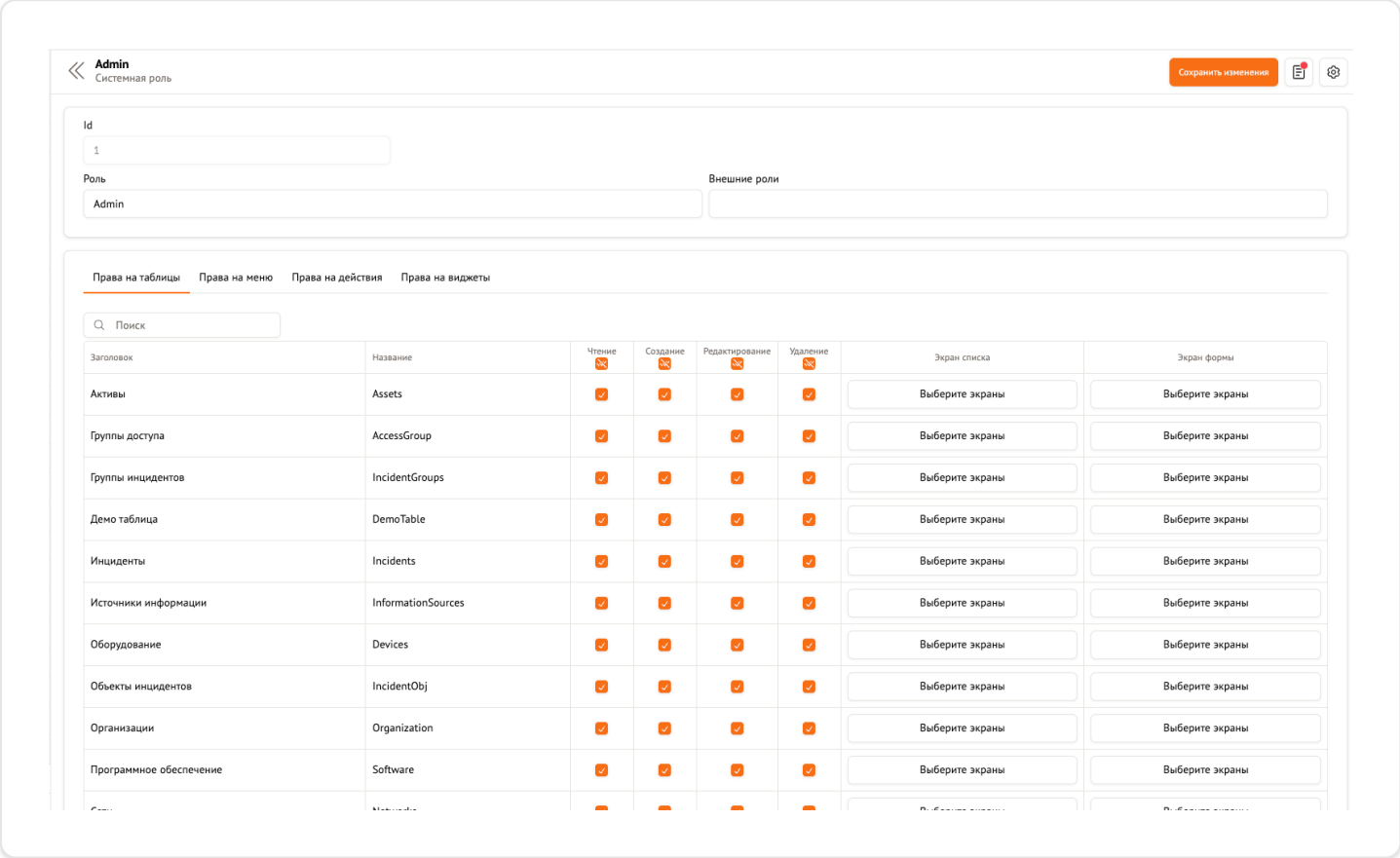


На экране отображается список всех зарегистрированных ролей. Каждая роль представлена в виде карточки с её названием. В списке отображается стандартная роль `Admin` .

Кнопка **«Создать роль»** позволяет добавить новую роль в систему.

7.1. Форма редактирования роли

При переходе к редактированию существующей роли открывается форма настройки её параметров.



На форме доступны следующие поля:

- Id — уникальный идентификатор роли (автоматически генерируется при создании);
- Роль — отображаемое имя роли;
- Внешние роли — поле для указания внешних ролей из интегрированных систем (например, Active Directory).

В нижней части формы расположены четыре вкладки для настройки прав:

- **Права на таблицы** — управление доступом к системным таблицам;
- **Права на меню** — управление доступом к пунктам меню;
- **Права на действия** — управление доступом к операциям;
- **Права на виджеты** — управление доступом к дашбордам и виджетам.

7.2. Настройка прав на таблицы

На вкладке **Права на таблицы** отображается таблица с перечнем всех системных таблиц. Для каждой таблицы можно установить следующие права:

- Чтение — возможность просмотра данных;
- Создание — возможность создания новых записей;

- Редактирование — возможность изменения существующих записей;
- Удаление — возможность удаления записей;
- Экран списка — выбор экрана, отображаемого при просмотре списка;
- Экран формы — выбор экрана, отображаемого при редактировании записи.

7.3. Настройка прав на меню

На вкладке **Права на меню** отображается список пунктов меню платформы. Для каждого пункта можно установить флаг доступа.
Доступные пункты включают:

- Администрирование: Настройка коннекторов, Системные таблицы, Действия, Категории действий, Виджеты, Списки, Роли, Пользователи, Меню, Логи, Интеграционные сервисы, Группы доступа, Периодические задачи, API сервисы, Рабочие процессы, Уведомления, Учетные записи, Проверка эндпоинтов, Песочница, Выгрузка структуры системы, Обновление системы, Лицензии приложения.

Флаг **Доступ** определяет, может ли пользователь с данной ролью открыть соответствующий раздел.

7.4. Настройка прав на действия

На вкладке **Права на действия** отображается список всех зарегистрированных действий. Для каждого действия можно установить флаг доступа.
Действия классифицируются по типу:

- `Wizard` — мастера;
- `Script` — скрипты.

Флаг **Доступ** определяет, может ли пользователь с данной ролью запустить действие.

7.5. Настройка прав на виджеты

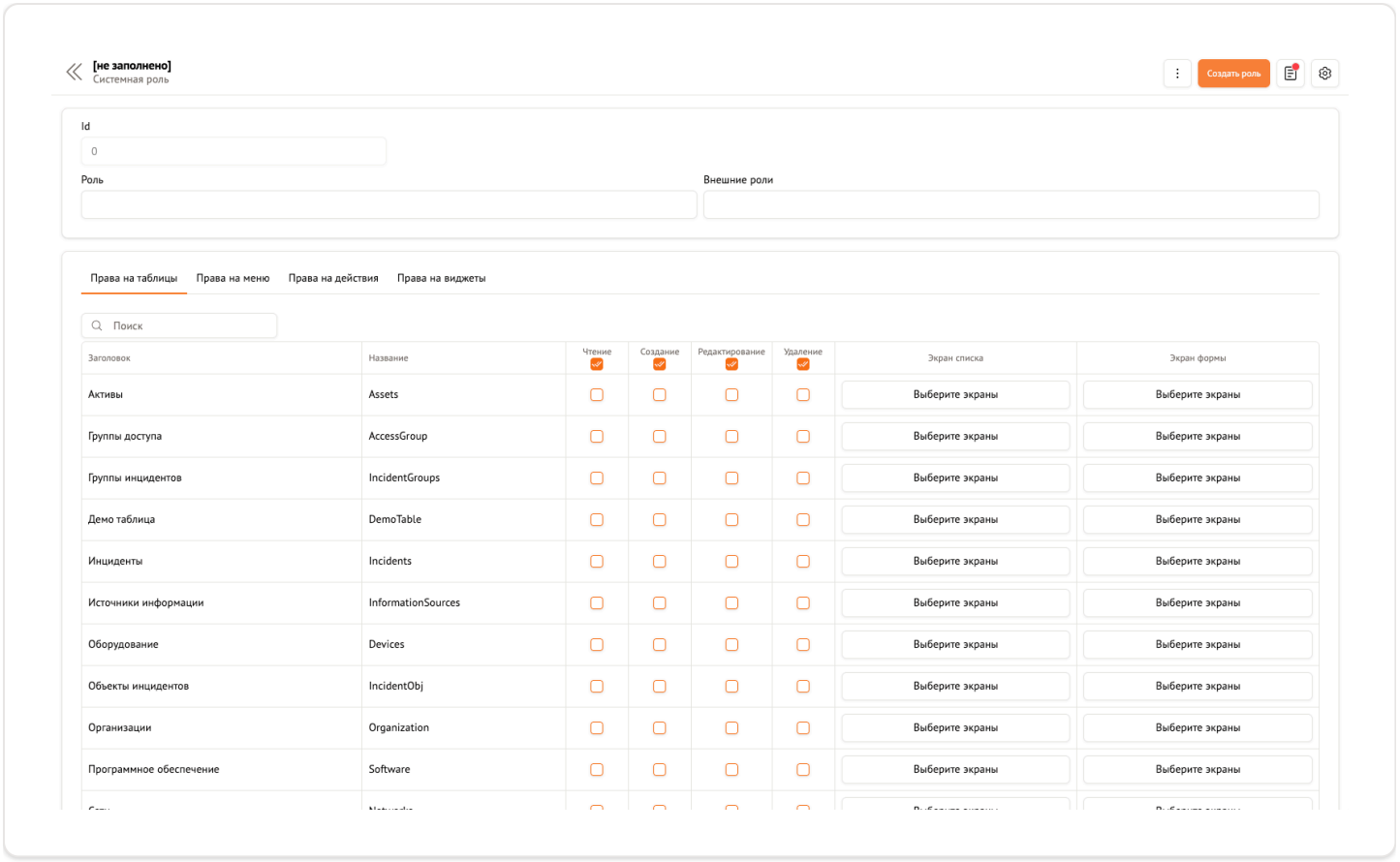
На вкладке **Права на виджеты** отображаются два столбца:

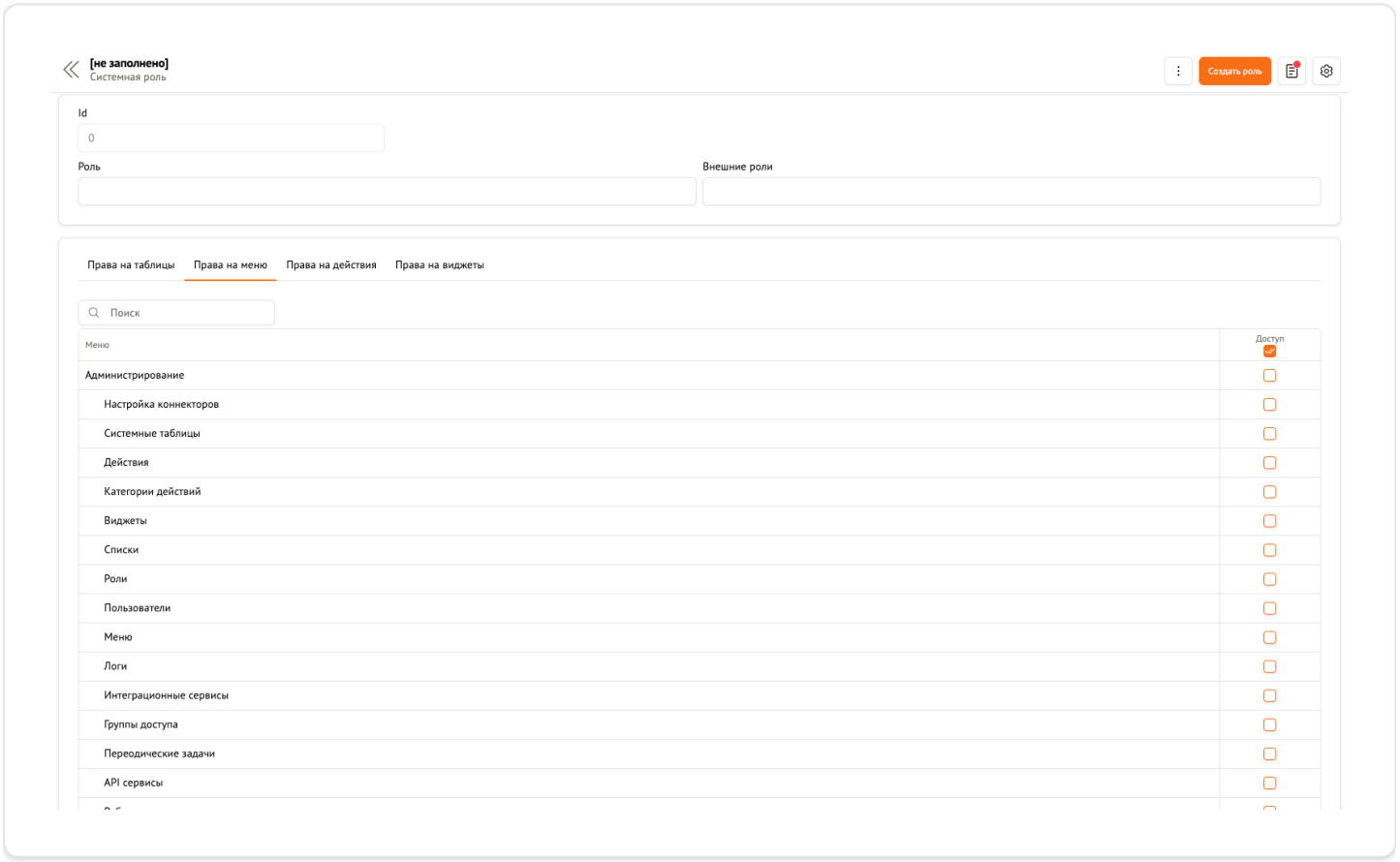
- **Доступные виджеты** — список всех доступных виджетов;
- **Выбранные виджеты** — список виджетов, которые будут доступны пользователю с данной ролью.

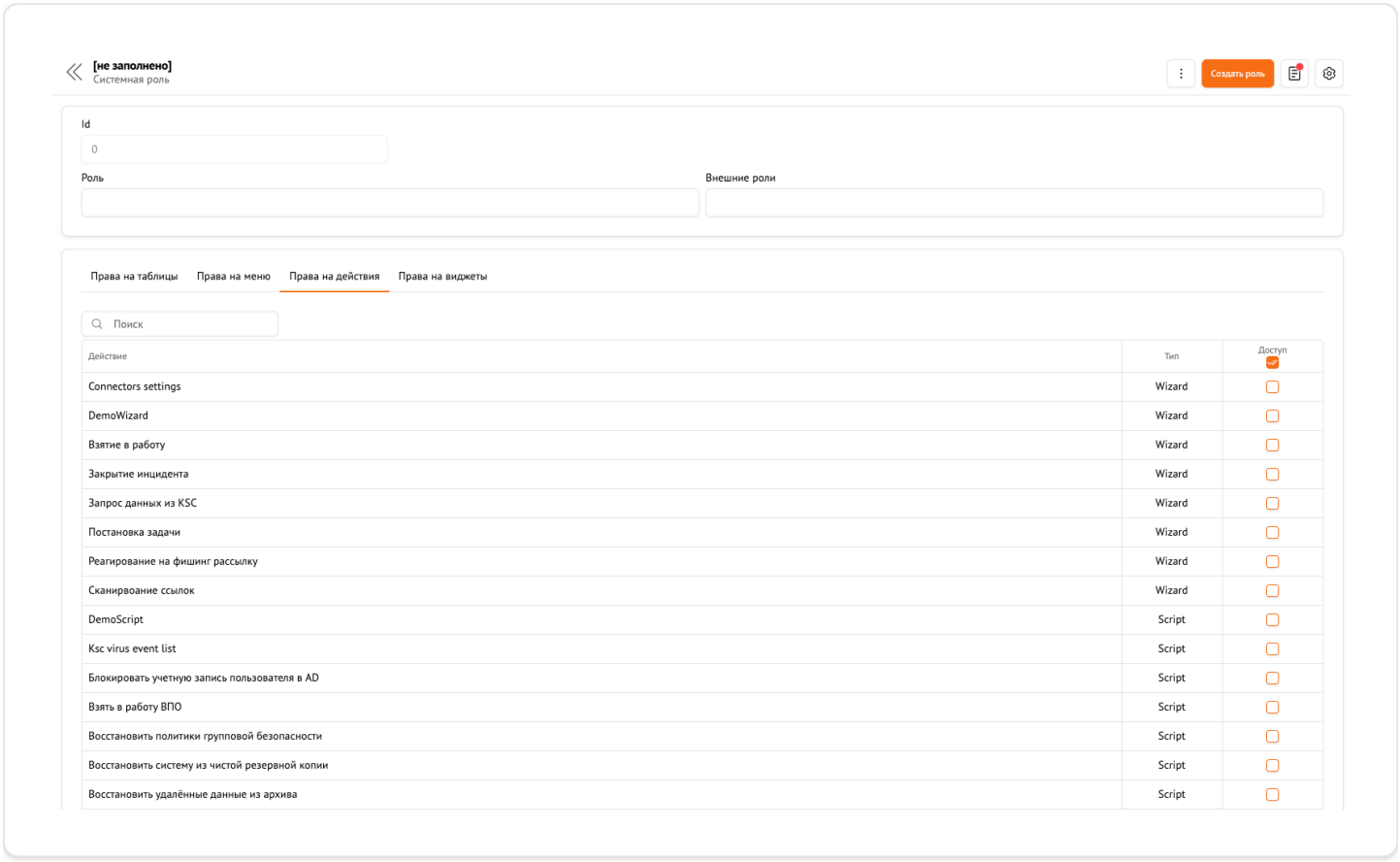
Администратор может перетаскивать виджеты из левого столбца в правый, чтобы предоставить доступ к ним пользователям.

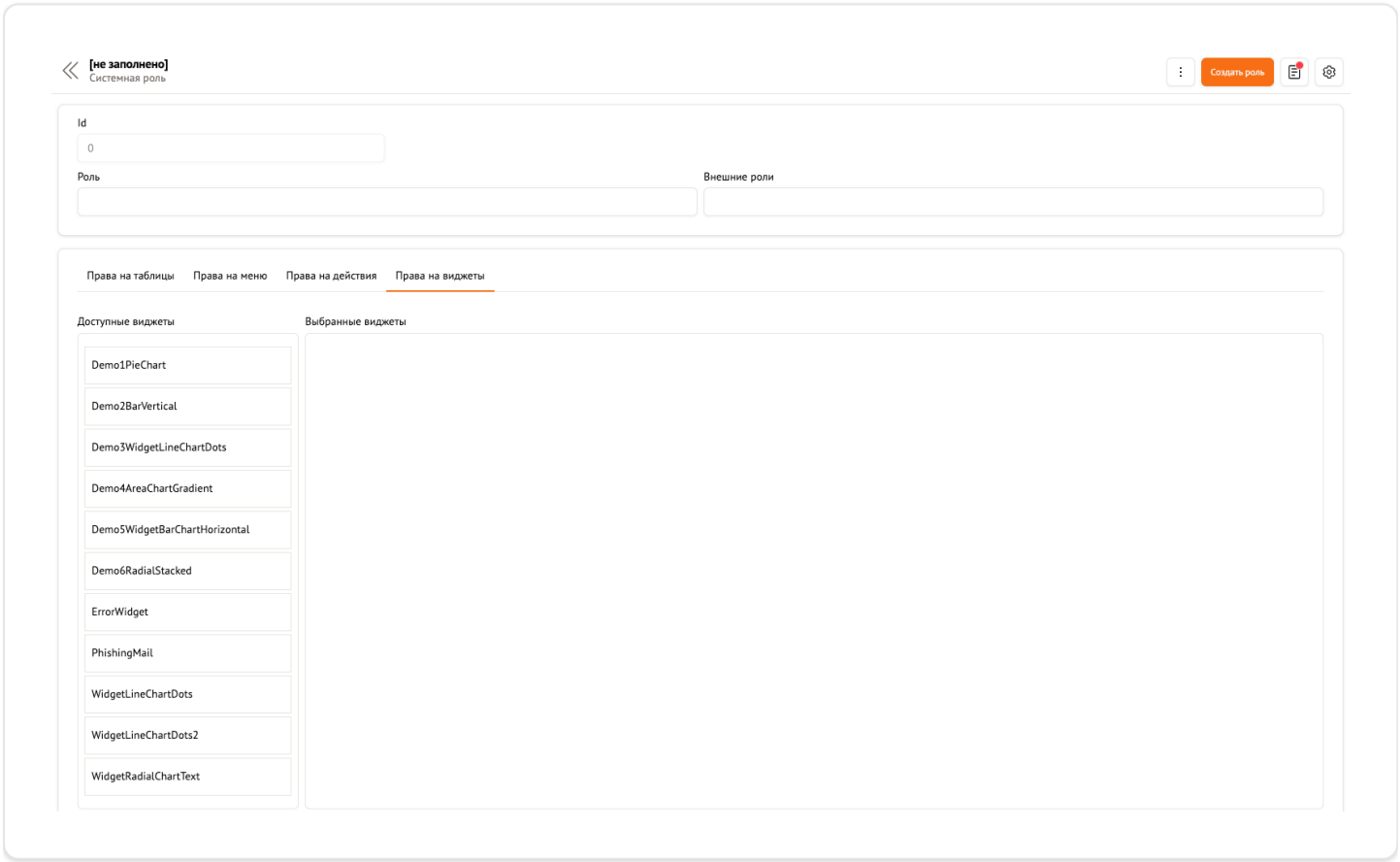
7.6. Создание новой роли

При нажатии кнопки **«Создать роль»** открывается форма создания новой роли.


[не заполнено]
Системная роль
Создать роль
Id
0
Роль
Внешние роли
Права на таблицы
Права на меню
Права на действия
Права на виджеты
Поиск
Заголовок
Название
Чтение
Создание
Редактирование
Удаление
Экран списка
Экран формы
Активы
Assets
Группы доступа
AccessGroup
Группы инцидентов
IncidentGroups
Демо таблица
DemoTable
Инциденты
Incidents
Источники информации
InformationSources
Оборудование
Devices
Объекты инцидентов
IncidentObj
Организации
Organization
Программное обеспечение
Software
Выберите экраны


[не заполнено]
Системная роль
Создать роль
Id
0
Роль
Внешние роли
Права на таблицы
Права на меню
Права на действия
Права на виджеты
Поиск
Меню
Доступ
Администрирование
Настройка коннекторов
Системные таблицы
Действия
Категории действий
Виджеты
Списки
Роли
Пользователи
Меню
Логи
Интеграционные сервисы
Группы доступа
Переодические задачи
API сервисы





На форме доступны следующие параметры:

- Id — автоматически генерируется;
- Роль — обязательное поле;
- Внешние роли — поле для указания внешних ролей (по умолчанию пусто).

Все остальные настройки выполняются через вкладки, аналогично редактированию существующей роли.
После заполнения формы и сохранения роль становится доступной для назначения пользователям.

7.7. Обобщение функциональности

Принцип работы ролей как функциональной единицы

1. Администратор создаёт новую роль через форму;
2. Задаёт имя и внешние роли;
3. Настраивает права доступа по таблицам, меню, действиям и виджетам;
4. Сохраняет роль;
5. Назначает роль пользователям в разделе «Пользователи».

Применимость

- Роли используются для реализации принципа наименьших привилегий;
- Позволяют организовать доступ к данным и функциям в зависимости от должности;
- Поддерживают интеграцию с внешними системами аутентификации;
- Используются в управлении группами доступа и политиками безопасности.

Все роли можно редактировать, копировать и удалять в любое время.

8. Раздел «Пользователи»

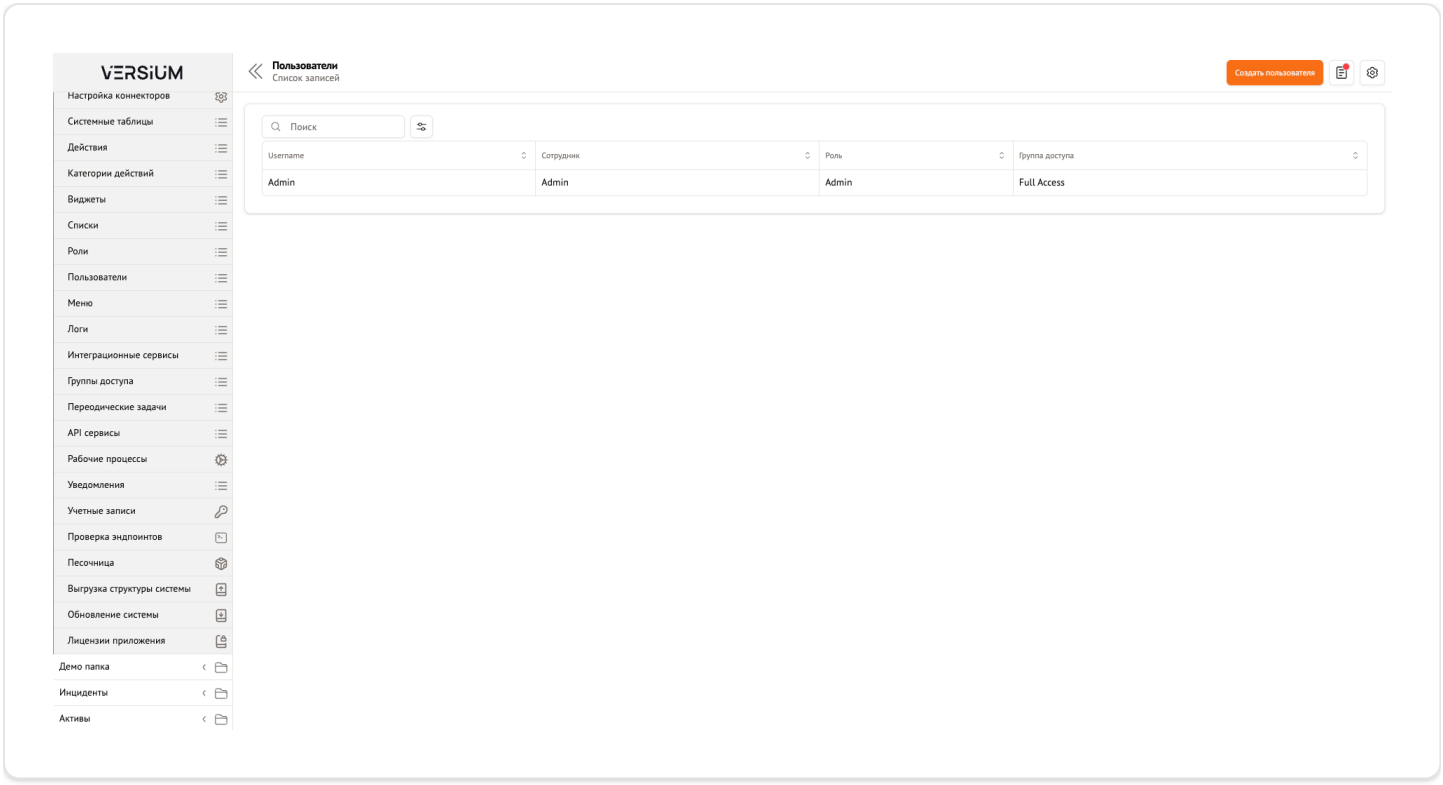
Раздел **«Пользователи»** предназначен для управления учётными записями пользователей системы, их правами доступа и параметрами аутентификации.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять учётные записи;
- Назначать роли и группы доступа;
- Настраивать режимы аутентификации;
- Управлять состоянием пользователей (блокировка, смена пароля).

Каждый пользователь представляет собой функциональную единицу, которая определяет доступ к платформе и её функционалу.



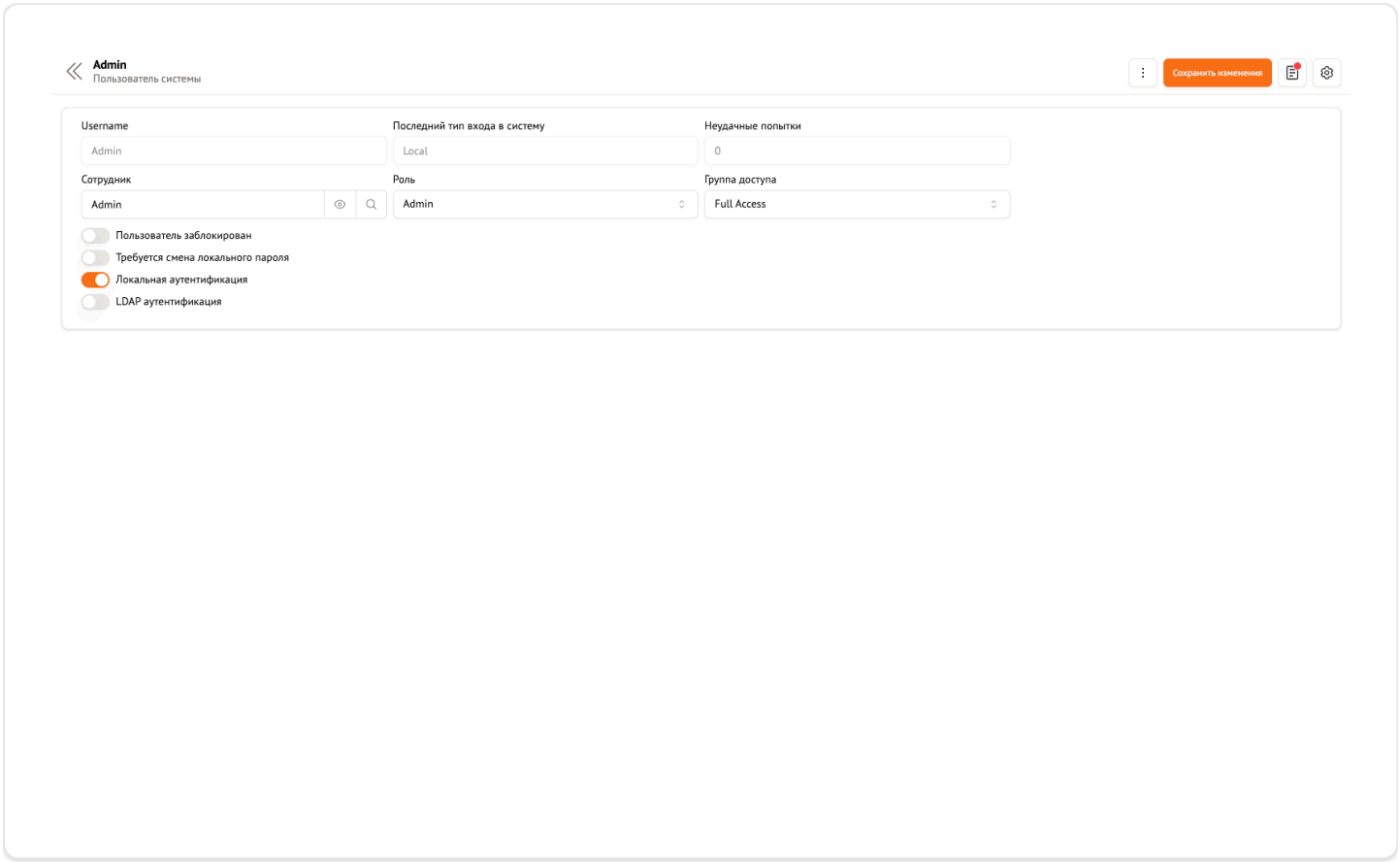
На экране отображается список всех зарегистрированных пользователей с возможностью поиска и сортировки. Каждая запись содержит:

- Username — имя пользователя;
- Сотрудник — связанный сотрудник из базы данных;
- Роль — назначенная роль;
- Группа доступа — группа, к которой принадлежит пользователь.

Кнопка **«Создать пользователя»** позволяет добавить новую учётную запись.

8.1. Форма редактирования пользователя

При переходе к редактированию существующего пользователя открывается форма настройки его параметров.



