



- [Основные положения](#)
- [SDK](#)
 - [Назначение и основные функции](#)
 - [Состав SDK](#)
 - [Использование SDK](#)
 - [Пример расширения функционала](#)
- [API](#)
 - [Общие принципы работы с API](#)
 - [Безопасность при работе с API](#)
 - [Авторизация](#)
 - [5 больших этапов](#)
 - [Идентификация клиента](#)
 - [Выполнение операции](#)
 - **Клиенты**
 - [Поиск клиента](#)
 - [Поиск операций созданных клиентом](#)
 - [Создание клиента](#)
 - [Изменение клиента](#)
 - **Операции**
 - [Создание операции](#)
 - [Изменение операции](#)
 - [Получение операции](#)
 - [Получение операции по идентификатору во внешней системе](#)
 - [Подтверждение операции](#)
 - **Типовые операции**
 - [Отправка перевода](#)
 - [Изменить перевод](#)
 - [Выдача перевода](#)
 - [Аннулирование перевода](#)
 - **Платежи**
 - [Пополнение карты Юнистрим](#)
 - [Пополнение в Тинькофф Банк](#)
 - [Пополнение карт VISA](#)
 - [Пополнение карт UZCARD](#)
 - [Пополнение карт Корти Милли](#)
 - [Пополнение банковской карты VISA/MasterCard/Мир/UZcard/КортиМилли](#)
 - [Оплата патента](#)
- [InfoApi](#)
 - **Справочник стран**
 - [Список стран](#)
 - [Страна по ISO-коду](#)
 - [Список регионов по операции](#)
 - [Регион по идентификатору](#)
 - [Метро в регионе](#)
 - **Справочник точек**
 - [Список точек](#)
 - [Поиск точек](#)
 - [Точка по идентификатору](#)
 - **Справочник контрагентов**
 - [Список контрагентов](#)
 - [Контрагент по идентификатору](#)
 - **Переводы**
 - [Информация о переводе](#)
 - [Расчет комиссии](#)
- [FAQ](#)
 - [Общие вопросы](#)
 - [Безопасность, авторизация и подтверждение операции](#)
 - [Клиенты](#)
 - [Операции](#)
- [Ошибки](#)

Основные положения

Чтобы получить **личные** учётные данные, необходимо заполнить заявку, заполняется 5 минут. Для получения моментального доступа, можете воспользоваться учётными данными из SDK - учётные данные предназначены исключительно для ознакомления с API и не могут использоваться в комплексном тестировании.

Заявку на подключение и акт тестирования можно скачать ниже.

Заявка на подключение к платёжной системе денежных переводов Юнистрим

Request connection to Unistream PS

Акт тестирования

- 1. В заявке необходимо указать тип подключения, если Вы используете API с данного сайта необходимо выбрать тип подключения GLOBAL 3.
- 2. Выбрать стенд подключения, тестовый/боевой. Если нужен доступ на боевой и на тестовый, заполняется две заявки.
- 3. Указать наименование организации.
- 4. Указать ответственных лиц.
- 5. Указать IP адреса с которых будут подключения до API.
- 6. Выбрать тип сертификата, пользовательский или корневой в случае собственного ЦС, указать на чье имя сертификаты.
- 7. Выбрать типы операций из предложенного списка.
- 8. Указать дату заполнения, подпись ответственного и печать.

SDK

Назначение и основные функции

SDK инкапсулирует транспорт обмена данными между Партнером и Юнистрим по протоколу G3. Обеспечивает процедуру авторизации и цифровой подписи пересылаемых сообщений. Имеет открытую архитектуру и имеет возможность расширения функционала. Имеет реализации на .NET Standard 2.0 и Java 10, что обеспечивает кросс-платформенность. Каждый публичный метод SDK реализует тот или иной функционал REST API. Некоторые, методы, пример, использующие цифровую подпись, могут использовать несколько вызовов REST API.

Состав SDK

- CsharpReleaseExample.zip - скомпилированный, рабочий пример под .NET 4.6, имеющий графический интерфейс и рабочие настройки (сертификат выдаётся на основании заявки на подключение).
- UnistreamApiSample.zip – исходный код, приведенного выше примера.
- Unistream-api-1.0.3.zip - библиотека на Java 10

Использование SDK

Подключить в проект следующие NuGet пакеты:

```
<package id="Newtonsoft.Json" version="12.0.1" targetFramework="net461" />
<package id="System.Security.Cryptography.Pkcs" version="4.5.2" targetFramework="net461" />
<package id="UnistreamApi" version="1.0.3" targetFramework="net461" />
```

Использование методов SDK:

```
using (var api = new UnistreamApiClient(config, m_Logger)) { }
```

