

**Инструкция по интеграции электронно-
логистической платформы «Портал»
с «1С:Предприятие 8»**

Версия 0.1.1

СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ.....	3
1.1. Сфера применения документа	3
1.2. Назначение интерфейса	3
1.3. Архитектура интерфейса	3
1.4. Авторизация.....	4
2. НАСТРОЙКА ИНТЕГРАЦИИ	5
3. МЕХАНИЗМ РАБОТЫ ИНТЕГРАЦИИ	22
3.1. Общая схема взаимодействия	22
3.2. Авторизация и выбор пользователя	22
3.3. Загрузка справочников	22
3.4. Создание и редактирование тендера из 1С.....	23
3.5. Синхронизация тендеров с площадки (кнопка «Обновить»)	23
3.6. Удаление тендера	24
3.7. Продление тендера.....	24
3.8. Автоподбор городов.....	24
4. МЕТОДЫ API.....	25
4.1. Проверка токена и получение профиля пользователя.....	25
4.2. Получение справочников	26
4.3. Поиск пунктов маршрута (городов).....	27
4.4. Получение списка тендеров	28
4.5. Получение полных данных тендера	30
4.6. Создание нового тендера.....	31
4.7. Редактирование тендера	33
4.8. Удаление тендера	36
4.9. Продление тендера.....	36
5. ОБРАБОТКА ОШИБОК И КОДЫ ОТВЕТА.....	39
5.1. Основные HTTP-коды ответов	39
5.2. Обработка ошибок в 1С	39

1. ВВЕДЕНИЕ

1.1. Сфера применения документа

Настоящая инструкция описывает порядок интеграции между электронной логистической платформой «Портал» (ShipperAPI версии 1.0.0) и конфигурацией 1С:Предприятие 8 с помощью специально разработанного расширения.

Интеграция позволяет автоматизировать обмен данными по тендерам на грузоперевозки в обе стороны: из 1С на площадку и обратно.

1.2. Назначение интерфейса

Интерфейс API обеспечивает взаимодействие по тендерам на грузоперевозки между электронно-логистической платформой «Портал» и системой «1С:Предприятие 8». Основные функции включают:

- создание и редактирование тендеров прямо из 1С;
- синхронизация списка тендеров и их актуального состояния;
- получение ставок, информации о перевозчике и водителе;
- работа со справочниками платформы (способы погрузки, грузы, типы кузова).
- безопасное удаление и продление тендеров.

1.3. Архитектура интерфейса

Взаимодействие между 1С:Предприятие 8 и платформой «Портал» построено на принципах REST API и осуществляется исключительно по защищенному протоколу HTTPS.

Для выполнения операций с ресурсами используются следующие стандартные HTTP-методы:

GET — получение информации (чтение данных без изменения состояния на сервере). Применяется для получения справочников (способы погрузки, грузы, типы кузова), поиска городов, списка тендеров, детальной информации по конкретному тендеру и ставок по тендеру.

POST — создание новых объектов. Используется при создании нового тендера, продлении тендера и загрузке файлов.

PATCH — частичное обновление существующего объекта. Применяется для редактирования (изменения) уже созданного тендера (сроки, цена, описание и т.д.).

DELETE — удаление объекта. Используется для отмены и полного удаления тендера с площадки.

Все данные в запросах и ответах (кроме бинарных файлов) передаются в формате JSON (кодировка UTF-8). Авторизация в каждом запросе выполняется через HTTP-заголовок: Authorization: Bearer <ключ_API>

При отправке данных в тело запроса обязательно указывается заголовок Content-Type: application/json.

В расширении 1С взаимодействие с API реализовано с помощью стандартных объектов HTTPСоединение и HTTPЗапрос. Это обеспечивает высокую надежность, простоту поддержки и полную совместимость с типовыми конфигурациями.

1.4. Авторизация

Для каждого запроса используется Bearer Token, который хранится в безопасном хранилище 1С.

2. НАСТРОЙКА ИНТЕГРАЦИИ

Подключение расширения происходит средствами типовой конфигурации. Для добавления системы необходимо перейти во вкладку «НСИ и администрирование», далее нажать «Печатные формы, отчеты и обработки» (Рис. 1), а затем «Расширение» (Рис. 2).

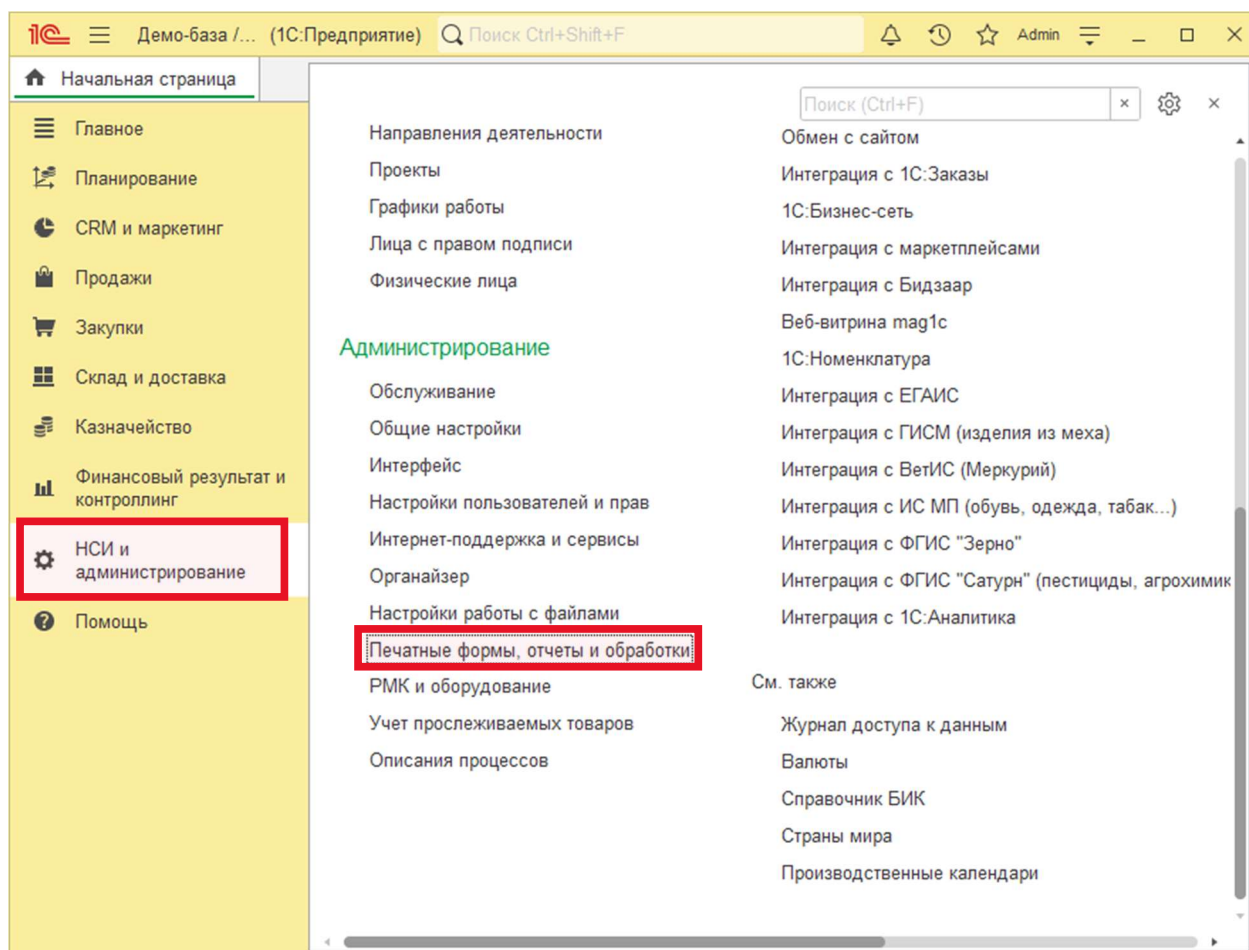


Рис. 1

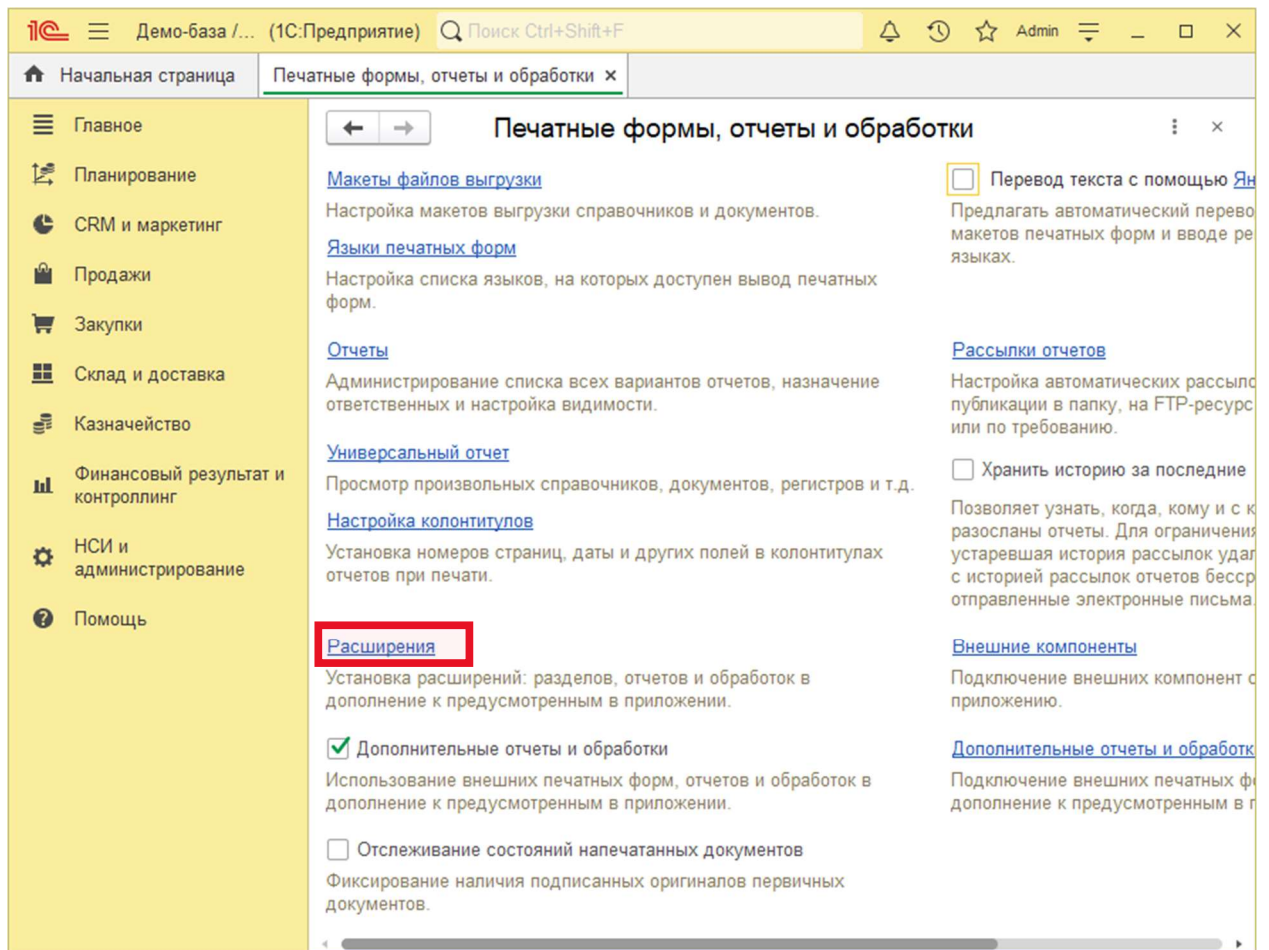


Рис. 2

После чего следует добавить конфигурацию из файла (Рис. 3).