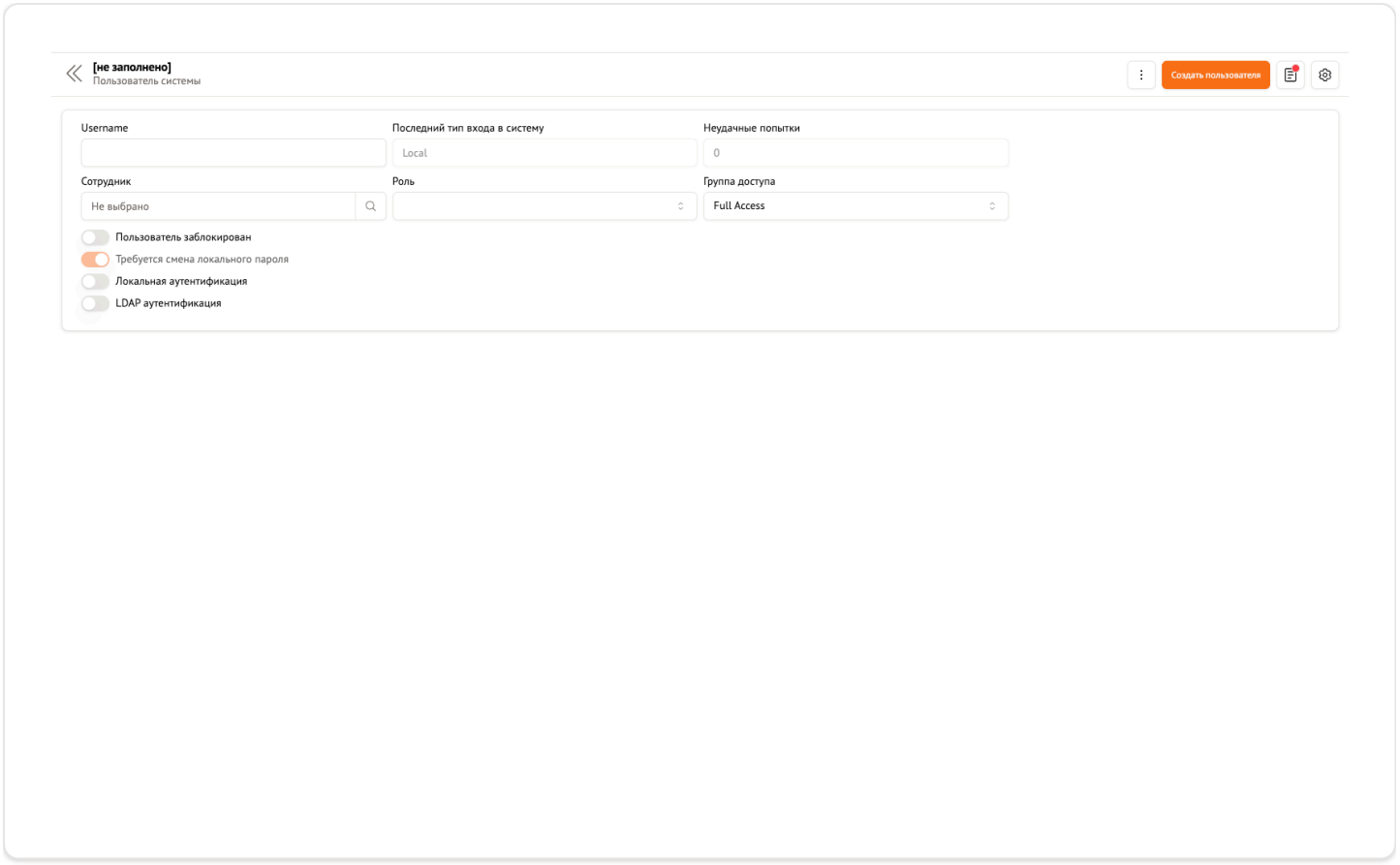
На форме доступны следующие поля:

- Username — имя пользователя (обязательное поле);
- Последний тип входа в систему — тип аутентификации (например, Local, LDAP);
- Неудачные попытки — количество неудачных попыток входа;
- Сотрудник — выбор сотрудника из справочника (может быть связан с данными из HR-системы);
- Роль — выбор роли, определяющей права доступа;
- Группа доступа — выбор группы, определяющей дополнительные ограничения или разрешения;
- Пользователь заблокирован — флаг, определяющий, заблокирован ли пользователь;
- Требуется смена локального пароля — флаг, указывающий, должен ли пользователь изменить пароль при первом входе;
- Локальная аутентификация — флаг, активирующий локальную проверку пароля;
- LDAP аутентификация — флаг, активирующий аутентификацию через LDAP-сервер.

После заполнения формы нажимается кнопка **«Сохранить изменения»**, что применяет настройки к пользователю.

8.2. Создание нового пользователя

При нажатии кнопки **«Создать пользователя»** открывается форма создания новой учётной записи.



На форме доступны следующие параметры:

- Username — обязательное поле;
- Сотрудник — выбор сотрудника (по умолчанию — «Не выбрано»);
- Роль — выбор роли (по умолчанию — пусто);
- Группа доступа — выбор группы (по умолчанию — «Full Access»);
- Флаги: блокировка, смена пароля, тип аутентификации.

После заполнения формы и сохранения учётная запись становится доступной для использования.

8.3. Обобщение функциональности

Принцип работы пользователей как функциональной единицы

1. Администратор создаёт новую учётную запись через форму;
2. Задаёт имя, роль, группу доступа и параметры аутентификации;
3. Сохраняет пользователя;
4. Пользователь может войти в систему с использованием своих учётных данных.

Применимость

- Пользователи используются для обеспечения индивидуального доступа к платформе;
- Поддерживают интеграцию с внешними системами аутентификации (LDAP, AD);
- Позволяют реализовывать политики безопасности (смена пароля, блокировка);
- Используются в управлении доступом к данным и функциям.

Все пользователи можно редактировать, копировать и удалять в любое время.

9. Раздел «Меню»

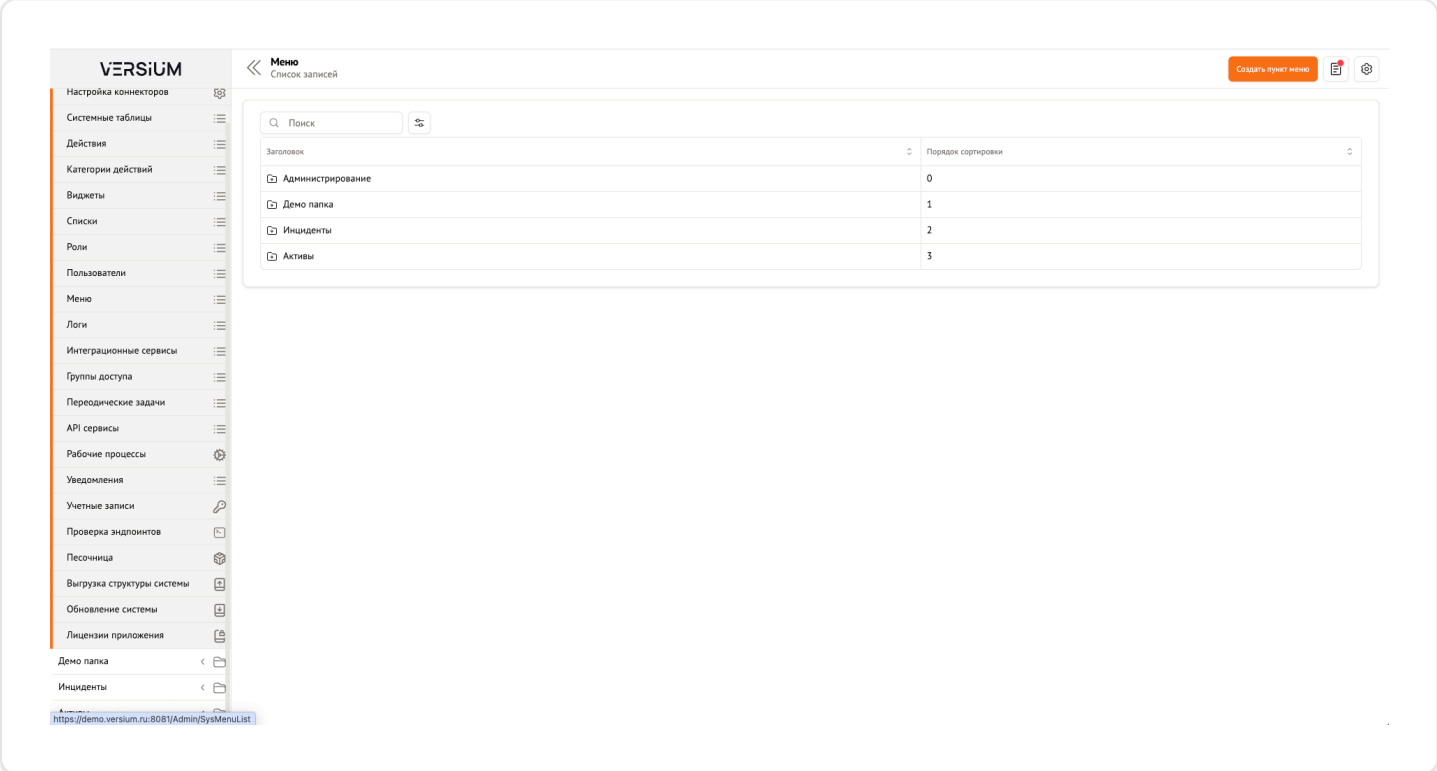
Раздел **«Меню»** предназначен для управления структурой пользовательского интерфейса, определяющей доступ к основным разделам платформы.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять пункты меню;
- Настраивать иерархию разделов (вложенность);
- Управлять порядком отображения элементов;
- Назначать URL и иконки для каждого пункта.

Каждый пункт меню представляет собой функциональную единицу, которая определяет доступ к конкретному разделу или внешнему ресурсу.



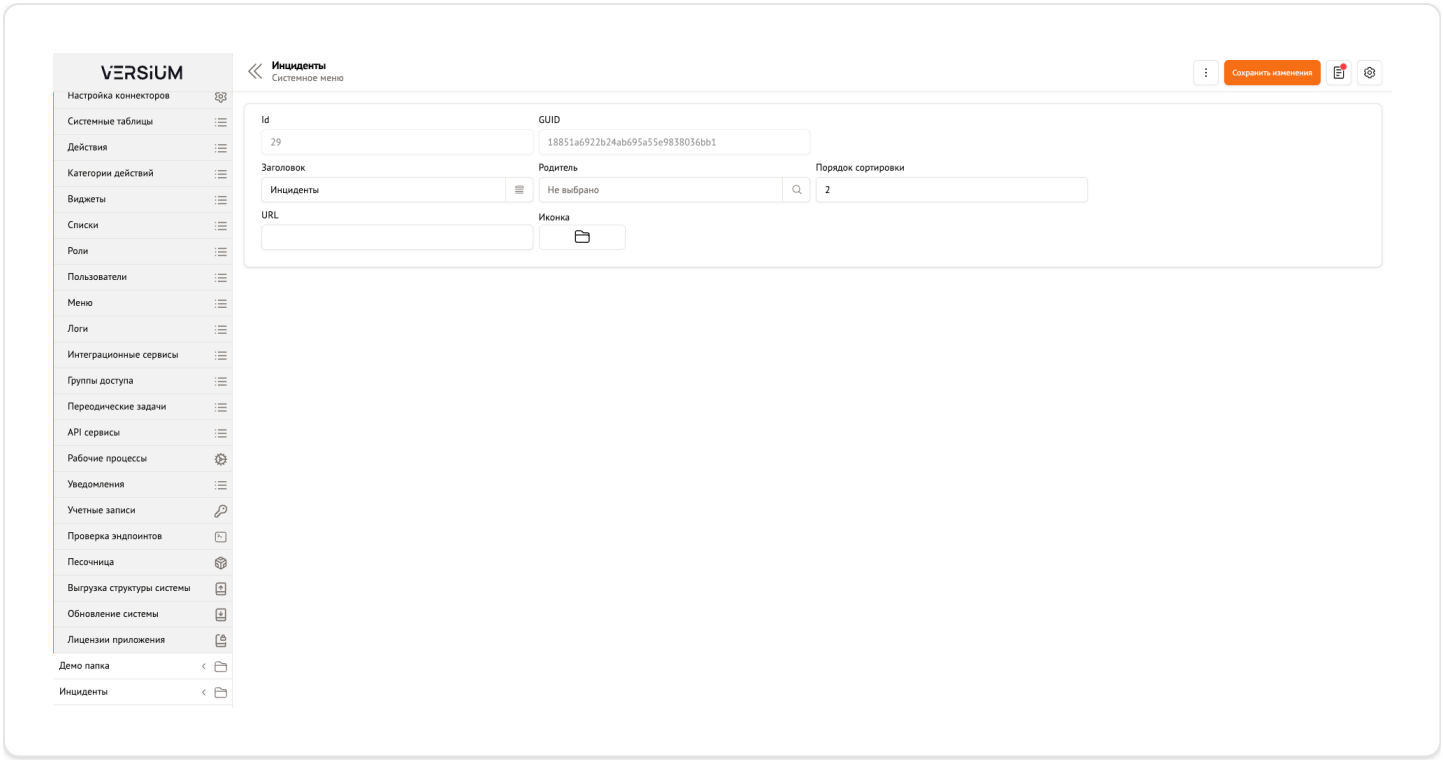
На экране отображается список всех зарегистрированных пунктов меню с возможностью поиска и сортировки. Каждая запись содержит:

- Заголовок — отображаемое название пункта;
- Порядок сортировки — числовое значение, определяющее положение пункта в списке (чем меньше значение, тем выше позиция).

Кнопка **«Создать пункт меню»** позволяет добавить новый элемент в структуру меню.

9.1. Форма редактирования пункта меню

При переходе к редактированию существующего пункта меню открывается форма настройки его параметров.



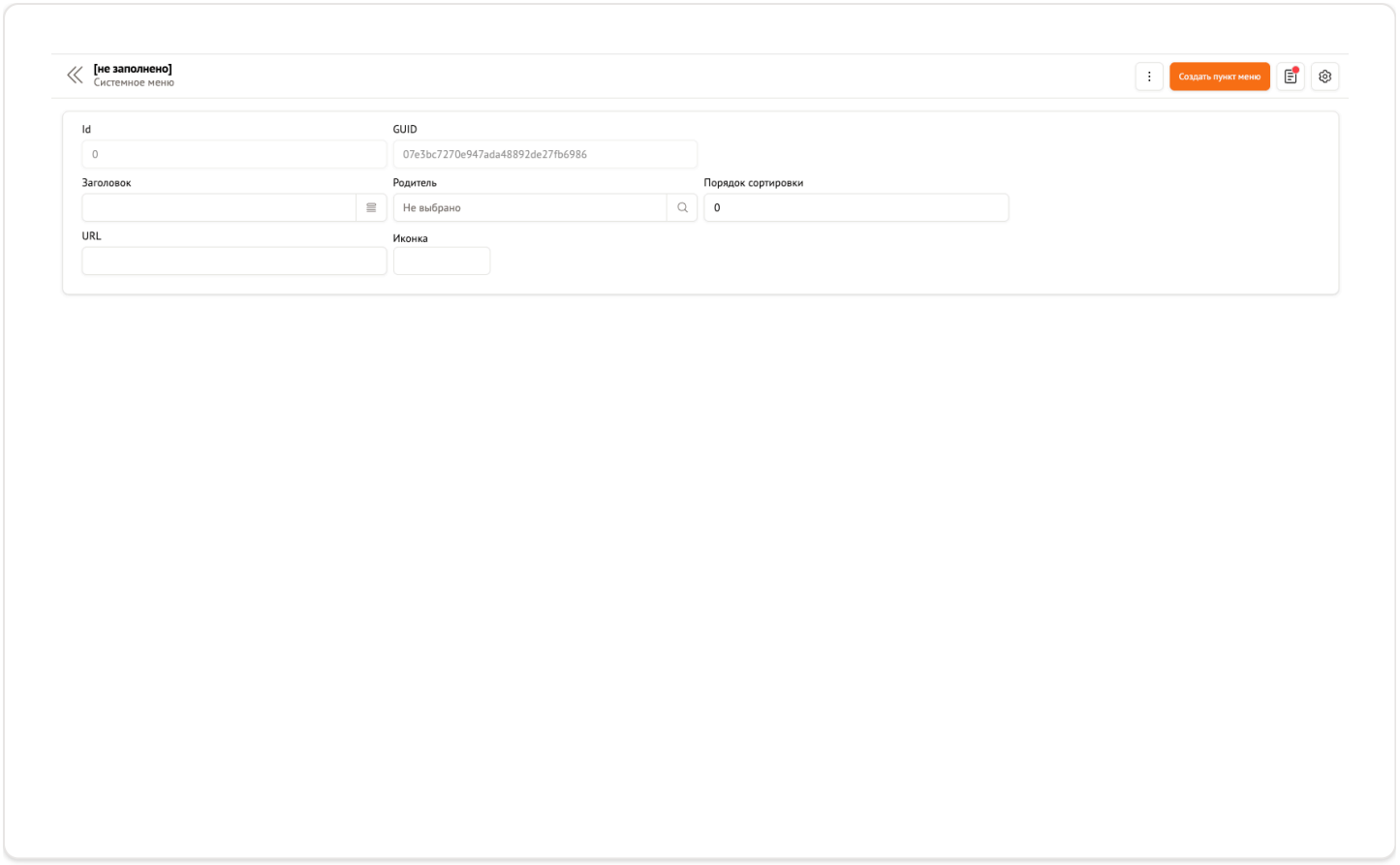
На форме доступны следующие поля:

- Id — уникальный идентификатор пункта (автоматически генерируется при создании);
- GUID — глобальный идентификатор пункта;
- Заголовок — отображаемое название пункта;
- Родитель — выбор родительского пункта для создания вложенной структуры;
- Порядок сортировки — числовое значение, определяющее положение пункта в списке;
- URL — адрес, на который перенаправляется пользователь при клике;
- Иконка — возможность назначить иконку для визуального отличия.

После заполнения формы нажимается кнопка **«Сохранить изменения»**, что применяет настройки к пункту меню.

9.2. Создание нового пункта меню

При нажатии кнопки **«Создать пункт меню»** открывается форма создания нового элемента.



На форме доступны следующие параметры:

- Id — автоматически генерируется;
- GUID — автоматически генерируется;
- Заголовок — обязательное поле;
- Родитель — выбор родительского пункта (по умолчанию — «Не выбрано»);
- Порядок сортировки — по умолчанию 0;
- URL — поле для указания адреса;
- Иконка — поле для выбора иконки.

После заполнения формы и сохранения пункт становится доступным в интерфейсе.

9.3. Обобщение функциональности

Принцип работы пунктов меню как функциональной единицы

1. Администратор создаёт новый пункт меню через форму;
2. Задаёт заголовок, URL и порядок отображения;
3. Выбирает родителя для создания вложенной структуры;
4. Сохраняет пункт;
5. Пункт отображается в боковой панели навигации для пользователей с соответствующими правами.

Применимость

- Пункты меню используются для организации доступа к разделам платформы;

- Поддерживают иерархическую структуру (вложенность);
- Позволяют интегрировать внешние ресурсы через URL;
- Используются в управлении пользовательским интерфейсом и настройке рабочих процессов.

Все пункты меню можно редактировать, копировать и удалять в любое время.

10. Раздел «Логи»

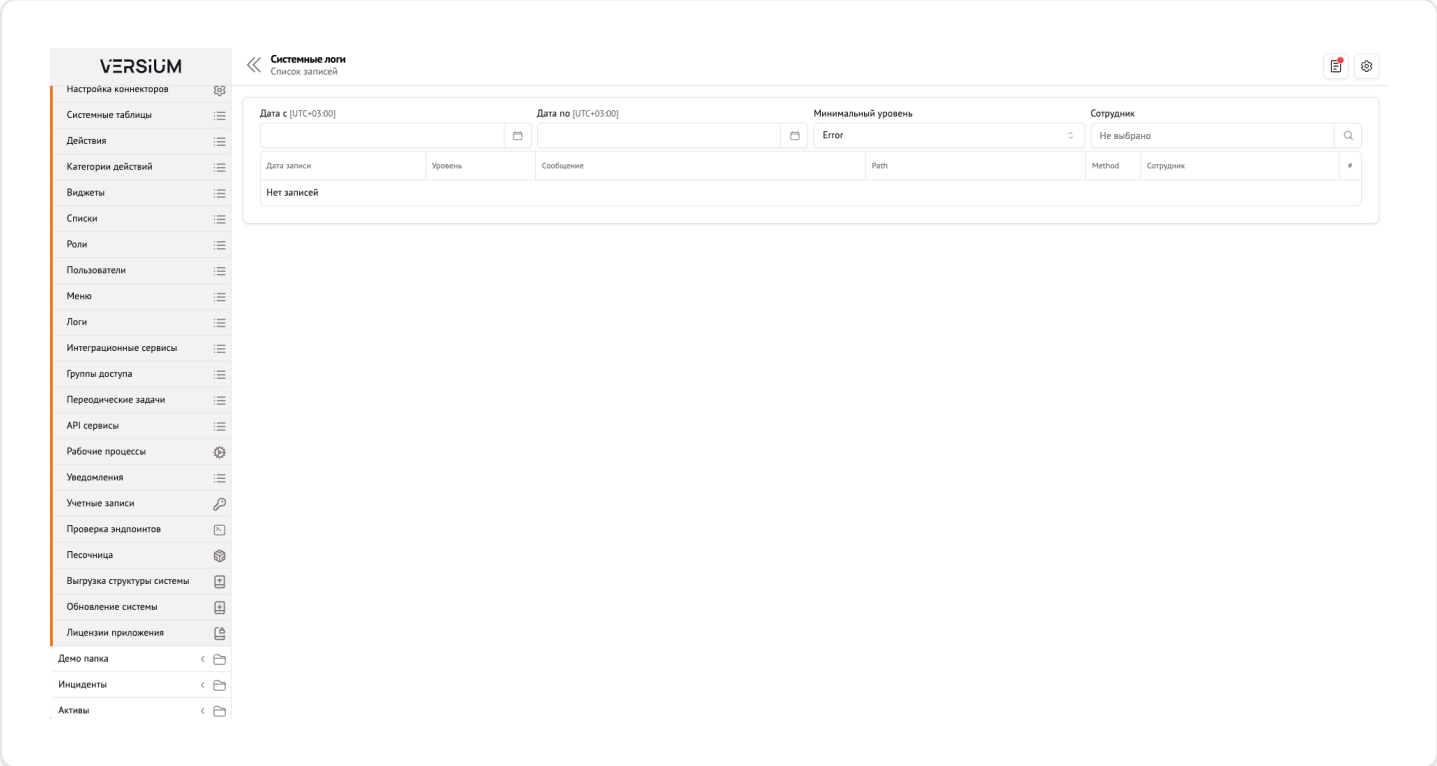
Раздел **«Логи»** предназначен для просмотра и анализа событий, происходящих в системе.

Назначение

Этот модуль позволяет администраторам:

- Просматривать системные события (ошибки, предупреждения, информационные сообщения);
- Фильтровать записи по времени, уровню важности и пользователю;
- Анализировать поведение системы и действия пользователей;
- Выявлять причины сбоев и неисправностей.

Каждая запись лога представляет собой функциональную единицу, содержащую информацию о конкретном событии.



На экране отображается список всех зарегистрированных записей лога с возможностью фильтрации и поиска. В верхней части формы расположены поля для настройки фильтра:

- Дата с — начало временного диапазона (в формате UTC+03:00);
- Дата по — конец временного диапазона (в формате UTC+03:00);
- Минимальный уровень — минимальный уровень серьёзности события (например, `Error`, `Warning`, `Info`);
- Сотрудник — выбор пользователя, чьи действия нужно отфильтровать.

После установки фильтров нажимается кнопка поиска (иконка лупы), что выводит соответствующие записи.

Таблица содержит следующие столбцы:

- Дата записи — временная метка события;
- Уровень — тип события (Error, Warning, Info);
- Сообщение — текстовое описание события;
- Path — путь к ресурсу или компоненту, где произошло событие;
- Method — HTTP-метод или действие, вызвавшее событие;
- Сотрудник — имя пользователя, связанный с событием;
- **— порядковый номер записи.**

10.1. Обобщение функциональности

Принцип работы логов как функциональной единицы

1. Администратор открывает раздел «Логи»;
2. Настраивает фильтры по дате, уровню и пользователю;
3. Инициирует поиск;
4. Анализирует полученные записи.

Применимость

- Логи используются для мониторинга стабильности платформы;
- Позволяют выявлять ошибки в работе системных компонентов;
- Поддерживают аудит действий пользователей;
- Используются при диагностике инцидентов и настройке систем безопасности.

Все записи логов доступны для просмотра, но не могут быть изменены или удалены напрямую.

11. Раздел «Интеграционные сервисы»

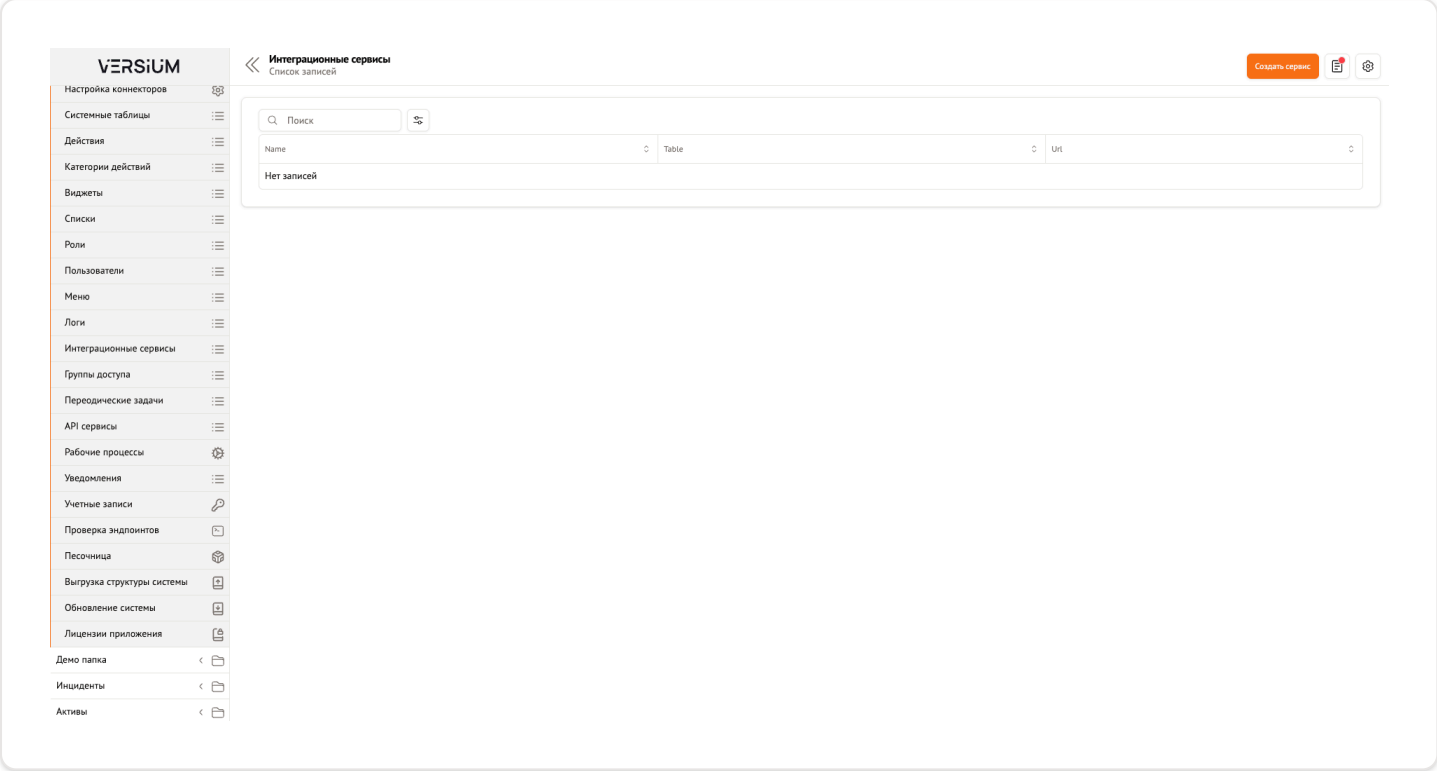
Раздел **«Интеграционные сервисы»** предназначен для управления внешними интеграциями платформы, позволяющими обмениваться данными с другими системами (например, SIEM, EDR, Active Directory).

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять интеграционные сервисы;
- Настраивать маппинг данных между источником и целевой таблицей;
- Управлять параметрами обработки и логированием;
- Интегрировать платформу с внешними API и системами.

Каждый интеграционный сервис представляет собой функциональную единицу, которая определяет способ и правила передачи данных.



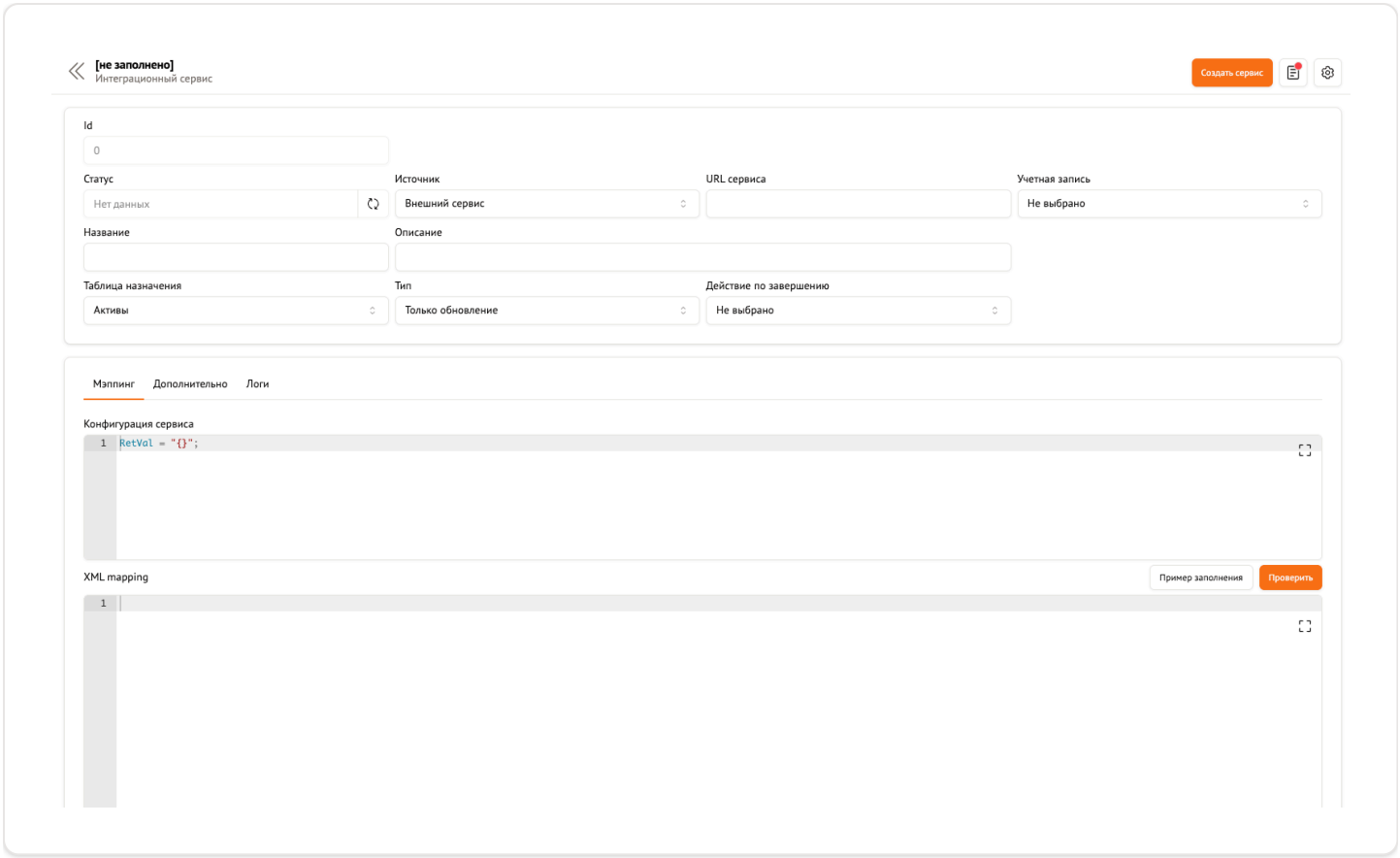
На экране отображается список всех зарегистрированных интеграционных сервисов с возможностью поиска и сортировки. Каждая запись содержит:

- Name — название сервиса;
- Table — целевая таблица, куда загружаются данные;
- Url — адрес внешнего источника данных.

Кнопка **«Создать сервис»** позволяет добавить новый интеграционный сервис.

11.1. Форма редактирования интеграционного сервиса

При переходе к редактированию существующего сервиса открывается форма настройки его параметров.