Полный состав методов SDK приведен ниже (их названия и назначения соответствуют основному описанию API)

```
public interface IUnistreamApiClient : IDisposable
{
    /// <summary>
    /// Create new client
    /// POST https://{hostname}/v2/clients/
    /// </summary>
    /// <returns>only HttpStatusCode (202 - OK)</returns>
    Task<Response<EmptyContent>> CreateClientAsync(ClientDto client, CancellationToken cancel);

    /// <summary>
    /// Get client by client id
    /// GET https://{hostname}/v2/clients/{clientId}
    /// </summary>
    Task<Response<ClientDto>> GetClientAsync(string clientId, CancellationToken cancel);

    /// <summary>
    /// Find client by document number
    /// GET https://{hostname}/v2/clients/search?documentNumber={documentNumber}&page={page}&rows={rows}
    /// </summary>
    /// <param name="documentNumber">Document number, i.e. 4321123456</param>
    /// <param name="page">Page number (optional)</param>
    /// <param name="rows">Line per page (optional)</param>
    /// <returns>Array of founded clients</returns>
    Task<Response<ClientsDto>> FindClientByDocNumberAsync(string documentNumber, int? page, int? rows, CancellationToken cancel);

    /// <summary>
    /// Find client by loyalty card number
    /// GET https://{hostname}/v2/clients/searchByLoyaltyCard?cardNumber={cardNumber}
    /// </summary>
    /// <param name="cardNumber">Loyalty card number</param>
    /// <returns>The founded client</returns>
    Task<Response<ClientDto>> FindClientByCardAsync(string cardNumber, CancellationToken cancel);

    /// <summary>
    /// Udate client info
    /// PATCH https://{hostname}/v2/clients/{clientId}
    /// </summary>
    /// <returns>only HttpStatusCode (200 - OK)</returns>
    Task<Response<EmptyContent>> UpdateClientAsync(string clientId, ClientDto client, CancellationToken cancel);

    /// <summary>
    /// Add new document to client
    /// POST https://{hostname}/v2/clients/{clientId}/documents
    /// </summary>
    /// <returns>only HttpStatusCode (200 - OK)</returns>
    Task<Response<EmptyContent>> AddDocumentAsync(string clientId, DocumentDto doc, CancellationToken cancel);

    /// <summary>
    /// Update clients document
    /// PATCH https://{hostname}/v2/clients/{clientId}/documents
```

```
/// </summary>
/// <returns>only HttpStatusCode (200 - OK)</returns>
Task<Response<EmptyContent>> UpdateDocumentAsync(string clientId, DocumentDto doc, CancellationToken cancel);

/// <summary>
/// Create new operation
/// </summary>
/// <param name="operInfo">Operation info</param>
/// <param name="operationType">Operation type</param>
/// <param name="operationId">Unique operation guid</param>
/// <returns>Operation guid and errors</returns>
Task<Response<OperationIdDto>> CreateOperationAsync(OperationInfoDto operInfo, string operationType, string operationId,
CancellationToken cancel);

/// <summary>
/// Create new command for operation
/// POST https://{hostname}/v2/operations/{operationType}/{id}
/// </summary>
/// <param name="operInfo">Operation info</param>
/// <param name="commandId">Unique command guid</param>
/// <returns>Command guid and errors</returns>
Task<Response<CommandIdDto>> UpdateOperationAsync(OperationInfoDto operInfo, string commandId, CancellationToken cancel);

/// <summary>
/// Get operations full info
/// GET https://{hostname}/v2/operations/{id}
/// </summary>
/// <param name="operationId">Operation guid</param>
Task<Response<OperationDto>> GetOperationAsync(string operationId, CancellationToken cancel);

/// <summary>
/// Revoke operation
/// GET https://{hostname}/v2/operations/{operationId}/revokeOperation
/// </summary>
/// <param name="operationId">Operation guid</param>
Task<Response<MessageDto>> RevokeOperationAsync(string operationId, CancellationToken cancel);

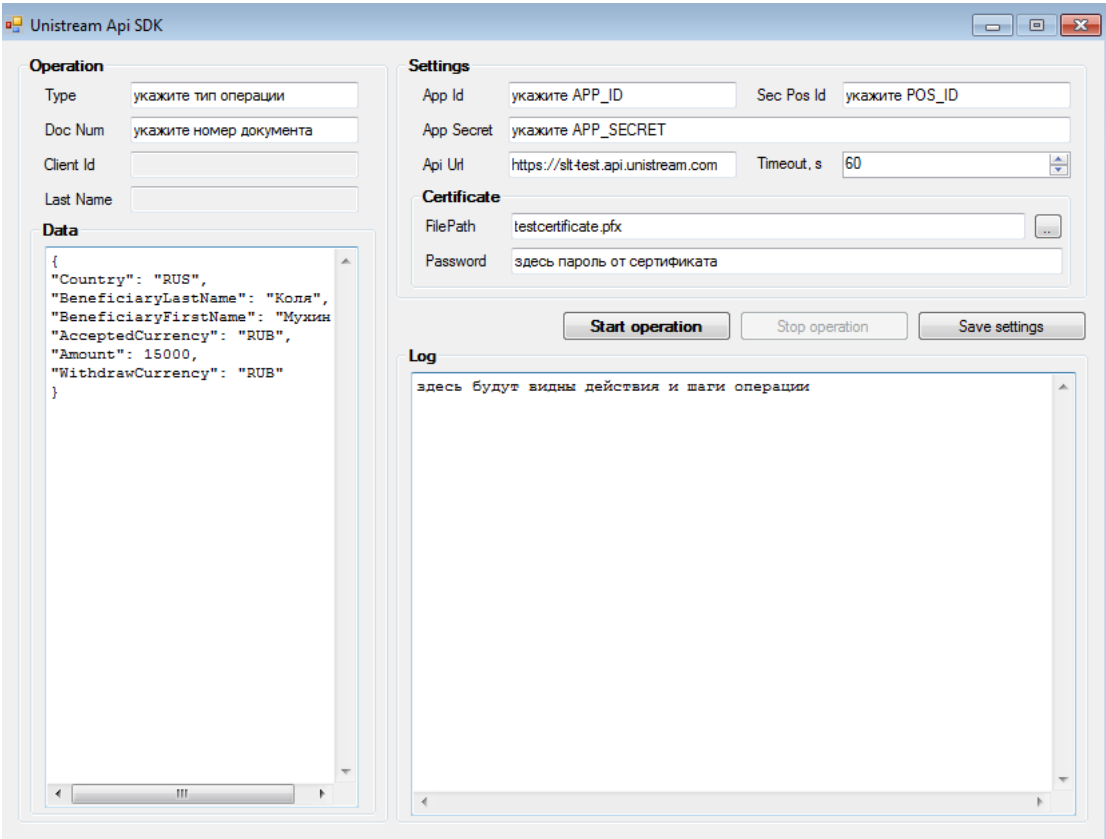
/// <summary>
/// Get operations info for display
/// GET https://{hostname}/v2/operations/{operationId}/details
/// </summary>
/// <param name="operationId">Operation guid</param>
Task<Response<OperationDetailsDto>> GetOperationDetailsAsync(string operationId, CancellationToken cancel);

/// <summary>
/// Confirm operation
/// POST https://{hostname}/v2/operations/{id}/confirm
/// </summary>
Task<Response<CommandIdDto>> ConfirmOperationAsync(OperationDto operation, CancellationToken cancel);

/// <summary>
/// Get current command status
/// GET https://{hostname}/v2/operations/{operationId}/commands/{commandId}
/// </summary>
Task<Response<CommandStatusDto>> GetCommandStatusAsync(string operationId, string commandId, CancellationToken cancel);
}
```

ILogger – необязательный параметр, при указании и реализации которого, SDK будет логировать все запросы и ответы на низком уровне.
Подробный пример использования SDK показан в методе UnistreamApiSample\UnistreamApiSample\MainForm.cs - doOperation()

Ниже скомпилированный, рабочий пример под .NET 4.6



App_ID, secPosID, app_secret, apiUrl, timeout, certificateFilePath, certifiactePassword – соответствует полям конфигурации API (класс UnistreamApiClientConfig), которая также необходима для создания экземпляра API.

Полный состав методов SDK приведен в блоке примеров справа(их названия и назначения соответствуют основному описанию API).

Пример расширения функционала

Рассмотрим пример создания функции добавления нового клиента.

Создаем свой класс и наследуем его от UnistreamApiClient, а затем добавляем в него метод, информацию (адрес API, параметры API) для которого, берем из основного описания API:

```
Public class My UnistreamApiClient : UnistreamApiClient { public async Task<Response<EmptyContent>> CreateClientAsync(ClientDto client, CancellationToken cancel) { return await PostDto<EmptyContent>(client, $"v2/clients", cancel); } }
```

Дело в том, что в классе UnistreamApiClient есть методы, обеспечивающие транспортные функции обращения к G3:

```
protected async Task<Response<T>> PostDto<T>(object dto, string path, CancellationToken cancel)
protected async Task<Response<T>> PutDto<T>(object dto, string path, CancellationToken cancel)
protected async Task<Response<T>> PatchDto<T>(object dto, string path, CancellationToken cancel)
protected async Task<Response<T>> GetDto<T>(string path, CancellationToken cancel)
protected async Task<byte[]> GetBytes(string path, CancellationToken cancel)
```

Жестких требований к передаваемым и принимаемым DTO объектам нет, так что их можно создавать самостоятельно.