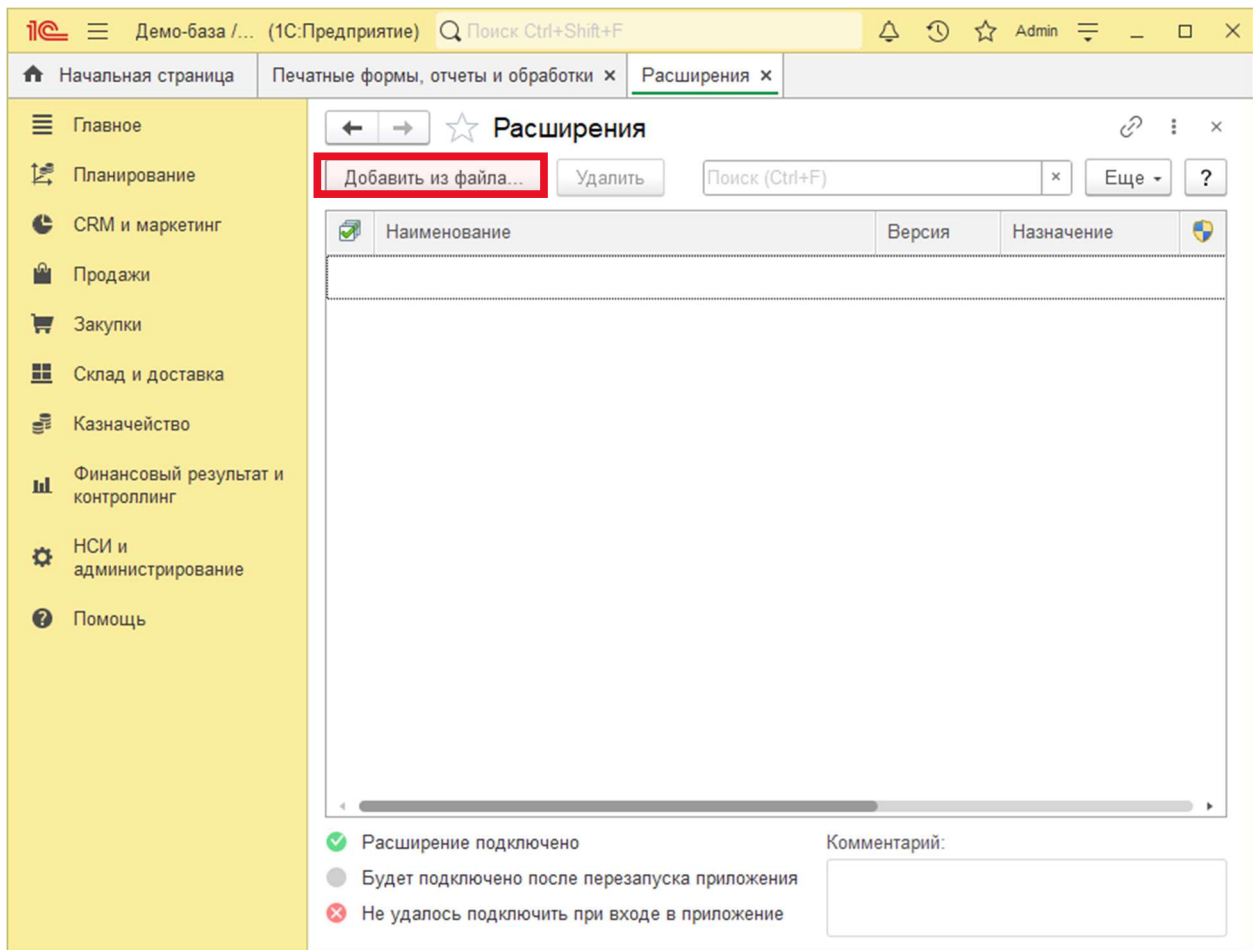


Рис. 3

Если появится следующее предупреждение, выберете «Продолжить» (Рис. 4).

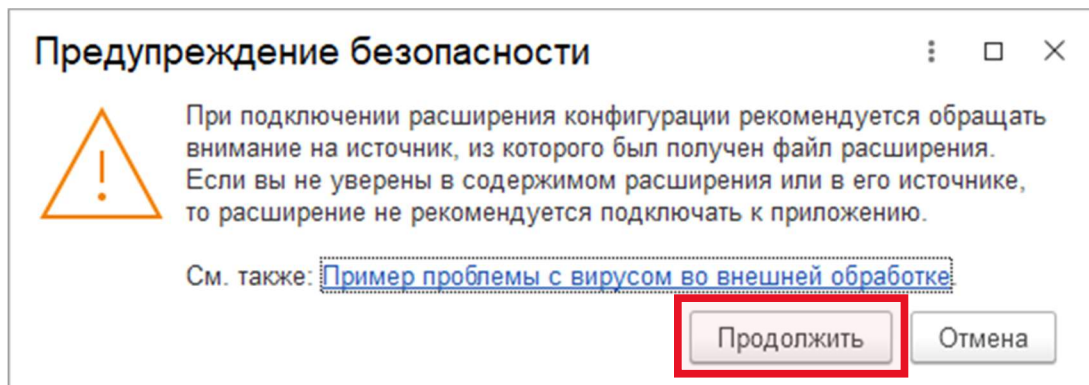


Рис. 4

Следующим шагом открываем добавляемое расширение (Рис. 5).

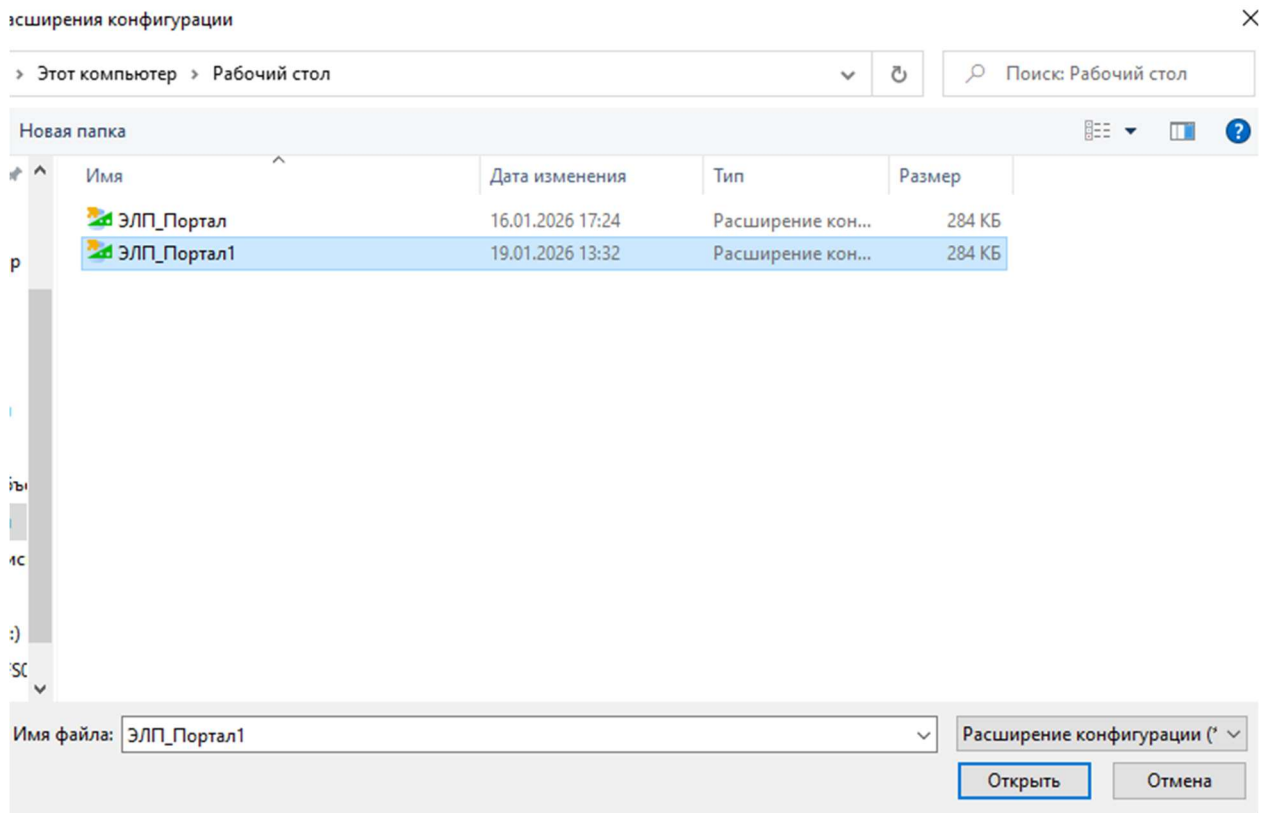


Рис. 5

После загрузки необходимо снять отметку безопасного режима и перезапустить конфигурацию (Рис. 6).

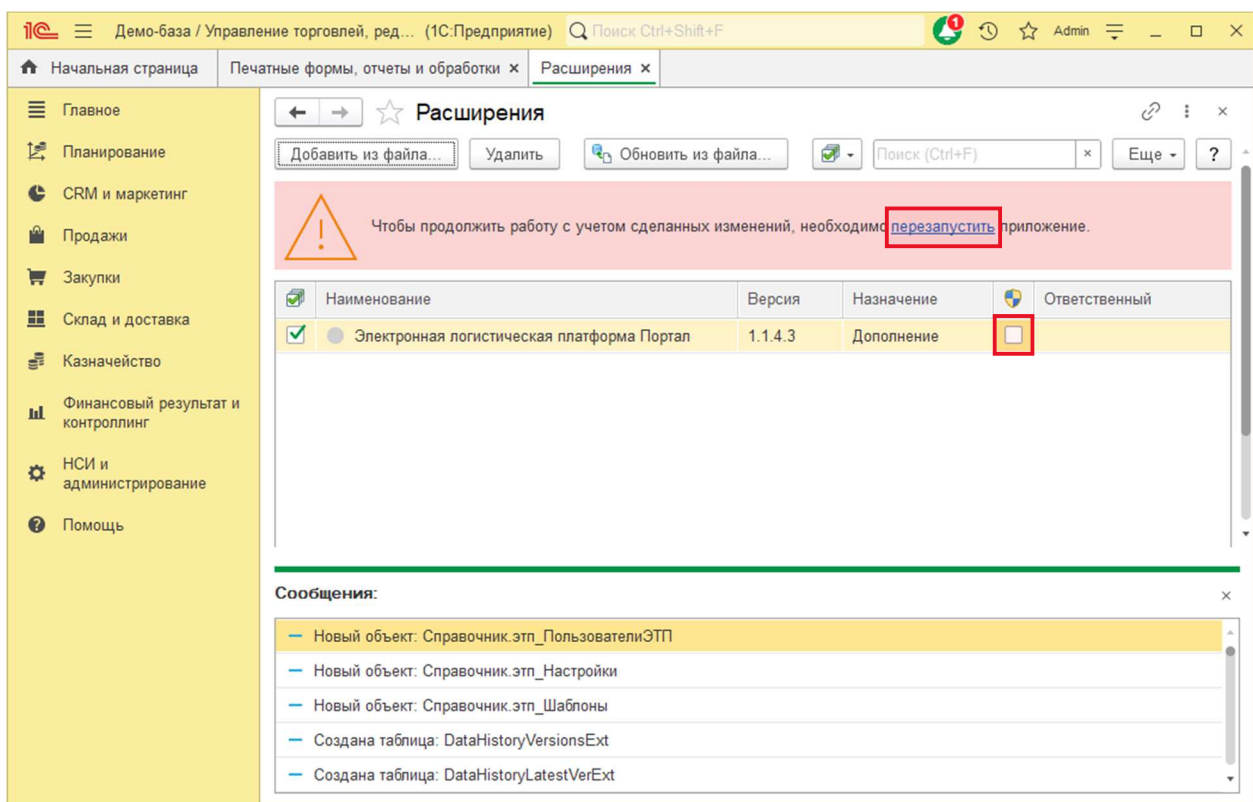


Рис. 6

При открытии появится новая вкладка с интегрируемой системой (Рис. 7).

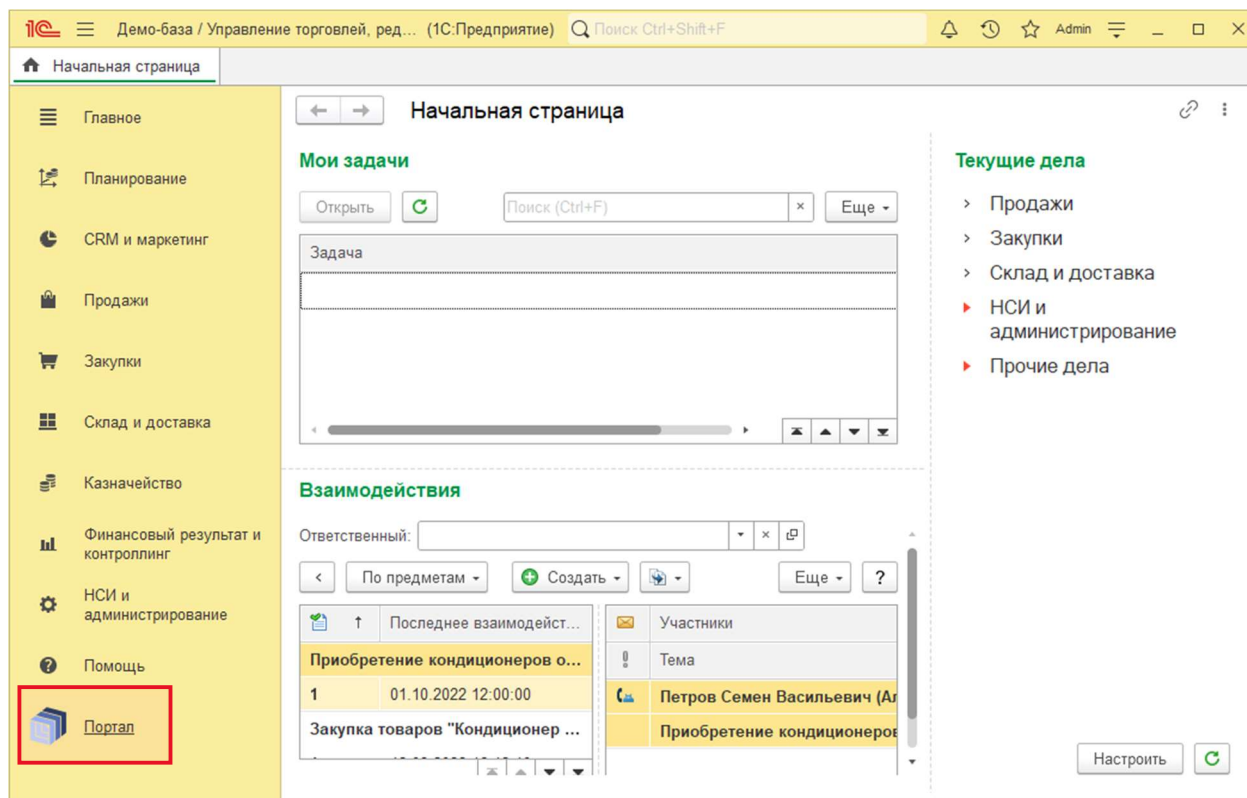


Рис. 7

«Портал» включает в себя три основных вкладки «Рабочее место», «Настройки платформы Портал», «Учетные записи платформы Портал» (Рис. 8). Чтобы начать работать с расширением, необходимо перейти либо в «Рабочее место» (при первом открытии выйдет предупреждение и автоматически откроются настройки (Рис. 9)), либо в «Настройки платформы Портал» (Рис. 10).