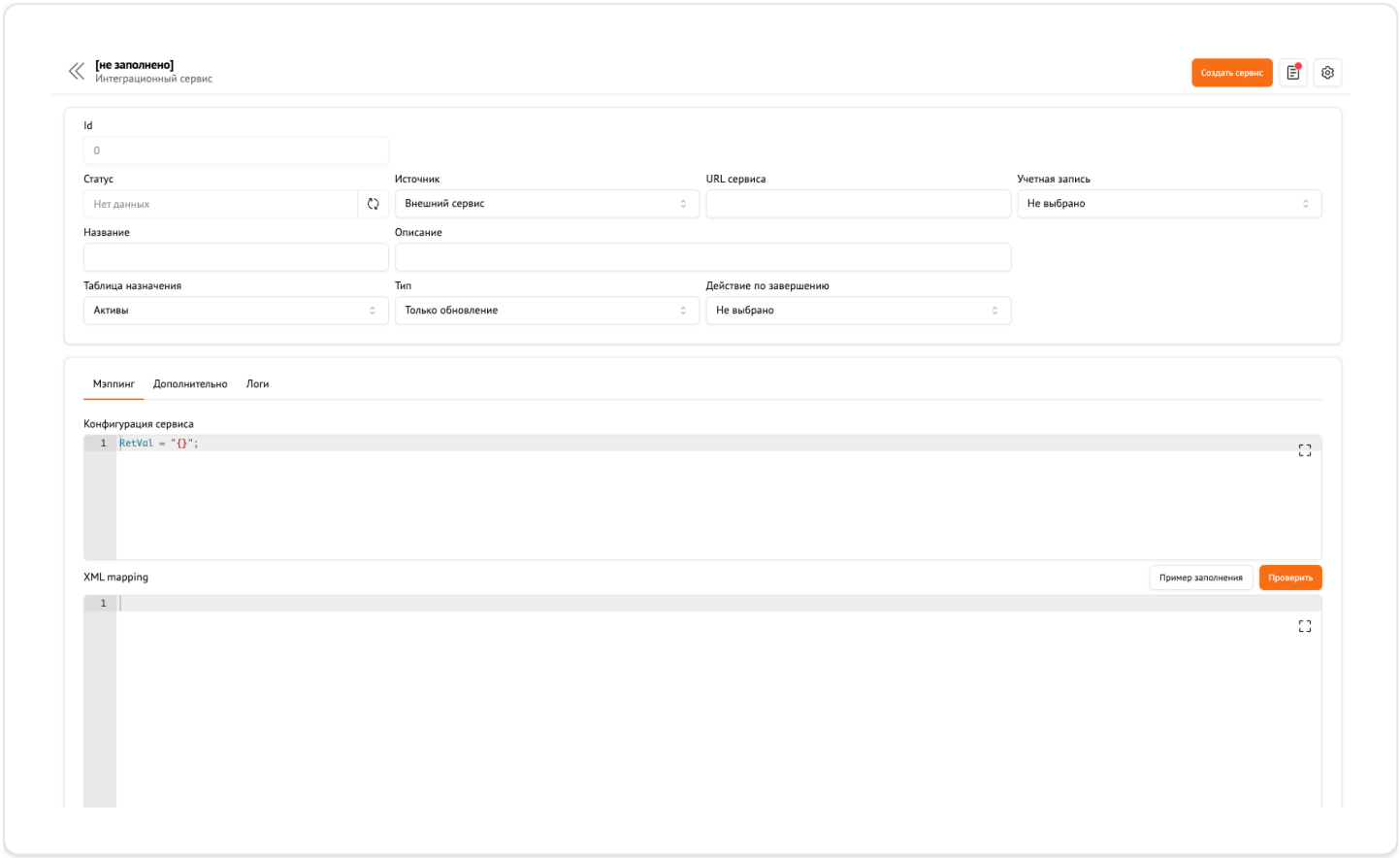


На форме доступны следующие поля:

- Id — уникальный идентификатор сервиса (автоматически генерируется при создании);
- Статус — текущее состояние сервиса (например, «Нет данных», «Активен»);
- Источник — тип источника данных (например, «Внешний сервис»);
- URL сервиса — адрес внешнего API или системы;
- Учетная запись — выбор учётной записи для аутентификации;
- Название — отображаемое имя сервиса;
- Описание — подробное описание назначения сервиса;
- Таблица назначения — выбор системной таблицы, куда будут загружаться данные;
- Тип — режим работы сервиса:
  - `Только обновление` — обновление существующих записей;
  - `Обновление и создание` — создание новых и обновление существующих;
- Действие по завершению — действие, выполняемое после завершения обработки (например, «Закрыть инцидент»).

11.2. Вкладка «Мэппинг»

На вкладке **Мэппинг** настраивается соответствие полей между источником данных и целевой таблицей.



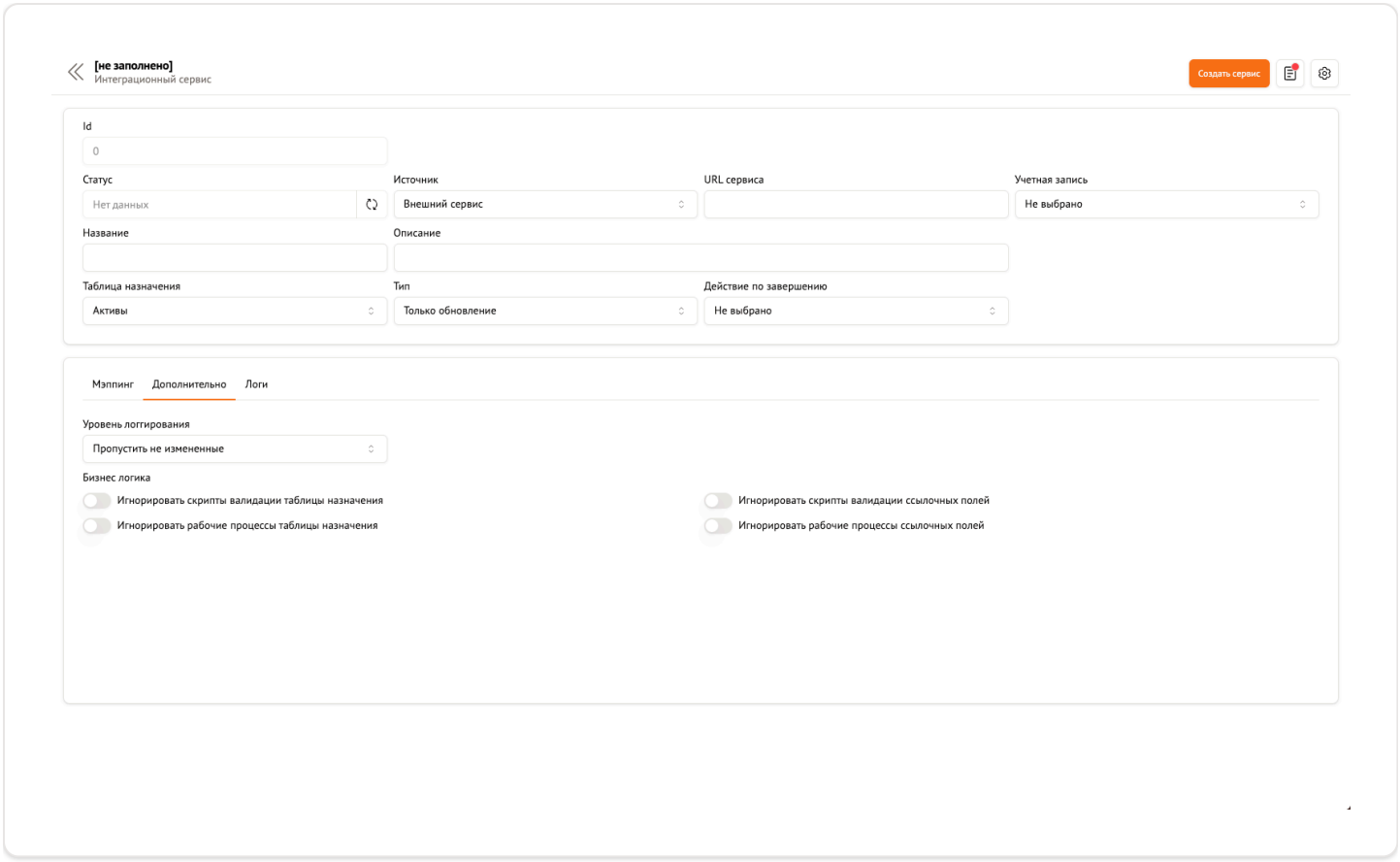
Включает два поля:

- **Конфигурация сервиса** — текстовое поле для ввода скрипта, определяющего логику обработки данных (например, преобразование значений, фильтрация);
- **XML mapping** — поле для ввода XML-описания структуры данных, используемой при парсинге ответа от внешнего сервиса.

Кнопка **«Пример заполнения»** предоставляет шаблон для корректного заполнения поля; Кнопка **«Проверить»** позволяет протестировать корректность мэппинга.

11.3. Вкладка «Дополнительно»

На вкладке **Дополнительно** настраиваются дополнительные параметры поведения сервиса.

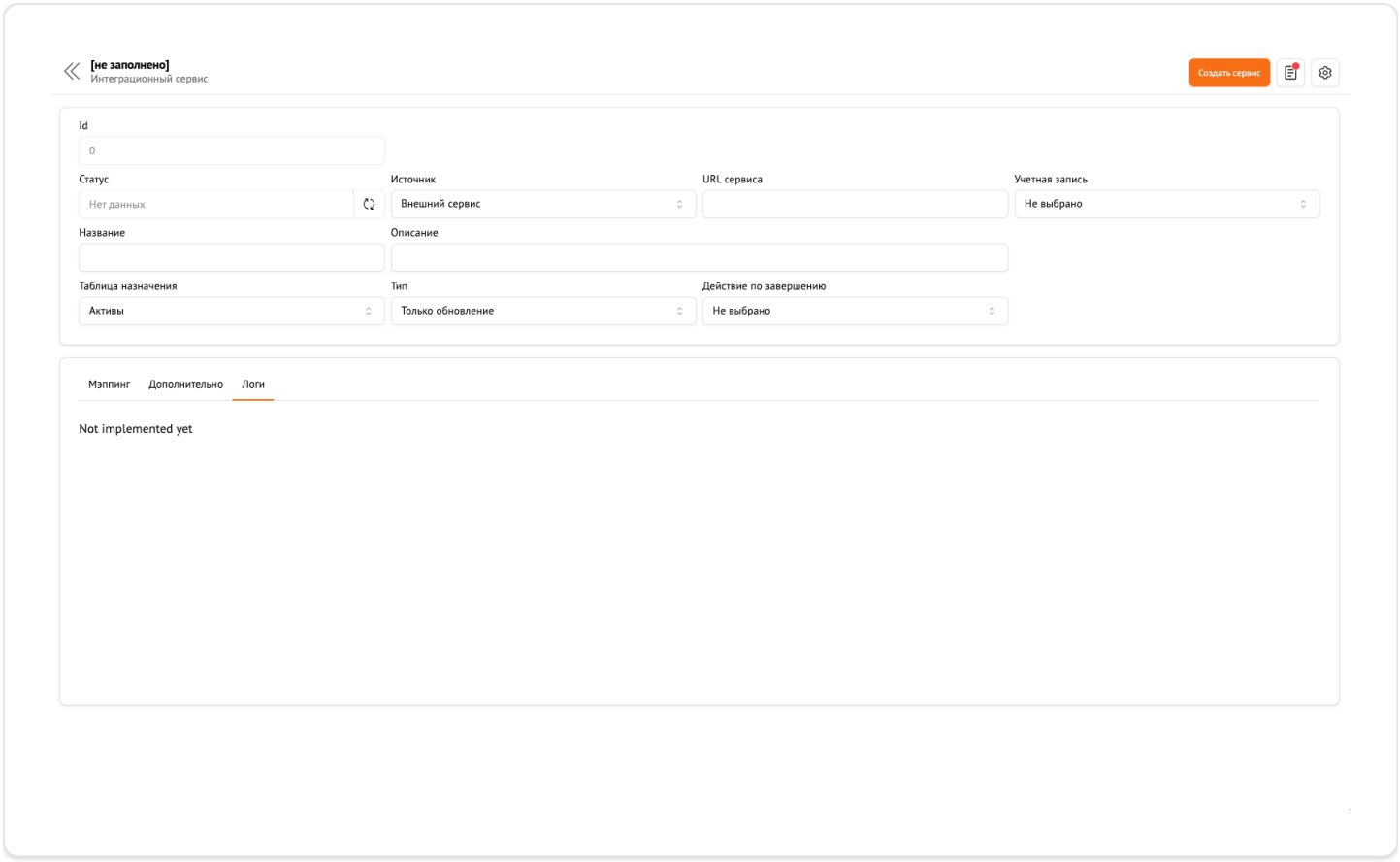


Включает следующие параметры:

- **Уровень логирования** — определяет детальность записей в логах:
  - Пропустить не изменённые — не логировать записи, которые не были изменены;
  - Логировать все — сохранять все операции;
  - Логировать изменения — только изменения.
- **Бизнес логика** — набор флагов для управления поведением:
  - Игнорировать скрипты валидации таблицы назначения;
  - Игнорировать рабочие процессы таблицы назначения;
  - Игнорировать скрипты валидации ссылочных полей;
  - Игнорировать рабочие процессы ссылочных полей.

11.4. Вкладка «Логи»

На вкладке **Логи** отображается история выполнения интеграционного сервиса.



В настоящее время функционал этой вкладки не реализован. На экране отображается сообщение: **Not implemented yet**.

Плановые возможности включают:

- Отображение истории запусков сервиса;
- Просмотр результатов выполнения (успешные/неудачные);
- Анализ ошибок и исключений;
- Фильтрацию по дате, статусу и другим параметрам.

11.5. Обобщение функциональности

Принцип работы интеграционных сервисов как функциональной единицы

1. Администратор создаёт новый сервис через форму;
2. Задаёт источник, URL, учетную запись и целевую таблицу;
3. Настраивает мэппинг данных и бизнес-логику;
4. Сохраняет сервис;
5. Сервис начинает периодически получать данные из внешнего источника и обновлять целевую таблицу.

Применимость

- Интеграционные сервисы используются для автоматизации сбора данных из внешних систем;
- Поддерживают работу с REST API, XML, JSON;
- Позволяют настраивать сложные правила обработки данных;

- Используются в процессах мониторинга, анализа и реагирования на инциденты.

Все интеграционные сервисы можно редактировать, копировать и удалять в любое время.

12. Раздел «Группы доступа»

Раздел **«Группы доступа»** предназначен для управления иерархическими группами пользователей, используемыми для организации прав доступа к объектам платформы.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять группы доступа;
- Организовывать иерархическую структуру (вложенность);
- Назначать группы пользователям и объектам;
- Управлять ограничениями на уровне групп.

Каждая группа доступа представляет собой функциональную единицу, которая используется для определения контекста доступа к данным и объектам.

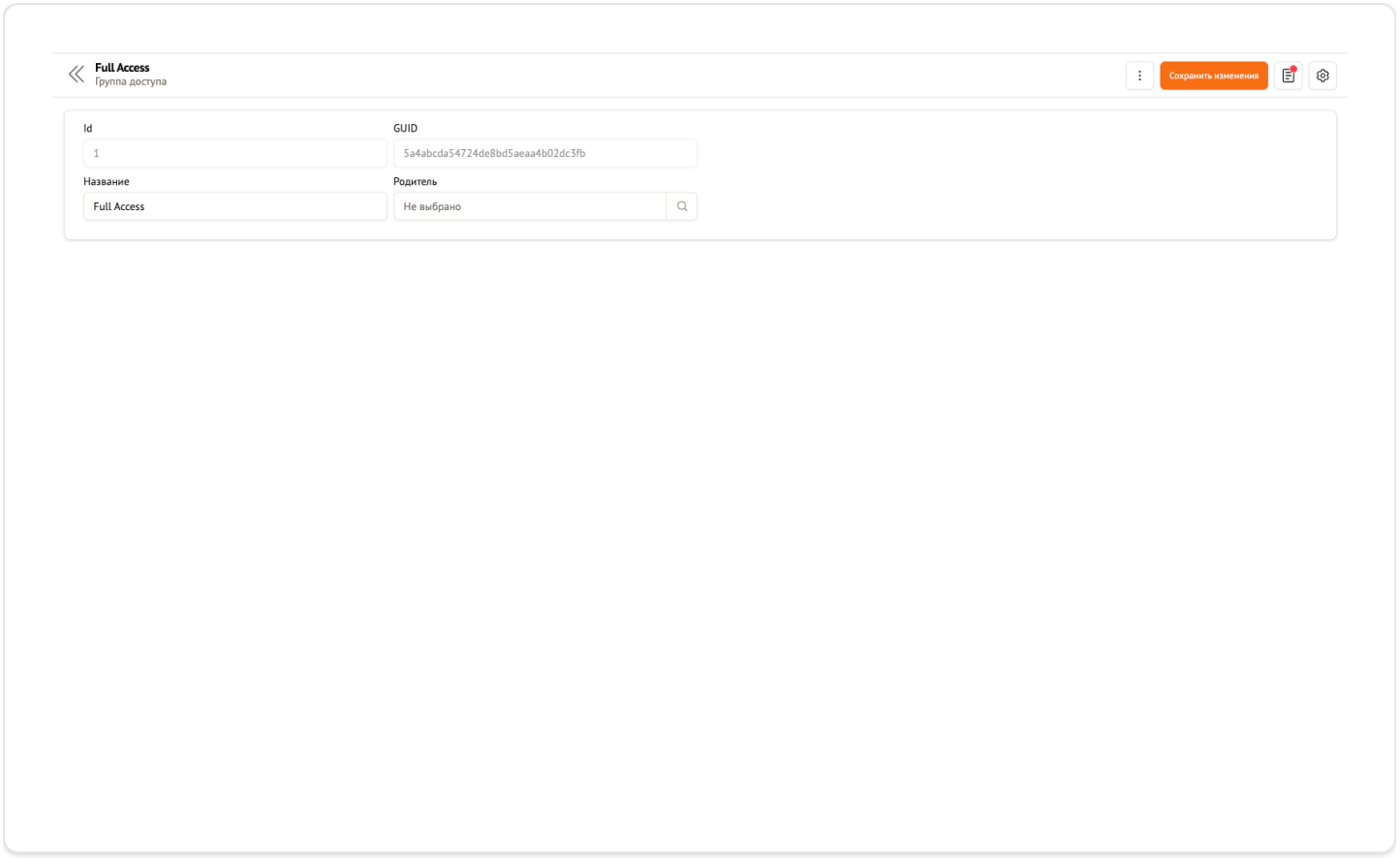
На экране отображается список всех зарегистрированных групп доступа с возможностью поиска и сортировки. Каждая запись содержит:

- FullName — отображаемое имя группы.

Кнопка **«Создать группу доступа»** позволяет добавить новую группу.

12.1. Форма редактирования группы доступа

При переходе к редактированию существующей группы открывается форма настройки её параметров.



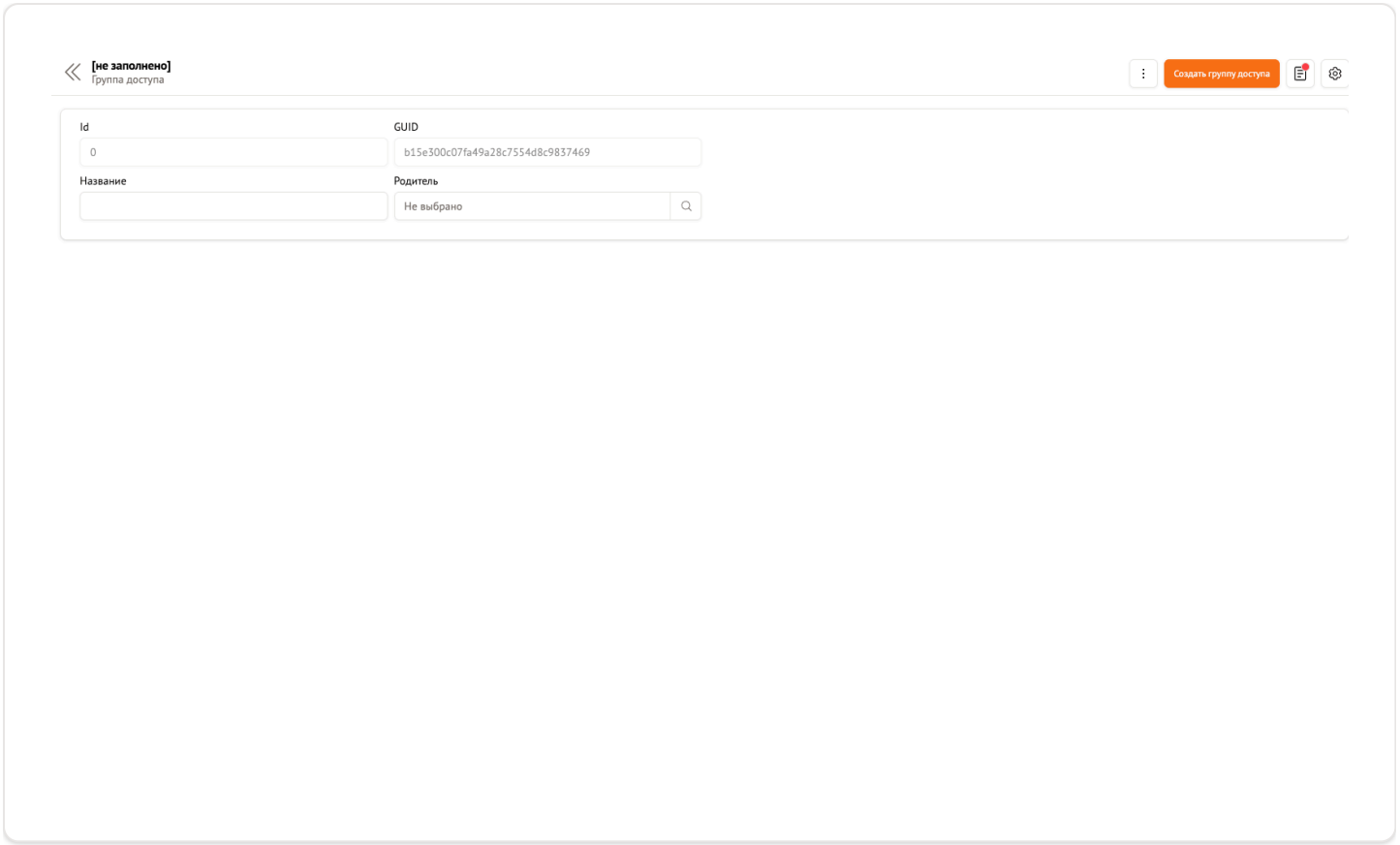
На форме доступны следующие поля:

- Id — уникальный идентификатор группы (автоматически генерируется при создании);
- GUID — глобальный идентификатор группы;
- Название — отображаемое имя группы;
- Родитель — выбор родительской группы для создания вложенной структуры.

После заполнения формы нажимается кнопка **«Сохранить изменения»**, что применяет настройки к группе.

12.2. Создание новой группы доступа

При нажатии кнопки **«Создать группу доступа»** открывается форма создания новой группы.



На форме доступны следующие параметры:

- Id — автоматически генерируется;
- GUID — автоматически генерируется;
- Название — обязательное поле;
- Родитель — выбор родительской группы (по умолчанию — «Не выбрано»).

После заполнения формы и сохранения группа становится доступной для назначения пользователям и объектам.

12.3. Обобщение функциональности

Принцип работы групп доступа как функциональной единицы

1. Администратор создаёт новую группу через форму;
2. Задаёт имя и родителя для создания иерархии;
3. Сохраняет группу;
4. Назначает группу пользователям или объектам в других разделах системы.

Применимость

- Группы доступа используются для реализации политик безопасности на уровне объектов;
- Поддерживают вложенность (например, «Департамент IT → Отдел безопасности»);
- Позволяют ограничивать доступ к данным по группам;
- Используются в настройке прав доступа к активам, инцидентам и другим объектам.

Все группы доступа можно редактировать, копировать и удалять в любое время.

13. Раздел «Периодические задачи»

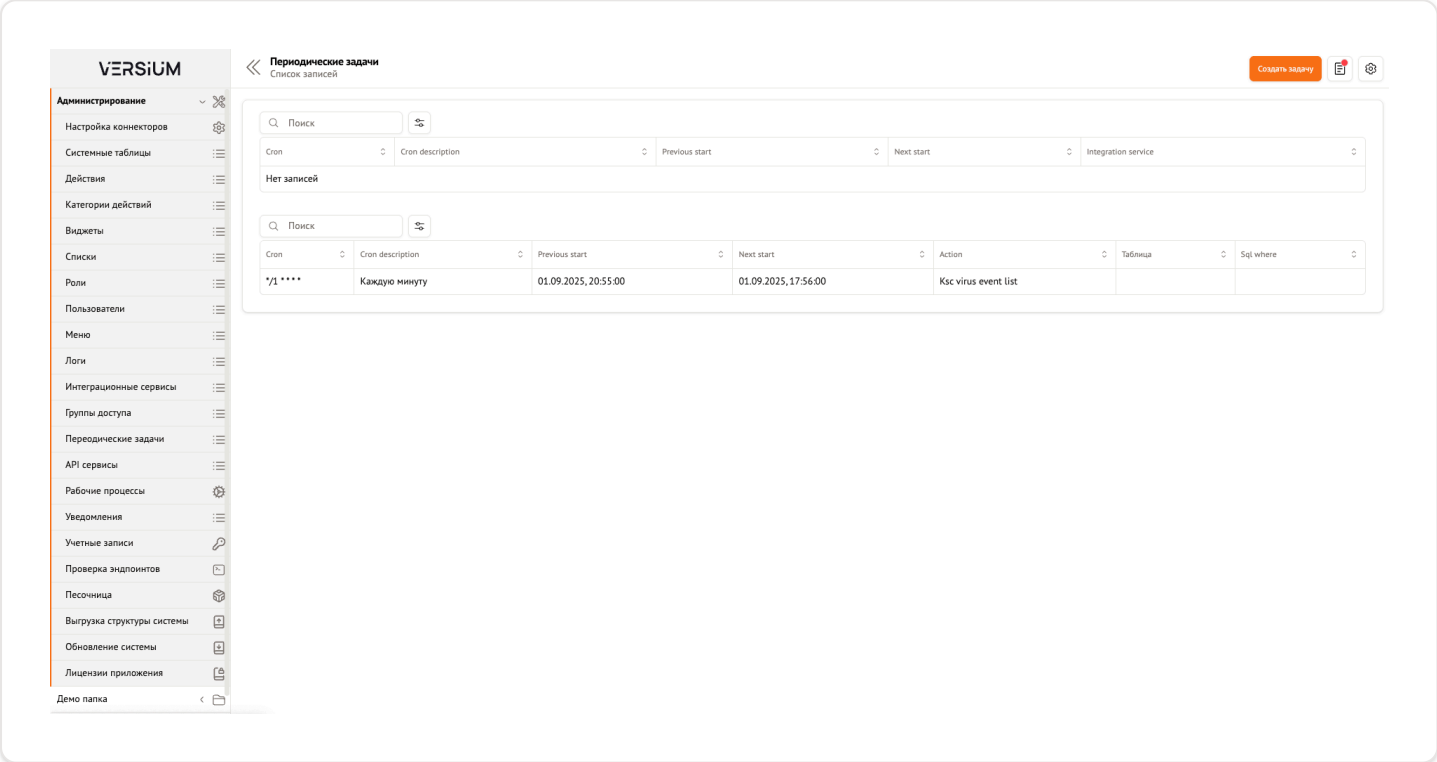
Раздел **«Периодические задачи»** предназначен для управления фоновыми процессами, которые выполняются автоматически по заданному расписанию.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять периодические задачи;
- Настраивать расписание выполнения (через cron-выражение);
- Управлять действиями и интеграциями, запускаемыми по расписанию;
- Мониторить историю запусков и состояние задач.

Каждая задача представляет собой функциональную единицу, которая выполняется в фоновом режиме и может быть связана с действием или интеграционным сервисом.



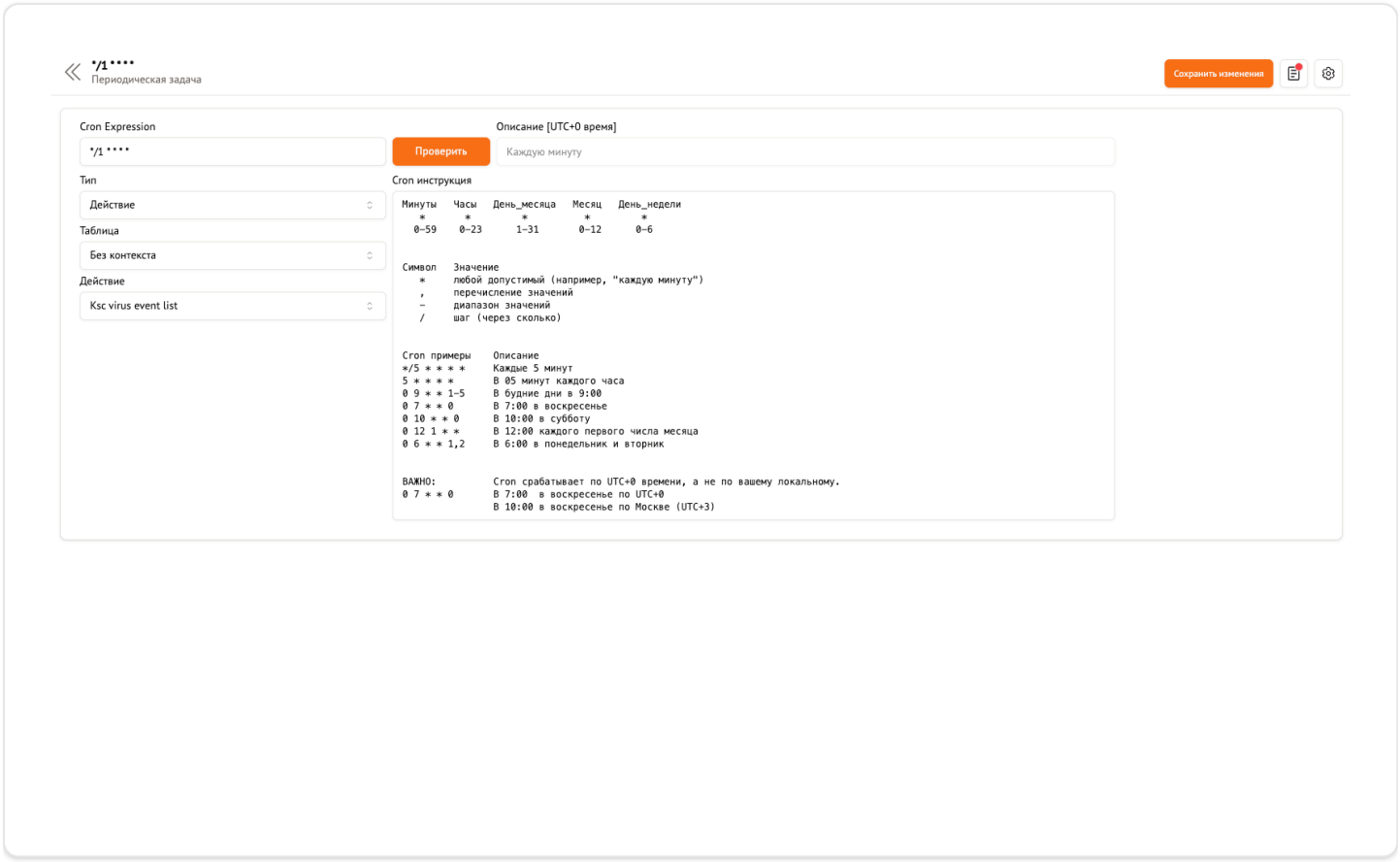
На экране отображается список всех зарегистрированных задач с возможностью поиска и сортировки. Каждая запись содержит:

- Cron — cron-выражение, определяющее расписание;
- Cron description — текстовое описание расписания;
- Previous start — дата и время последнего запуска;
- Next start — дата и время следующего запуска;
- Action — действие, которое будет выполнено;
- Таблица — целевая таблица (если применимо);
- Sql where — условие выборки данных (если применимо).

Кнопка **«Создать задачу»** позволяет добавить новую задачу.

13.1. Форма редактирования периодической задачи

При переходе к редактированию существующей задачи открывается форма настройки её параметров.