API

API предназначено для интеграции услуг, предоставляемых Юнистрим, в автоматизированные системы партнёров. Это позволяет избежать двойного ввода и, при необходимости, реализовать выполнение определённых действий при изменении статуса операции. API создано на основе современных технологий для кроссплатформенной работы и открытых стандартов, что позволяет реализовать работу с API на любой современной платформе.

URL для подключения к API:

```
https://api.unistream.ru - production
https://slt-test.api.unistream.com - {sandbox}
```

Общие принципы работы с API

Все запросы к API выполняются асинхронно. При создании операции партнёр передаёт уникальный идентификатор, который представляет собой GUID и генерируется партнёром (в документации обозначается {id}). В дальнейшем, все манипуляции с созданной операцией выполняются по переданному идентификатору.

Обратите внимание, что каждому обращению к API, команде присваивается уникальный идентификатор, генерируемый системой Юнистрим. Этот идентификатор определяет именно конкретное обращение, а не операцию в целом. Не следует использовать такой идентификатор для получения статуса всей операции. Также следует отметить, что, в силу асинхронной сущности API, результат обработки команды возвращается моментально. Однако положительный ответ означает лишь, что запрос прошёл аутентификацию и имеет корректный формат. Операция, инициированная или изменённая таким запросом, может завершиться неудачно в процессе выполнения в системе Юнистрим. Именно поэтому требуется периодически проверять статус выполняемой операции.

Безопасность при работе с API

Весь обмен данными между сервером партнёра и сервером Юнистрим, предоставляющим API, осуществляется по соединению, защищённому SSL (HTTPS). **Работы с API по открытому соединению (HTTP) не допускается.** Партнёр должен проверять валидность сертификата сервера Юнистрим при каждом подключении!

Кроме этого, каждый запрос к API должен сопровождаться специальным заголовком Authorization, содержащим электронную подпись запроса и заголовком X-Unistream-Security-PosId, содержащим идентификатор точки предоставления услуги. Данные для подписи формируются из специальных заголовков X-Unistream-* и данных запроса при помощи алгоритма HMAC-SHA256. В качестве ключа подписи используется секретный код приложения (application secret). Подробнее узнать о формировании этого заголовка, а также ознакомиться с примерами можно в соответствующем разделе где указаны примера кода.

Реквизитами, необходимыми для работы с API, являются:

- ApplicationID – HEX-строка длиной 20 символов.
- Application Secret – BASE64-строка (перед использованием переводится в массив байт).
- POS ID – число.

Первые два реквизита идентифицируют партнёра в целом. Последний реквизит идентифицирует точку оказания услуг (касса, дополнительный офис и т.п.). Партнёр может запросить необходимое ему количество POSID. Для работы с API необходим как минимум один такой идентификатор. Для получения доступа необходимо заполнить заявку и передать её Вашему менеджеру.

Авторизация

Каждый запрос к API должен содержать аутентификационную информацию чтобы идентифицировать контрагента выполняющего запрос. Это достигается составлением строкового представления запроса из HTTP заголовков сообщения и MD5 хеша его содержимого, которые подписываются секретным ключом доступа, привязанным к идентификатору приложения.

Также контрагент должен добавлять в каждый запрос к API HTTP заголовок **"X-Unistream-Security-Posid"** содержащий идентификатор точки предоставления услуг. Значение данного заголовка также участвует в аутентификационной информации,"X-Unistream-Security-cashierLogin" необходим для идентификации операциониста инициирующего создание операции.

Передача подписи через стандартный HTTP заголовок Authorization


```
"Authorization: UNIHMAC " + APPLICATION_ID + ":" + "  
base64(hmac-sha256(APPLICATION_SECRET,  
to-upper(VERB) + "\n"  
+ CONTENT-MD5 + "\n"  
+ DATE + "\n"  
+ to-lower(url-decode(PATH-AND-QUERY)))  
+ X-UNISTREAM-HEADERS >>)"
```

Для получения ответов на английском языке, необходимо передавать в заголовках значение «Accept-Language: en»

Параметр	Описание
VERB	Название http метода заглавными буквами
CONTENT-MD5	Хэш код md5 содержимого запроса, для GET запросов - пустая строка
DATE	Заголовок дата запроса в универсальном формате RFC1123
PATH-AND-QUERY	URL-декодированный путь к ресурсу строчными буквами
X-UNISTREAM-HEADERS	Канкатенация значений заголовков, соответствующих маске X-Unistream-*. Заголовки сортируются по возрастанию по названию, приведенному к строчным буквам. Значение каждого заголовка начинается с символа "\n"

Пример:

Параметр	Значение	Примечание
VERB	"GET"	Метод GET
CONTENT-MD5	""	Пустая строка, т.к. метод GET
DATE	"Mon, 16 Nov 2015 07:14:02 GMT"	Дата
PATH-AND-QUERY	"/v1/definitions/operation?x=2 a"	Путь к ресурсу
X-UNISTREAM-HEADERS	"\n" + "123456" + "\n" + "a772"	Значение заголовков X-Unistream-Security-Posid и X-Unistream-Security-SomeOther

 Личные данные для авторизации и сертификат будут отправлены Вам на е-мил после заполнения заявки на подключение, сертификат потребуется при подтверждении операции.

5 больших этапов

• На первом этапе вызывается метод создания операции (состав: GUID, инфо о клиенте, входные данные для выбранной операции). Если в ответ приходит HTTP статус 202, то это означает что запрос принят и поставлен в очередь на обработку.
Иначе возвращается код ошибки и JSON описание ошибок допущенных партнером.

• Затем необходимо периодически (рекомендуется раз в 2 секунд) запрашивать статус операции для продолжения работы.
В начале может возвращаться статус 404, он означает что операция еще не зарегистрирована и надо продолжать опрашивать сервер.

После регистрации возможны 4 статуса (**Created**, **Accepted**, **Rejected**, **Failed**),необходимо дождаться статуса **Accepted** (означает, что прошли предварительные внутренние проверки и операция принята Юнистрим, готова для подтверждения).

• Методом GET необходимо загрузить себе документ (в нем содержаться данные об операции) для подписи своим сертификатом, после подписи нужно передать их методом подтверждение операции.

• После проверки подписи на стороне "Юнистрим", операция переводится в статус Confirmed и ее уже нельзя изменить (до этого шага возможна корректировка данных).