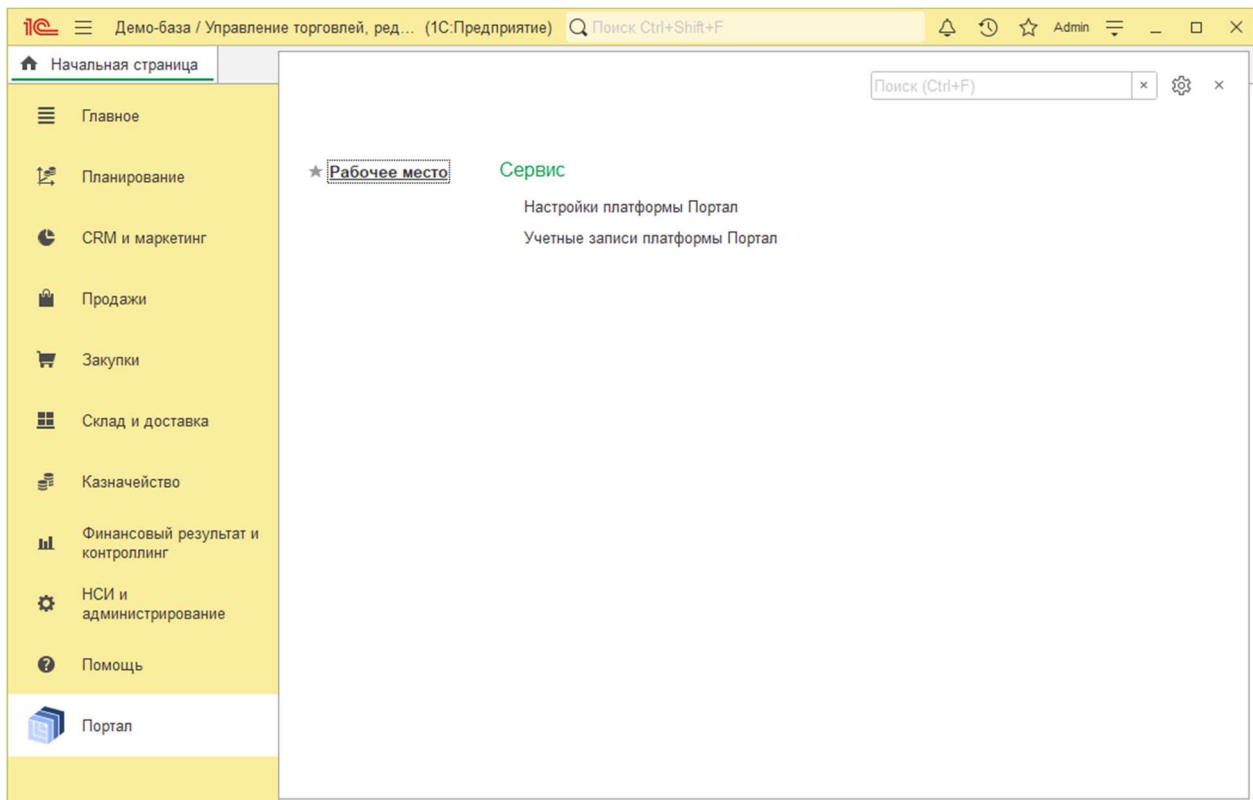


Рис. 8

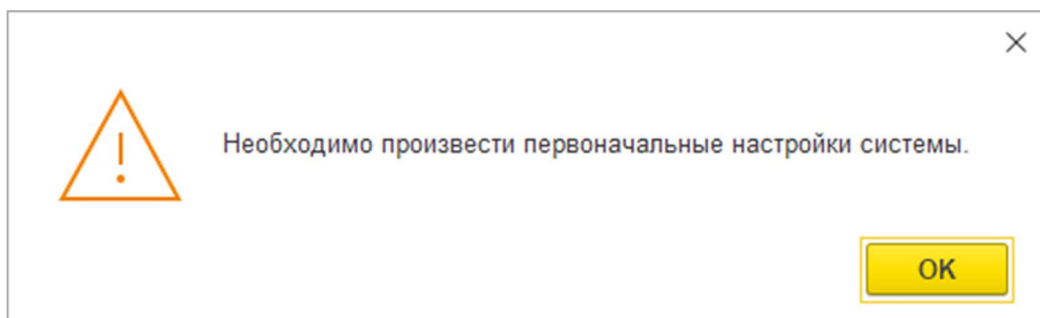


Рис. 9

Заполняем поле «Адрес API», при необходимости меняем поля «Порт» и «Таймаут». (Рис. 10)

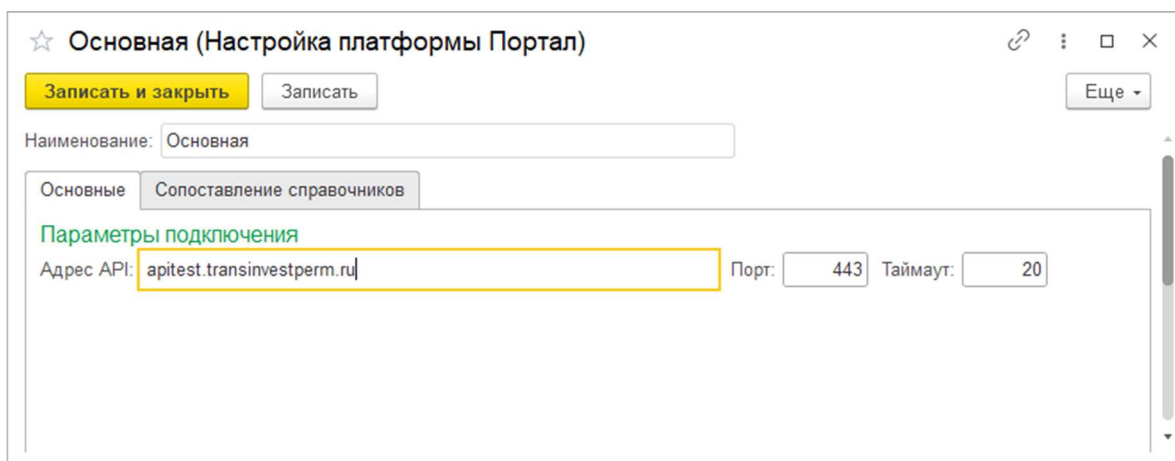


Рис. 10

Следующим шагом сопоставляем необходимые справочники конфигурации с данным электронно-логистической платформы «Портал» (Рис. 11, Рис. 12).

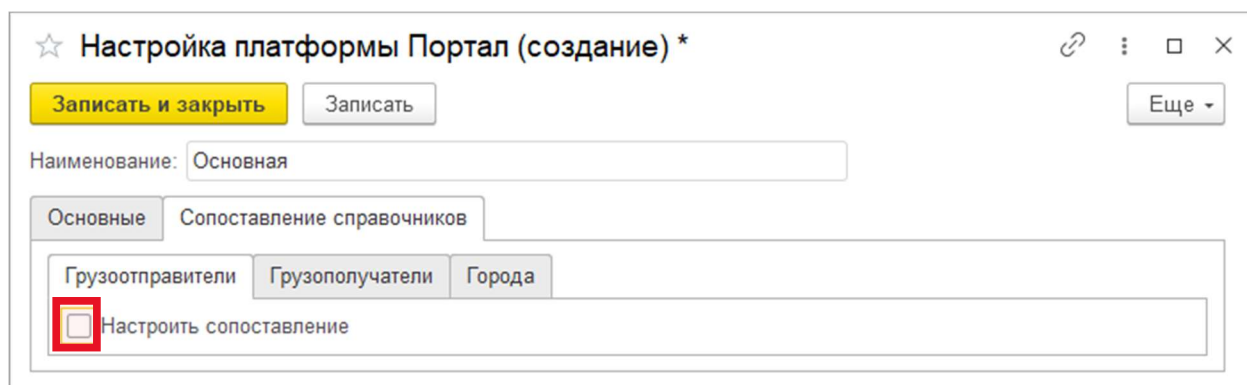


Рис. 11

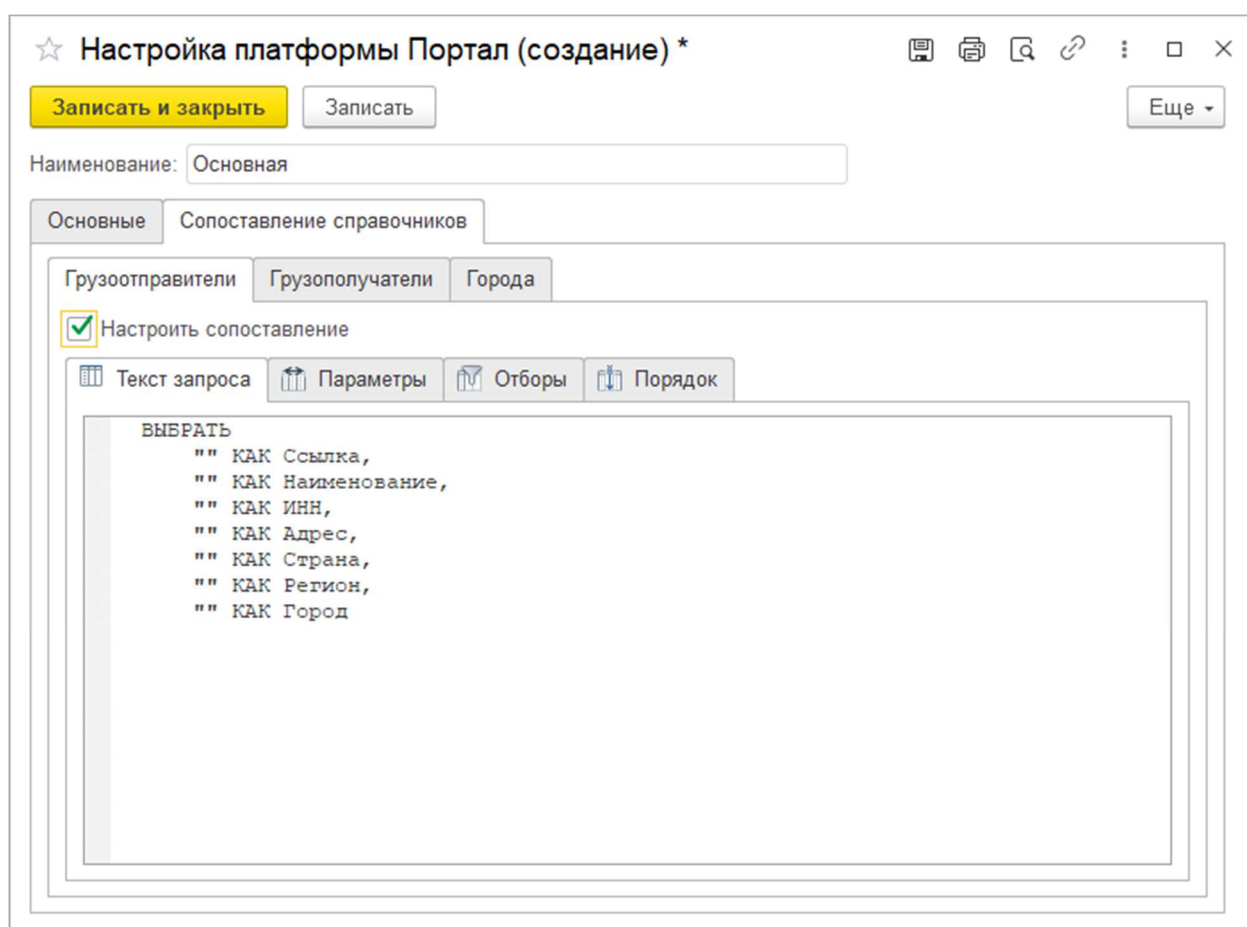


Рис. 12

Определяем ссылку на справочник, затем наименование, ИНН. Далее из табличной части «Контактная информация» выбираем «Представление» как «Адрес», страну, регион и город. После этого устанавливаем связи (Рис. 13).