На форме доступны следующие поля:

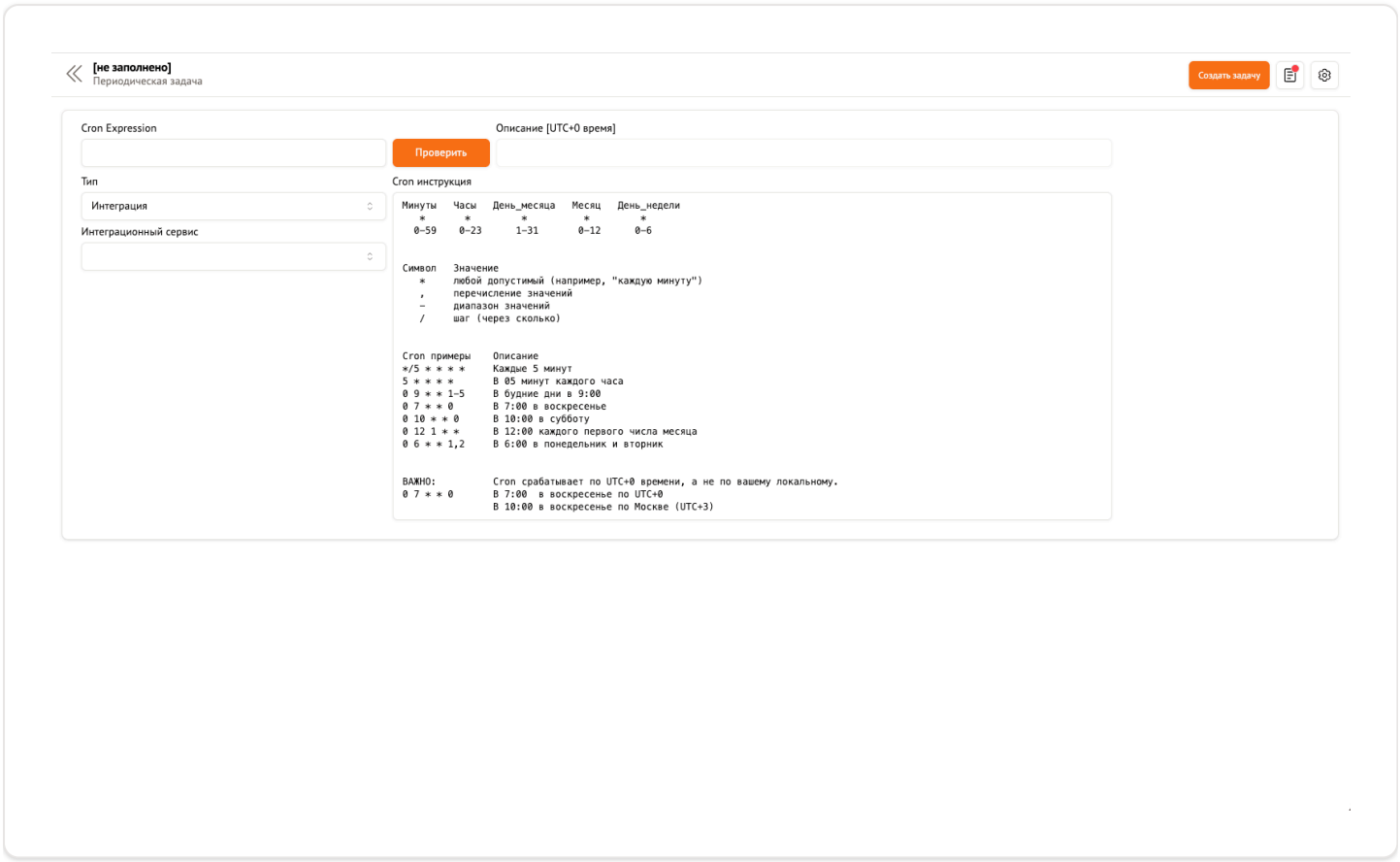
- Cron Expression — строка с cron-выражением (например, `*/1 * * * *`);
- Описание [UTC+0 время] — текстовое описание расписания;
- Тип — тип задачи:
  - `Действие` — запуск системного действия;
  - `Интеграция` — запуск интеграционного сервиса.
- Таблица — выбор целевой таблицы (по умолчанию — «Без контекста»);
- Действие — выбор действия, которое будет выполнено;
- Интеграционный сервис — выбор сервиса, который будет запущен.

Кнопка **«Проверить»** позволяет проверить корректность cron-выражения и получить его текстовое описание.
В правой части формы отображается справочная информация:

- Символы и их значения (*, /, -, ,);
- Примеры cron-выражений;
- Важное замечание: задачи срабатывают по UTC+0, а не по локальному времени.

13.2. Создание новой периодической задачи

При нажатии кнопки **«Создать задачу»** открывается форма создания новой задачи.



На форме доступны следующие параметры:

- Cron Expression — обязательное поле;
- Описание [UTC+0 время] — рекомендуется заполнить;
- Тип — выбор типа задачи;
- Таблица — выбор целевой таблицы (если применимо);
- Действие — выбор действия (если тип — «Действие»);
- Интеграционный сервис — выбор сервиса (если тип — «Интеграция»).

После заполнения формы и сохранения задача становится активной и будет выполняться по расписанию.

13.3. Обобщение функциональности

Принцип работы периодических задач как функциональной единицы

1. Администратор создаёт новую задачу через форму;
2. Задаёт cron-выражение и тип задачи;
3. Выбирает действие или интеграционный сервис;
4. Сохраняет задачу;
5. Задача начинает выполняться по расписанию.

Применимость

- Периодические задачи используются для автоматизации повторяющихся операций;
- Поддерживают работу с cron-выражениями (стандарт Unix);
- Позволяют запускать действия и интеграции в заданное время;
- Используются в процессах мониторинга, обновления данных и аналитики.

Все периодические задачи можно редактировать, копировать и удалять в любое время.

14. Раздел «API сервисы»

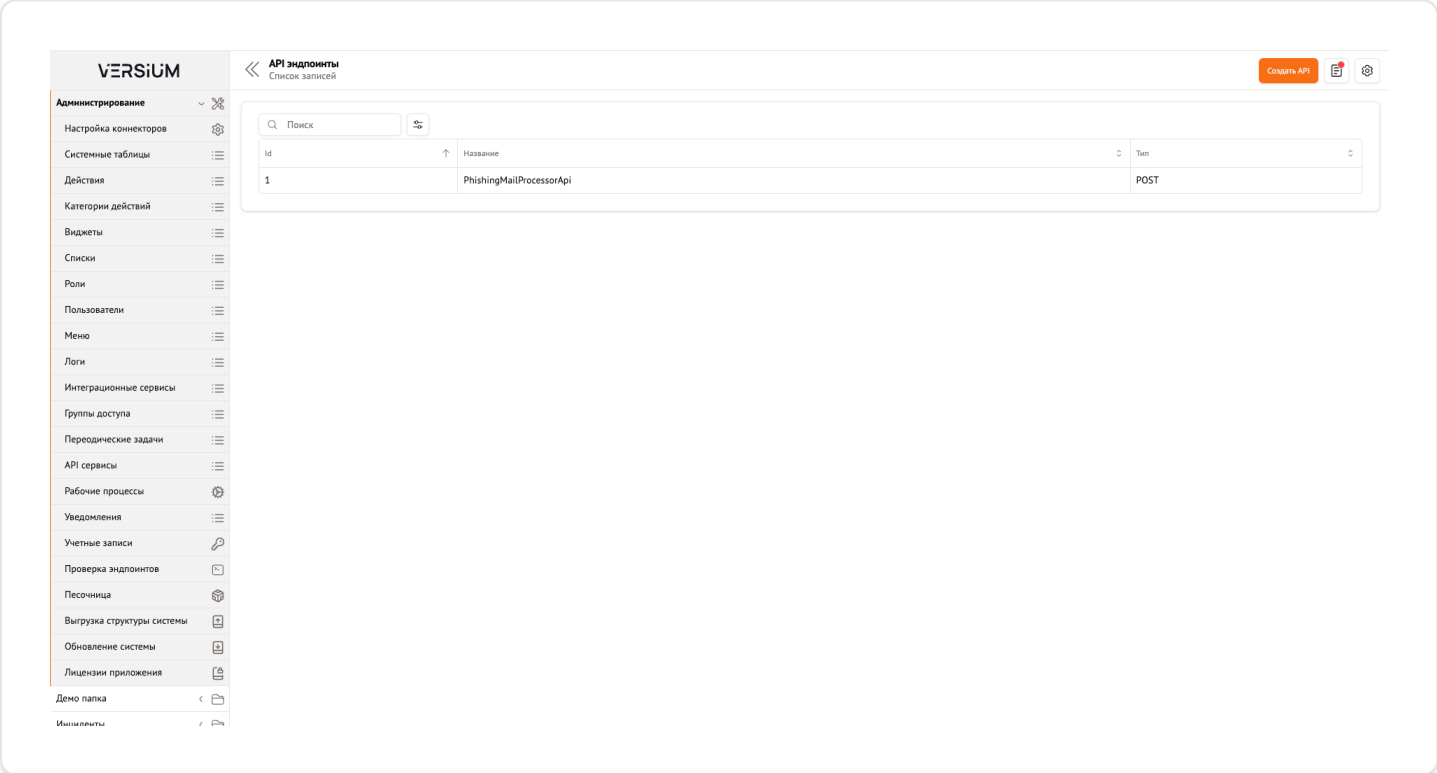
Раздел **«API сервисы»** предназначен для управления пользовательскими API-эндпоинтами, позволяющими интегрировать платформу с внешними системами через HTTP-запросы.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять API-эндпоинты;
- Настраивать обработку входящих запросов (GET, POST и др.);
- Писать пользовательский код для обработки данных;
- Интегрировать платформу с внешними приложениями и системами.

Каждый API-эндпоинт представляет собой функциональную единицу, которая отвечает на HTTP-запросы и выполняет заданную логику.



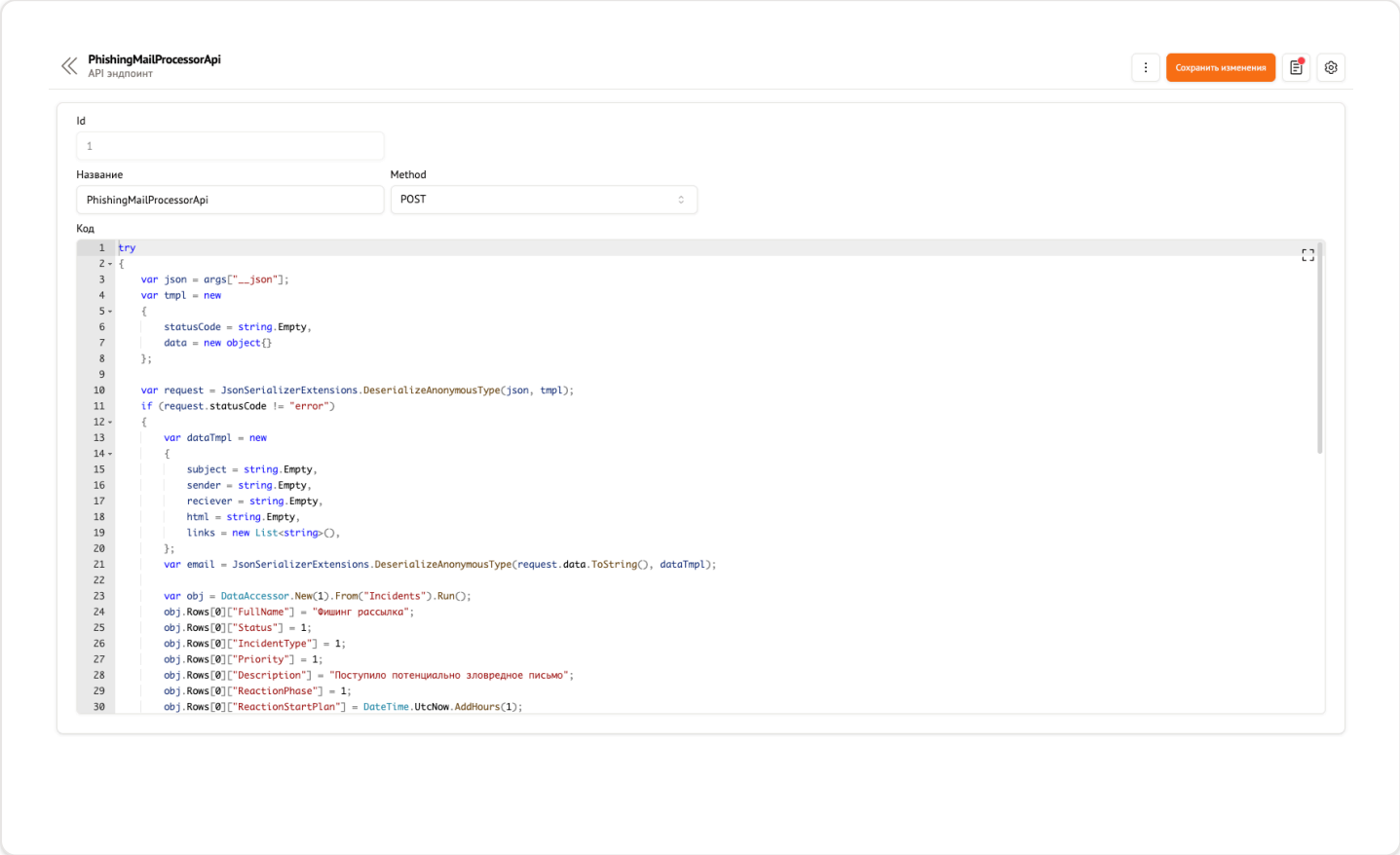
На экране отображается список всех зарегистрированных API-эндпоинтов с возможностью поиска и сортировки. Каждая запись содержит:

- Id — уникальный идентификатор эндпоинта;
- Название — отображаемое имя;
- Тип — HTTP-метод (например, `POST`, `GET`).

Кнопка **«Создать API»** позволяет добавить новый эндпоинт.

14.1. Форма редактирования API-эндпоинта

При переходе к редактированию существующего эндпоинта открывается форма настройки его параметров.



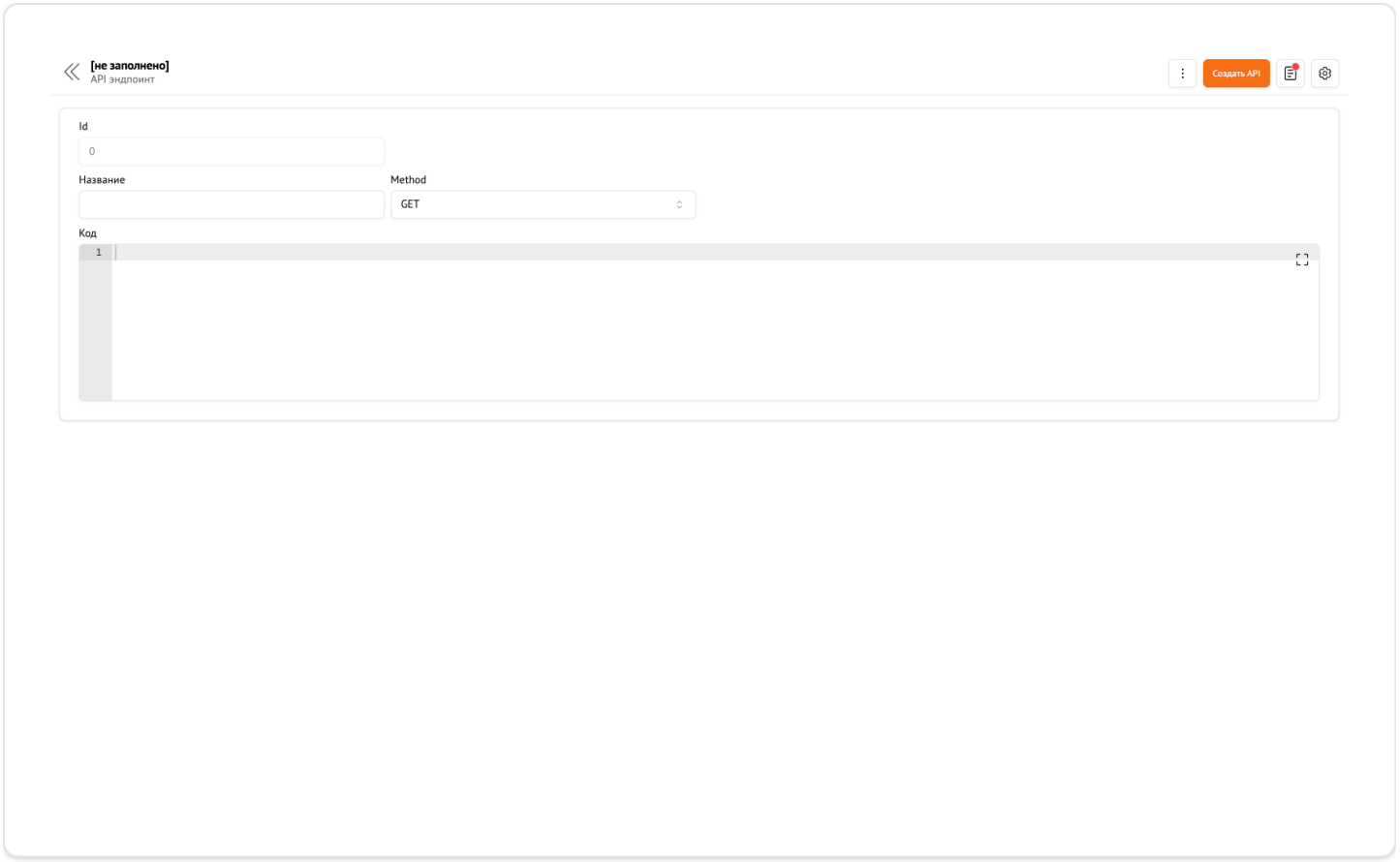
На форме доступны следующие поля:

- Id — уникальный идентификатор эндпоинта (автоматически генерируется при создании);
- Название — отображаемое имя эндпоинта;
- Method — тип HTTP-запроса:
  - `GET` — получение данных;
  - `POST` — отправка данных;
  - `PUT` — обновление данных;
  - `DELETE` — удаление данных.
- Код — поле для ввода исходного кода на C#, определяющего логику обработки запроса.

После заполнения формы нажимается кнопка **«Сохранить изменения»**, что применяет настройки к эндпоинту.

14.2. Создание нового API-эндпоинта

При нажатии кнопки **«Создать API»** открывается форма создания нового эндпоинта.



На форме доступны следующие параметры:

- Id — автоматически генерируется;
- Название — обязательное поле;
- Method — выбор типа HTTP-запроса (по умолчанию — `GET`);
- Код — поле для ввода кода (по умолчанию пусто).

После заполнения формы и сохранения эндпоинт становится доступным для вызова по URL.

14.3. Обобщение функциональности

Принцип работы API-эндпоинтов как функциональной единицы

1. Администратор создаёт новый эндпоинт через форму;
2. Задаёт название, метод и логику обработки;
3. Вводит код для выполнения действий при запросе;
4. Сохраняет эндпоинт;
5. Внешние системы могут вызывать эндпоинт по соответствующему URL.

Применимость

- API-эндпоинты используются для интеграции платформы с внешними системами;
- Поддерживают стандартные HTTP-методы и JSON-данные;
- Позволяют реализовывать сложную бизнес-логику;
- Используются в процессах автоматизации, мониторинга и анализа.

Все API-эндпоинты можно редактировать, копировать и удалять в любое время.

15. Раздел «Рабочие процессы»

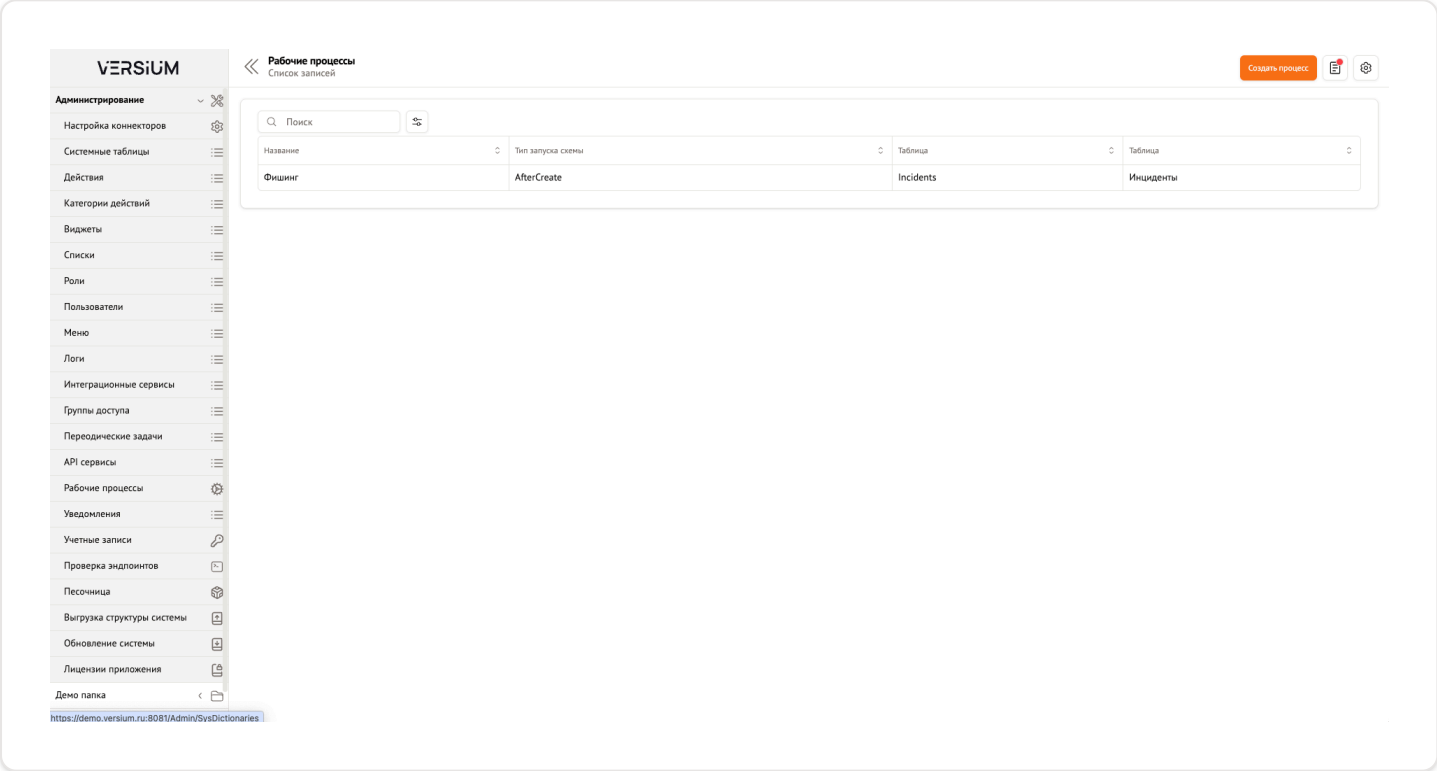
Раздел **«Рабочие процессы»** предназначен для управления автоматизированными рабочими потоками (workflow), которые определяют последовательность действий при обработке инцидентов, активов и других объектов платформы.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять рабочие процессы;
- Настраивать последовательность шагов и условий перехода;
- Автоматизировать стандартные бизнес-процессы (например, реагирование на фишинг);
- Интегрировать действия, уведомления и внешние системы в единую цепочку.

Каждый рабочий процесс представляет собой функциональную единицу, которая запускается по заданному событию и выполняет серию операций.



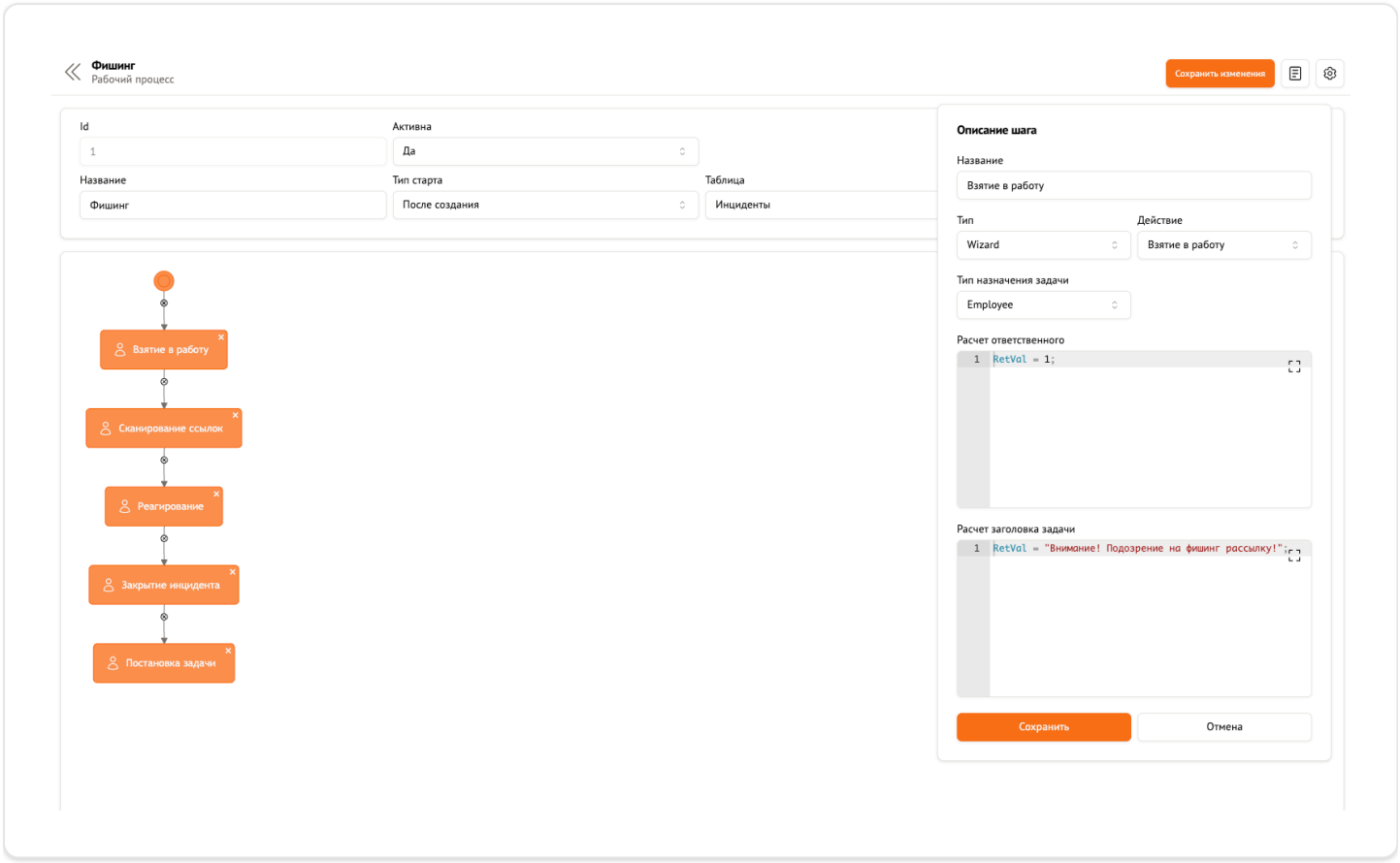
На экране отображается список всех зарегистрированных рабочих процессов с возможностью поиска и сортировки. Каждая запись содержит:

- Название — отображаемое имя процесса;
- Тип запуска схемы — событие, при котором запускается процесс (например, `AfterCreate`, `BeforeUpdate`);
- Таблица — объект, к которому применяется процесс (например, `Incidents`);
- Таблица (второй столбец) — внутреннее имя таблицы.

Кнопка **«Создать процесс»** позволяет добавить новый рабочий процесс.

15.1. Форма редактирования рабочего процесса

При переходе к редактированию существующего процесса открывается форма настройки его параметров.



На форме доступны следующие поля:

- Id — уникальный идентификатор процесса (автоматически генерируется при создании);
- Активна — флаг, определяющий, запускается ли процесс;
- Название — отображаемое имя процесса;
- Тип старта — событие, при котором запускается процесс:
  - `После создания` — после создания записи;
  - `Перед обновлением` — перед изменением записи;
  - `После обновления` — после изменения записи;
  - `По расписанию` — по cron-выражению.
- Таблица — выбор объекта, к которому применяется процесс (например, `Инциденты`).

В нижней части формы расположен **визуальный дизайнер процессов**, позволяющий создавать и редактировать графическую схему рабочего процесса.

15.2. Дизайнер рабочих процессов

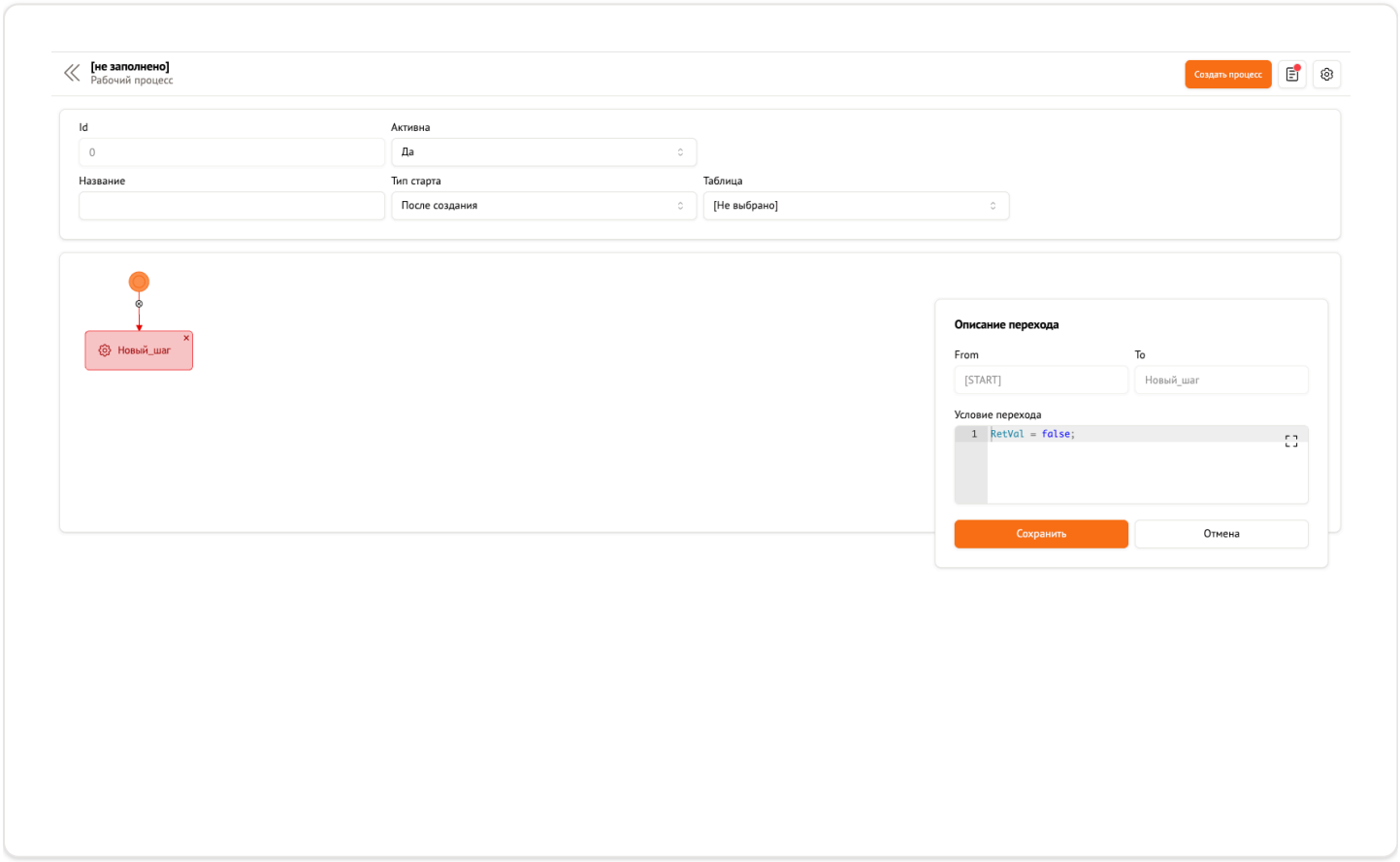
Встроенная система поддерживает **визуальный конструктор рабочих процессов**, что позволяет создавать процессы любой сложности без программирования.
Дизайнер предоставляет следующие возможности:

- **Добавление шагов**: пользователь может перетаскивать элементы (например, «Задача», «Условие», «Действие») на поле и настраивать их параметры;
- **Настройка шагов**: при клике на шаг открывается панель свойств, где можно:
  - Задать название шага;
  - Выбрать тип шага (например, `Wizard` , `Script`);
  - Выбрать действие, которое будет выполнено;
  - Настроить назначение задачи (например, «Сотруднику», «Группе»);
  - Ввести код для расчёта ответственного или заголовка задачи;
- **Настройка переходов**: между шагами можно настраивать условия перехода (например, `RetVal = true;`);
- **Поддержка вложенных логик**: возможно создание ветвлений, повторяющихся блоков, параллельных потоков.

Процесс может содержать любое количество шагов и уровней вложенности, что делает его подходящим для реализации сложных бизнес-правил.

15.3. Создание нового рабочего процесса

При нажатии кнопки **«Создать процесс»** открывается форма создания нового процесса.



На форме доступны следующие параметры:

- Id — автоматически генерируется;
- Активна — по умолчанию «Да»;

- Название — обязательное поле;
- Тип старта — выбор события запуска;
- Таблица — выбор объекта.

После сохранения открывается дизайнер, где можно начать создание схемы процесса.