• Выполнение операции завершается статусом **Completed** или **Failed**, следует периодически опрашивать метод получения информации об операции до получения финального статуса.

Идентификация клиента

В API используется справочник клиентов и в запросе на создание операции передаётся идентификатор клиента и документ в секции ClientContext. Справочник клиентов позволяет производить поиск клиента по уникальному идентификатору (GUID), либо по первичному документу (как правило, паспорт). Поддерживается добавление новых документов, и ограниченное изменение данных клиента и его документов.

Обратите внимание, что не все данные у клиента можно изменить через API!

Также для клиентов из справочника есть возможность получать список их операций.

Выполнение операций подразумевает три уровня идентификации:

1. Без идентификации – большинство платежей в адрес юридических лиц (пополнение мобильного, оплата онлайн-игр и т.п.) на сумму до 15 000 руб.
2. Упрощённая идентификация – денежные переводы на сумму до 600 000 руб. в день и платежи в адрес юридических лиц на сумму до 100 000 руб.
3. Полная идентификация - денежные переводы на сумму более 600 т.р. в день.

Указанные суммы лимитов являются динамическими, зависят от типа операции, страны назначения, нормативных актов Банка России и Федеральных законов Российской Федерации.

СХЕМА ИДЕНТИФИКАЦИИ

Возвращает схему идентификации (т.е. набор правил идентификации) для заданной страны. Например, для полной идентификации в России (отправка переводов на сумму более 600 т.р. в день) необходим паспорт гражданина РФ (Passport.RUS) или паспорт иностранного гражданина (Passport.ARM), если предъявлен паспорт иностранного гражданина, необходим дополнительный документ, подтверждающий право нахождения на территории страны, например, миграционная карта (MigrationCard.RUS) или вид на жительство в РФ (ResidencePermit.RUS).

GET https://{hostname}/v2/identification/schemas/{countryCode}

Пример ответа схемы идентификации

```
{
  "full": {
    "person": {
      "id": "person.full",
      "type": "object",
      "properties": {
        "firstName": {
          "type": "string",
          "minLength": 1,
          "maxLength": 50
        },
        "CountryOfResidenceCode": {
          "type": "string",
          "minLength": 3,
          "maxLength": 3,
          "pattern": "^[a-zA-Z]+$"
        }
      },
      "required": [
        "firstName"
      ]
    },
    "documents": [
      {
        "name": "Identity",
        "rules": [
          {
            "oneof": "Passport.*, ForeignPassport.RUS"
          }
        ]
      },
      {
        "name": "Addmitance",
        "rules": [
          {
            "oneof": "MigrationCard.RUS, ResidencePermit.RUS",
            "condition": [
              {
                "ref": "Identity",
                "oneof": "Passport.(\"RUS,BLR\")"
              }
            ]
          }
        ]
      }
    ]
  }
}
```

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
countryCode	string	Трехбуквенный ISO-код страны	Обязательное поле

Описание ответа

Возвращает схему идентификации для страны в виде коллекции поддерживаемых уровней идентификации, каждый из которых связан с набором требований, предъявляемых к полноте и составу идентификационных данных клиента.

JSON-схема документа

Получение JSON-схемы по идентификатору типа документа.

GET https://{hostname}/v2/identification/schemas/documents/{id}

Пример ответа JSON-схемы по идентификатору типа документа

```
{
  "$schema": "http://json-schema.org/schema#",
  "title": "Passport of Russian Federation citizen",
  "type": "object",
  "properties": {
    "Series": {
      "title": "Document series",
      "description": "Document series, 4 digits",
      "pattern": "/^\\d{4}$/i",
      "type": "string",
      "maxLength": 4,
      "minLength": 4
    },
    "Number": {
      "title": "Document number",
      "description": "Document number, 6 digits",
      "pattern": "/^\\d{6}$/i",
      "type": "string",
      "maxLength": 6,
      "minLength": 6
    },
    "Issuer": {
      "title": "Issuer",
      "description": "The state authority who issued the document",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    }
  }
}
```

```
{
  "IssuerDepartmentCode": {
    "title": "Issuer department code",
    "description": "Issuer department code, format 000-000",
    "pattern": "/^\\d{3}-\\d{3}$/i",
    "type": "string",
    "maxLength": 7,
    "minLength": 7
  },
  "IssueDate": {
    "title": "Issue date",
    "description": "The issue date of the document",
    "type": "string",
    "format": "date-time"
  }
},
"additionalProperties": false,
"required": [
  "Series",
  "Number",
  "Issuer",
  "IssuerDepartmentCode",
  "IssueDate"
]
}
```

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
id	string	Идентификатор типа документа. Состоит из класса документа и ISO-кода страны его выдачи. Например, Passport.RUS - паспорт гражданина России.	Обязательно

Описание ответа

JSON-схема для запрошенного типа документа

ПРЕДВАРИТЕЛЬНЫЙ УРОВЕНЬ ИДЕНТИФИКАЦИИ КЛИЕНТА

Для целей отладки API предоставляет метод, позволяющий оценить получаемый уровень идентификации по данным клиента. Этот метод может быть полезен для отладки ошибок идентификации, когда система на попытку создания операции ответила, что фактический уровень идентификации не соответствует ожидаемому: «Current client level is Simple. Required level is Full.»

```
POST https://{hostname}/v2/identification/IdentifyAssume
```

Описание запроса

Контекст клиента

Поле	Тип данных	Описание
ClientId	globally unique identifier	Идентификатор клиента
Documents	Коллекция элементов типа String	Список документов клиента для идентификации, например Passport.RUS.4515341411
Phone	string	Идентификатор клиента

Пример запроса на определение предварительного уровня идентификации:

```
{
  "clientId": "aae18ffd-8396-4536-87a4-e71958d67dcf",
  "documents": [
    "Passport.RUS.4515341411"
  ]
}
```

Пример ответа на определение предварительного уровня идентификации:

```
{
  "clientId": "aae18ffd-8396-4536-87a4-e71958d67dcf",
  "identificationLevel": "Full"
}
```

Описание ответа

Коллекция элементов типа Object

Выполнение операции

После получения необходимых данных от пользователя и их валидации на стороне клиентского ПО, следует приступить к непосредственному выполнению операции. Оно происходит в два этапа:

1. Создание операции на сервере Юнистрим и валидация данных сервером.
2. Выполнение операции.

Рассмотрим этапы подробнее.