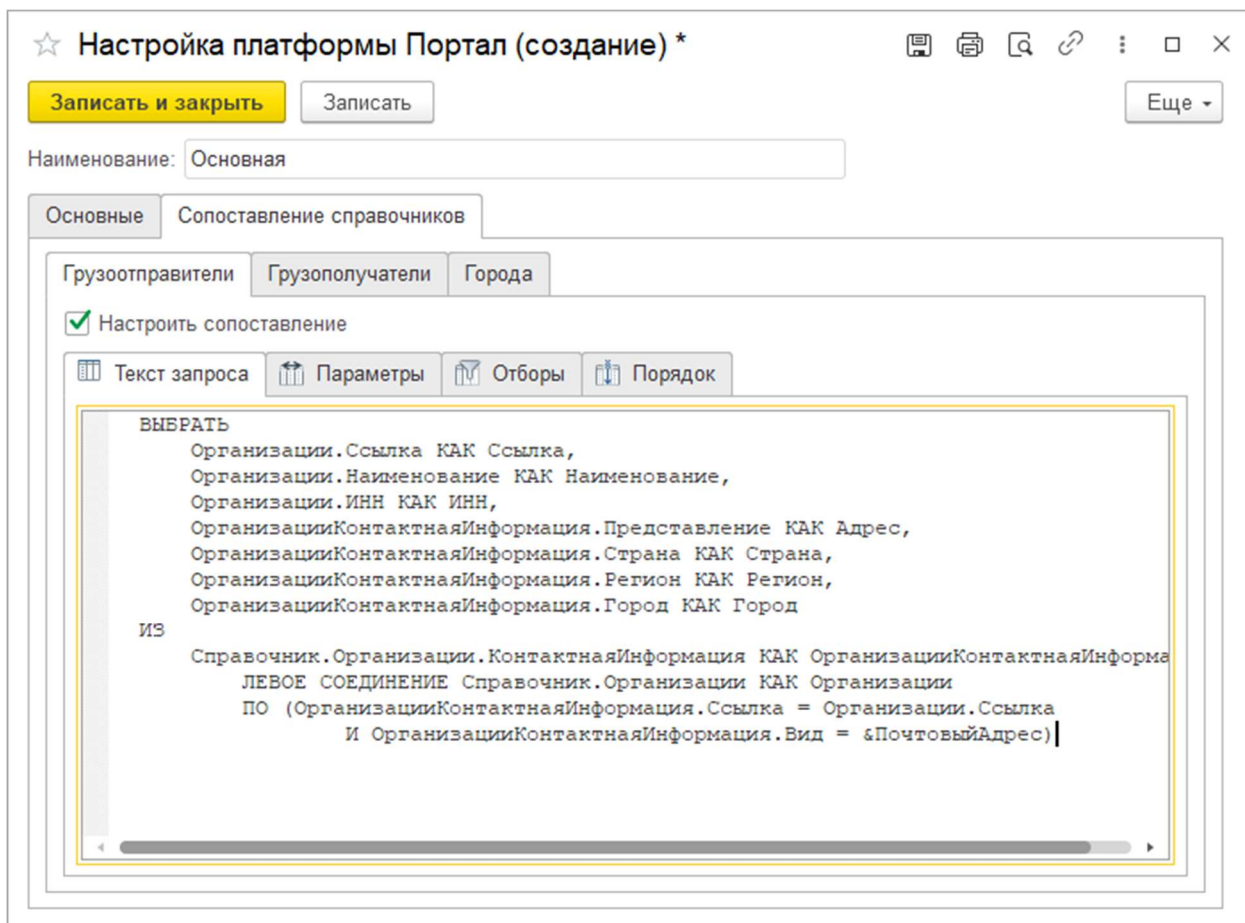


Рис. 13

Продолжая настройку, переходим в следующие вкладки и настраиваем параметр «Почтовый адрес» (Рис. 14), также по необходимости можно настроить отбор и порядок (Рис. 15).

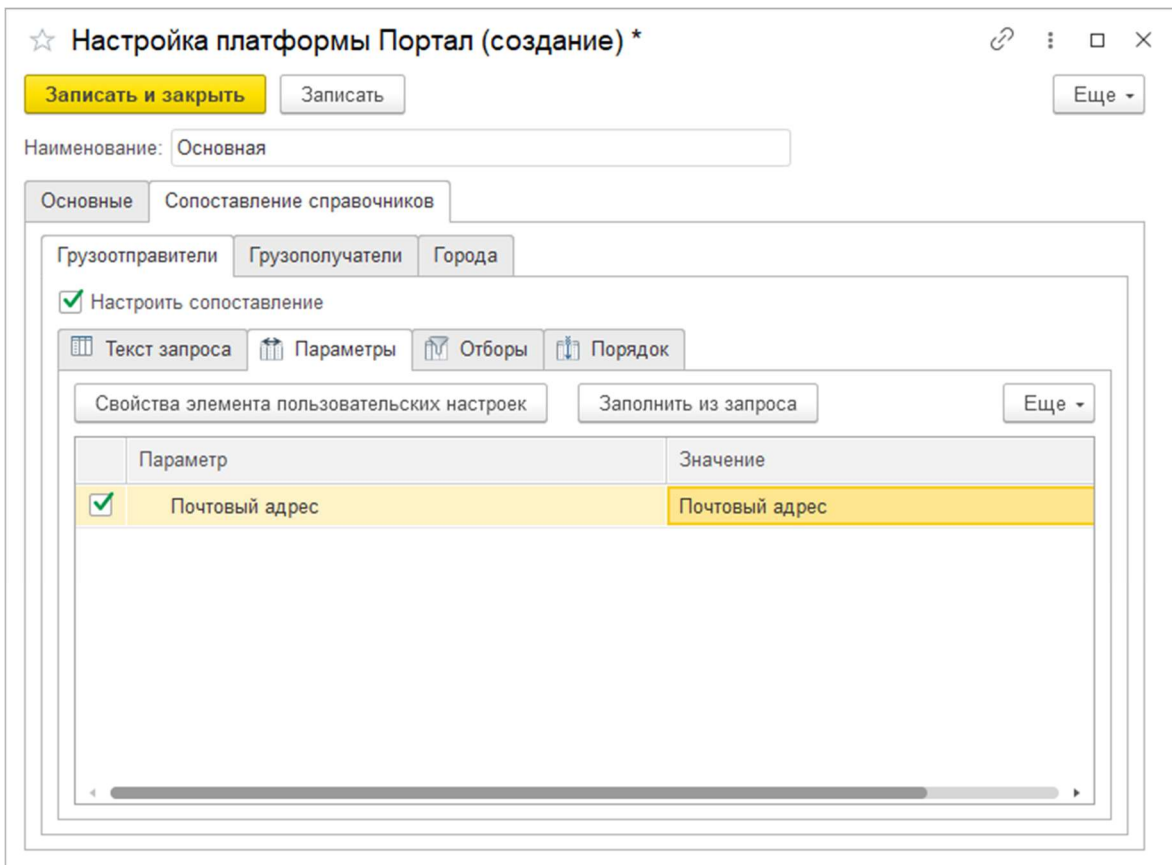


Рис. 14

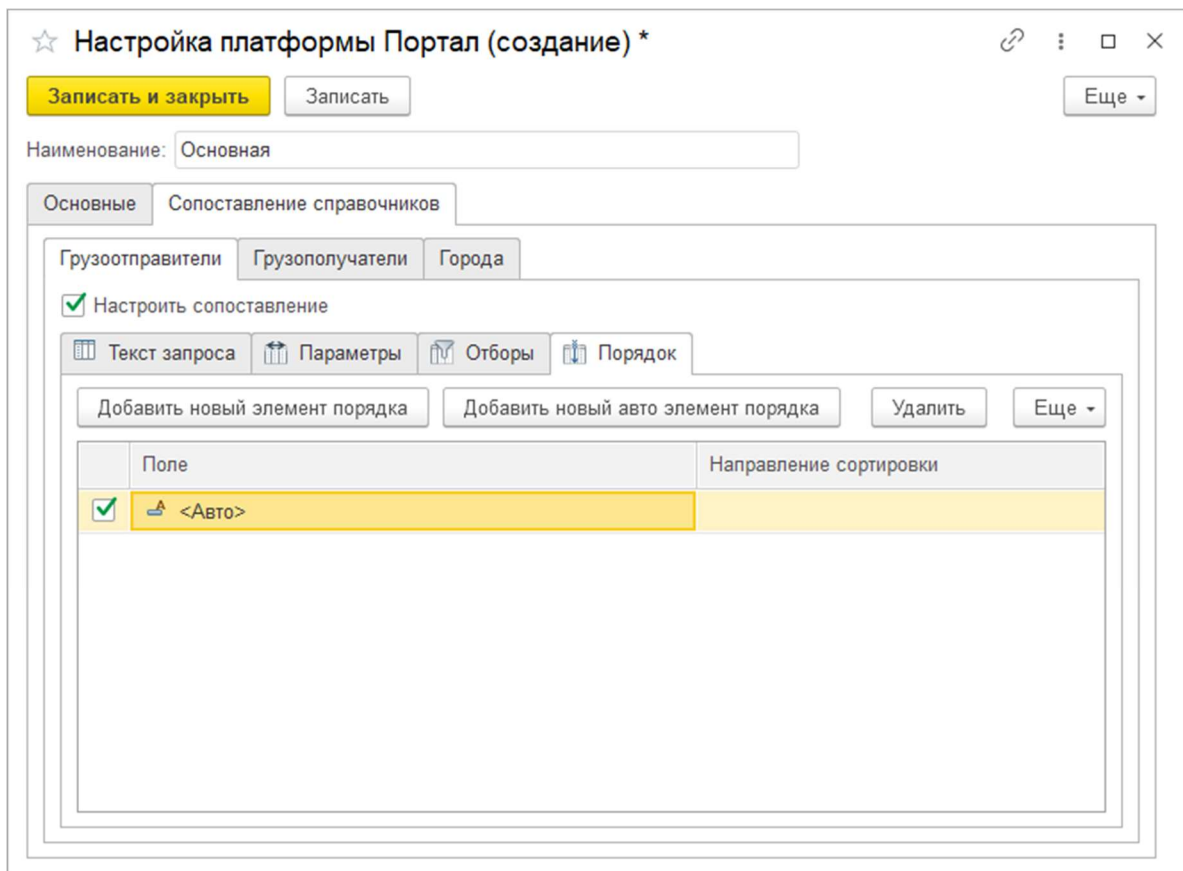


Рис. 15

Аналогично проделываем с грузополучателями (Рис. 16, Рис. 17, Рис. 18).