15.4. Обобщение функциональности

Принцип работы рабочих процессов как функциональной единицы

1. Администратор создаёт новый процесс через форму;
2. Задаёт название, тип старта и таблицу;
3. Использует визуальный дизайнер для создания схемы;
4. Настраивает шаги, действия и условия перехода;
5. Сохраняет процесс;
6. Процесс автоматически запускается при наступлении заданного события.

Применимость

- Рабочие процессы используются для автоматизации стандартных операций;
- Поддерживают визуальное проектирование и редактирование;
- Позволяют интегрировать действия, уведомления и внешние системы;
- Используются в процессах реагирования на инциденты, мониторинга и анализа.

Все рабочие процессы можно редактировать, копировать и удалять в любое время.

16. Раздел «Уведомления»

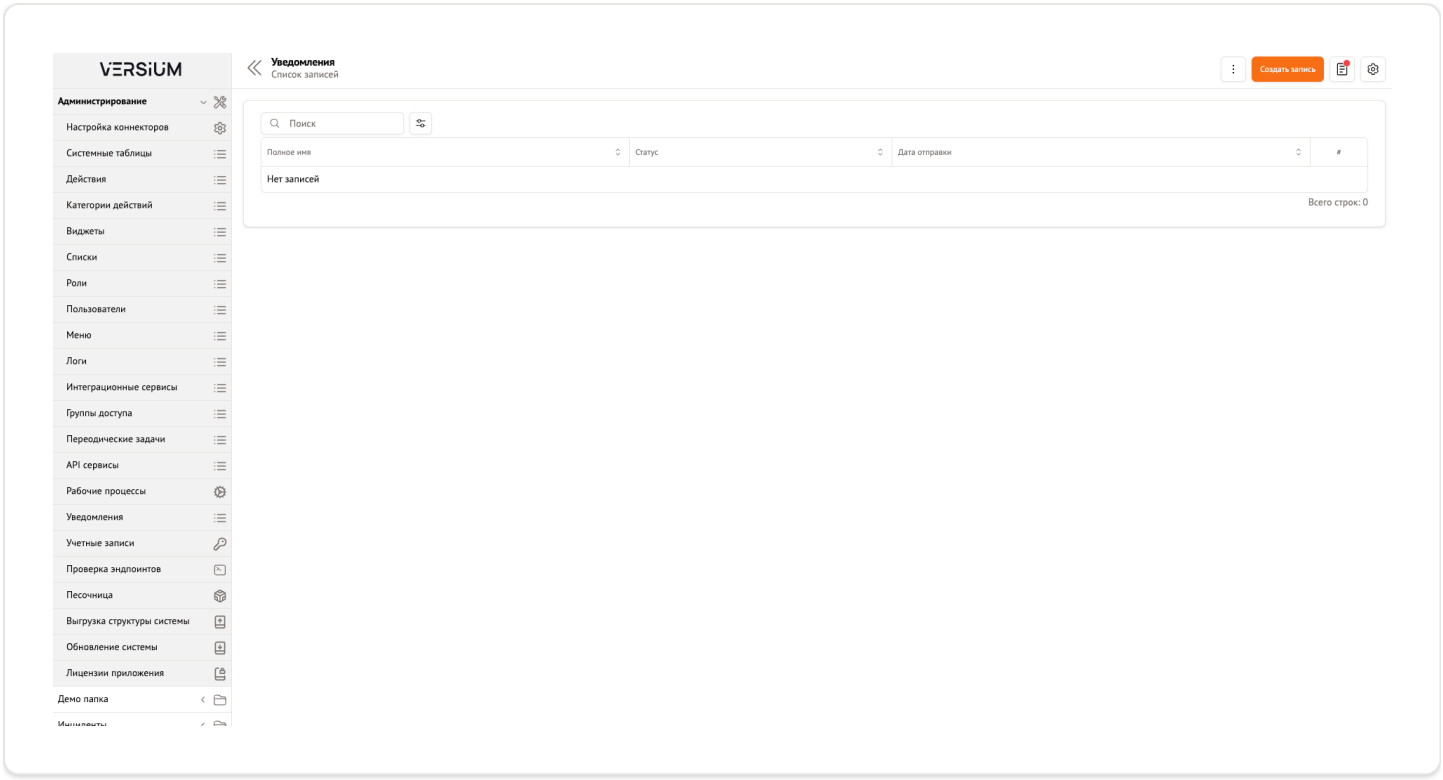
Раздел **«Уведомления»** предназначен для управления системой оповещений, позволяющей информировать пользователей о важных событиях и действиях в платформе.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять записи уведомлений;
- Отслеживать статус отправки сообщений;
- Управлять историей уведомлений;
- Анализировать результаты доставки.

Каждое уведомление представляет собой функциональную единицу, которая фиксирует факт попытки или фактической отправки сообщения пользователю.



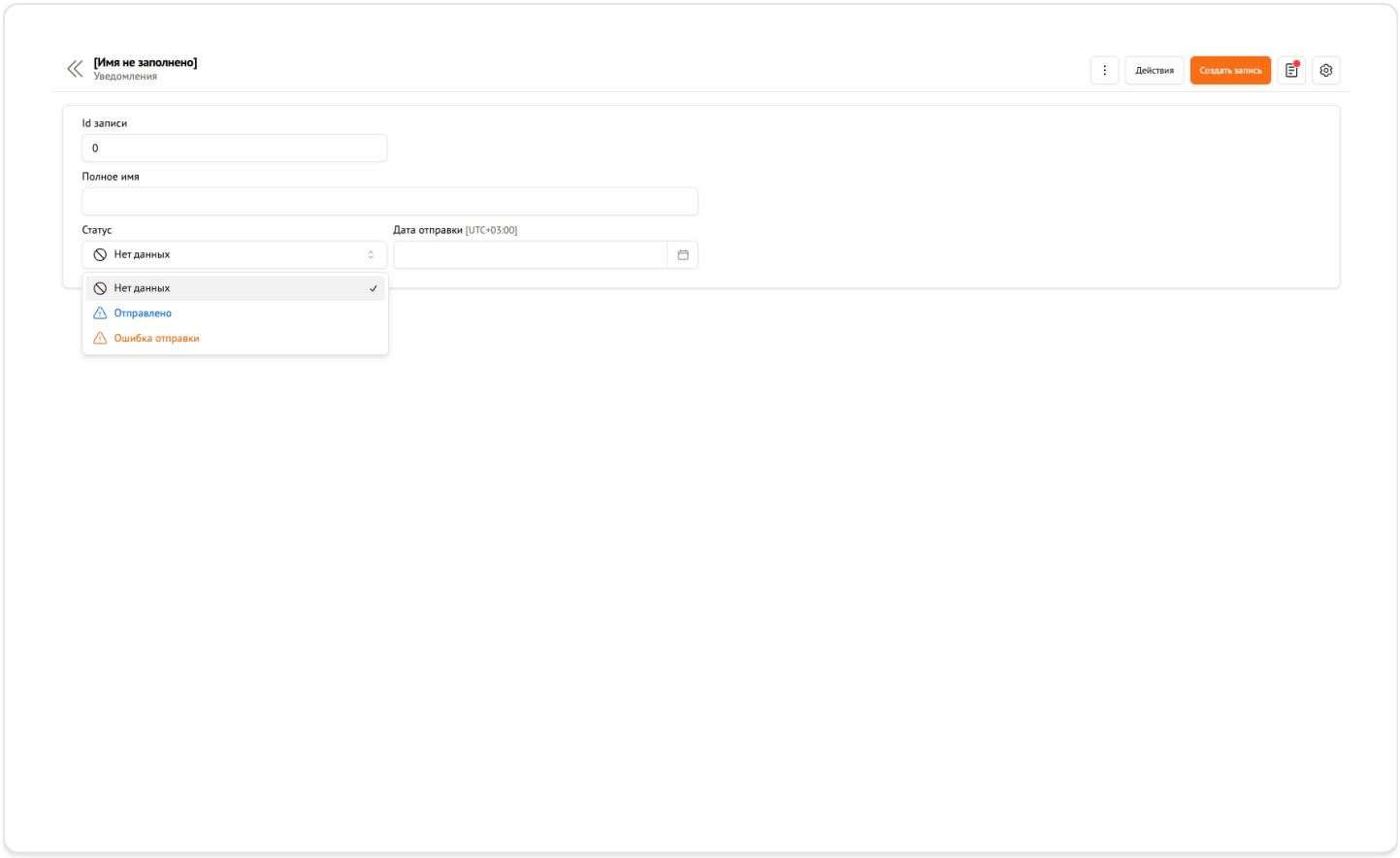
На экране отображается список всех зарегистрированных уведомлений с возможностью поиска и сортировки. Каждая запись содержит:

- Полное имя — отображаемое название уведомления;
- Статус — текущее состояние (например, «Нет данных», «Отправлено», «Ошибка отправки»);
- Дата отправки — временная метка события.

Кнопка **«Создать запись»** позволяет добавить новую запись уведомления.

16.1. Форма редактирования уведомления

При переходе к редактированию существующего уведомления открывается форма настройки его параметров.



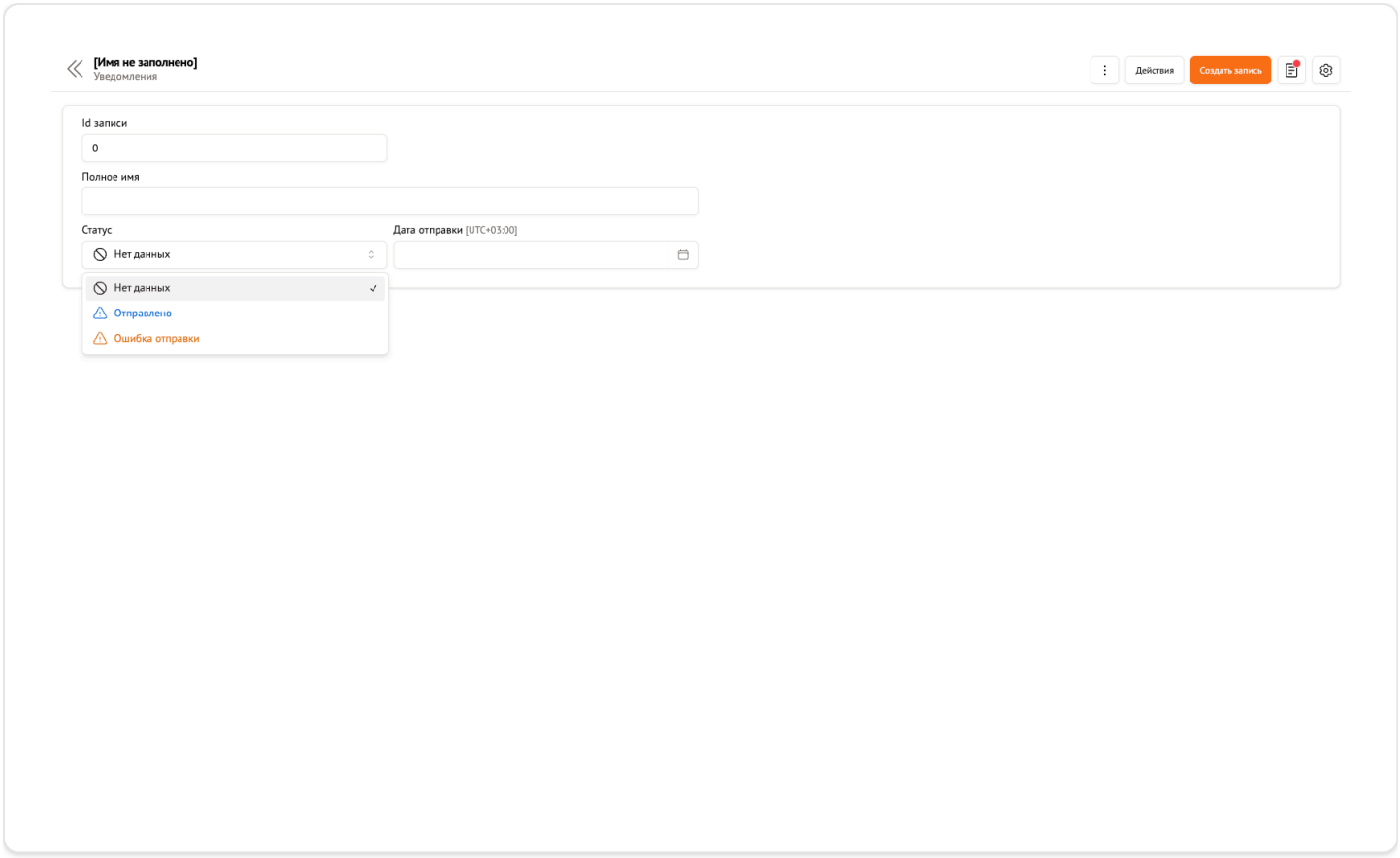
На форме доступны следующие поля:

- Id записи — уникальный идентификатор уведомления (автоматически генерируется при создании);
- Полное имя — отображаемое название уведомления;
- Статус — выбор состояния:
  - `Нет данных` — уведомление не было отправлено;
  - `Отправлено` — уведомление успешно доставлено;
  - `Ошибка отправки` — произошла ошибка при доставке.
- Дата отправки [UTC+03:00] — дата и время события (по умолчанию пусто).

После заполнения формы нажимается кнопка **«Создать запись»**, что сохраняет изменения.

16.2. Создание нового уведомления

При нажатии кнопки **«Создать запись»** открывается форма создания нового уведомления.



На форме доступны следующие параметры:

- Id записи — автоматически генерируется;
- Полное имя — обязательное поле;
- Статус — выбор из доступных значений;
- Дата отправки — поле для указания времени события.

После заполнения формы и сохранения запись становится доступной для просмотра и анализа.

16.3. Обобщение функциональности

Принцип работы уведомлений как функциональной единицы

1. Администратор создаёт новую запись уведомления через форму;
2. Задаёт название, статус и дату;
3. Сохраняет запись;
4. Запись отображается в списке и может быть использована для анализа.

Применимость

- Уведомления используются для аудита и контроля процессов доставки сообщений;
- Поддерживают отслеживание успешных и неудачных попыток отправки;
- Позволяют анализировать эффективность коммуникации внутри системы;
- Используются в рабочих процессах и автоматизации.

Все записи уведомлений можно редактировать, копировать и удалять в любое время.

17. Раздел «Учетные записи»

Раздел **«Учетные записи»** предназначен для управления системными учётными записями, используемыми для аутентификации и интеграции с внешними системами.

Назначение

Этот модуль позволяет администраторам:

- Создавать, редактировать и удалять учётные записи;
- Хранить логины, пароли и параметры подключения к внешним сервисам;
- Обеспечивать безопасное хранение конфиденциальных данных;
- Использовать учётные записи в коннекторах, интеграционных сервисах и API.

Каждая учётная запись представляет собой функциональную единицу, которая используется для авторизации в сторонних системах.

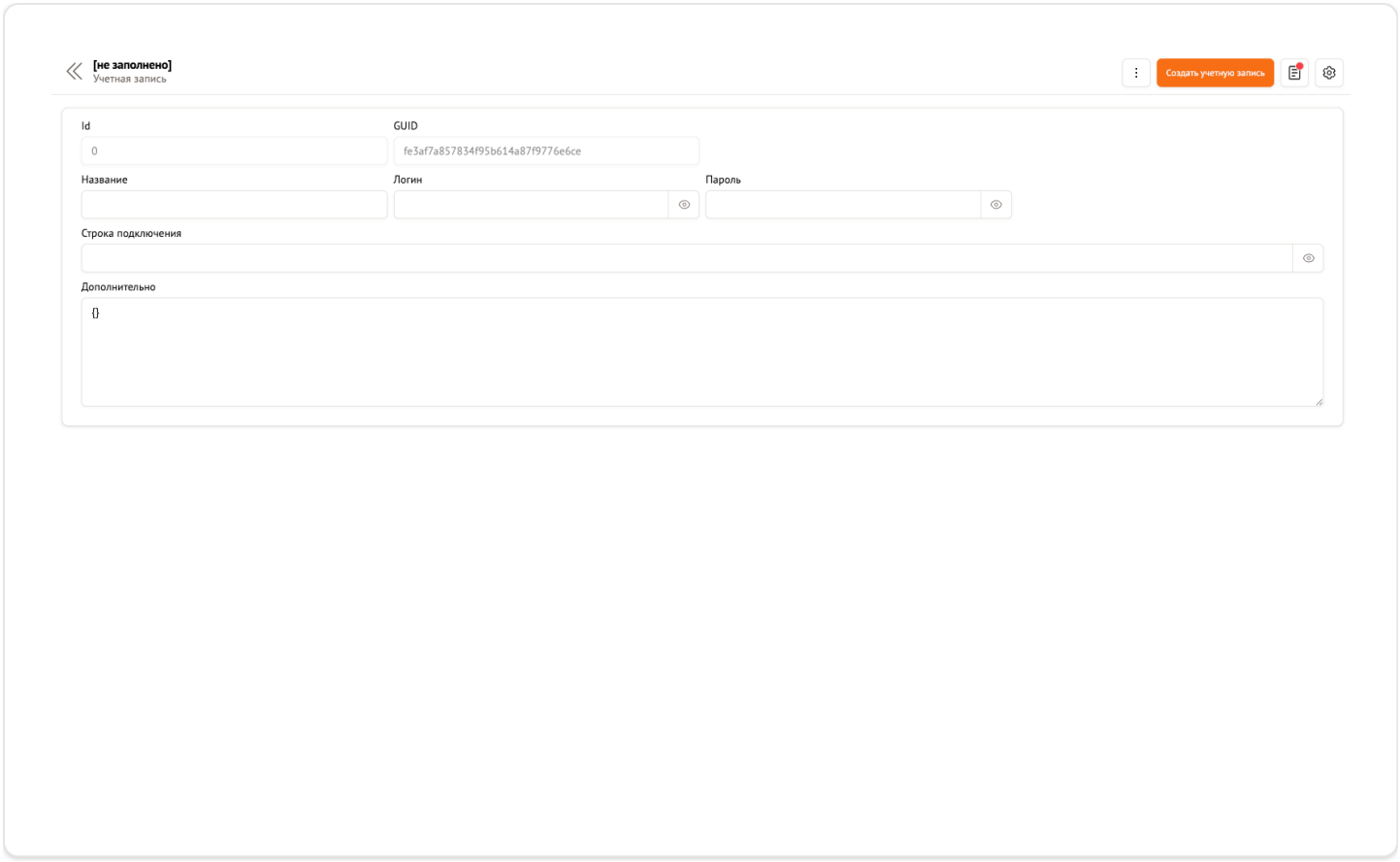
На экране отображается список всех зарегистрированных учётных записей с возможностью поиска и сортировки. Каждая запись содержит:

- Название — отображаемое имя записи.

Кнопка **«Создать учетную запись»** позволяет добавить новую учётную запись.

17.1. Форма редактирования учетной записи

При переходе к редактированию существующей учётной записи открывается форма настройки её параметров.



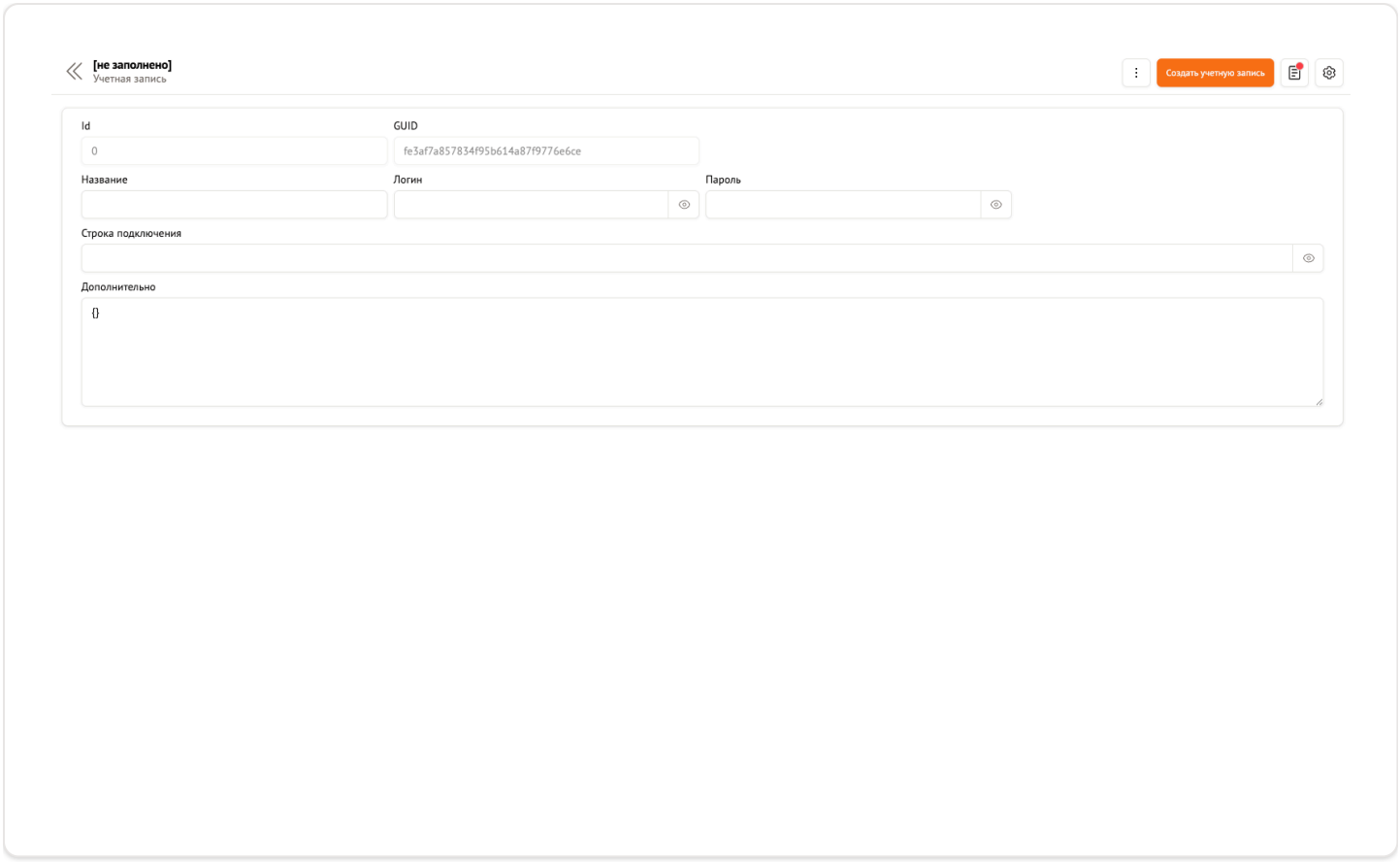
На форме доступны следующие поля:

- Id — уникальный идентификатор записи (автоматически генерируется при создании);
- GUID — глобальный идентификатор записи;
- Название — отображаемое имя записи;
- Логин — имя пользователя для входа в внешнюю систему;
- Пароль — пароль для входа (вводится в зашифрованном виде);
- Строка подключения — дополнительные параметры подключения (например, URL, порт, база данных);
- Дополнительно — текстовое поле для хранения произвольных данных в формате JSON.

После заполнения формы нажимается кнопка **«Создать учетную запись»**, что сохраняет изменения.

17.2. Создание новой учетной записи

При нажатии кнопки **«Создать учетную запись»** открывается форма создания новой записи.



На форме доступны следующие параметры:

- Id — автоматически генерируется;
- GUID — автоматически генерируется;
- Название — обязательное поле;
- Логин — поле для указания имени пользователя;
- Пароль — поле для указания пароля;
- Строка подключения — поле для указания дополнительных параметров;
- Дополнительно — поле для указания произвольных данных.

После заполнения формы и сохранения учётная запись становится доступной для использования в коннекторах и других модулях.

17.3. Обобщение функциональности

Принцип работы учетных записей как функциональной единицы

1. Администратор создаёт новую учётную запись через форму;
2. Задаёт название, логин, пароль и параметры подключения;
3. Сохраняет запись;
4. Использует запись в коннекторах или интеграционных сервисах.

Применимость

- Учетные записи используются для безопасного хранения данных аутентификации;
- Поддерживают шифрование паролей;

- Позволяют избежать хранения конфиденциальных данных в открытых формах;
- Используются в настройке коннекторов, интеграций и API.

Все учетные записи можно редактировать, копировать и удалять в любое время.

18. Раздел «Проверка эндпоинтов»

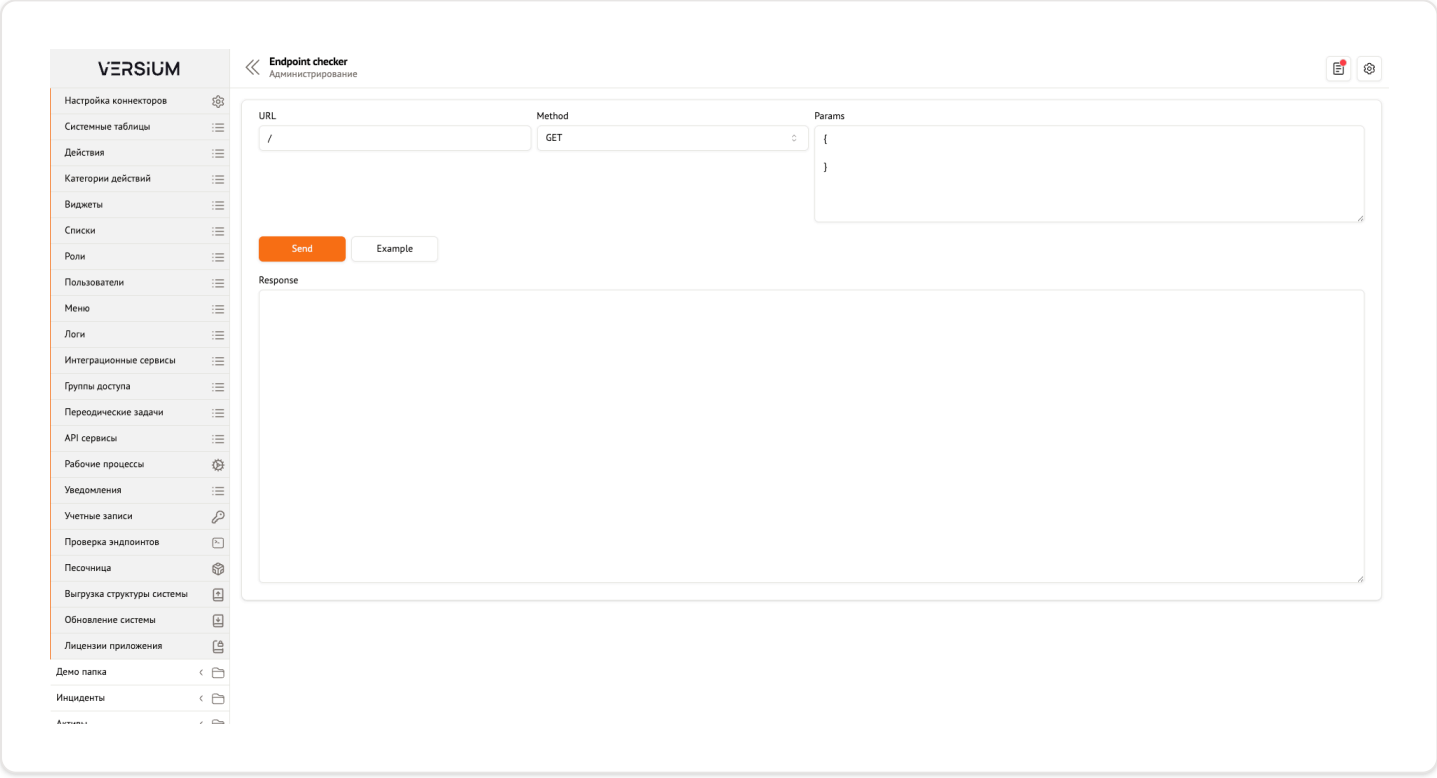
Раздел **«Проверка эндпоинтов»** предназначен для тестирования доступности и корректности работы внутренних и внешних HTTP-ресурсов (API, веб-сервисы, микросервисы) прямо из административной панели.

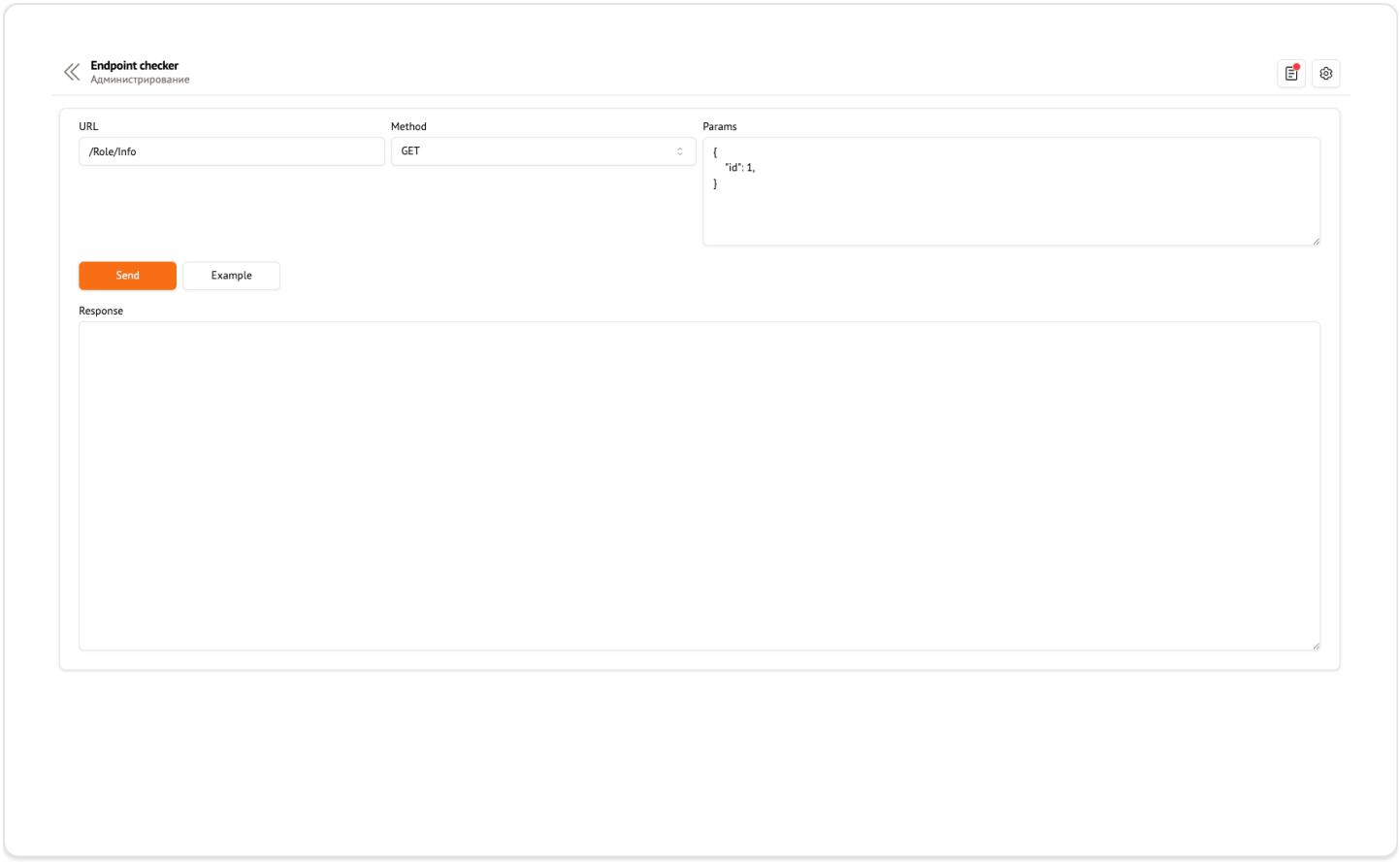
Назначение

Этот модуль позволяет администраторам:

- Отправлять HTTP-запросы к любым URL;
- Проверять ответ сервера (статус, тело, заголовки);
- Тестировать интеграции и API перед их использованием;
- Диагностировать проблемы с подключением и обработкой данных.

Каждый запрос представляет собой функциональную единицу, которая имитирует обращение к эндпоинту и отображает результат.





На экране отображается форма для настройки и отправки HTTP-запроса. Форма содержит следующие поля:

- **URL** — адрес, к которому будет отправлен запрос (например, `/Role/Info`);
- **Method** — тип HTTP-запроса: `GET`, `POST`, `PUT`, `DELETE`;
- **Params** — поле для указания параметров запроса в формате JSON (например, `{ "id": 1 }`);
- **Response** — область для отображения ответа от сервера (включая статус, заголовки и тело).

Кнопка **Send** запускает запрос; кнопка **Example** предоставляет пример запроса.

18.1. Обобщение функциональности

Принцип работы проверки эндпоинтов как функциональной единицы

1. Администратор задаёт URL, метод и параметры запроса;
2. Нажимает кнопку **Send**;
3. Система отправляет запрос и отображает ответ;
4. Анализируется результат (статус код, содержимое тела).