Для создания операции необходимо выполнить соответствующий вызов, передав уникальный идентификатор заявки. Все операции в API выполняются асинхронно, поэтому результатом вызова будет результат создания, но не проверки заявки. **Для получения статуса проверки заявки следует вызвать метод получения информации об операции, передавая идентификатор заявки.** Ожидаемый статус операции (в случае успешной проверки) – Accepted. Если проверка ещё не проведена, следует повторить вызов позже. Рекомендуемый интервал опроса операции – 3 секунды. В любом случае, не следует вызывать соответствующий метод чаще, чем раз в 3 секунды. Если проверка прошла успешно, следует проверить массив Documents объекта Operation, который возвращается в ответ на запрос статуса заявки. Если в этом массиве присутствуют объекты, их следует распечатать. Объекты представляют собой документы формата PDF закодированные в BASE64.


После того, как подтверждение на выполнение операции получено от клиента, можно переходить к подтверждению операции в системе Юнистрим. Для этого используется соответствующий метод API (метод описан в отдельном блоке подтверждение операции), в который следует передать идентификатор операции и результат подписания массива байт, полученных по ссылке из

поля `signDocument` (данные, полученные при проверке статуса заявки), закрытым ключом сертификата партнёра. **Обратите внимание, что документ для подписи (ссылка на который присутствует в `signDocument`) следует интерпретировать как массив байт! Не пытайтесь работать с полученным документом как с текстовым файлом, это приведёт к его перекодировке и, как следствие, некорректной подписи.**

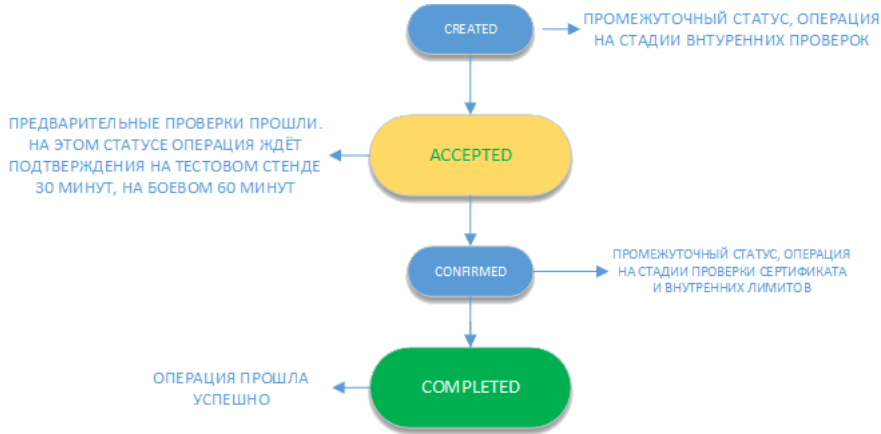
Тем не менее, имеется возможность перевести документ для подписи в понятный человеку вид для осуществления дополнительного контроля. Документ представляет собой текстовый файл в кодировке UTF-16 (или в кодировке UTF-8, если при запросе на его получение передавалось значение `utf-8` в заголовке `Accept-Encoding`).

После подтверждения следует проверить статус операции, как было описано выше. Ожидаемый статус операции – `Confirmed`.

Обратите внимание, что фактически операция выполнена только после того, как статус заявки изменился на `Completed`! После подтверждения операция всё ещё может завершиться с ошибкой. Следует корректно обрабатывать такую ситуацию и предоставлять оператору информацию о состоянии заявок.

 Не путайте опрос операции с опросом команды, необходимо проверять статус самой операции.

Удачная схема жизни операции по статусам



Клиенты

Перед созданием операции Вам нужно убедиться, что клиент есть в нашей базе, для этого Вы можете воспользоваться поиском, если клиент будет найден, убедитесь, что найденный клиент из базы совпадает с фактическим, если клиент не найден, то необходимо создать клиента.

Поиск клиента

Перед тем как создать клиента, ищем его в базе "Юнистрим", поиск возможен по серии и номеру документа, так и по идентификатору (GUID) клиента

ПРИМЕР ЗАПРОСА НА ПОИСК КЛИЕНТА ПО ДОКУМЕНТУ

GET <https://{hostname}/v2/clients/search?documentNumber=222277777&page=1&rows=1>

ПРИМЕР ЗАПРОСА НА ПОИСК КЛИЕНТА ПО ИДЕНТИФИКАТОРУ (GUID)

GET <https://{hostname}/v2/clients/{clientId}>

Пример ответа на поиск клиента по GUID и по номеру документа:

```
{
  "documents": [
    {
      "type": "Passport.RUS",
      "uniqueNumber": "Passport.RUS.373255555",
      "legend": "Паспорт гражданина РФ серия 3732 номер 555555",
      "fields": {
        "Series": "3732",
        "Number": "555555",
        "Issuer": "ОТДЕЛОМ №2 УФСБ РОССИИ",
        "IssuerDepartmentCode": "777-777",
        "IssueDate": "2002-07-07T00:00:00",
        "state": "RUS"
      }
    }
  ],
  "confidants": [],
  "id": "77777777-3333-7777-3333-777777777777",
  "countryOfResidence": "RUS",
  "firstName": "Виктор",
  "lastName": "Федоров",
  "middleName": "Петрович",
  "gender": "Male",
  "birthPlace": "г.Омск",
  "birthDate": "1988-01-23T00:00:00",
  "phoneNumber": "7977777777",
  "taxpayerIndividualIdentificationNumber": "",
  "address": {
    "addressString": "",
    "apartment": "",
    "building": "777",
    "city": "г.Омск",
    "countryCode": "RUS",
    "house": "7",
    "postcode": "777777",
    "state": "г.Омск",
  }
}
```

```
    "street": "ул.Верхняя Масловка"
  },
  "loyaltyCardNumber": null,
  "kazId": ""
}
```

Поиск операций созданных клиентом

Список операций, созданных клиентом с возможностью разбивки на страницы

```
GET https://{hostname}/v2/operations?page={page}&pageSize={pageSize}&from={from}&to={to}&clientId={clientId}
```

Пример ответа:

```
{
  "items": [
    {
      "id": "ee57b73b-2d1f-4202-91e7-f9ab99edeaf9",
      "createTime": "2019-02-14T15:19:53.511Z",
      "title": "Отправить перевод",
      "legend": "188429725733 3 600.00 RUB Ласо Карл Мазинов",
      "status": "Completed",
      "clientContext": {
        "clientId": "f934295f-3b73-480e-a510-8fe998197af3",
        "documents": [
          {
            "series": "2222",
            "number": "222222"
          }
        ],
        "loyaltyCardNumber": "",
        "smsConfirmationId": null
      },
      "operationType": "Transfer"
    }
  ],
  "page": 1,
  "pageSize": 20,
  "totalItemCount": 1,
  "totalPageCount": 1
}
```

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
page	integer	номер страницы	значение по умолчанию: 1
pageSize	integer	количество операций на страницу	
from	date	дата начала периода выборки	
to	date	дата конца периода выборки	
clientId	globally unique identifier	идентификатор клиента (GUID)	