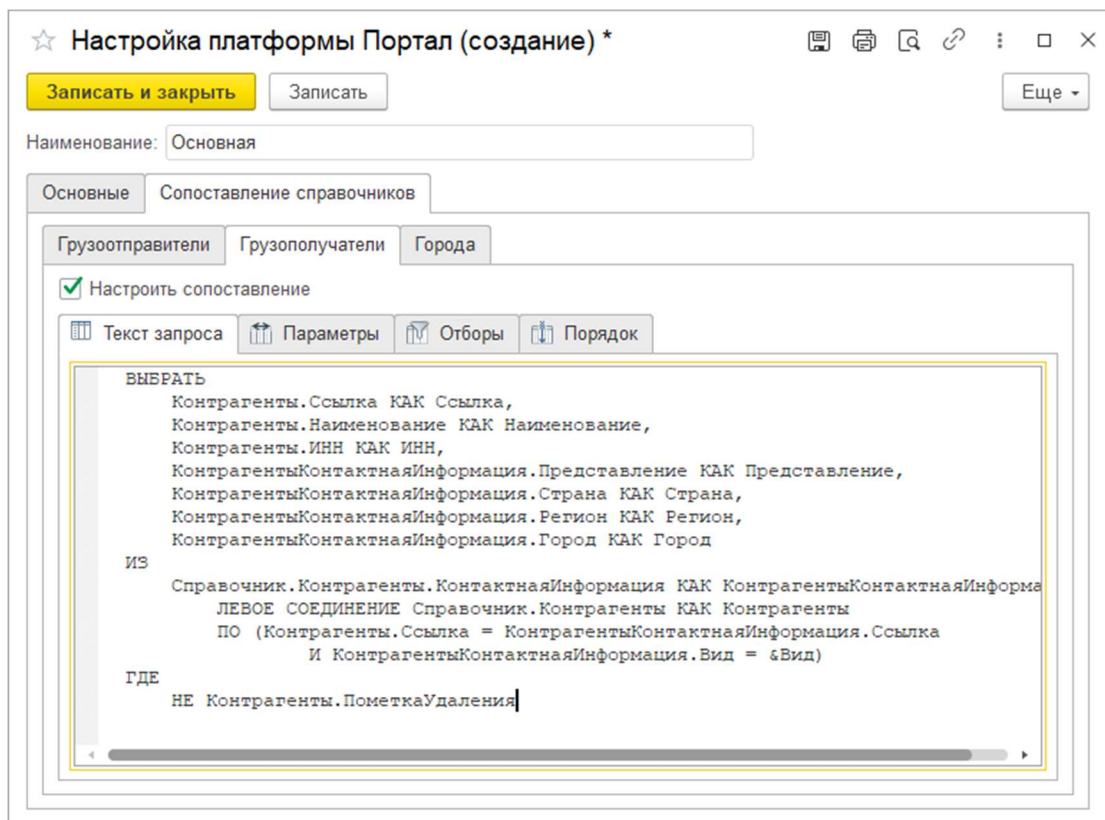


Рис. 16

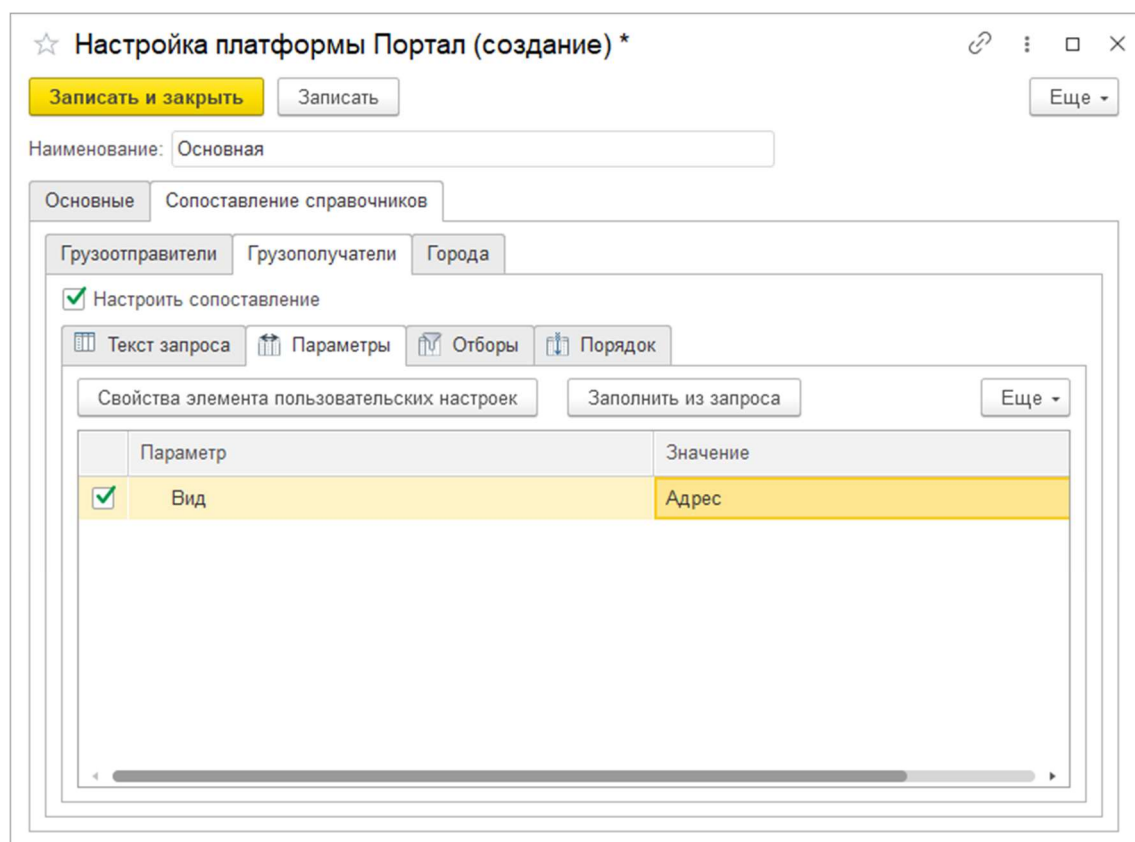


Рис. 17

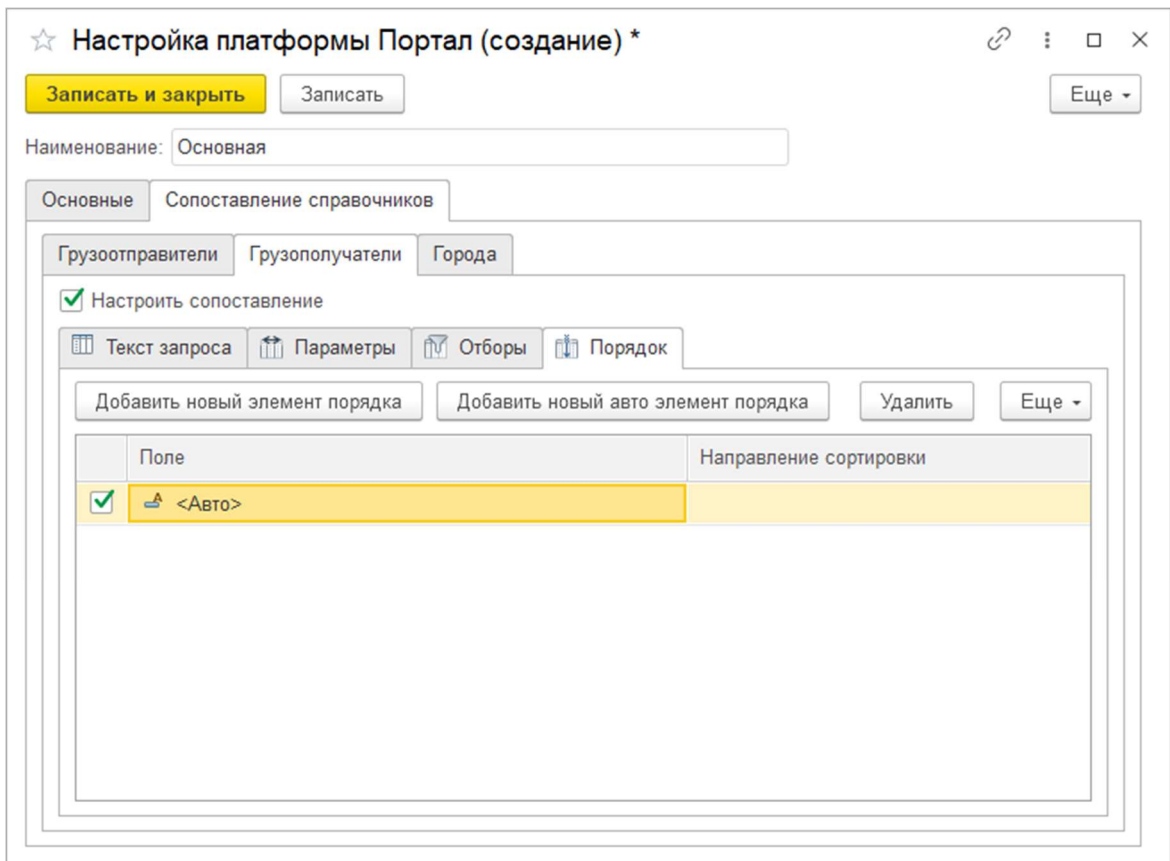


Рис. 18

После этого переходим в «Учетные записи платформы Портал» (Рис. 19).

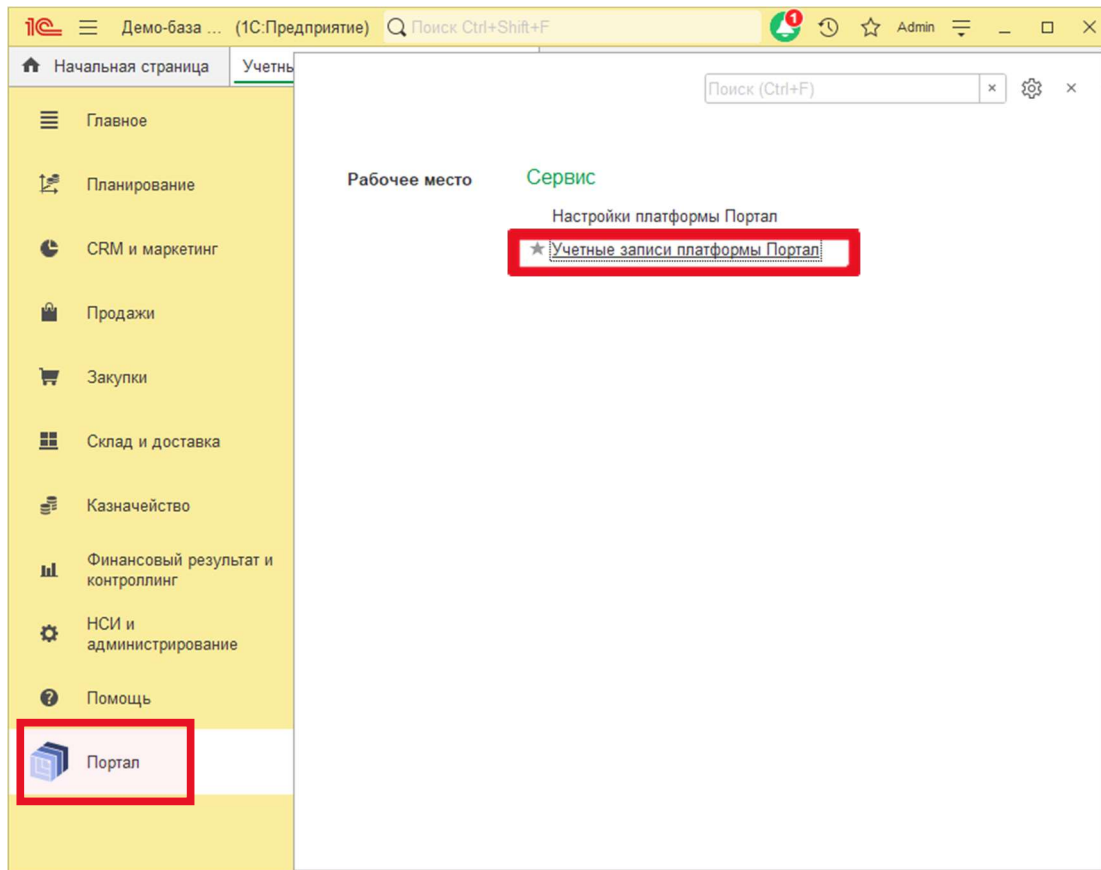


Рис. 19

Создаем учетную запись (Рис. 20).

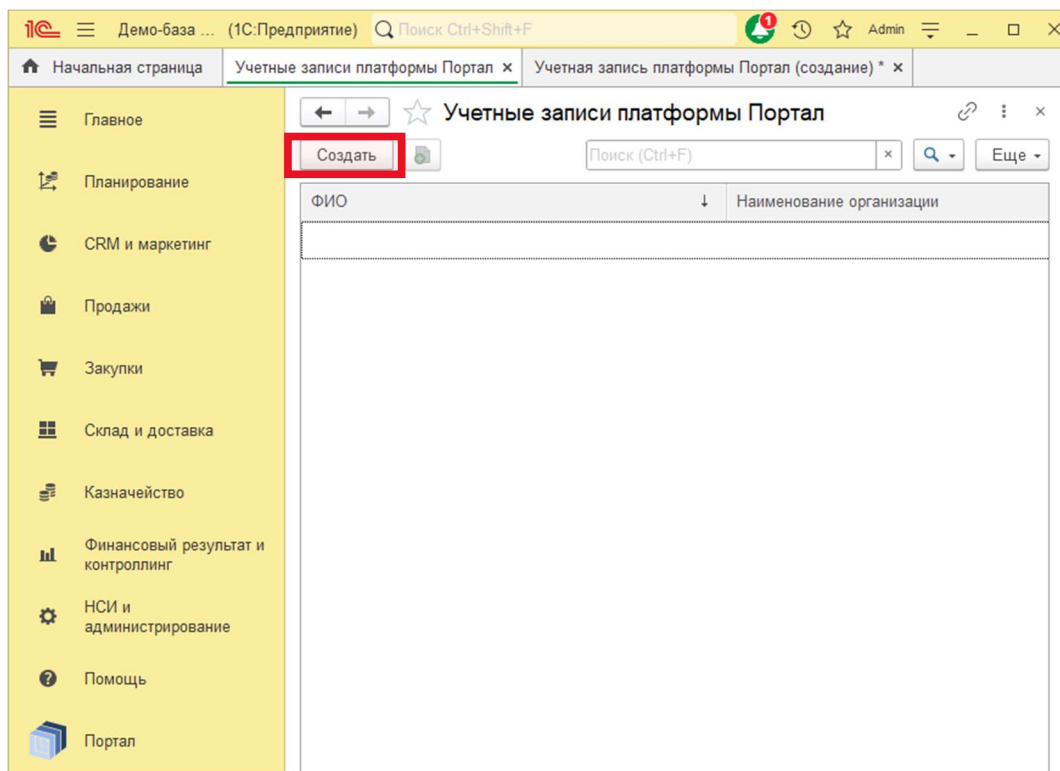


Рис. 20

Выбираем пользователя, заполняем ФИО, данные об организации и прописываем ключ к API (Рис. 21).

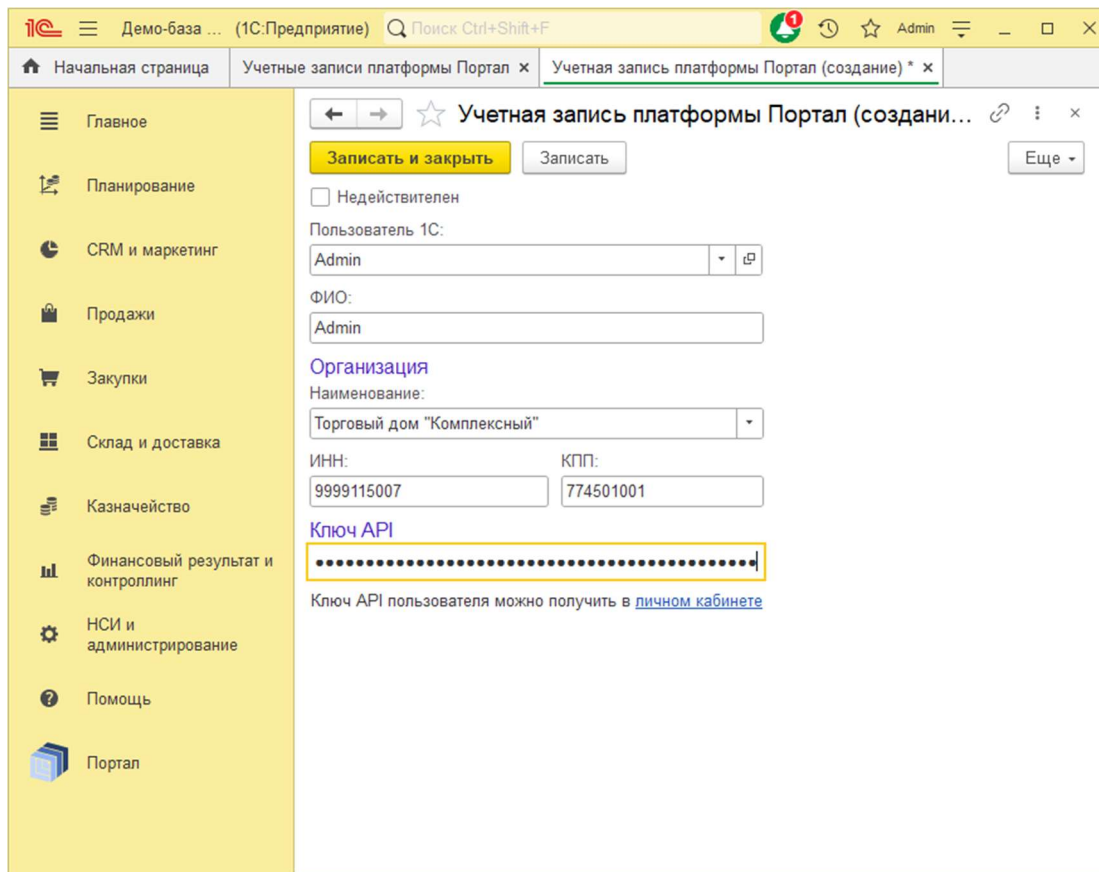


Рис. 21

Ключ API можно получить перейдя в личный кабинет «Портала», далее «Настройки» – «API»(Рис. 22). После «Сгенерировать» и «Копировать» (Рис. 23).