Применимость

- Используется для тестирования API платформы и внешних систем;
- Поддерживает стандартные HTTP-методы и JSON-параметры;
- Позволяет быстро диагностировать ошибки интеграций;
- Используется при разработке и настройке коннекторов, API-эндпоинтов и интеграционных сервисов.

Все запросы выполняются в реальном времени и не сохраняются в истории.

19. Раздел «Песочница»

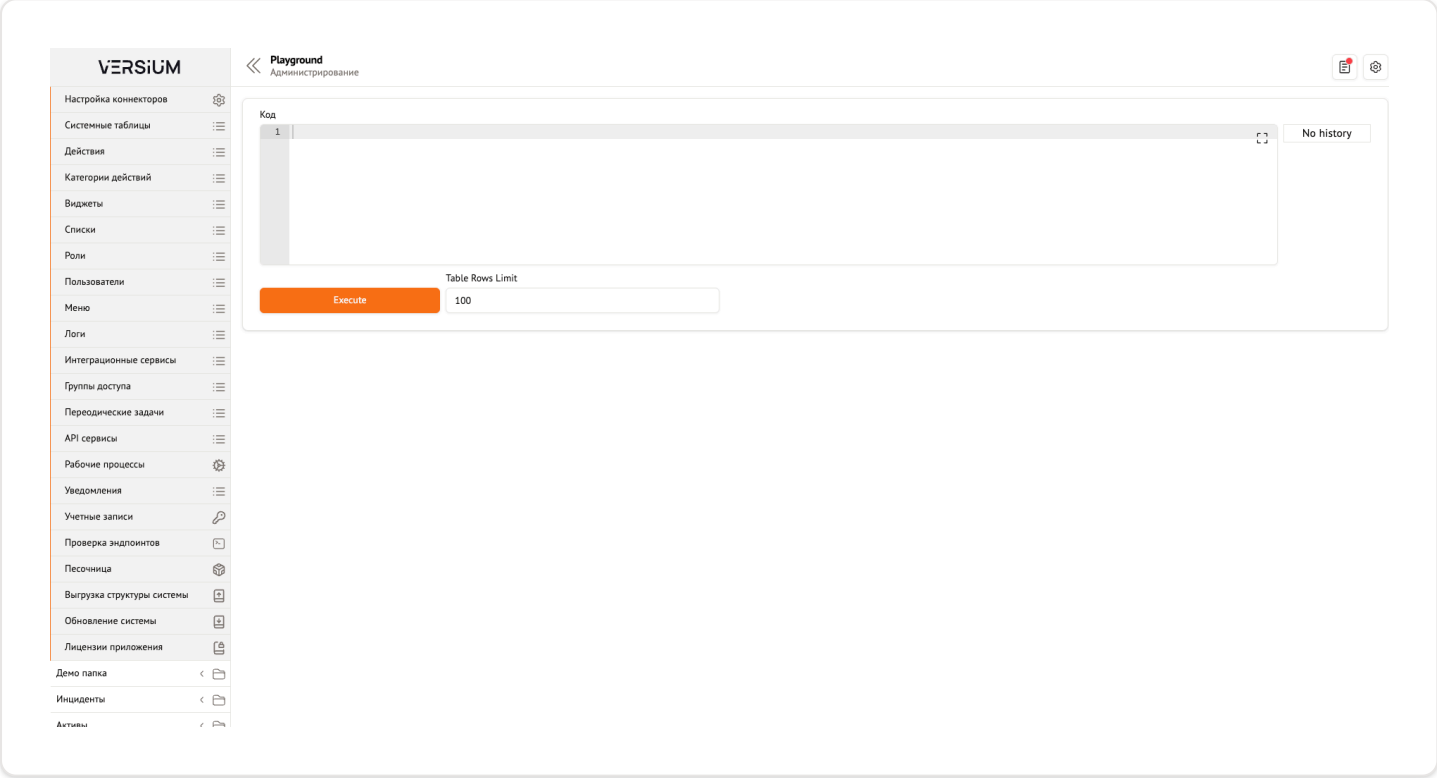
Раздел **«Песочница»** предназначен для безопасного тестирования и отладки пользовательского кода (например, скриптов) без влияния на основную систему.

Назначение

Этот модуль позволяет администраторам и разработчикам:

- Писать и выполнять произвольные скрипты на C#;
- Тестировать логику обработки данных;
- Отлаживать действия, рабочие процессы и интеграции;
- Исследовать API и структуру данных платформы.

Каждый запуск скрипта выполняется в изолированной среде, что предотвращает повреждение или изменение реальных данных.



На экране отображается интерфейс песочницы, состоящий из следующих элементов:

- **Код** — поле для ввода исходного кода на C#;
- **Table Rows Limit** — ограничение количества строк, возвращаемых при запросах к таблицам (по умолчанию — 100);
- **Execute** — кнопка для запуска скрипта;
- **No history** — кнопка, отображающая историю выполненных скриптов (если доступна).

После ввода кода и нажатия кнопки **Execute** система выполняет скрипт и отображает результат в виде вывода или ошибок.

19.1. Обобщение функциональности

Принцип работы песочницы как функциональной единицы

1. Администратор вводит код на C#;
2. Устанавливает ограничение на количество строк;
3. Нажимает кнопку **Execute**;
4. Система выполняет скрипт в изолированной среде;
5. Выводит результат или сообщение об ошибке.

Применимость

- Песочница используется для тестирования сложных логик;
- Поддерживает работу с данными из системных таблиц;
- Позволяет отлаживать скрипты перед их внедрением;
- Используется в процессах разработки, обучения и диагностики.

Все выполненные скрипты можно сохранять, копировать и повторно запускать.

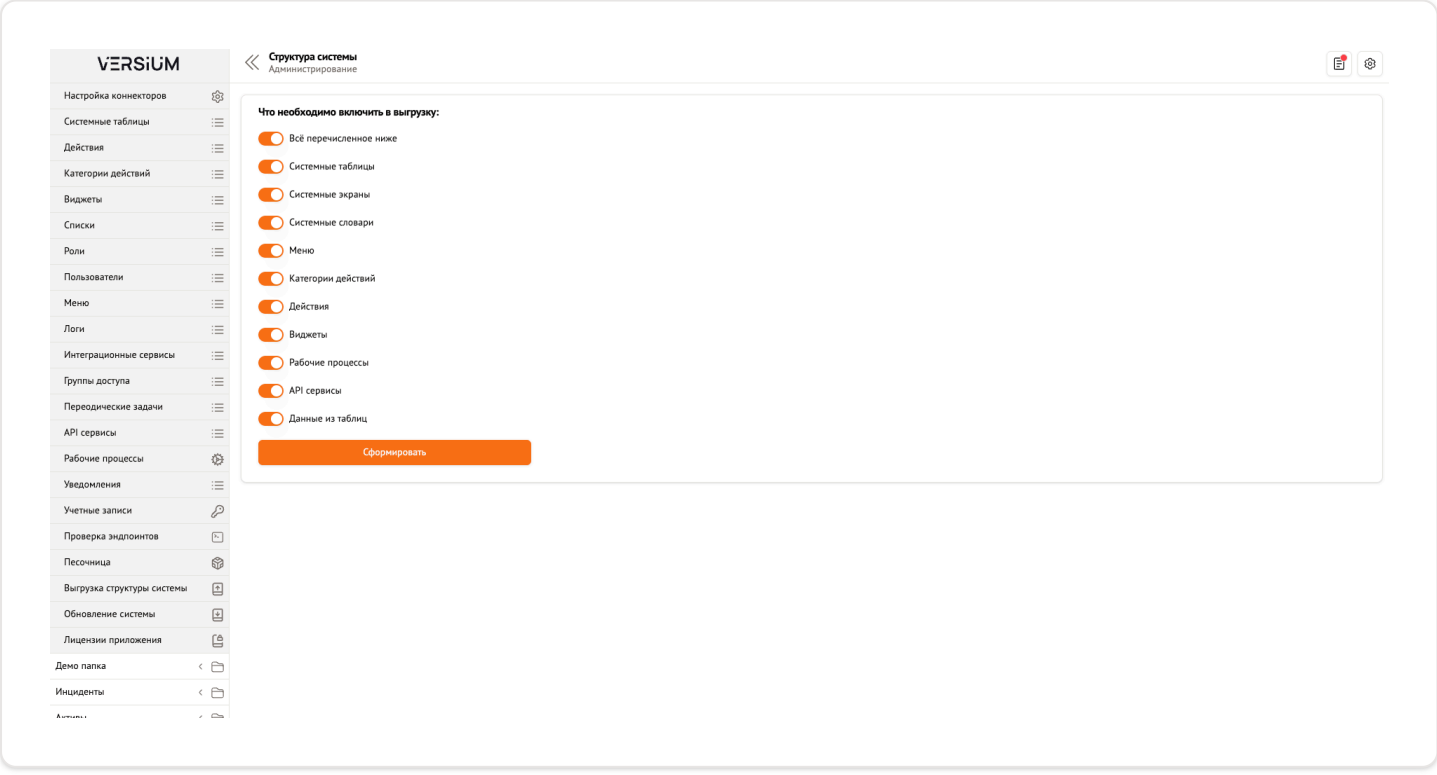
20. Раздел «Выгрузка структуры системы»

Раздел **«Выгрузка структуры системы»** предназначен для экспорта конфигурации платформы в формате JSON или XML, что позволяет осуществлять резервное копирование, миграцию или восстановление настройки.

Назначение

Этот модуль позволяет администраторам:

- Выбирать компоненты конфигурации для экспорта;
- Формировать архив с текущей структурой системы;
- Использовать экспорт для резервного копирования или передачи настройки в другую среду.



На экране отображается список компонентов, которые можно включить в выгрузку:

- Всё перечисленное ниже — выбор всех элементов;
- Системные таблицы — структура базовых данных;
- Системные экраны — настройки отображения данных;
- Системные словари — справочники значений;
- Меню — структура пользовательского интерфейса;
- Категории действий — классификация операций;
- Действия — системные действия;
- Виджеты — аналитические элементы;
- Рабочие процессы — автоматизированные потоки;
- API сервисы — внешние интерфейсы;
- Данные из таблиц — содержимое системных таблиц (по умолчанию не выбрано).

Кнопка **«Сформировать»** запускает процесс экспорта выбранных данных.

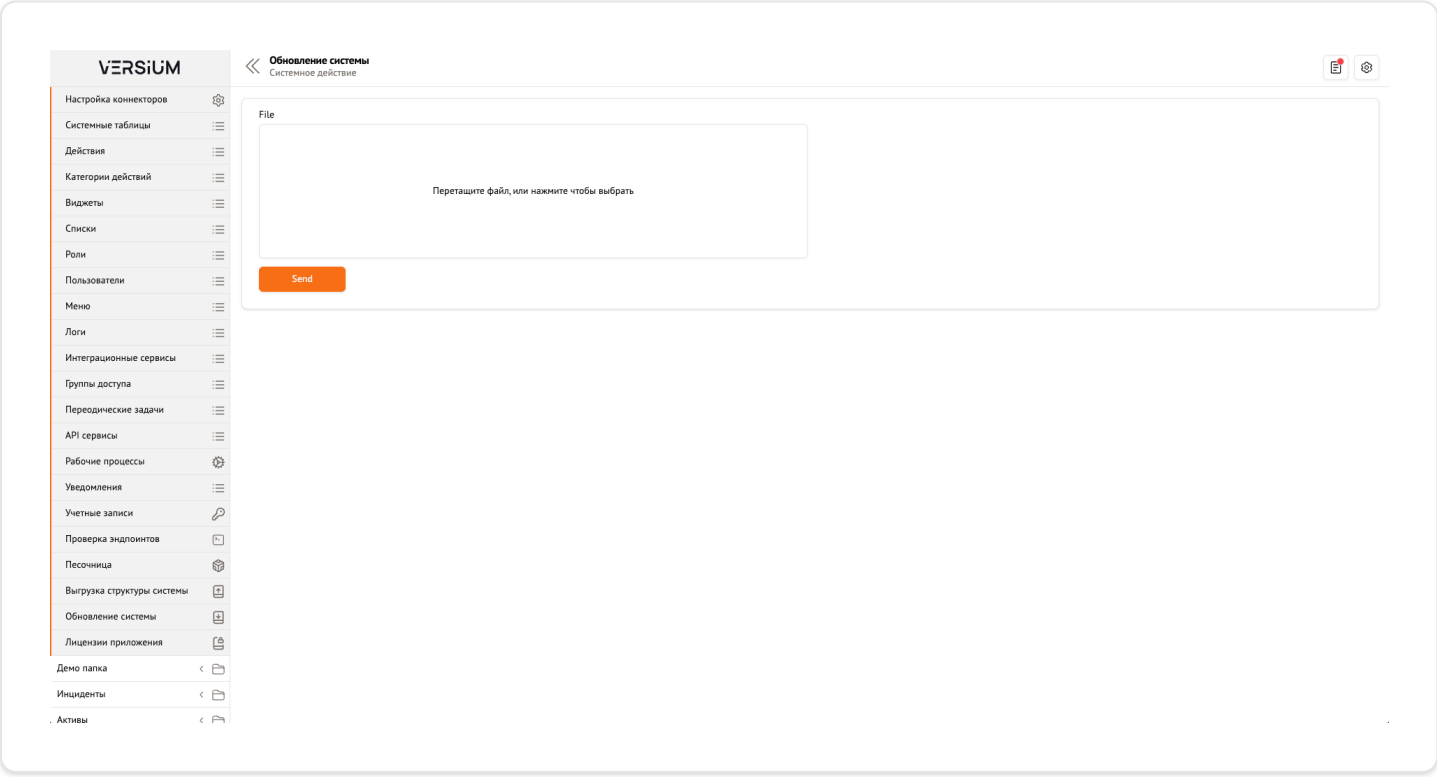
21. Раздел «Обновление системы»

Раздел **«Обновление системы»** предназначен для установки обновлений платформы, включая исправления ошибок, новые функции и улучшения производительности.

Назначение

Этот модуль позволяет администраторам:

- Загружать файл обновления;
- Устанавливать новую версию платформы;
- Обеспечивать актуальность системы.



На экране отображается форма для загрузки файла обновления:

- Поле **File** — область для перетаскивания или выбора файла обновления;
- Кнопка **Send** — отправка файла на сервер для установки.

После загрузки система автоматически начнёт процесс обновления, который может потребовать перезапуск сервисов.

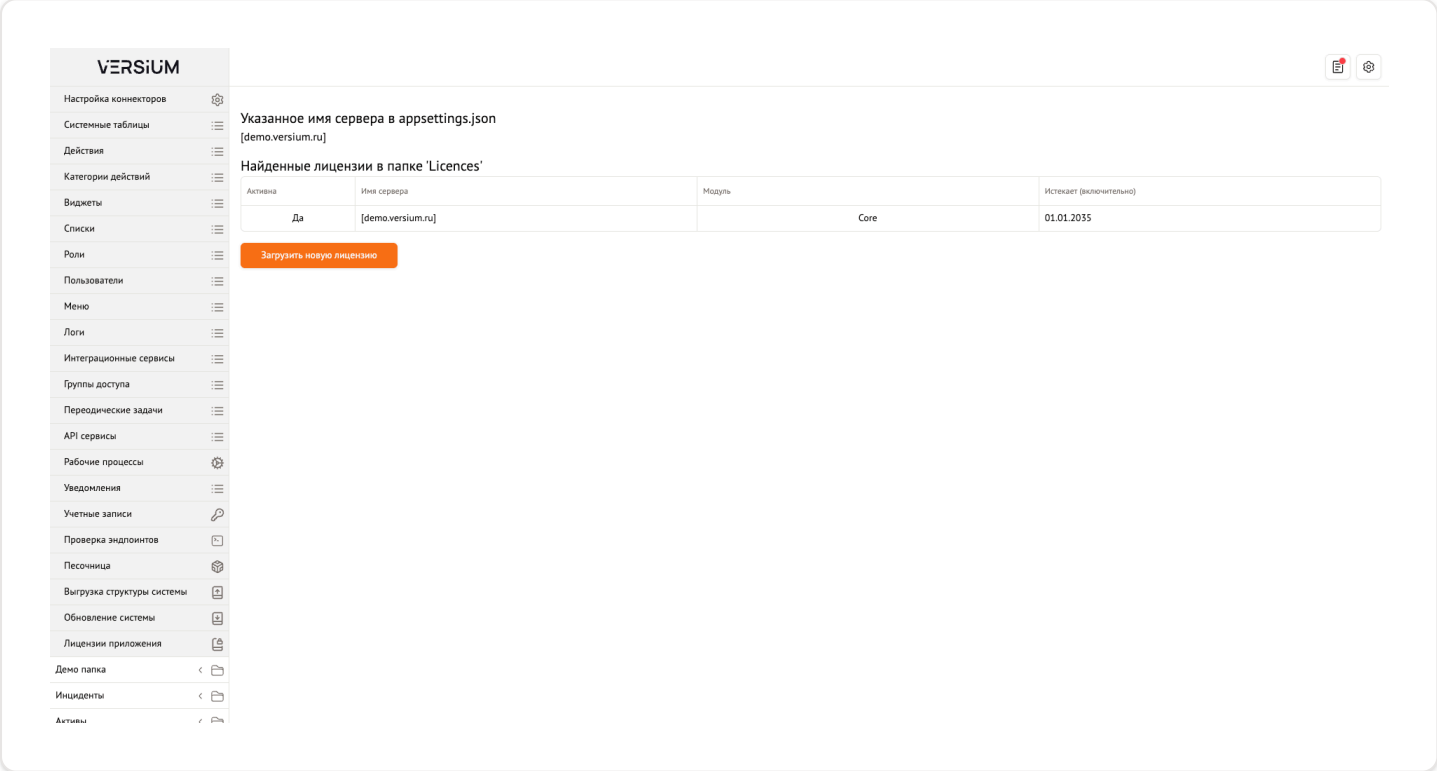
22. Раздел «Лицензии приложения»

Раздел **«Лицензии приложения»** предназначен для управления лицензионными ключами, активацией модулей и проверкой срока действия.

Назначение

Этот модуль позволяет администраторам:

- Просматривать текущие лицензии;
- Загружать новые лицензионные файлы;
- Управлять активацией модулей.



На экране отображается информация о найденных лицензиях:

- Указанное имя сервера — значение из `appsettings.json` ;
- Активна — статус лицензии (да/нет);
- Имя сервера — домен, к которому привязана лицензия;
- Модуль — название модуля (например, Core);
- Истекает (включительно) — дата окончания срока действия.

Кнопка **«Загрузить новую лицензию»** позволяет выбрать и загрузить новый лицензионный файл.

23. Обобщение функциональности

Принцип работы административных разделов как функциональных единиц

1. Администратор переходит в нужный раздел через боковое меню;
2. Выполняет соответствующие действия:
   - Экспортирует конфигурацию;
   - Загружает обновление;
   - Управляет лицензиями.
3. Сохраняет изменения и проверяет результат.

Применимость

- Эти разделы используются для обеспечения надёжности, безопасности и легального использования платформы;
- Поддерживают резервное копирование и миграцию;
- Позволяют управлять жизненным циклом системы;
- Используются в процессах поддержки и технического обслуживания.

Все операции доступны только пользователям с ролью администратора.

24. Раздел «Инциденты»

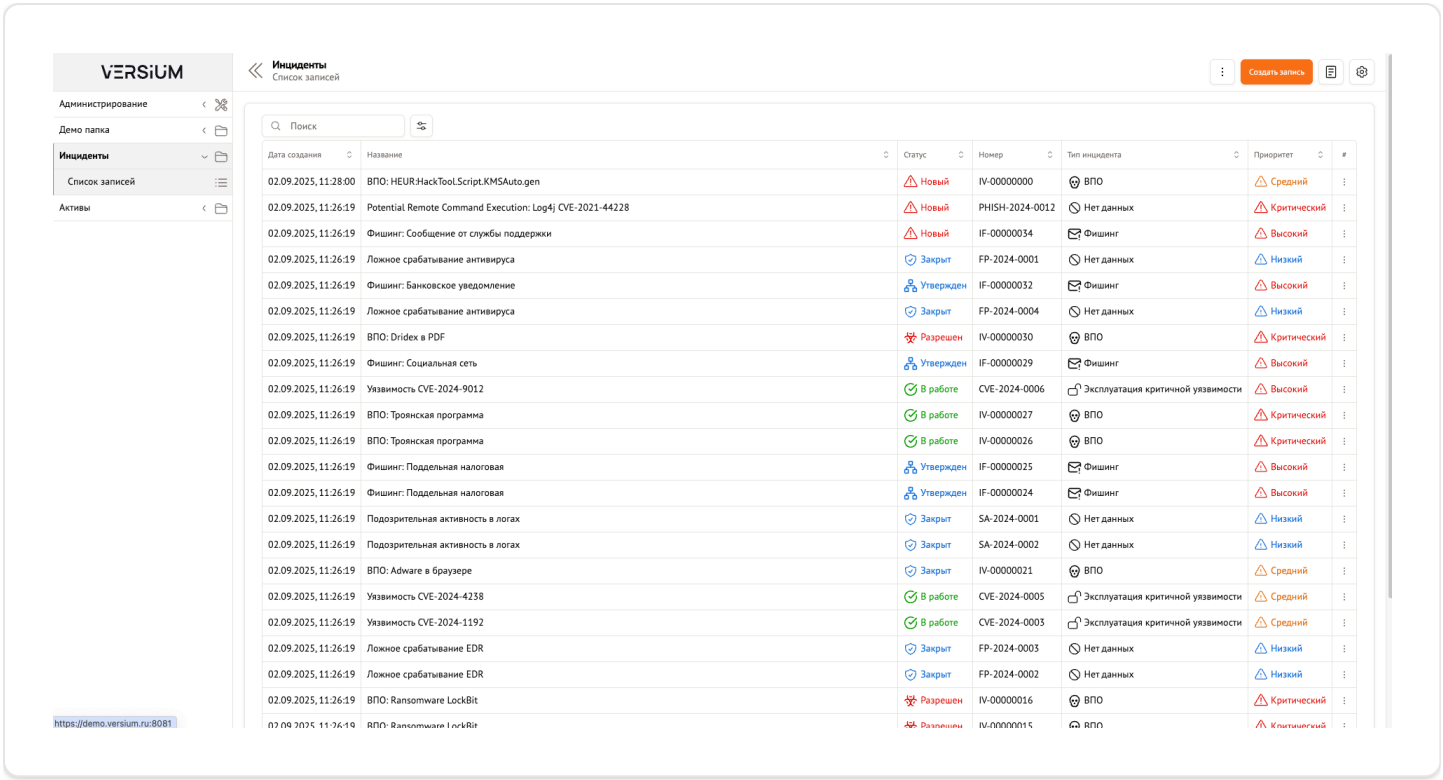
Раздел **«Инциденты»** предназначен для отображения, просмотра и управления всеми зарегистрированными инцидентами информационной безопасности (ИБ) в системе.

Назначение

Этот модуль является центральным рабочим пространством для аналитиков и специалистов по ИБ. Он позволяет:

- Отслеживать все инциденты в реальном времени;
- Фильтровать и сортировать записи по критериям;
- Просматривать детали каждого инцидента;
- Управлять статусом, приоритетом и ответственными лицами.

Каждый инцидент представляет собой функциональную единицу, которая фиксирует событие, требующее реагирования.



На экране отображается список всех инцидентов с возможностью поиска и сортировки. Каждая запись содержит следующие поля:

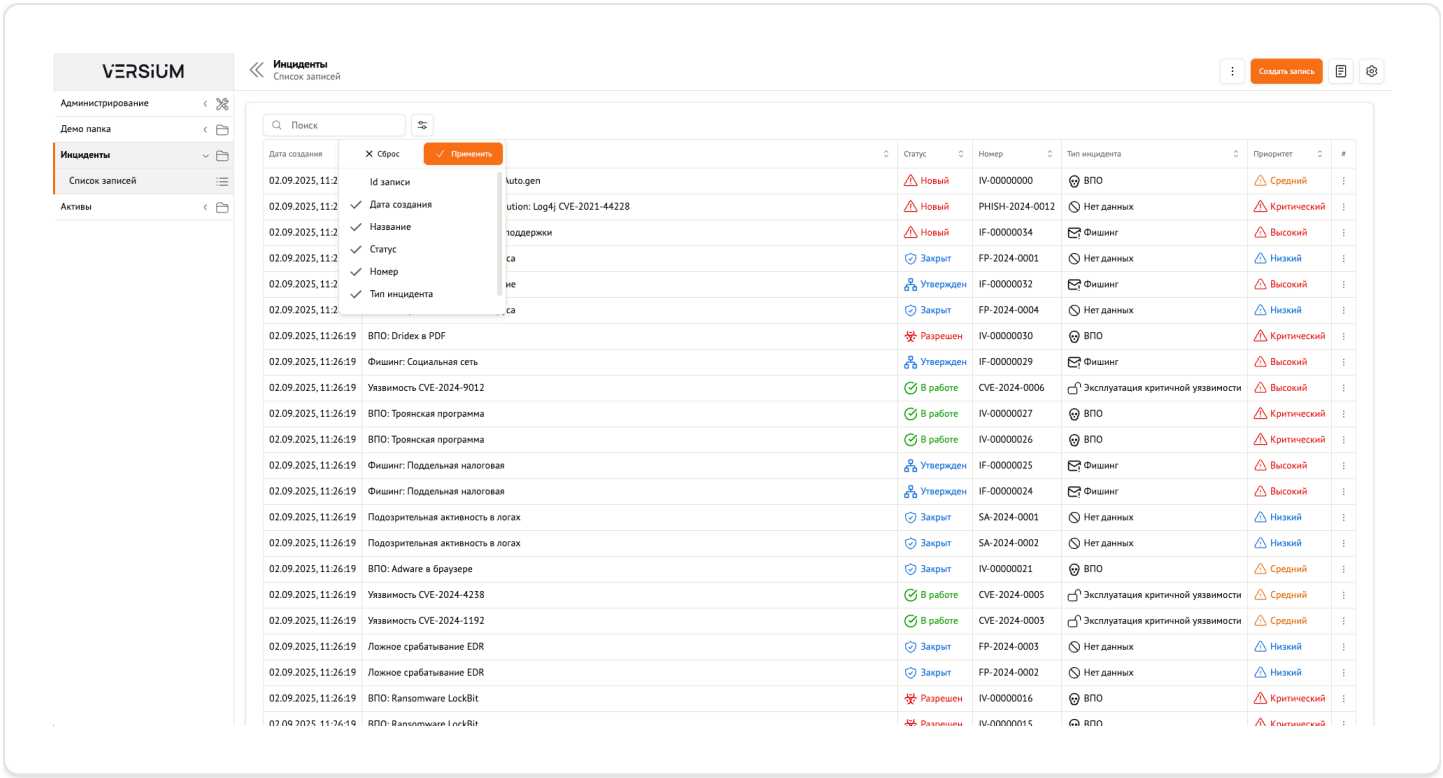
- Дата создания — временная метка события;
- Название — краткое описание инцидента;
- Статус — текущее состояние (например, «Новый», «В работе», «Закрыт»);
- Номер — уникальный идентификатор инцидента;
- Тип инцидента — категория события (например, «Фишинг», «ВПО», «Уязвимость»);
- Приоритет — уровень важности (например, «Критический», «Высокий», «Средний», «Низкий»).

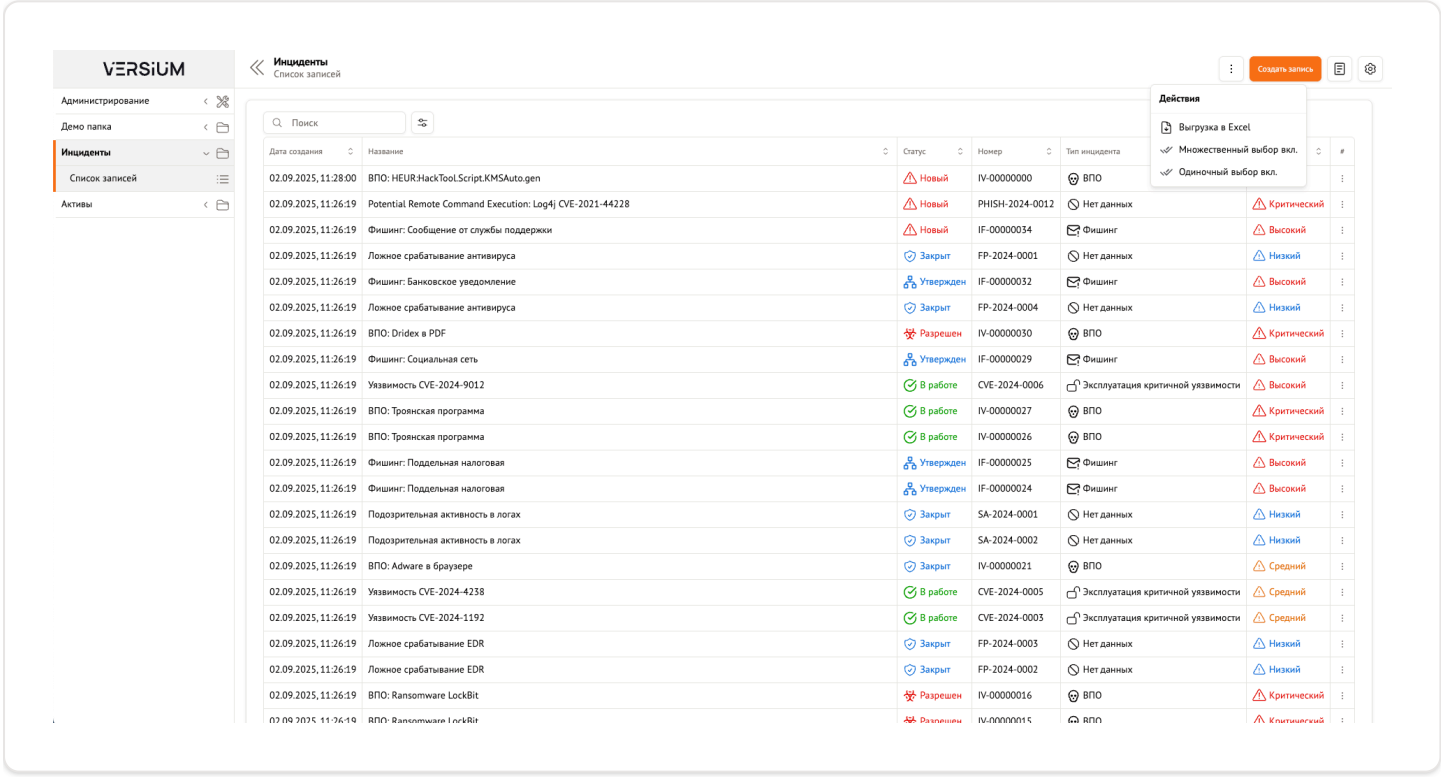
Кнопка **«Создать запись»** позволяет добавить новый инцидент вручную.

24.1. Функциональность списка инцидентов

В верхней части формы расположен блок фильтрации:

- Поле **Поиск** — позволяет выполнять текстовый поиск по названию или описанию;
- Кнопка фильтрации (иконка фильтра) — открывает меню выбора полей для фильтрации.





Доступные поля для фильтрации:

- Id записи;
- Дата создания;
- Название;
- Статус;
- Номер;
- Тип инцидента.

После выбора полей и нажатия кнопки **«Применить»** список обновляется с учётом выбранных критериев.

Выгрузка отчёта в Excel

incidents_export_excel.png

В правом верхнем углу экрана расположена кнопка **«Действия»**, которая открывает выпадающее меню с дополнительными опциями.