Описание ответа

Запрошенная страница списка операций созданных клиентом

Имя	Тип данных	Описание
Items	Коллекция элементов типа OperationListItemModel	Коллекция с элементами списка
page	integer	количество операций на страницу
TotalPages	integer	дата начала периода выборки
TotalCount	date	дата конца периода выборки
pageSize	integer	количество операций на страницу

Создание клиента

При создании клиента Вам нужно учитывать его резиденство и необходимые документы для создания операции. Узнать схему идентификации по стране и возможные документы для создания клиента можно из следующего метода:

```
GET https://{hostname}/v2/identification/schemas/{Alpha-3 code}
```

Пример запроса на создание клиента:

```
{
  "id": "6edaba8c-ad5a-4ced-a4f2-bc9401efe59e",
  "countryOfResidence": "RUS",
  "firstName": "Андрей",
  "lastName": "Блазов",
  "middleName": "Петрович",
  "gender": "Male",
  "birthPlace": "г.Омск",
  "birthDate": "1988-01-23T00:00:00",
}
```

```
{
  "phoneNumber": "79299720011",
  "taxpayerIndividualIdentificationNumber": "",
  "kazId": null,
  "address": {
    "addressString": "г.Омск, ул.Ленина, 33",
    "apartment": "",
    "building": "33",
    "city": "г.Омск",
    "countryCode": "RUS",
    "house": "25",
    "postcode": "644088",
    "state": "г.Омск",
    "street": "улица Ленина"
  },
  "documents": [
    {
      "type": "Passport.RUS",
      "fields": {
        "Series": "0991",
        "Number": "992283",
        "Issuer": "МВД г.Омск,отделом №2",
        "IssuerDepartmentCode": "550-005",
        "IssueDate": "2008-07-07T00:00:00",
        "expiryDate": "2055-09-23T00:00:00",
        "state": "RUS"
      }
    }
  ]
}
```

URL запрос на создание клиента

POST https://{hostname}/v2/clients/

Описание запроса

Данные для создания клиента

Поле	Тип данных	Описание	Дополнительно
Id	string	Уникальный идентификатор клиента	
CountryOfResidence	string	ISO-код страны, резидентом которой является клиент	
FirstName	string	Имя	
LastName	string	Фамилия	
MiddleName	string	Отчество	
Gender		Пол (Male-мужчина, Female-женщина)	
BirthPlace	string	Место рождения	
BirthDate	date	Дата рождения	
PhoneNumber	string	Номер контактного телефона	
TaxpayerIndividualIdentificationNumber	string	Индивидуальный номер налогоплательщика (ИНН)	
KazId	string	Индивидуальный Идентификационный Номер в республике Казахстан	
Address		Информация об адресе	
Documents		Список документов	

Описание ответа

В случае успеха возвращает HTTP-код 201 без контента.

Изменение клиента

Уже существующему клиенту можно добавить документ и изменить часть данных:

1) ПРИМЕР ЗАПРОСА НА ДОБАВЛЕНИЕ ДОКУМЕНТА КЛИЕНТУ

POST https://{hostname}/v2/clients/6edaba0c-ad5a-4ced-a4f2-bc9401efe59e/documents

Пример запроса на добавление документа уже существующему клиенту:

```
{
  "type": "Passport.RUS",
  "fields": {
    "Series": "1234",
    "Number": "123456",
    "Issuer": "МВД г.Тестовый",
    "IssuerDepartmentCode": "123-456",
    "IssueDate": "2002-01-31T00:00:00",
    "expiryDate": "2055-09-23T00:00:00",
    "state": "RUS"
  }
}
```

2) ПРИМЕР ЗАПРОСА НА ИЗМЕНЕНИЕ УЖЕ СУЩЕСТВУЮЩЕГО ДОКУМЕНТА У КЛИЕНТА

PATCH https://{hostname}/v2/clients/6edaba0c-ad5a-4ced-a4f2-bc9401efe59e/documents

Пример запроса на изменение уже существующего документа у клиента:

```
{
  "type": "Passport.RUS",
  "fields": {
    "Series": "1234",
    "Number": "123456",
    "Issuer": "МВД г.Тестовый",
    "IssuerDepartmentCode": "123-456",
    "IssueDate": "2002-01-31T00:00:00",
    "expiryDate": "2055-09-23T00:00:00",
    "state": "RUS"
  }
}
```

3) ПРИМЕР ЗАПРОСА НА ИЗМЕНЕНИЕ ДАННЫХ АДРЕСА, РЕЗИДЕНТСТВА, НОМЕРА ТЕЛЕФОНА, ИНН

```
PATCH https://{hostname}/v2/clients/6edaba0c-ad5a-4ced-a4f2-bc9401efe59e
```

Пример запроса на изменение данных адреса, резидентства, номера телефона, ИНН:

```
{
  "Address": {
    "AddressString": "",
    "Apartment": "",
    "Building": "",
    "City": "г.Москва",
    "CountryCode": "RUS",
    "House": "20",
    "Postcode": "",
    "State": "г.Москва",
    "Street": "ул.Верхняя Масловка"
  },
  "PhoneNumber": "79990801122",
  "TaxpayerIndividualIdentificationNumber": "",
  "CountryOfResidence": "RUS"
}
```

! У клиента нельзя менять тип документа, серию и номер документа (касается всех типов паспортов), ФИО, пол, дату рождения. В миграционной карте (MigrationCard.RUS) можно изменить - VisaNumber, TenureStart, TenureEnd. В разрешении на временное проживание (TemporaryResidencePermit.RUS) можно изменить - Issuer, ValidFrom, ExpiryDate, IssuerDepartmentCode.

Операции

Процесс проведения операции через API начинается с вызова метода создания операции. Клиент API должен сгенерировать идентификатор операции, предоставить информацию о клиенте и контексте безопасности, а также подготовить входные данные для операции в соответствии с описанием типа операции.

Если метод создания операции вернул HTTP статус-код 202, это означает, что запрос на создание был принят системой и поставлен в очередь. В противном случае будет возвращена информация об ошибке.

После успешного создания операции необходимо выполнять периодический опрос её статуса методом получения информации об операции. Этот метод возвращает объект с информацией об операции, в поле Status которого содержится текущий статус операции.

Возможные статусы операции на этом этапе:

- Created - операция успешно создана (промежуточный статус, следует продолжать опрос)
- Accepted - первичная обработка операции прошла успешно, операция готова к подтверждению (окончательный статус)
- Rejected - первичная обработка операции завершилась с ошибкой, обычно этот статус означает, что во входных данных для операции допущена ошибка (окончательный статус)
- Failed - при первичной обработке операции произошла ошибка (окончательный статус)

Кроме того следует учитывать, что поскольку обработка запроса на создание операции происходит асинхронно, то с момента успешного выполнения метода создания операции до фактического создания может пройти некоторое время. В течение этого времени запрос информации об операции будет возвращать HTTP статус-код 404 (NotFound).