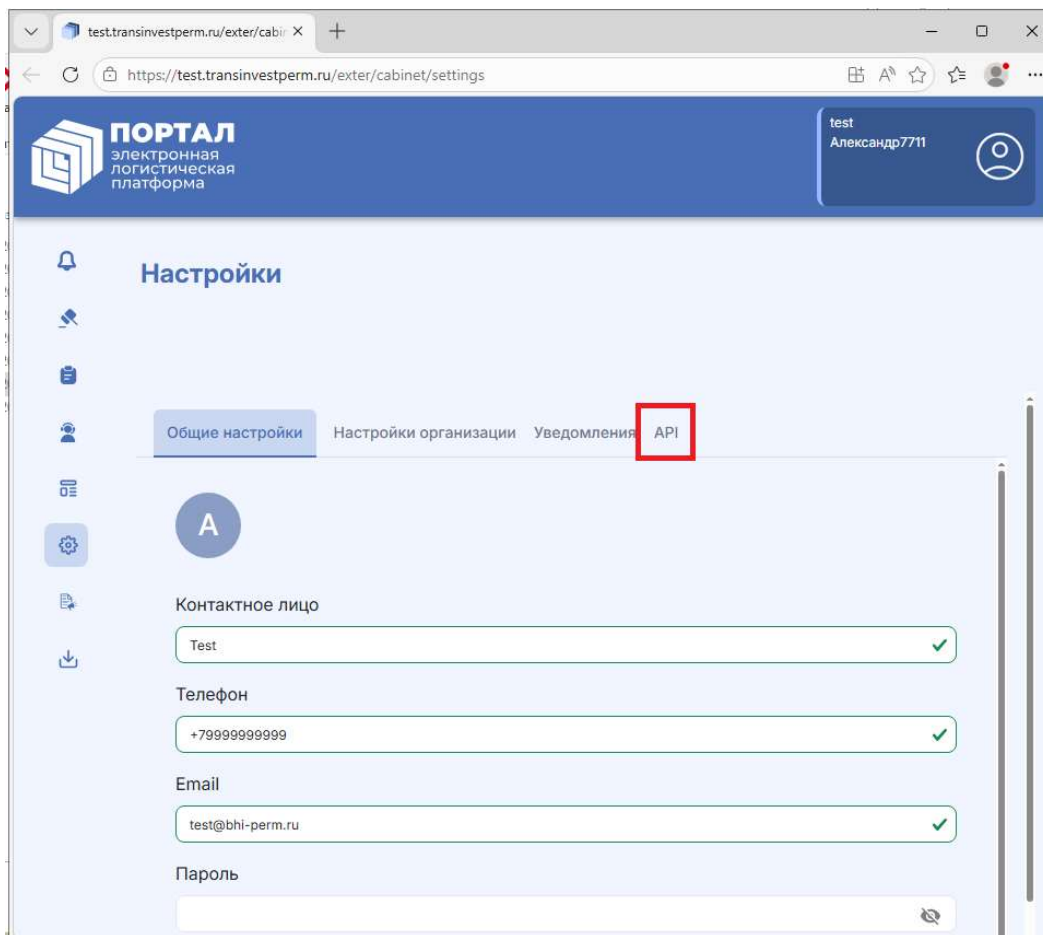


Рис. 22

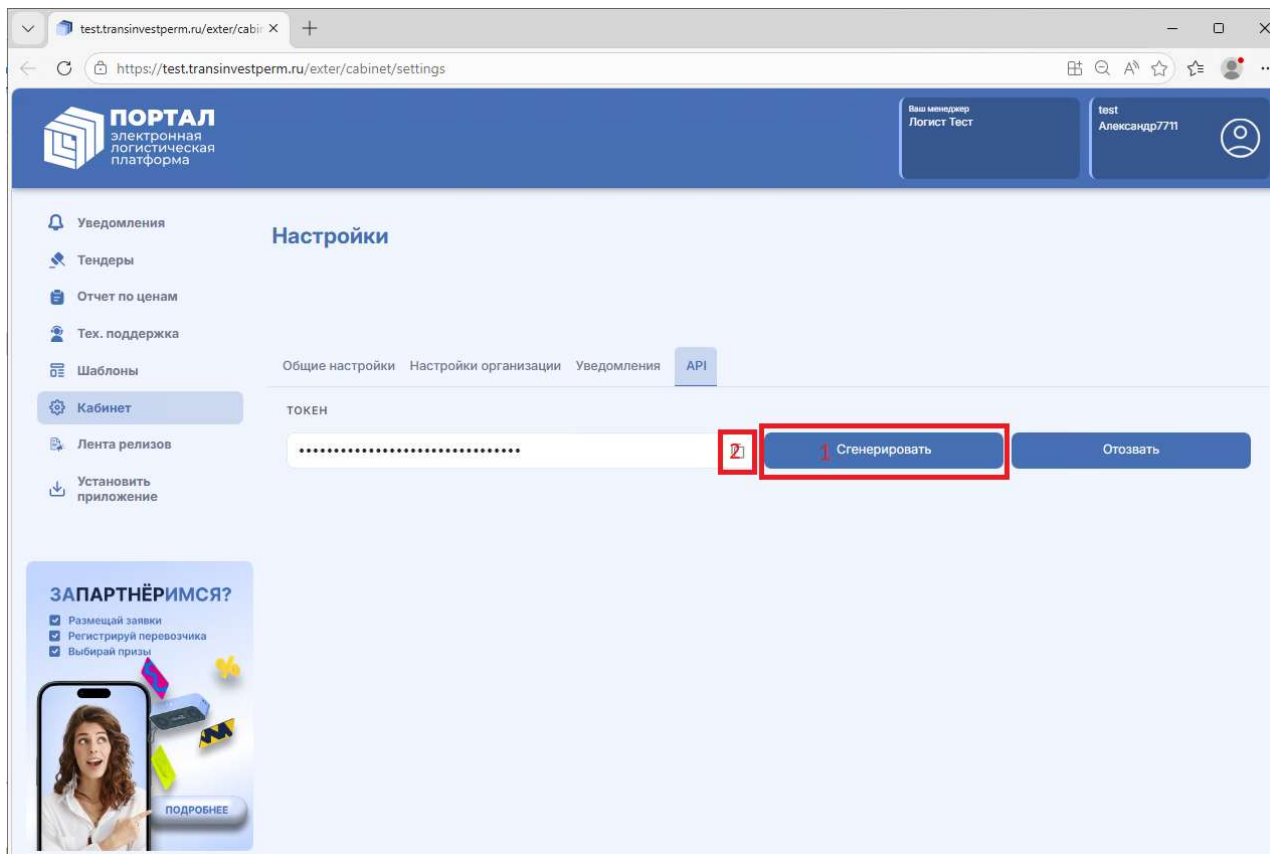


Рис. 23

В результате рабочее место настроено (Рис. 24). Нажимаем «Обновить», чтобы загрузить тендеры.

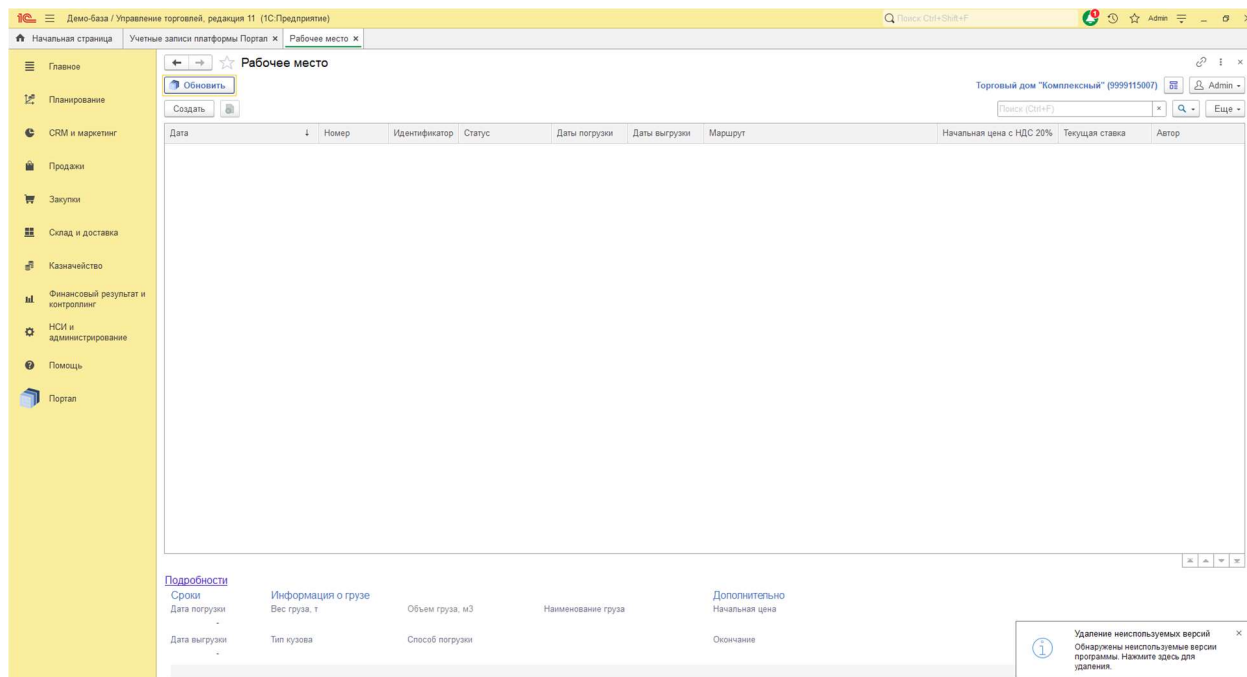


Рис. 24

Теперь есть возможность просматривать, создавать, редактировать, отменять, продлевать тендер из конфигурации, а также обновлять только статусы и ставки тендера (Рис. 25, Рис. 26, Рис. 27, Рис. 28).

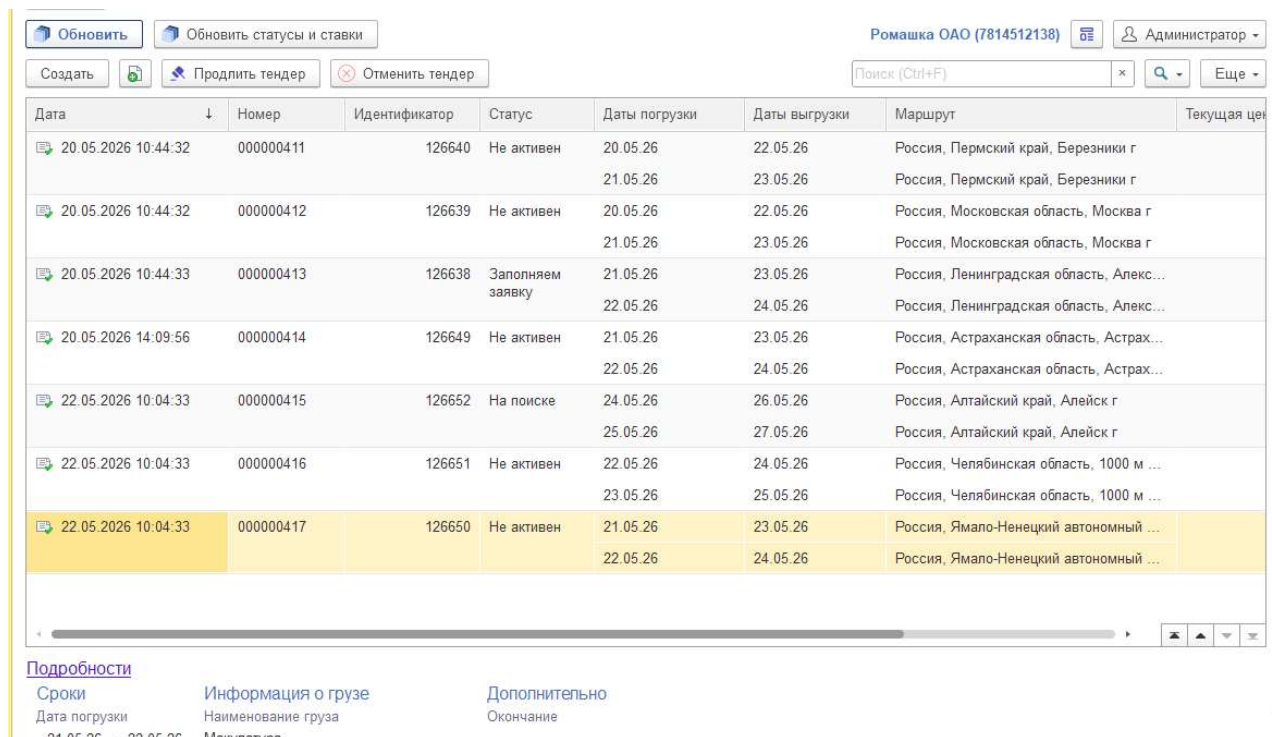


Рис. 25

☆ Тендер 000000409 от 20.05.2026 10:44:32

Основное История ставок История статусов

Записать и закрыть Записать

Номер: 000000409 Дата создания документа: 20.05.2026 10:44:32 Тип тендера: Простой Статус: Заявка