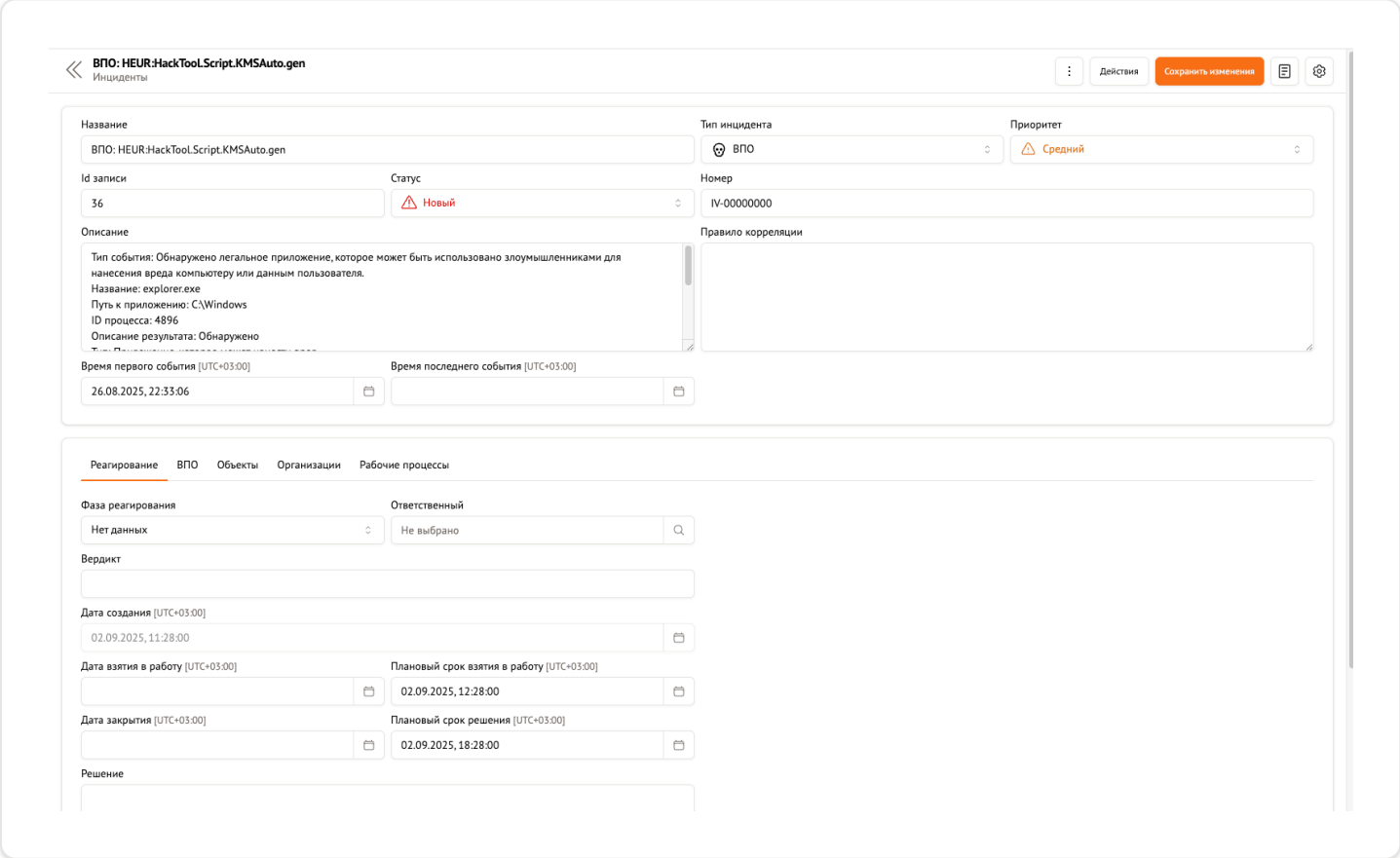
В этом меню доступна функция **«Выгрузка в Excel»**, позволяющая экспортировать текущий список инцидентов в формате `.xlsx`. Эта функция особенно полезна для:

- Генерации отчётов для руководства;
- Анализа данных вне платформы;
- Архивации информации.

При нажатии на эту опцию система формирует файл, содержащий все видимые поля из текущего списка, включая фильтры и сортировку. Выгрузка выполняется в фоновом режиме, и пользователь получает ссылку для скачивания.

24.2. Карточка инцидента

При клике на строку инцидента открывается его карточка с подробной информацией.



На форме отображаются следующие данные:

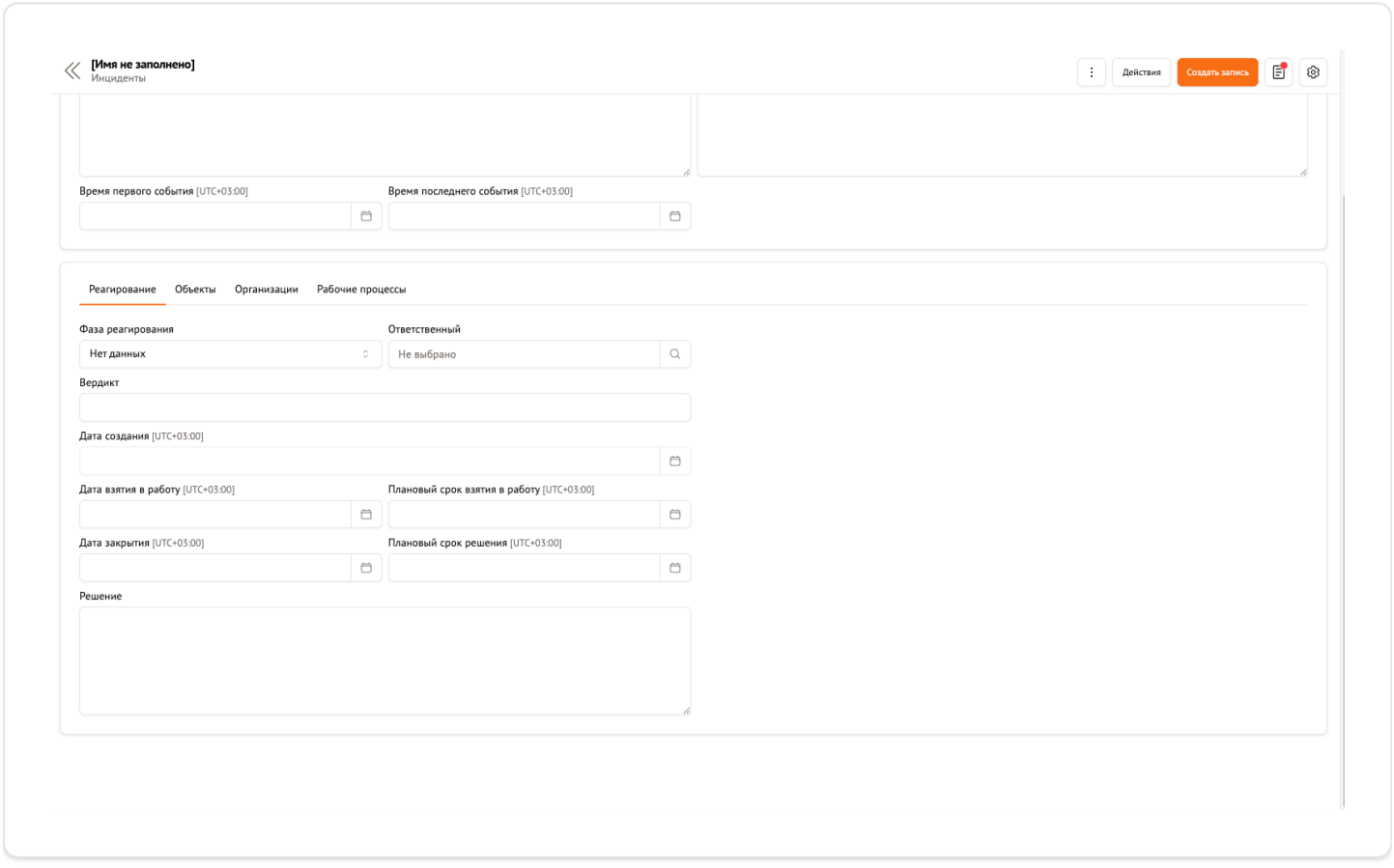
- **Общая информация**:
  - Название — отображаемое имя инцидента;
  - Тип инцидента — категория события;

- Приоритет — уровень важности;
- Id записи — уникальный идентификатор;
- Статус — текущее состояние;
- Номер — внутренний номер инцидента;
- Описание — подробное описание события;
- Правило корреляции — правило, по которому был сгенерирован инцидент;
- Время первого события — дата и время начала инцидента;
- Время последнего события — дата и время последнего действия.

- **Вкладки**:
  - Регистрирование — базовая информация о событии;
  - ВПО — данные о вредоносном программном обеспечении;
  - Объекты — связанные активы и системы;
  - Организации — организации, затронутые инцидентом;
  - Рабочие процессы — автоматизированные потоки, запущенные по этому инциденту.

24.3. Создание нового инцидента

При нажатии кнопки **«Создать запись»** открывается форма создания нового инцидента.



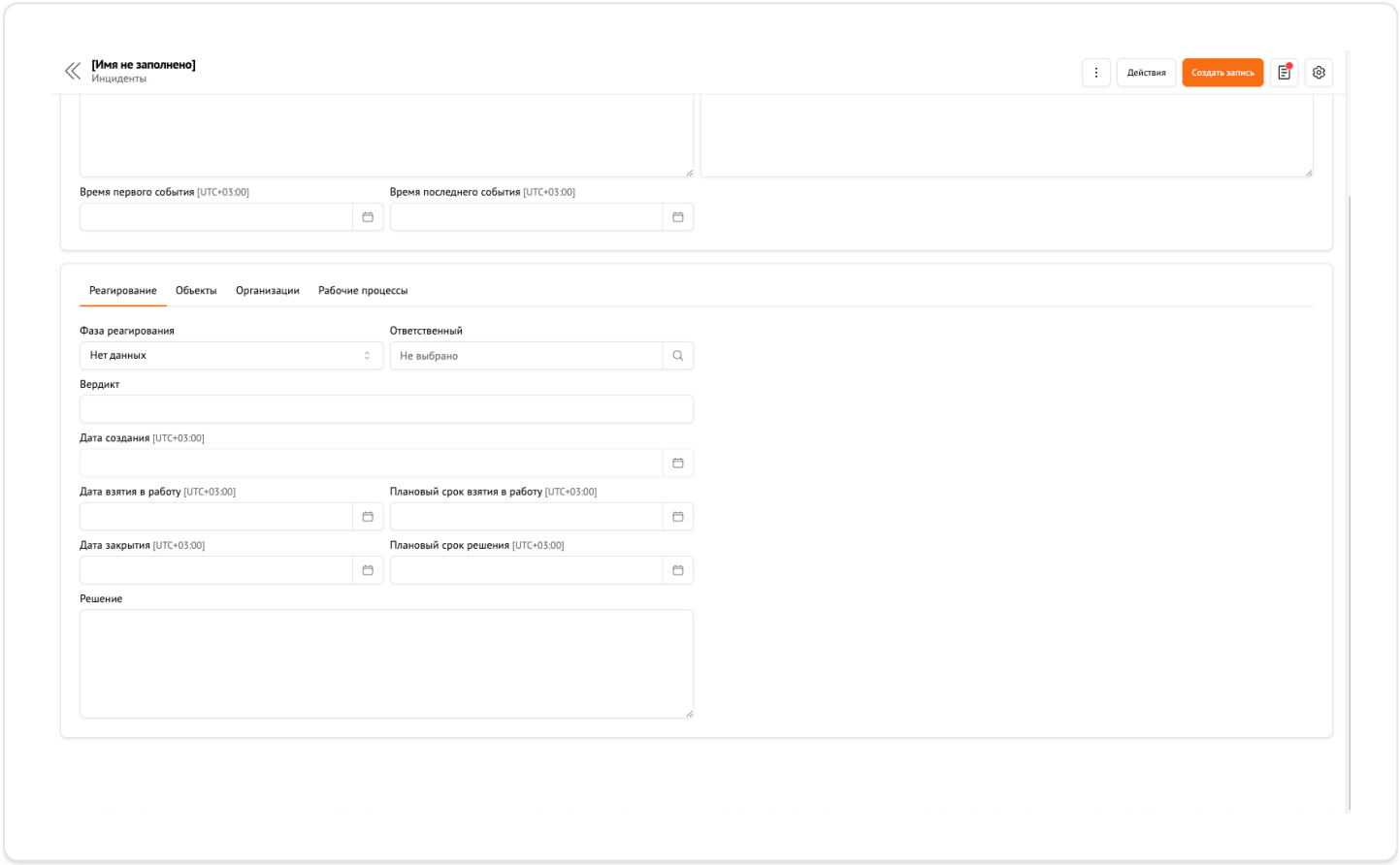
Форма состоит из двух частей: общих данных и вкладок.

Общие данные

- Название — отображаемое имя инцидента;
- Тип инцидента — выбор категории (например, «Фишинг», «ВПО»);
- Приоритет — выбор уровня важности;
- Id записи — автоматически генерируется;
- Статус — по умолчанию «Новый»;
- Номер — автоматически генерируется;
- Описание — текстовое поле для описания события;
- Правило корреляции — поле для указания правила, по которому был создан инцидент;
- Время первого события — дата и время начала инцидента;
- Время последнего события — дата и время последнего действия.

Вкладки

Вкладка «Реагирование»

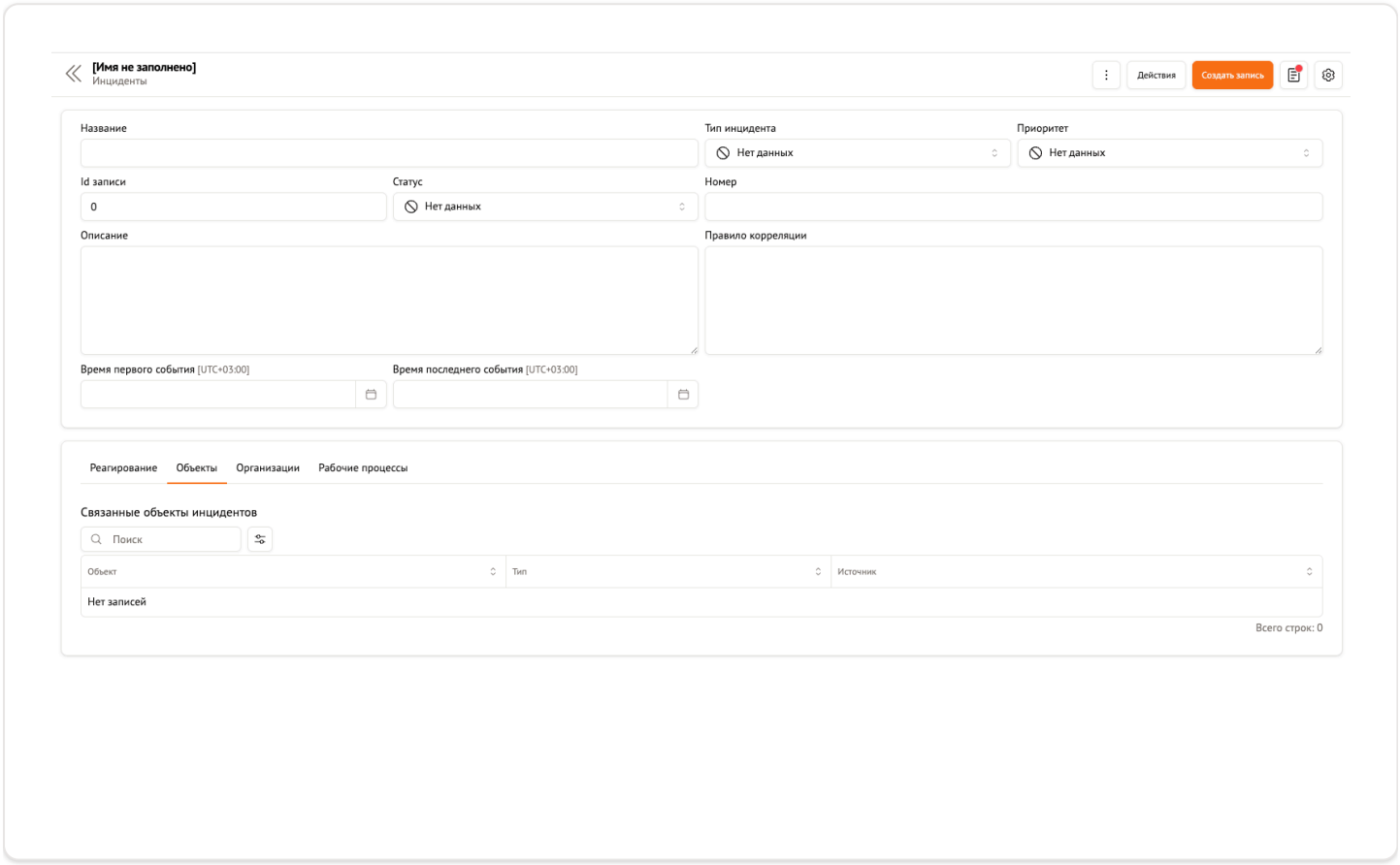


На этой вкладке настраиваются параметры реагирования:

- Фаза реагирования — выбор этапа (например, «Подозрительная активность», «Проверка», «Подтверждение»);
- Ответственный — выбор сотрудника, ответственного за инцидент;
- Вердикт — текстовое поле для вывода по результатам анализа;
- Дата создания — временная метка;
- Дата взятия в работу — дата назначения ответственного;

- Плановый срок взятия в работу — планируемая дата начала работы;
- Дата закрытия — дата завершения инцидента;
- Плановый срок решения — планируемая дата завершения;
- Решение — текстовое поле для описания принятых мер.

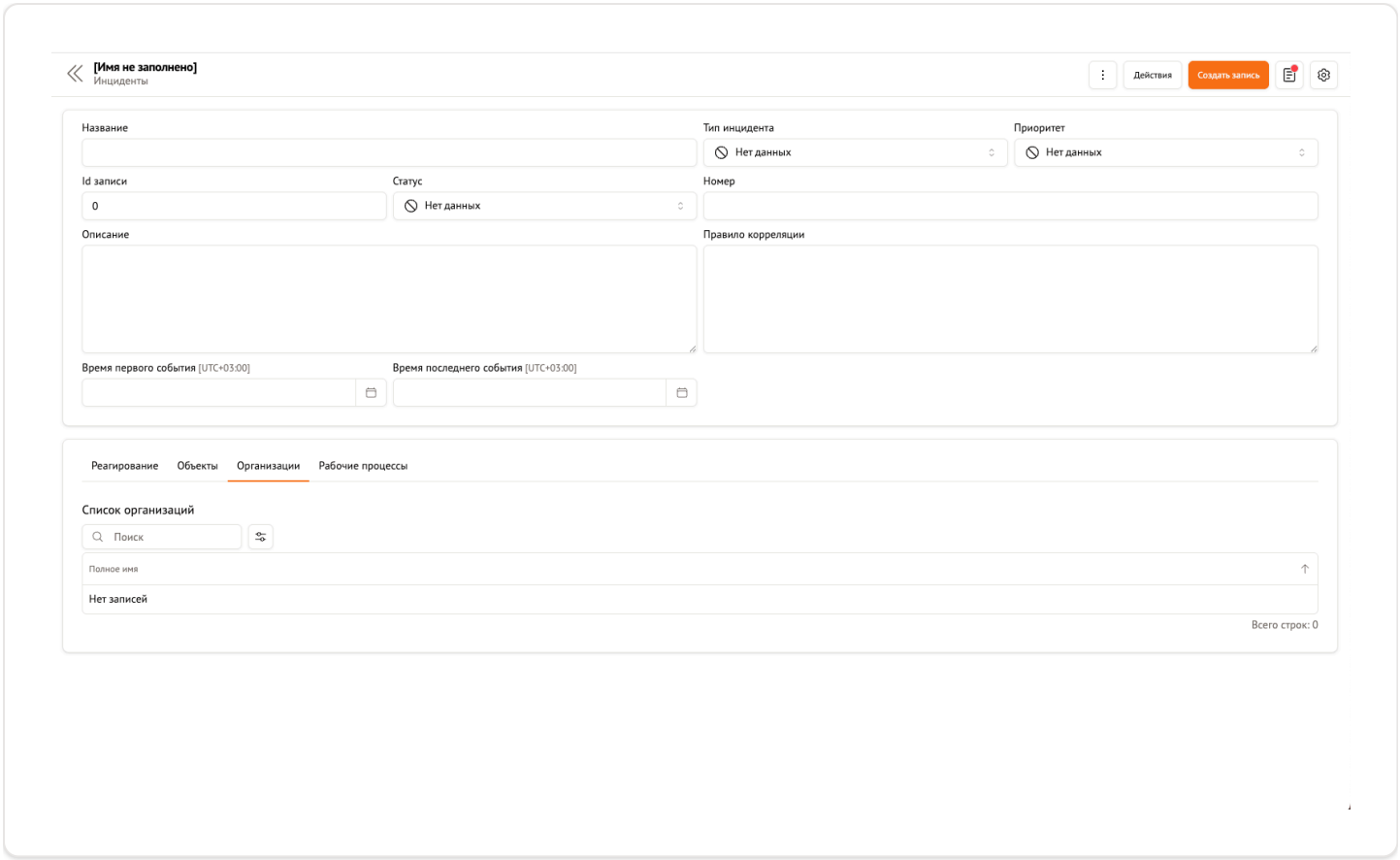
Примечание: Список полей в данной вкладке может отличаться в зависимости от типа инцидента и настроек конкретной инсталляции платформы. Администратор может настраивать состав полей через раздел «Системные таблицы» и «Экраны».
Вкладка «Объекты»



На этой вкладке отображается список объектов, связанных с инцидентом:

- Связанные объекты инцидентов — таблица с перечнем активов, устройств или систем, затронутых инцидентом;
- Поиск — возможность поиска по имени или типу объекта;
- Объект — отображаемое имя;
- Тип — тип объекта (например, «Актив», «Хост»);
- Источник — источник информации о объекте.

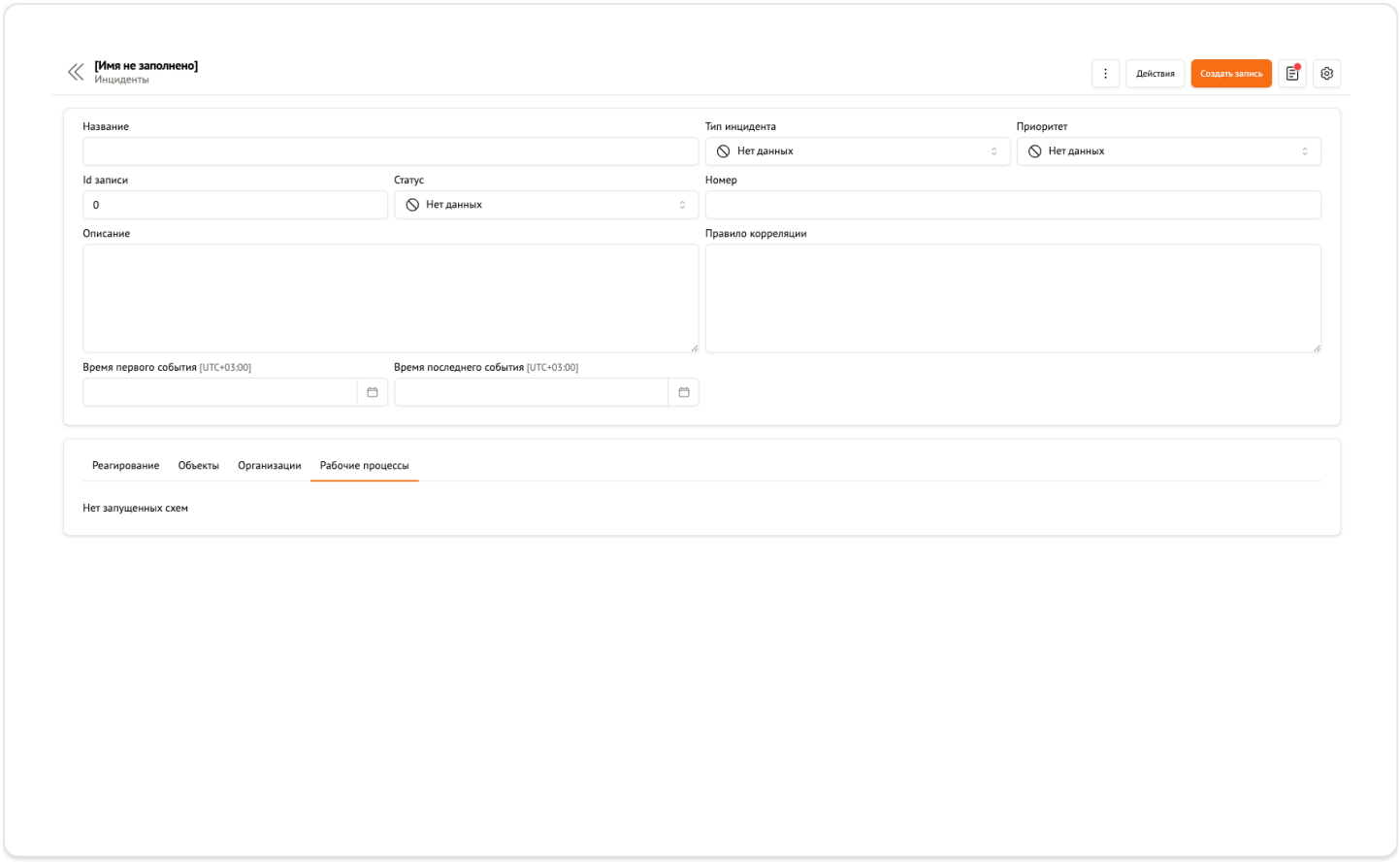
Вкладка «Организации»



На этой вкладке отображается список организаций, затронутых инцидентом:

- Список организаций — таблица с перечнем организаций;
- Поиск — возможность поиска по имени;
- Полное имя — отображаемое название организации.

Вкладка «Рабочие процессы»



На этой вкладке отображается список запущенных рабочих процессов по данному инциденту:

- Нет запущенных схем — сообщение, если процессы не запущены;
- При наличии процессов — отображаются их названия, статусы и параметры.

Примечание: Рабочие процессы создаются и настраиваются в **визуальном дизайнере рабочих процессов**, описанном в разделе «Рабочие процессы». Любой созданный процесс может быть привязан к любому типу инцидента, что позволяет реализовывать стандартные процедуры реагирования для различных категорий событий.

24.4. Обобщение функциональности

Принцип работы раздела «Инциденты» как функциональной единицы

1. Администратор или аналитик переходит в раздел «Инциденты»;
2. Просматривает список инцидентов;
3. Использует фильтры для узкой выборки;
4. Переходит в карточку инцидента для детального анализа;
5. Управляет статусом, приоритетом и ответственными лицами.

Применимость

- Раздел используется для мониторинга и реагирования на инциденты;
- Поддерживает работу с различными типами событий (фишинг, ВПО, уязвимости);
- Интегрируется с рабочими процессами и автоматизацией;
- Используется в процессах анализа, классификации и закрытия инцидентов.

Все операции доступны пользователям с соответствующими правами.

25. Раздел «Активы»

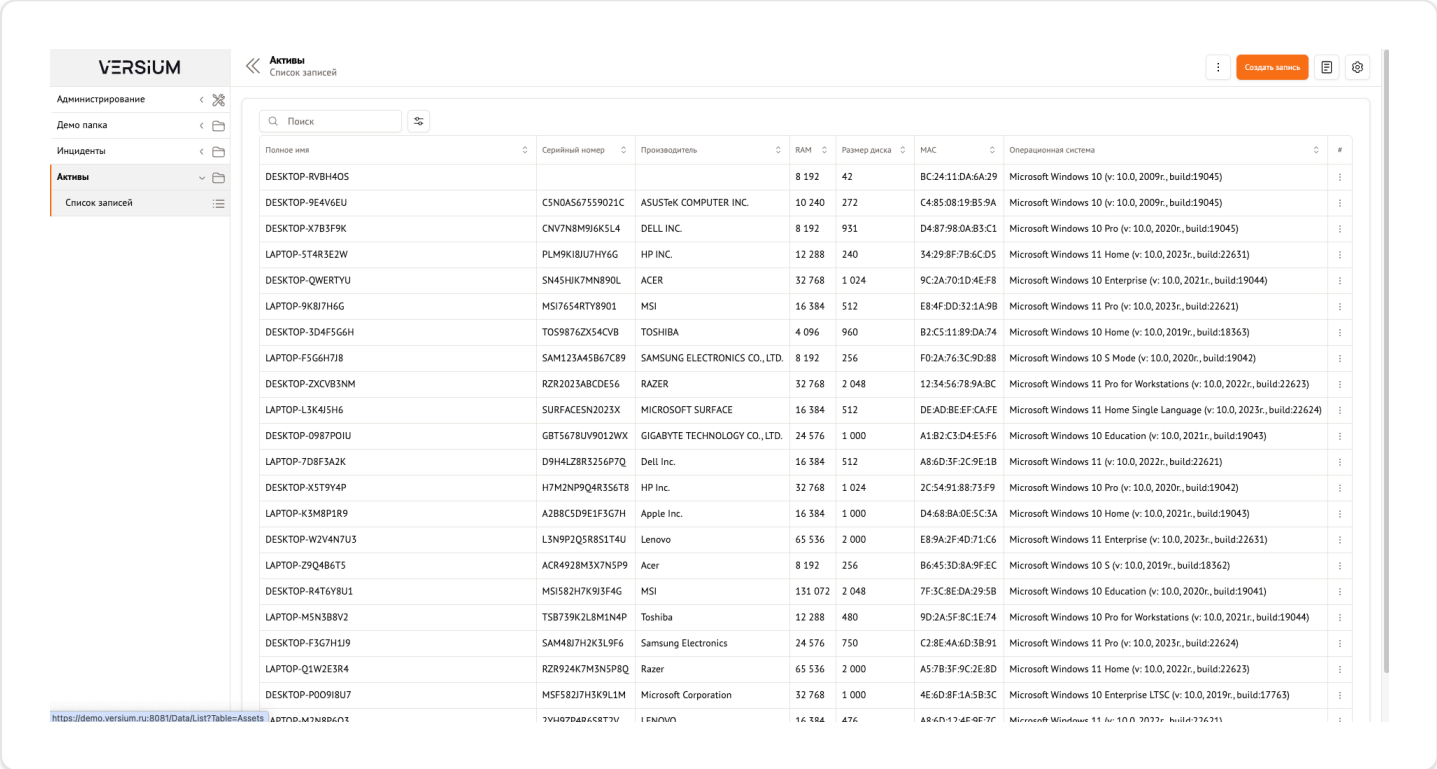
Раздел **«Активы»** предназначен для управления базой ИТ-активов (компьютеры, серверы, мобильные устройства, сетевое оборудование и др.), сбора их конфигурационных данных и отслеживания состояния.

Назначение

Этот модуль позволяет администраторам и аналитикам:

- Вести единую базу всех активов;
- Получать информацию о аппаратной и программной составляющей;
- Отслеживать изменения в конфигурации;
- Анализировать риски и уязвимости на уровне активов.

Каждый актив представляет собой функциональную единицу, которая может быть связана с инцидентами, пользователями и рабочими процессами.



На экране отображается список всех зарегистрированных активов с возможностью поиска и сортировки. Каждая запись содержит следующие поля:

- Полное имя — отображаемое имя актив (например, `DESKTOP-9E4V6EU`);
- Серийный номер — уникальный идентификатор устройства;
- Производитель — компания-изготовитель;
- RAM — объём оперативной памяти;
- Размер диска — объём жёсткого диска;
- MAC — MAC-адрес;

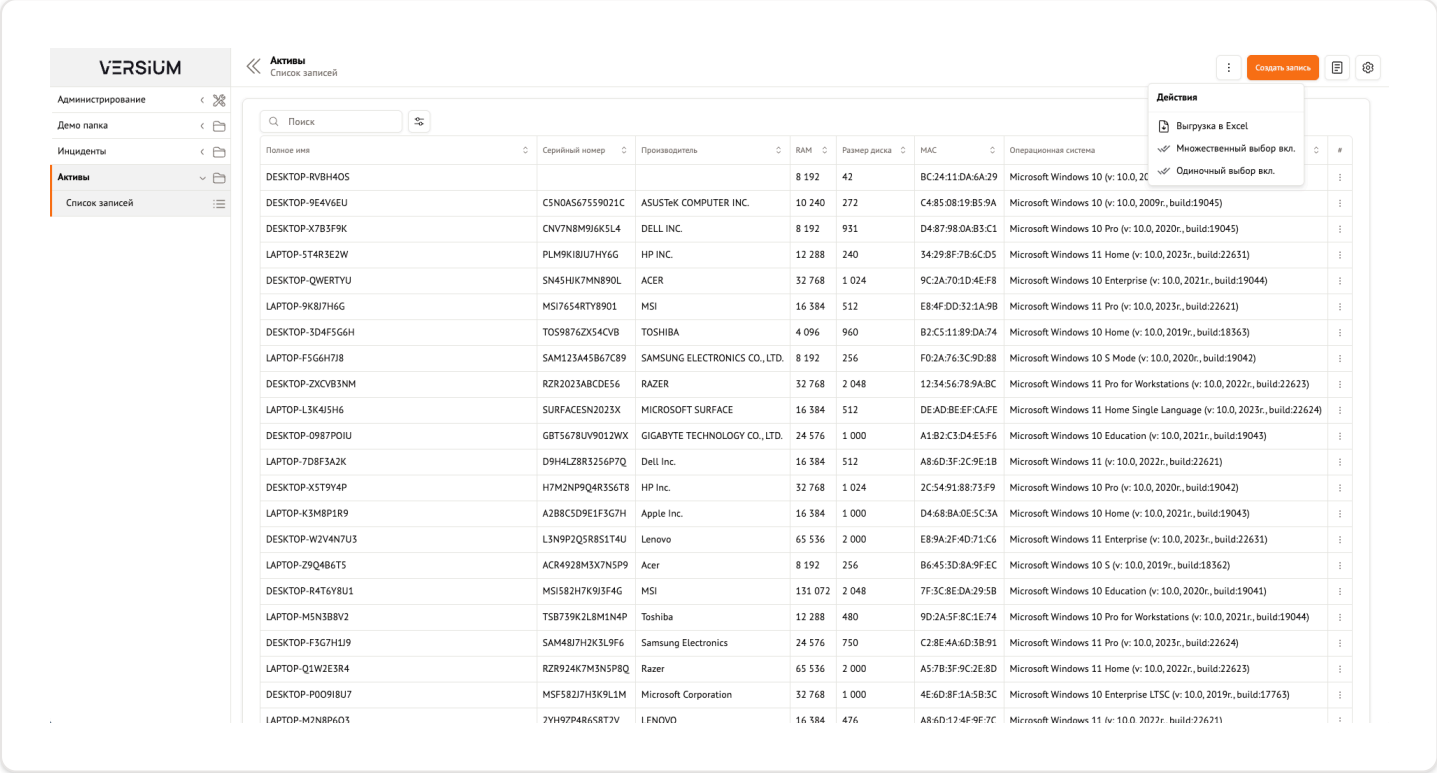
- Операционная система — установленная ОС и её версия.