Для получения человекочитаемого описания операции (например, для отображения в UI) может быть вызван метод получения деталей операции.

Для запуска дальнейшей обработки принятой (статус Accepted) операции ее следует подтвердить. Для этого нужно загрузить документ для подписи, подписать его и передать подпись в метод подтверждения операции.

Успешное подтверждение переводит операцию в статус Confirmed. В любой момент до подтверждения операция может быть изменена. Изменение подтвержденной операции не допускается.

Методы подтверждения и изменения операции работают асинхронно путем создания соответствующей команды и постановки ее в очередь. Информация о созданной команде возвращается как результат вызова метода. Клиент должен ожидать окончания операции путем периодического опроса метода возврата статуса команды.

Выполнение подписанной операции будет завершено со статусом Completed или Failed. Следует периодически опрашивать метод получения информации об операции до получения финального статуса.

Полный список операций, созданных клиентом, может быть получен методом получения списка операций.

Создание операции

Метод возвращает следующие коды HTTP-статуса:
400 - запрос на создание операции не принят (причина отказа будет указана в поле Errors ответа).
200 - запрос на создание операции принят.
Создание операций происходит асинхронно, поэтому после получения запроса на создание проходит некоторое время прежде чем операция становится доступна.

В параметрах создаваемой операции присутствует массив документов клиента. Структура каждого из передаваемых документов определяется соответствующей JSON-схемой, а список необходимых документов задается схемой идентификации .

Пример запроса на создание операции

```
{
  "clientContext": {
    "clientId": "c56aa180-65aa-42ec-a945-5fd21dec0538",
    "documents": [
      "Passport.RUS.1234123456",
      "Visa.RUS.121234567"
    ],
    "phone": null,
    "firstName": null,
    "lastName": null,
    "middleName": null,
    "confidantId": null,
    "loyaltyCardNumber": null
  },
  "data": {
    "field1": "value1",
    "field2": "value2"
  },
  "externalId": null,
  "paymentInstrument": null
}
```

POST https://{hostname}/v2/operations/{operationType}/{id}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Описание запроса

Параметры создаваемой операции

Поле	Тип данных	Описание	Дополнительно
ClientContext	Контекст клиента	Идентификатор клиента и номера документов, которые он предоставил. При создании операции клиент должен быть создан заранее. Передается Guid клиента - ClientContext.	Обязательно для заполнения
Data	Коллекция элементов типа Object	Поля операции в формате JSON (зависят от типа операции)	Обязательно для заполнения
ExternalId	string	Уникальный идентификатор операции во внешней системе	необязательно для заполнения

Пример ответа:

```
{
  "operationId": "465d99de-b84a-4795-88ec-6c6998e7f0b8",
  "errors": null
}
```

Описание ответа

Поле	Тип данных	Описание
OperationId	globally unique identifier	Идентификатор созданной операции
Errors	Коллекция элементов типа Object	Ошибки в формате JSON. Если ошибок нет, то null.

Изменение операции

Изменение входных данных операции

PUT https://{hostname}/v2/operations/{id}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Описание запроса

Данные для обновления операции

Поле	Тип данных	Описание
ClientContext	Контекст клиента	Идентификатор клиента и номера документов, которые он предоставил. При создании операции одно из полей Client или ClientContext должно быть заполнено.

Поле	Тип данных	Описание
Data	Коллекция элементов типа Object	Поля операции в формате JSON (зависят от типа операции)

Описание ответа

Информация о команде обновления, помещенной в очередь для последующего опроса статуса команды

Поле	Тип данных	Описание
CommandId	globally unique identifier	Идентификатор команды
Errors	Коллекция элементов типа Object	Ошибки в формате JSON. Если ошибок нет, то null

 Операцию нельзя изменить на статусах CONFIRM и COMPLETED !

Получение операции

Создание операций происходит асинхронно, поэтому после получения запроса на создание проходит некоторое время прежде чем операция становится доступна. Все это время запрос операции возвращает код 404 (не найдено).

GET https://{hostname}/v2/operations/{id}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример ответа:

```
{
  "client": {
    "documents": [
      {
        "expiryDate": "2022-09-23T00:00:00",
        "extraValues": [],
        "issueDate": "2002-09-23T00:00:00",
        "issuer": "УВД РАМЕНСКОГО РАЙОНА ГОРОДА МОСКВЫ",
        "issuerDepartmentCode": "123-001",
        "number": "123321",
        "series": "1234",
        "type": 35
      }
    ],
    "source": null,
    "id": null,
    "countryOfResidence": "RUS",
    "firstName": "Иван",
    "lastName": "Иванович",
    "middleName": "Иванов",
    "gender": "Male",
    "birthPlace": "Москва",
    "birthDate": "1981-01-01T00:00:00",
    "phoneNumber": "79261234567",
    "taxpayerIndividualIdentificationNumber": "123456789101",
    "address": {
      "addressString": "",
      "apartment": "5",
      "building": "2",
      "city": "Москва",
      "countryCode": "RUS",
      "house": "20",
      "postcode": "",
      "state": "",
      "street": "Верхняя Масловка"
    },
    "loyaltyCardNumber": null,
    "kazId": null
  },
  "clientContext": {
    "clientId": "c56a4180-65aa-42ec-a945-5fd21dec0938",
    "documents": [
      {
        "series": "1234",
        "number": "567890"
      }
    ],
    "loyaltyCardNumber": null,
    "smsConfirmationId": null
  },
  "securityContext": {
    "agentId": 1,
    "bankId": 1,
    "cashierId": 1,
    "cashWindow": 1,
    "cashierUniqueId": "9ac63b91-e8de-42ed-90e4-c1880d3df99",
    "cashierLogin": "sample string 1",
    "cashierInfo": {
      "id": 1,
      "login": "sample string 1",
      "userName": "sample string 2",

```

```
    "uniqueId": "sample string 3",
    "postId": 1,
    "post": "sample string 4",
    "blocked": true,
    "firstName": "sample string 5",
    "lastName": "sample string 6",
    "middleName": "sample string 7",
    "bankIds": [
      {
        "bankId": 1,
        "parentBankId": 1
      },
      {
        "bankId": 1,
        "parentBankId": 1
      }
    ]
  },
  "cashAccounts": [
    {
      "account": "sample string 1",
      "currency": "sample string 2"
    },
    {
      "account": "sample string 1",
      "currency": "sample string 2"
    }
  ],
  "cashName": "sample string 2",
  "absCashName": "sample string 3",
  "absCashNumber": "sample string 4"
},
"data": null,
"id": "f55fc831-e7a2-46c7-9f6d-0533946d3898",
"type": "sample string 1",
"status": "sample string 2",
"statusChangedTime": "2019-02-17T11:57:54.0763737+03:00",
"statusMessage": "sample string 3",
"title": "sample string 4",
"legend": "sample string 5",
"dataErrors": null,
"signDocument": "http://webapihelppage6.com",
"documents": [
  "http://webapihelppage1.com",
  "http://webapihelppage2.com"
],
"clearingContext": {
  "agentFee": 10.0,
  "agentFeeCurrency": "RUB",
  "amount": 1000.0,
  "clearingRef": "1234567890",
  "currency": "RUB",
  "totalFee": 25.0,
  "totalFeeCurrency": "RUB"
},
"isNeedSmsConfirmation": true,
"acceptedTime": "2019-02-17T11:57:54.0763737+03:00"
}
```