Погрузка Город: Россия, Приморский край, Артем г Дата с: 21.05.2026 по: 22.05.2026

Выгрузка Город: Россия, Приморский край, Артем г Дата с: 23.05.2026 по: 24.05.2026

Начальная цена с НДС 22%: 52 632,00 Текущая цена: 48 131,00

Информация о грузе
 Наименование груза: Макулатура Вес груза, т: 2,000 Объем груза, м3: 2,000 Стоимость груза: 50 000,00
 Тип кузова: ТЕНТ/РЕФ/ИЗОТЕРМ Способ погрузки/выгрузки: ЗАД/ЗАД Длительность тендера в днях: Только текущий день
 Комментарий: ком

Данные перевозчика
 Перевозчик: ООО ТРАНСИНВЕСТ
 Данные авто: Модель: марка: awdwdad21313 не Гос. номер: aa0101
 Прицеп: aa0101
 Водитель: ФИО: Иванов Иван Иванович
 Паспортные данные: Серия паспорта: 5722 Номер паспорта: 212341 Дата выдачи: 01.01.2025
 Кем выдан: Отделом
 Водительское удостоверение: Серия ВУ: 0122 Номер ВУ: 234522 Дата выдачи ВУ: 01.01.2025
 Телефон: +79999999997 Доп. телефон: +79999999991

Автор: Администратор ID тендера: 126643 Итоговая стоимость: 48 131,00

Рис. 26

← → ☆ Рабочее место

Обновить Обновить статусы и ставки

Создать Продлить тендер Отменить тендер

Ромашка ОАО (7814512138) Администратор

Поиск (Ctrl+F)

Дата	Номер	Идентифика...	Статус	Даты погрузки	Даты выгрузки	Маршрут	Текущая цен
20.05.2026 8:50:00	000000407	126648	Заполняем	21.05.26	22.05.26	Россия, Пермский край, Пермь г	
20.05.2026 14:09:56	000000414					Россия, Калужская область, Кондрово г	
22.05.2026 10:04:33	000000417					Россия, Астраханская область, Астрах...	
22.05.2026 10:04:33	000000416					Россия, Астраханская область, Астрах...	
22.05.2026 10:04:33	000000415					Россия, Ямало-Ненецкий автономный ...	
22.05.2026 10:04:33	000000415					Россия, Ямало-Ненецкий автономный ...	
22.05.2026 11:07:03	000000418	126653	Заполняем заявку	25.05.26	27.05.26	Россия, Челябинская область, 1000 м ...	
22.05.2026 11:07:03	000000418	126653	Заполняем заявку	29.05.26	03.06.26	Россия, Челябинская область, 1000 м ...	
22.05.2026 16:17:28	000000419	126654	Ставка	01.06.26	08.06.26	Россия, Алтайский край, Алейск г	
22.05.2026 16:17:28	000000419	126654	Ставка	23.05.26	27.05.26	Россия, Алтайский край, Алейск г	
22.05.2026 16:17:28	000000419	126654	Ставка	23.05.26	26.05.26	Россия, Астраханская область, Камен...	
22.05.2026 16:17:28	000000419	126654	Ставка	24.05.26	08.06.26	Россия, Ямало-Ненецкий автономный ...	
22.05.2026 16:17:28	000000419	126654	Ставка	24.05.26	26.05.26	Россия, Республика Дагестан, Алмало с	
22.05.2026 16:17:28	000000419	126654	Ставка	24.05.26	26.05.26	Россия, Республика Дагестан, Алмало с	

☆ Рабочее место

Начальная ставка: 6 000,00

Погрузка с: 23.05.2026 Погрузка по: 24.05.2026

Выгрузка с: 26.05.2026 Выгрузка по: 26.05.2026

Длительность тендера в днях: Только текущий день

Продлить тендер

Рис. 27

← → ☆ Рабочее место

Обновить Обновить статусы и ставки Ромашка ОАО (7814512138) Администратор

Создать Продлить тендер Отменить тендер Поиск (Ctrl+F) Еще

Дата	Номер	Идентифика... ↓	Статус	Даты погрузки	Даты выгрузки	Маршрут	Текущая цен
20.05.2026 8:50:00	000000407	126648	Заполняем заявку	21.05.26	22.05.26	Россия, Пермский край, Пермь г Россия, Калужская область, Кондрово г	
20.05.2026 14:09:56	000000414	126649	Не активен	21.05.26 22.05.26	23.05.26 24.05.26	Россия, Астраханская область, Астрах... Россия, Астраханская область, Астрах...	
22.05.2026 10:04:33	000000417	126650	Не активен	21.05.26 22.05.26	23.05.26 24.05.26	Россия, Ямало-Ненецкий автономный ... Россия, Ямало-Ненецкий автономный ...	
22.05.2026 10:04:33	000000416	126651	Не активен	22.05.26 23.05.26	24.05.26 25.05.26	Россия, Челябинская область, 1000 м ... Россия, Челябинская область, 1000 м ...	
22.05.2026 10:04:33	000000415	126652	На поиске	25.05.26 25.05.26	26.05.26 27.05.26	Россия, Алтайский край, Алейск г Россия, Алтайский край, Алейск г	
22.05.2026 11:07:03	000000418	126653	Заполняем заявку	29.05.26 01.06.26	03.06.26 08.06.26	Россия, Астраханская область, Камен... Россия, Ямало-Ненецкий автономный ...	
22.05.2026 16:17:28	000000419	126654	Ставка	23.05.26 24.05.26	26.05.26 26.05.26	Россия, Республика Дагестан, Алмало с Россия, Республика Дагестан, Алмало с	

Подробности

Рис. 28

3. МЕХАНИЗМ РАБОТЫ ИНТЕГРАЦИИ

3.1. Общая схема взаимодействия

Интеграция работает по принципу двустороннего обмена. Из 1С – на площадку (создание, редактирование, удаление, продление тендеров). С площадки – в 1С (получение списка тендеров, актуализация данных, загрузка справочников).

Все операции выполняются через модуль `этп_АРІ`, а бизнес-логика сосредоточена в модуле `этп_ИнтеграцияСервер`.

3.2. Авторизация и выбор пользователя

При открытии «Рабочего места» система считывает пользователя ЭТП (справочник `этп_ПользователиЭТП`). Пользователь сохраняется в `ПараметрыСеанса.этп_ТекущийПользователь`. Сразу выполняется проверка токена через метод `ПроверкаТокена (GET /shipperapi/v1/auth/me)`. Если токен недействителен – пользователь видит сообщение об ошибке.

3.3. Загрузка справочников

Справочники платформы («Способы погрузки», «Наименование грузов», «Типы кузова») автоматически загружаются в базу 1С при записи элемента справочника «Настройки платформы Портал» (`этп_Настройки`). При сохранении настроек автоматически вызывается процедура `ЗагрузитьСправочникиЭТП()` модуля `этп_ИнтеграцияСервер`.

Внутри этой процедуры:

- получается токен текущего пользователя ЭТП;
- выполняются запросы к АРІ (`ПолучитьЗначенияСправочника`) для типов 1, 2 и 3.

Полученные данные записываются или обновляются в соответствующих справочниках 1С (этп_СпособПогрузки, этп_НаименованиеГруза, этп_ТипКузова) по коду (поле Код = id с платформы).

3.4. Создание и редактирование тендера из 1С

При записи документа этп_Тендер:

- В ПередЗаписьюНаСервере формируется структура данных (ПолучитьСтруктуруНовогоТендера; или ПолучитьСтруктуруРедактированияТендера);
- в ПриЗаписиНаСервере вызывается либо СоздатьТендер (POST), либо РедактироватьТендер (PATCH);
- при успешном ответе от сервера в документ записывается id тендера и сохраняется связь в регистре этп_ИдентификаторыОбъектов.

3.5. Синхронизация тендеров с площадки (кнопка «Обновить»)

В обработке «Рабочее место» процедура ОбновитьНаСервере выполняет следующее:

- получает список тендеров постранично (ПолучитьСписокТендеров);
- для каждого тендера запрашивает полные данные (ПолучитьДанныеТендера);
- вызывает СоздатьИлиОбновитьТендерПоПолнымДанным;
- внутри этой процедуры ищет документ по id и либо обновляет существующий, либо создаёт новый.

3.6. Удаление тендера

При удалении документа этп_Тендер в процедуре ПередУдалением автоматически выполняется УдалитьТендер (DELETE /shipperapi/v1/tenders/{id}).

3.7. Продление тендера

Из формы списка вызывается специальная форма продления. Данные отправляются методом ПродлитьТендер (POST /shipperapi/v1/tenders/{id}/extension).

3.8. Автоподбор городов

На форме документа при вводе в поля «Город погрузки» / «Город выгрузки» срабатывает автоподбор через ПолучитьПунктыМаршрута (GET /shipperapi/v1/route-points).

4. МЕТОДЫ API

4.1. Проверка токена и получение профиля пользователя

Назначение: Проверка действительности ключа API и получение информации о текущем пользователе.

Тип запроса: GET

Эндпоинт: /shipperapi/v1/auth/me

Параметры: отсутствуют

Пример ответа JSON:

```
{
  "id": 123,
  "login": "shipper",
  "contact_name": "Иванов Иван Иванович",
  "email": "user@example.com",
  "phone": "+70000000000",
  "organization": {
    "id": 5,
    "name": "ООО Пример",
    "inn": "5900000000",
    "kpp": "590001001"
  }
}
```

4.1.1. Использование метода в расширении

Пример запроса:

```
Запрос = Новый HTTPЗапрос("/shipperapi/v1/auth/me");
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);

Результат = Соединение.ВызватьHTTPМетод("GET", Запрос);
РезультатСтрокой = Результат.ПолучитьТелоКакСтроку();
```

4.2. Получение справочников

Назначение: Получение значений справочников платформы (способы погрузки, наименования грузов, типы кузова).

Тип запроса: GET

Эндпоинт:

/shipperapi/v1/loading-methods (тип 1)

/shipperapi/v1/cargo-types (тип 2)

/shipperapi/v1/body-types (тип 3)

Параметры: отсутствуют

Пример ответа JSON:

```
{
  "data": [
    {
      "id": 6,
      "value": "Задняя"
    },
    {
      "id": 7,
      "value": "Боковая"
    }
  ]
}
```

4.2.1. Использование метода в расширении

Пример запроса:

```
Если ТипСправочника = "1" Тогда
  Ресурс = "/shipperapi/v1/loading-methods";
ИначеЕсли ТипСправочника = "2" Тогда
  Ресурс = "/shipperapi/v1/cargo-types";
ИначеЕсли ТипСправочника = "3" Тогда
  Ресурс = "/shipperapi/v1/body-types";
КонецЕсли;

Запрос = Новый HTTPЗапрос(Ресурс);
```

```
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);
```

```
Результат = Соединение.ВызватьHTTPМетод("GET", Запрос);
```

```
РезультатСтрокой = Результат.ПолучитьТелоКакСтроку();
```

4.3. Поиск пунктов маршрута (городов)

Назначение: Поиск городов/пунктов маршрута для автоподбора.

Тип запроса: GET

Эндпоинт: /shipperapi/v1/route-points

Таблица 1 – Параметры запроса.

Поле	Тип	Описание
search	string	Строка поиска по названию города
page	integer	Номер страницы
per_page	integer	Количество записей на странице

Пример ответа JSON:

```
{
  "data": [
    {
      "id": 1,
      "name": "Пермь",
      "region_name": "Пермский край",
      "country_name": "Россия"
    }
  ],
  "links": {
    "self": "/shipperapi/v1/route-points?page=1&per_page=20"
  },
  "meta": {
    "total": 1000,
    "count": 20,
    "per_page": 20,
    "has_more": true
  }
}
```

4.3.1. Использование метода в расширении

Пример запроса:

```
Ресурс = "/shipperapi/v1/route-points?search=" +  
КодироватьСтроку(ТекстПоиска,  
СпособКодированияСтроки.URLBКодировкеURL);  
Ресурс = Ресурс + "&page=" + Формат(Page, "ЧГ=0") + "&per_page=" +  
Формат(PerPage, "ЧГ=0");  
  
Запрос = Новый HTTPЗапрос(Ресурс);  
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);  
  
Результат = Соединение.ВызватьHTTPМетод("GET", Запрос);  
РезультатСтрокой = Результат.ПолучитьТелоКакСтроку();
```

4.4. Получение списка тендеров

Назначение: Получение списка идентификаторов тендеров, доступных грузоотправителю.