Кнопка **«Создать запись»** позволяет добавить новый актив вручную или импортировать данные из внешних источников.

25.1. Функциональность списка активов

В верхней части формы расположен блок поиска:

- Поле **Поиск** — позволяет выполнять текстовый поиск по имени, серийному номеру или другим полям.



Полное имя	Серийный номер	Производитель	RAM	Размер диска	MAC	Операционная система
DESKTOP-RVBH4OS			8 192	42	BC:24:11:DA:6A:29	Microsoft Windows 10 (v: 10.0, 20
DESKTOP-9E4V6EU	C5N0AS67559021C	ASUSTeK COMPUTER INC.	10 240	272	C4:85:08:19:B5:9A	Microsoft Windows 10 (v: 10.0, 2009г., build:19045)
DESKTOP-X7B3F9K	CNV7N8M9J6K5L4	DELL INC.	8 192	931	D4:87:98:0A:B3:C1	Microsoft Windows 10 Pro (v: 10.0, 2020г., build:19045)
LAPTOP-5T4R3E2W	PLM9KI8JU7HY6G	HP INC.	12 288	240	34:29:8F:7B:6C:D5	Microsoft Windows 11 Home (v: 10.0, 2023г., build:22631)
DESKTOP-QWERTYU	SN45HJK7MN890L	ACER	32 768	1 024	9C:2A:70:1D:4E:F8	Microsoft Windows 10 Enterprise (v: 10.0, 2021г., build:19044)
LAPTOP-9K8J7H6G	MSI7654RTY8901	MSI	16 384	512	E8:4F:DD:32:1A:9B	Microsoft Windows 11 Pro (v: 10.0, 2023г., build:22621)
DESKTOP-3D4F5G6H	TOS9876ZX54CVB	TOSHIBA	4 096	960	B2:C5:11:89:DA:74	Microsoft Windows 10 Home (v: 10.0, 2019г., build:18363)
LAPTOP-F5G6H7J8	SAM123A45B67C89	SAMSUNG ELECTRONICS CO., LTD.	8 192	256	F0:2A:76:3C:9D:88	Microsoft Windows 10 S Mode (v: 10.0, 2020г., build:19042)
DESKTOP-ZXCVB3NM	RZR2023ABCDE56	RAZER	32 768	2 048	12:34:56:78:9A:BC	Microsoft Windows 11 Pro for Workstations (v: 10.0, 2022г., build:22623)
LAPTOP-L3K4J5H6	SURFACESN2023X	MICROSOFT SURFACE	16 384	512	DE:AD:BE:EF:CA:FE	Microsoft Windows 11 Home Single Language (v: 10.0, 2023г., build:22624)
DESKTOP-0987POIU	GBT5678UV9012WX	GIGABYTE TECHNOLOGY CO., LTD.	24 576	1 000	A1:B2:C3:D4:E5:F6	Microsoft Windows 10 Education (v: 10.0, 2021г., build:19043)
LAPTOP-7D8F3A2K	D9H4LZ8R3256P7Q	Dell Inc.	16 384	512	A8:6D:3F:2C:9E:1B	Microsoft Windows 11 (v: 10.0, 2022г., build:22621)
DESKTOP-X5T9Y4P	H7M2NP9Q4R3S6T8	HP Inc.	32 768	1 024	2C:54:91:88:73:F9	Microsoft Windows 10 Pro (v: 10.0, 2020г., build:19042)
LAPTOP-K3M8P1R9	A2B8C5D9E1F3G7H	Apple Inc.	16 384	1 000	D4:68:BA:0E:5C:3A	Microsoft Windows 10 Home (v: 10.0, 2021г., build:19043)
DESKTOP-W2V4N7U3	L3N9P2Q5R8S1T4U	Lenovo	65 536	2 000	E8:9A:2F:4D:71:C6	Microsoft Windows 11 Enterprise (v: 10.0, 2023г., build:22631)
LAPTOP-Z9Q4B6T5	ACR4928M3X7N5P9	Acer	8 192	256	B6:45:3D:8A:9F:EC	Microsoft Windows 10 S (v: 10.0, 2019г., build:18362)
DESKTOP-R4T6Y8U1	MSI582H7K9J3F4G	MSI	131 072	2 048	7F:3C:8E:DA:29:5B	Microsoft Windows 10 Education (v: 10.0, 2020г., build:19041)
LAPTOP-M5N3B8V2	TSB739K2L8M1N4P	Toshiba	12 288	480	9D:2A:5F:8C:1E:74	Microsoft Windows 10 Pro for Workstations (v: 10.0, 2021г., build:19044)
DESKTOP-F3G7H1J9	SAM48J7H2K3L9F6	Samsung Electronics	24 576	750	C2:8E:4A:6D:3B:91	Microsoft Windows 11 Pro (v: 10.0, 2023г., build:22624)
LAPTOP-Q1W2E3R4	RZR924K7M3N5P8Q	Razer	65 536	2 000	A5:7B:3F:9C:2E:8D	Microsoft Windows 11 Home (v: 10.0, 2022г., build:22623)
DESKTOP-P0O9I8U7	MSF582J7H3K9L1M	Microsoft Corporation	32 768	1 000	4E:6D:8F:1A:5B:3C	Microsoft Windows 10 Enterprise LTSC (v: 10.0, 2019г., build:17763)
LAPTOP-M2N8P6Q3	2YH97P4R6S8TJV	LENOVO	16 384	476	A8:6D:12:4F:9E:7C	Microsoft Windows 11 (v: 10.0, 2022г., build:22621)

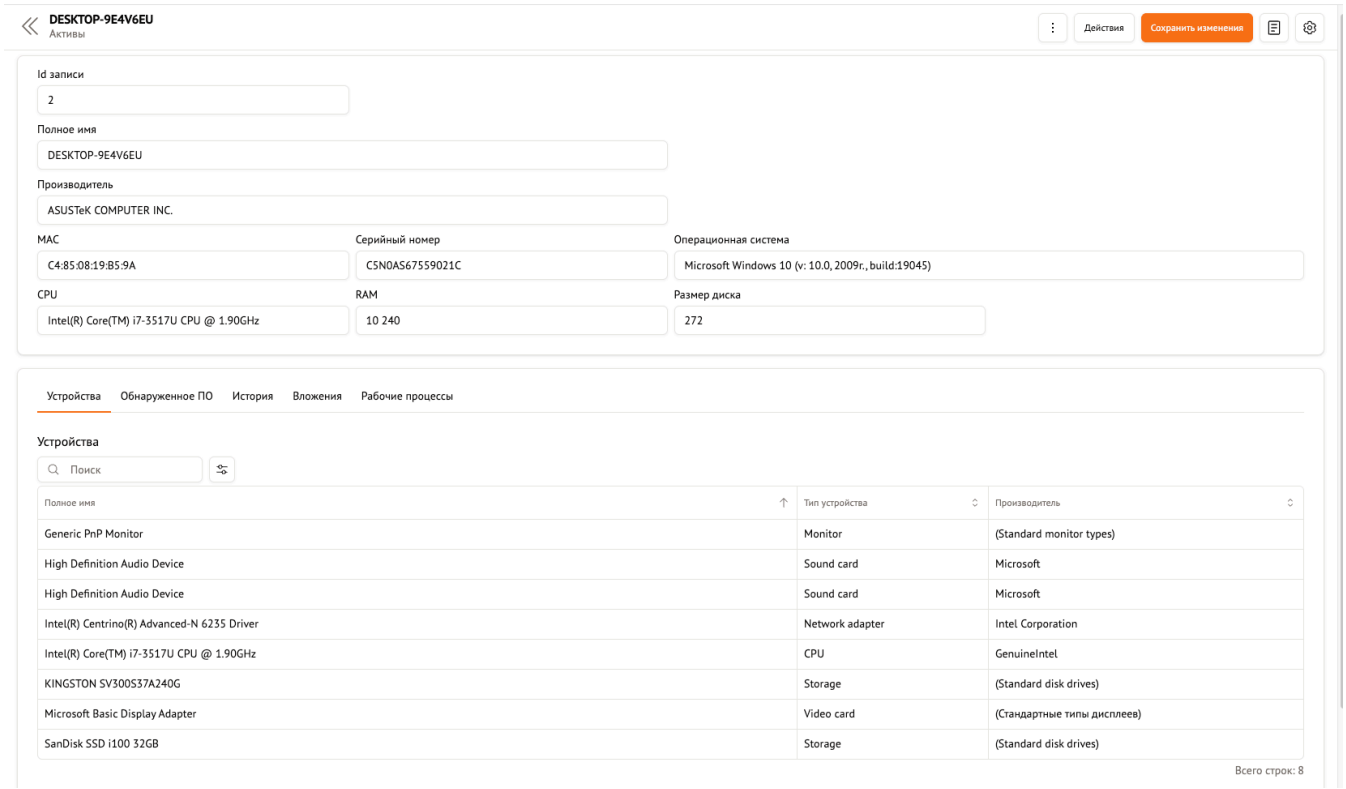
В правом верхнем углу расположена кнопка **«Действия»**, которая открывает выпадающее меню с дополнительными опциями.
В этом меню доступна функция **«Выгрузка в Excel»**, позволяющая экспортировать текущий список активов в формате `.xlsx`. Эта функция особенно полезна для:

- Генерации отчётов для руководства;
- Аудита IT-инфраструктуры;
- Архивации информации.

При нажатии на эту опцию система формирует файл, содержащий все видимые поля из текущего списка, включая фильтры и сортировку. Выгрузка выполняется в фоновом режиме, и пользователь получает ссылку для скачивания.

25.2. Карточка актива

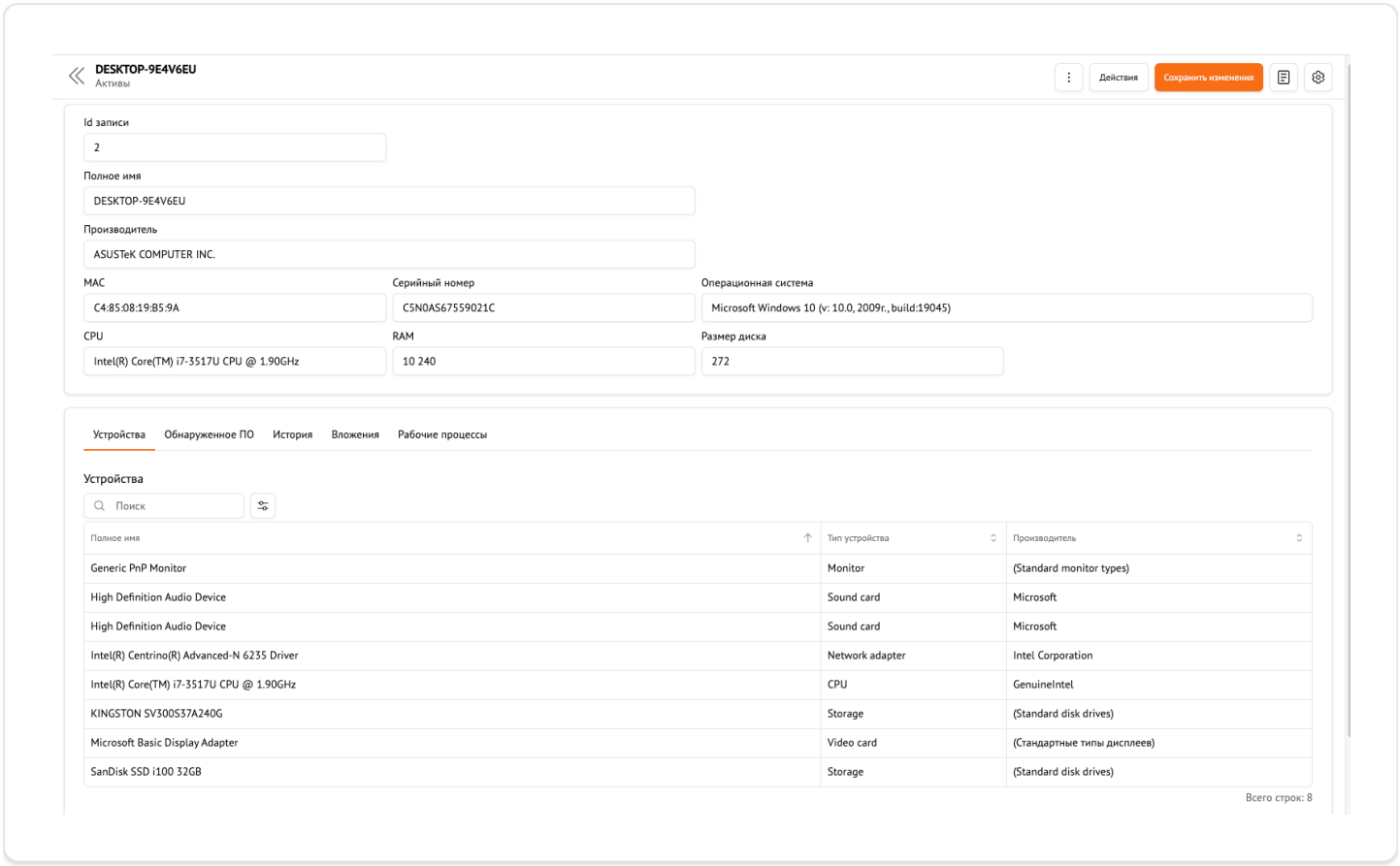
При клике на строку актив открывается его карточка с подробной информацией.



На форме отображаются основные параметры актив:

- Id записи — уникальный идентификатор;
- Полное имя — отображаемое имя;
- Производитель — компания-изготовитель;
- MAC — MAC-адрес;
- Серийный номер — уникальный идентификатор;
- Операционная система — установленная ОС;
- CPU — процессор;
- RAM — объём оперативной памяти;
- Размер диска — объём жёсткого диска.

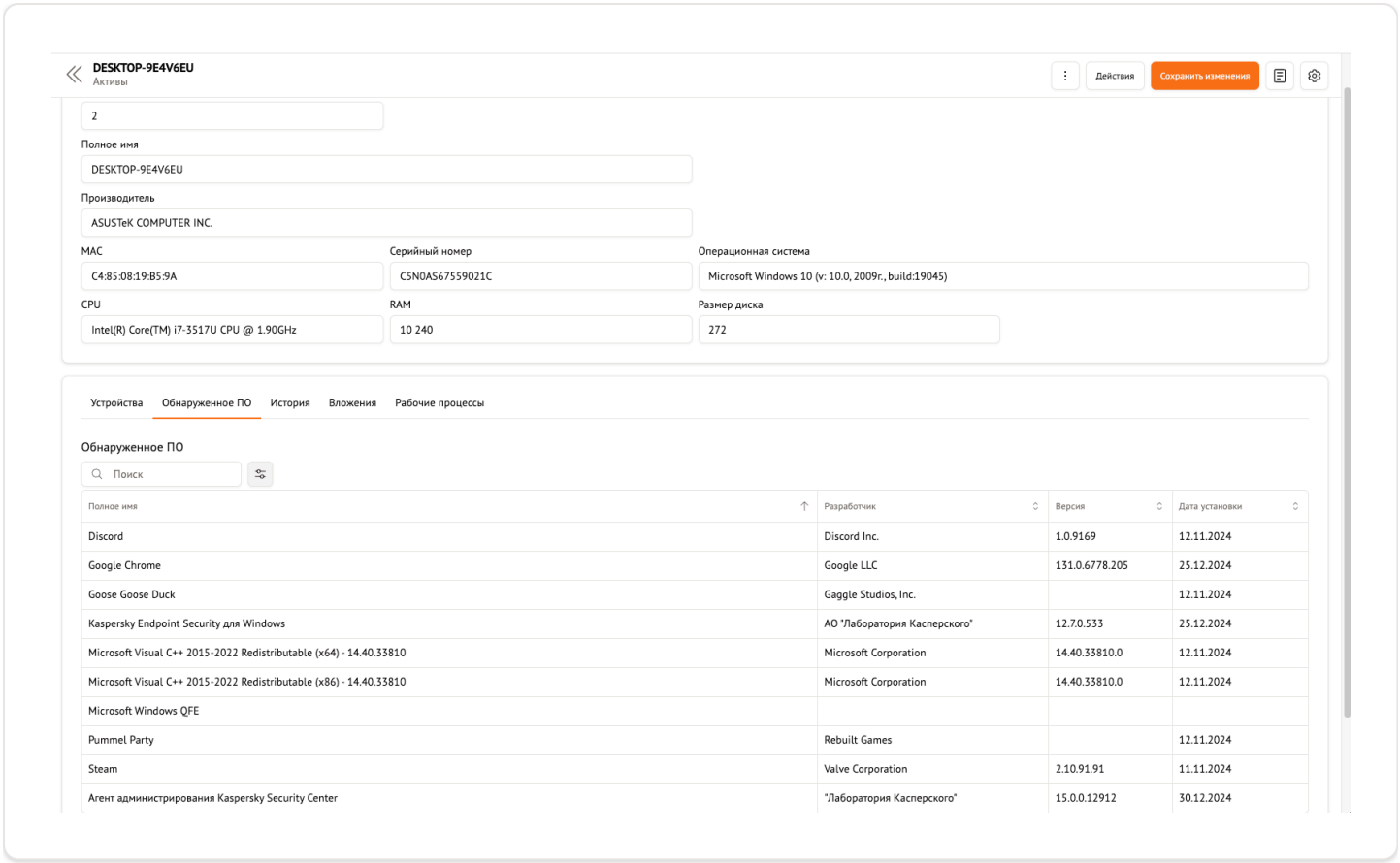
Нижняя часть формы содержит вкладки, которые могут быть настроены администратором через раздел **Администрирование → Системные таблицы → Assets → Экраны → Форма**.
Вкладка «Устройства»



На этой вкладке отображается список физических устройств, подключённых к активу:

- Полное имя — название устройства;
- Тип устройства — категория (например, Monitor, Sound card, Network adapter);
- Производитель — компания-изготовитель.

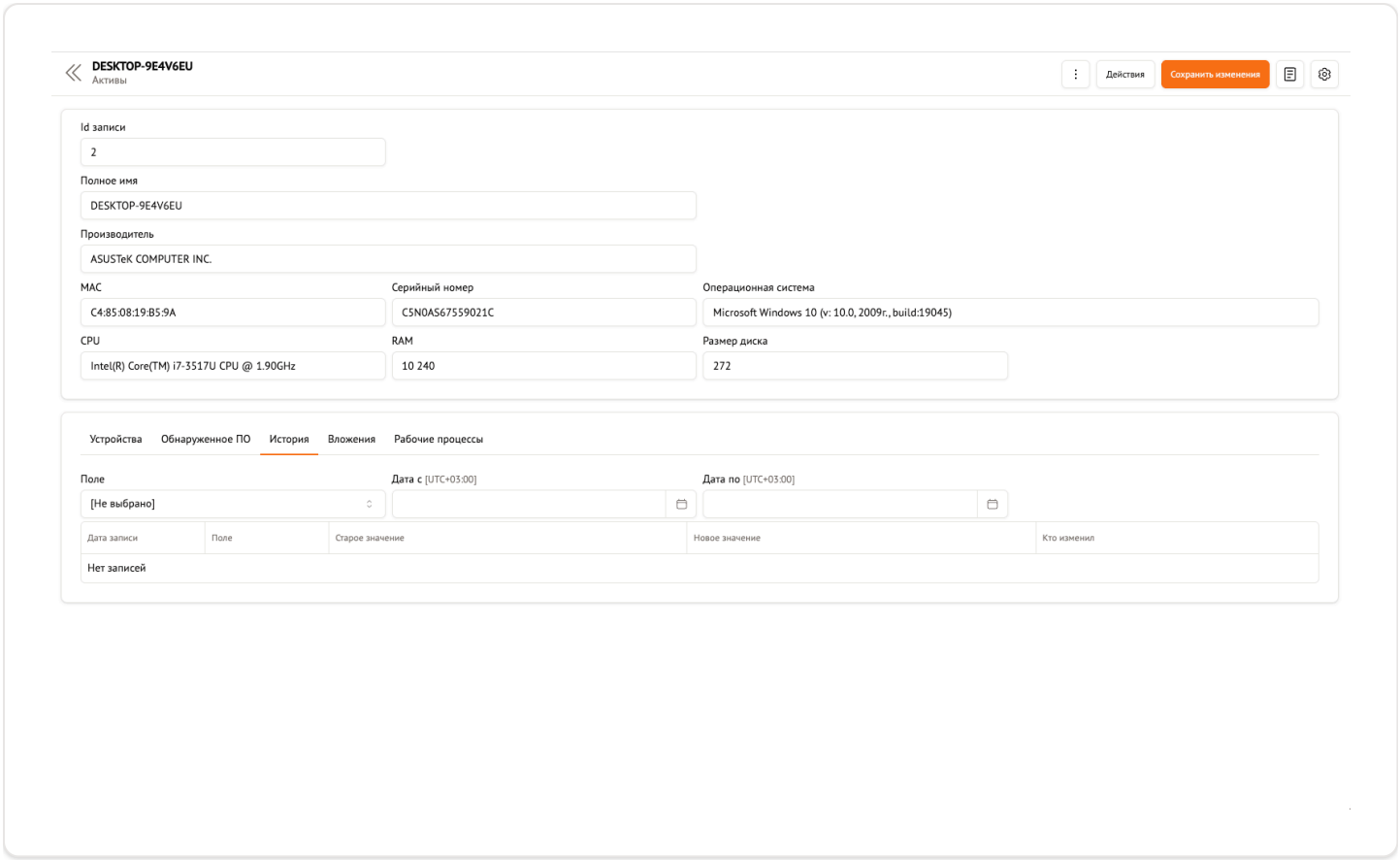
Вкладка «Обнаруженное ПО»



На этой вкладке отображается список программного обеспечения, установленного на активе:

- Полное имя — название программы;
- Разработчик — компания-разработчик;
- Версия — установленная версия;
- Дата установки — дата установки программы.

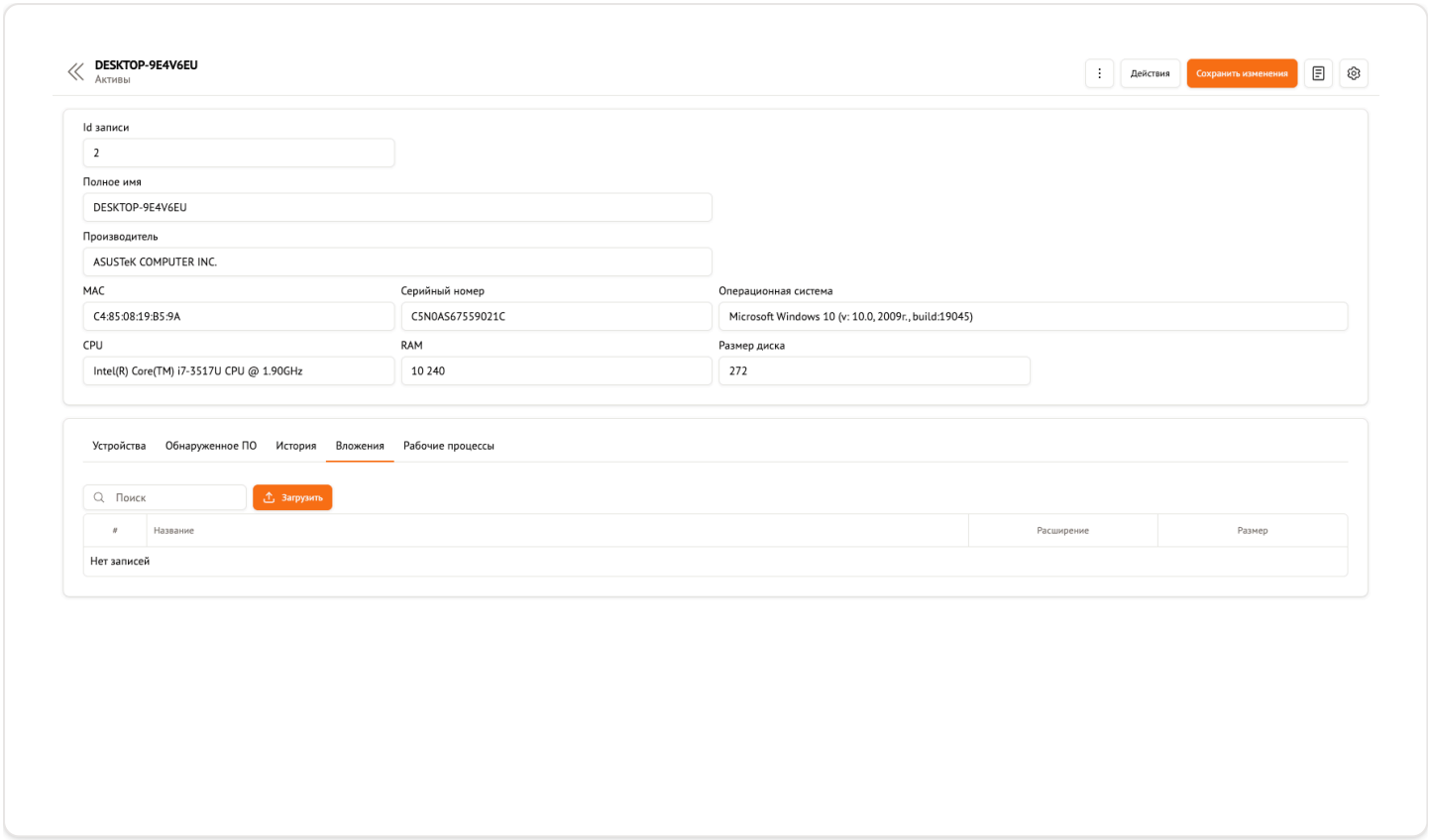
Вкладка «История»



На этой вкладке отображается история изменений значений полей актив:

- Поле — название изменённого поля;
- Старое значение — предыдущее значение;
- Новое значение — новое значение;
- Кто изменил — пользователь, внесший изменения;
- Дата записи — временная метка события.

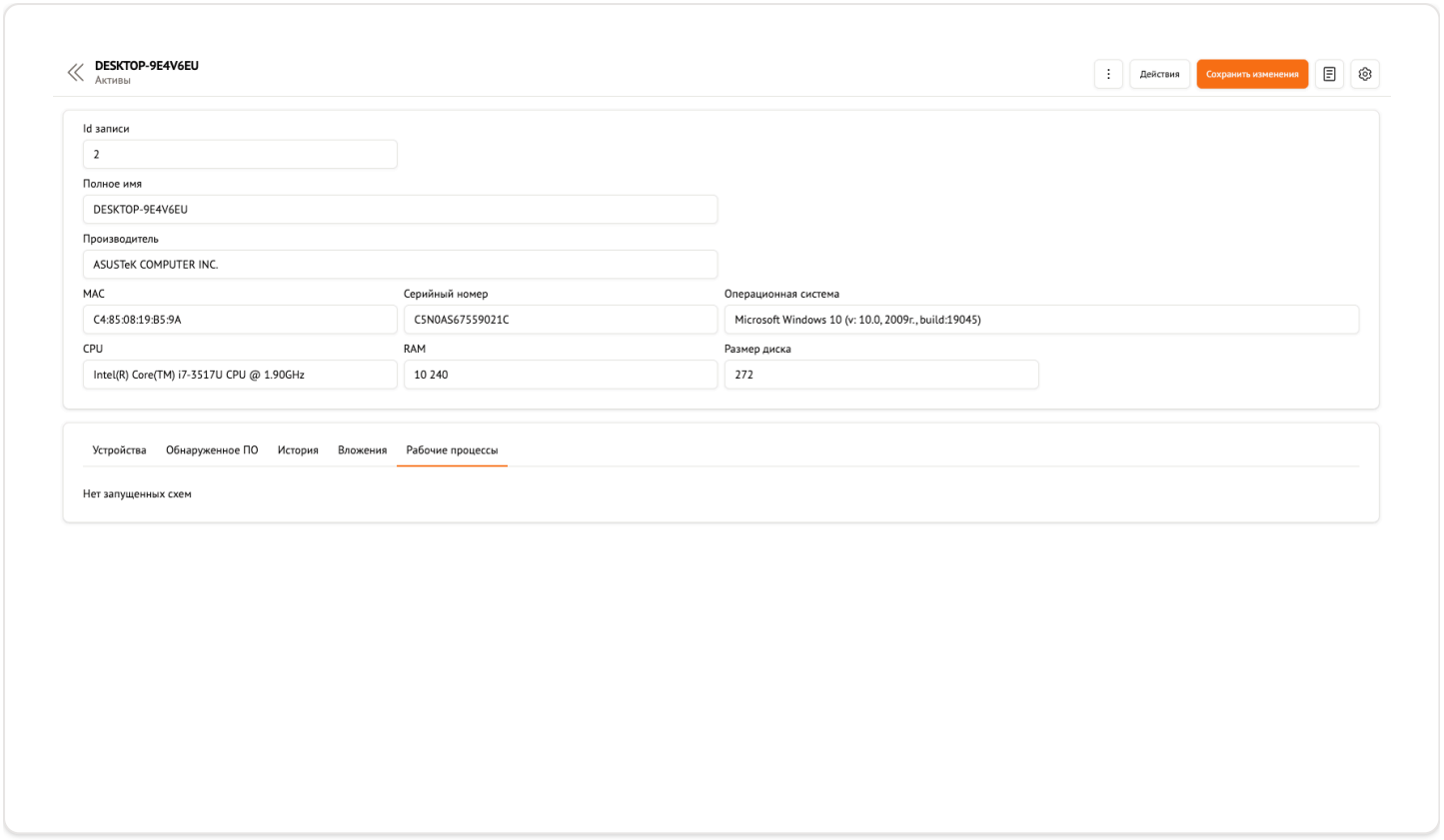
Вкладка «Вложения»



На этой вкладке можно прикреплять файлы к активу (например, фотографии, документы).
Функциональность:

- Кнопка **«Загрузить»** — позволяет выбрать и загрузить файлы;
- Таблица отображает список прикреплённых файлов с указанием:
  - Название файла;
  - Расширение;
  - Размер;
- Поддерживается возможность удаления файлов (через контекстное меню).

Вкладка «Рабочие процессы»



На этой вкладке отображается список запущенных рабочих процессов по данному активу.

- Если процессов нет — отображается сообщение **«Нет запущенных схем»**;
- При наличии процессов — отображаются их названия, статусы и параметры.

Примечание: Состав вкладок и полей в карточке актива может быть полностью настроен администратором. Это осуществляется через раздел **Администрирование → Системные таблицы → Assets → Экраны → Форма**, где можно:

- Выбирать поля, которые будут отображаться в карточке;
- Группировать поля по вкладкам;
- Настраивать порядок отображения;
- Управлять видимостью полей (включая динамическую логику через скрипты);
- Добавлять новые вкладки (например, «Комментарии», «Риски», «Связанные инциденты») и заполнять их нужным набором полей.

Такой подход позволяет адаптировать интерфейс под специфические требования организации без необходимости программирования.

25.3. Обобщение функциональности

Принцип работы раздела «Активы» как функциональной единицы

1. Администратор или аналитик переходит в раздел «Активы»;
2. Просматривает список активов;
3. Использует фильтры и поиск для узкой выборки;
4. Переходит в карточку актива для детального анализа;

5. Управляет данными и связями с другими объектами.

Применимость

- Раздел используется для создания и поддержания единой базы IT-активов;
- Поддерживает интеграцию с EDR, SIEM и другими системами мониторинга;
- Позволяет анализировать конфигурацию и риски на уровне устройств;
- Используется в процессах аудита, инвентаризации и реагирования на инциденты.

Все операции доступны пользователям с соответствующими правами.