Тип запроса: GET

Эндпоинт: /shipperapi/v1/tenders

Таблица 2 – Параметры запроса.

Поле	Тип	Описание
page	integer	Номер страницы
per_page	integer	Количество записей на странице

Пример ответа JSON:

```
{  
  "data": [  
    {  
      "id": 1,  
      "reference_number": "1042",  
      "cargo_description": "Товарная позиция",  
      "status": {  
        "code": "search_ts",
```

```

    "label": "На поиске"
  },
  "bidding_end": "2024-01-15T18:00:00+03:00",
  "opening_price": 150000,
  "origin_point": {
    "id": 1,
    "name": "Пермь",
    "region_name": "Пермский край",
    "country_name": "Россия"
  },
  "destination_point": {
    "id": 2,
    "name": "Москва",
    "region_name": "Московская область",
    "country_name": "Россия"
  },
  "bids_count": 5
}
],
"links": {
  "self": "/shipperapi/v1/tenders?page=1&per_page=20",
  "first": "/shipperapi/v1/tenders?page=1&per_page=20",
  "prev": null,
  "next": "/shipperapi/v1/tenders?page=2&per_page=20"
},
"meta": {
  "total": 100,
  "count": 20,
  "per_page": 20,
  "has_more": true
}
}

```

4.4.1. Использование метода в расширении

Пример запроса:

```

Ресурс = "/shipperapi/v1/tenders?page=" + Формат(Page, "ЧГ=0") +
"&per_page=" + Формат(PerPage, "ЧГ=0");

```

```

Запрос = Новый HTTPЗапрос(Ресурс);
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);

```

```

Результат = Соединение.ВызватьHTTPМетод("GET", Запрос);

```

```
РезультатСтрокой = Результат.ПолучитьТелоКакСтроку();
```

4.5. Получение полных данных тендера

Назначение: Получение полной информации по одному тендеру.

Тип запроса: GET

Эндпоинт: /shipperapi/v1/tenders/{id}

Пример ответа JSON:

```
{
  "id": 1,
  "reference_number": "1042",
  "type": {
    "code": "simple",
    "label": "Простой"
  },
  "status": {
    "code": "search_ts",
    "label": "На поиске"
  },
  "cargo_description": "Товарная позиция",
  "cargo_value": 1000000,
  "weight": 20.5,
  "volume": 50.5,
  "origin_point": {
    "id": 1,
    "name": "Пермь",
    "region_name": "Пермский край",
    "country_name": "Россия"
  },
  "destination_point": {
    "id": 2,
    "name": "Москва",
    "region_name": "Московская область",
    "country_name": "Россия"
  },
  "loading_date_start": "2026-05-22",
  "loading_date_end": "2026-05-23",
  "unloading_date_start": "2026-05-25",
  "unloading_date_end": "2026-05-26",
}
```

```
"duration_days": 3,
"body_type": {
  "id": 16,
  "value": "Тент"
},
"loading_method": {
  "id": 6,
  "value": "Задняя"
},
"cargo_type": {
  "id": 2,
  "value": "Картон плоских слоев"
},
"bidding_start": "2026-05-20T10:00:00+03:00",
"bidding_end": "2026-05-20T18:00:00+03:00",
"opening_price": 150000
}
```

4.5.1. Использование метода в расширении

Пример запроса:

```
Запрос = Новый HTTPЗапрос("/shipperapi/v1/tenders/" +
Формат(ИдентификаторТендера, "ЧН=0; ЧГ=0"));
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);

Результат = Соединение.ВызватьHTTPМетод("GET", Запрос);
РезультатСтрокой = Результат.ПолучитьТелоКакСтроку();
```

4.6. Создание нового тендера

Назначение: Создание нового тендера на платформе.

Тип запроса: POST

Эндпоинт: /shipperapi/v1/tenders

Пример запроса JSON

```
{
  "type": "simple",
  "weight": 20.5,
```

```
"volume": 50.5,  
"bidding_start": "2026-05-20T10:00:00+03:00",  
"bidding_end": "2026-05-20T18:00:00+03:00",  
"origin_point_id": 1,  
"destination_point_id": 2,  
"loading_date_start": "2026-05-22",  
"loading_date_end": "2026-05-23",  
"unloading_date_start": "2026-05-25",  
"unloading_date_end": "2026-05-26",  
"opening_price": 150000,  
"cargo_value": 1000000,  
"duration_days": 3,  
"loading_method_id": 6,  
"cargo_description": "Товарная позиция",  
"body_type_id": 16,  
"cargo_type_id": 2  
}
```

Пример ответа JSON:

```
{  
  "id": 1,  
  "reference_number": "1042",  
  "type": {  
    "code": "simple",  
    "label": "Простой"  
  },  
  "status": {  
    "code": "search_ts",  
    "label": "На поиске"  
  },  
  "cargo_description": "Товарная позиция",  
  "origin_point": {  
    "id": 1,  
    "name": "Пермь"  
  },  
  "destination_point": {  
    "id": 2,  
    "name": "Москва"  
  },  
  "loading_date_start": "2026-05-22",  
  "opening_price": 150000  
}
```

4.6.1. Использование метода в расширении

Пример запроса:

```
Запрос = Новый HTTPЗапрос("/shipperapi/v1/tenders");
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);
Запрос.Заголовки.Вставить("Content-type", "application/json");

СтрокаJS = ОбщегоНазначения.ЗначениеBJSON(ДанныеТендера);
Запрос.УстановитьТелоИзСтроки(СтрокаJS);

Результат = Соединение.ВызватьHTTPМетод("POST", Запрос);
РезультатСтрокой = Результат.ПолучитьТелоКакСтроку();
```

Структура данных нового тендера:

```
Строка =
"{
|   ""type"": "",
|   ""weight"": 0,
|   ""volume"": 0,
|   ""bidding_start"": "",
|   ""bidding_end"": "",
|   ""origin_point_id"": 0,
|   ""destination_point_id"": 0,
|   ""loading_date_start"": "",
|   ""loading_date_end"": "",
|   ""unloading_date_start"": "",
|   ""unloading_date_end"": "",
|   ""opening_price"": 0,
|   ""cargo_value"": 0,
|   ""duration_days"": 0,
|   ""loading_method_id"": 0,
|   ""cargo_description"": "",
|   ""body_type_id"": 0,
|   ""cargo_type_id"": 0
|}";
```

4.7. Редактирование тендера

Назначение: Частичное обновление существующего тендера.

Тип запроса: PATCH

Эндпоинт: /shipperapi/v1/tenders/{id}

Пример запроса JSON:

```
{
  "type": "simple",
  "weight": 20.5,
  "volume": 50.5,
  "bidding_start": "2026-05-20T10:00:00+03:00",
  "bidding_end": "2026-05-20T18:00:00+03:00",
  "origin_point_id": 1,
  "destination_point_id": 2,
  "loading_date_start": "2026-05-22",
  "loading_date_end": "2026-05-23",
  "unloading_date_start": "2026-05-25",
  "unloading_date_end": "2026-05-26",
  "opening_price": 150000,
  "cargo_value": 1000000,
  "duration_days": 3,
  "loading_method_id": 6,
  "cargo_description": "Товарная позиция",
  "body_type_id": 16,
  "cargo_type_id": 2
}
```

Пример ответа JSON:

```
{
  "id": 1,
  "reference_number": "1042",
  "type": {
    "code": "simple",
    "label": "Простой"
  },
  "status": {
    "code": "search_ts",
    "label": "На поиске"
  },
  "cargo_description": "Товарная позиция",
  "origin_point": {
    "id": 1,
    "name": "Пермь"
  }
}
```

```
},  
"destination_point": {  
  "id": 2,  
  "name": "Москва"  
},  
"loading_date_start": "2026-05-22",  
"opening_price": 150000  
}
```

4.7.1. Использование метода в расширении

Пример запроса:

```
ИД = ДанныеТендера.tender_id;  
Запрос = Новый HTTPЗапрос("/shipperapi/v1/tenders/" + Формат(ИД, "ЧН=0;  
ЧГ=0"));  
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);  
Запрос.Заголовки.Вставить("Content-type", "application/json");  
  
СтруктураДляPATCH = Новый Структура;  
Для Каждого КЗ Из ДанныеТендера Цикл  
  Если КЗ.Ключ <> "tender_id" Тогда  
    СтруктураДляPATCH.Вставить(КЗ.Ключ, КЗ.Значение);  
  КонецЕсли;  
КонецЦикла;  
  
СтрокаJS = ОбщегоНазначения.ЗначениеJSON(СтруктураДляPATCH);  
Запрос.УстановитьТелоИзСтроки(СтрокаJS);  
  
Результат = Соединение.ВызватьHTTPМетод("PATCH", Запрос);  
РезультатСтрокой = Результат.ПолучитьТелоКакСтроку();
```

Структура данных для редактирования тендера:

```
Строка =  
{  
  ""tender_id"": 0,  
  ""type"": "",  
  ""weight"": 0,  
  ""volume"": 0,  
  ""bidding_start"": "",  
  ""bidding_end"": "",  
  ""origin_point_id"": 0,
```

```
|"destination_point_id": 0,  
|"loading_date_start": "",  
|"loading_date_end": "",  
|"unloading_date_start": "",  
|"unloading_date_end": "",  
|"opening_price": 0,  
|"cargo_value": 0,  
|"duration_days": 0,  
|"loading_method_id": 0,  
|"cargo_description": "",  
|"body_type_id": 0,  
|"cargo_type_id": 0  
|}";
```

4.8. Удаление тендера

Назначение: Полное удаление тендера.

Тип запроса: DELETE

Эндпоинт: /shipperapi/v1/tenders/{id}

Ответ: Код 204 при успешном удалении

4.8.1. Использование метода в расширении

Пример запроса:

```
Ресурс = "/shipperapi/v1/tenders/" + Формат(ИдентификаторТендера,  
"ЧН=0; ЧГ=0");  
Запрос = Новый HTTPЗапрос(Ресурс);  
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);  
  
Результат = Соединение.ВызватьHTTPМетод("DELETE", Запрос);
```

4.9. Продление тендера

Назначение: Продление срока действия тендера.

Тип запроса: POST

Эндпоинт: /shipperapi/v1/tenders/{id}/extension

Пример запроса JSON:

```
{
  "duration_days": 5,
  "loading_date_start": "2026-05-22",
  "loading_date_end": "2026-05-23",
  "unloading_date_start": "2026-05-25",
  "unloading_date_end": "2026-05-26",
  "opening_price": 150000
}
```

Пример ответа JSON:

```
{
  "id": 1,
  "reference_number": "1042",
  "type": {
    "code": "simple",
    "label": "Простой"
  },
  "status": {
    "code": "search_ts",
    "label": "На поиске"
  },
  "cargo_description": "Товарная позиция",
  "origin_point": {
    "id": 1,
    "name": "Пермь"
  },
  "destination_point": {
    "id": 2,
    "name": "Москва"
  },
  "loading_date_start": "2026-05-22",
  "opening_price": 150000
}
```

4.9.1. Использование метода в расширении

Пример запроса:

```
Ресурс = "/shipperapi/v1/tenders/" + Формат(ИдентификаторТендера,
"ЧН=0; ЧГ=0") + "/extension";
Запрос = Новый HTTPЗапрос(Ресурс);
Запрос.Заголовки.Вставить("Authorization", "Bearer " + КлючAPI);
Запрос.Заголовки.Вставить("Content-type", "application/json");

СтрокаJS = ОбщегоНазначения.ЗначениеBJSON(ДанныеПродления);
Запрос.УстановитьТелоИзСтроки(СтрокаJS);

Результат = Соединение.ВызватьHTTPМетод("POST", Запрос);
РезультатСтрокой = Результат.ПолучитьТелоКакСтроку();
```

Структура данных для продления тендера:

```
Строка =
"{
|   ""duration_days"": 0,
|   ""loading_date_start"": """",
|   ""loading_date_end"": """",
|   ""unloading_date_start"": """",
|   ""unloading_date_end"": """",
|   ""opening_price"": 0
}";
```

5. ОБРАБОТКА ОШИБОК И КОДЫ ОТВЕТА

5.1. Основные HTTP-коды ответов

Таблица 3 – HTTP-коды

Код	Название	Когда возникает
200	OK	Успешное выполнение GET / PATCH
201	Created	Тендер успешно создан
204	No Content	Успешное удаление или продление
401	Unauthorized	Неверный или отсутствующий токен авторизации, истекший срок действия токена
403	Forbidden	У пользователя нет прав для выполнения запрошенного действия
404	Not Found	Тендер с указанным ID не существует, некорректный эндпоинт
422	Unprocessable Entity	Запрос синтаксически корректен, но содержит семантические ошибки. Ошибка валидации данных.
500	Internal Server Error	Непредвиденная ошибка на стороне сервера

5.2. Обработка ошибок в 1С

Все ошибки обрабатываются функцией `ПолучитьТекстОшибкиAPI`.

Пример ответа сервера с ошибкой валидации:

```
{
  "type": "https://tools.ietf.org/html/rfc2616#section-10",
  "title": "Invalid input",
  "status": 422,
  "detail": "The given data was invalid.",
  "errors": {
    "weight": ["The weight field must be greater than 0."],
```

```
"origin_point_id": ["The selected origin point id is invalid."]  
}  
}
```