Описание ответа

Информация об операции

Поле	Тип данных	Описание
Client	Информация о клиенте	Полные данные клиента. При создании операции одно из полей Client или ClientContext должно быть заполнено.
ClientContext	Контекст клиента	Идентификатор клиента и номера документов, которые он предоставил. При создании операции одно из полей Client или ClientContext должно быть заполнено.
SecurityContext	Информация о контексте безопасности	Данные о пункте и кассире
Data	Коллекция элементов типа Object	Поля операции в формате JSON (зависят от типа операции)
Id	globally unique identifier	Идентификатор операции
Type	string	Тип операции (Transfer, PayOutTransfer, CashToUnistreamCard и т.н)
Status	string	Статус операции: Created, Accepted, Confirmed, Completed, Failed, Cancelled, Rejected.
StatusChangedTime	date	Дата и время последнего изменения статуса операции
StatusMessage	string	Подсказка для кассира в связи с переходом операции в текущий статус (например если операция подтверждена кассиру будет рекомендовано отпустить клиента и перейти к слдующей операции)
Title	string	Тип операции для отображения на UI
Legend	string	Описание операции для отображения на UI
DataErrors	Коллекция элементов типа Object	Ошибки. Словарь в формате JSON. Имя поля совпадает с именем поля в Data, значение - текст ошибки
SignDocument	URI	Документ, содержащий байты, электронную подпись которых необходимо передать при подтверждении операции
Documents	Коллекция элементов типа Uri	Документы, которые необходимо распечатать и передать на подпись клиенту (в большинстве случаев - заявление)
ClearingContext	Информация по клирингу	Информация по клирингу (сумма, комиссия и т.п.)

Поле	Тип данных	Описание
IsNeedSmsConfirmation	boolean	Нужно ли окно ввода sms отображать
AcceptedTime	date	Дата и время обработки операции

Получение операции по идентификатору во внешней системе

Создание операций происходит асинхронно, поэтому после получения запроса на создание проходит некоторое время прежде чем операция становится доступна. Все это время запрос операции возвращает код 404 (не найдено).

GET https://{hostname}/v2/operations/external/{externalId}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример ответа:

```
{
  "client": {
    "documents": [
      {
        "expiryDate": "2022-09-23T00:00:00",
        "extraValues": [],
        "issueDate": "2002-09-23T00:00:00",
        "issuer": "УВД РАМЕНСКОГО РАЙОНА ГОРОДА МОСКВЫ",
        "issuerDepartmentCode": "123-001",
        "number": "123321",
        "series": "1234",
        "type": 35
      }
    ],
    "source": null,
    "id": null,
    "countryOfResidence": "RUS",
    "firstName": "Иван",
    "lastName": "Иванович",
    "middleName": "Иванов",
    "gender": "Male",
    "birthPlace": "Москва",
    "birthDate": "1981-01-01T00:00:00",
    "phoneNumber": "79261234567",
    "taxpayerIndividualIdentificationNumber": "123456789101",
    "address": {
      "addressString": "",
      "apartment": "5",
      "building": "2",
      "city": "Москва",
      "countryCode": "RUS",
      "house": "20",
      "postcode": "",
      "state": "",
      "street": "Верхняя Масловка"
    },
    "loyaltyCardNumber": null,
    "kazId": null
  },
  "clientContext": {
    "clientId": "c56a4180-65aa-42ec-a945-5fd21dec0538",
    "documents": [
      {
        "series": "1234",
        "number": "567890"
      }
    ],
    "loyaltyCardNumber": null,
    "smsConfirmationId": null
  },
  "securityContext": {
    "agentId": 1,
    "bankId": 1,
    "cashierId": 1,
    "cashWindow": 1,
    "cashierUniqueId": "9ac63b91-e0de-42ed-90e4-c1880dd3df99",
    "cashierLogin": "sample string 1",
    "cashierInfo": {
      "id": 1,
      "login": "sample string 1",
      "userName": "sample string 2",
      "uniqueId": "sample string 3",
      "postId": 1,
      "post": "sample string 4",
      "blocked": true,
      "firstName": "sample string 5",
      "lastName": "sample string 6",
      "middleName": "sample string 7",
      "bankIds": [
        {
          "bankId": 1,
          "parentBankId": 1
        },
        {
          "bankId": 1,
          "parentBankId": 1
        }
      ]
    }
  }
}
```

```
    },
    "cashAccounts": [
      {
        "account": "sample string 1",
        "currency": "sample string 2"
      },
      {
        "account": "sample string 1",
        "currency": "sample string 2"
      }
    ],
    "cashName": "sample string 2",
    "absCashName": "sample string 3",
    "absCashNumber": "sample string 4"
  },
  "data": null,
  "id": "f55fc831-e7a2-46c7-9f6d-8533946d3898",
  "type": "sample string 1",
  "status": "sample string 2",
  "statusChangedTime": "2019-02-17T11:57:54.0763737+03:00",
  "statusMessage": "sample string 3",
  "title": "sample string 4",
  "legend": "sample string 5",
  "dataErrors": null,
  "signDocument": "http://webapihelppage6.com",
  "documents": [
    "http://webapihelppage1.com",
    "http://webapihelppage2.com"
  ],
  "clearingContext": {
    "agentFee": 10.0,
    "agentFeeCurrency": "RUB",
    "amount": 1000.0,
    "clearingRef": "1234567890",
    "currency": "RUB",
    "totalFee": 25.0,
    "totalFeeCurrency": "RUB"
  },
  "isNeedSmsConfirmation": true,
  "acceptedTime": "2019-02-17T11:57:54.0763737+03:00"
}
```

Описание ответа

Информация об операции

Поле	Тип данных	Описание
Client	Информация о клиенте	Полные данные клиента. При создании операции одно из полей Client или ClientContext должно быть заполнено.
ClientContext	Контекст клиента	Идентификатор клиента и номера документов, которые он предоставил. При создании операции одно из полей Client или ClientContext должно быть заполнено.
SecurityContext	Информация о контексте безопасности	Данные о пункте и кассире
Data	Коллекция элементов типа Object	Поля операции в формате JSON (зависят от типа операции)
Id	globally unique identifier	Идентификатор операции
Type	string	Тип операции (Transfer, PayOutTransfer, CashToUnistreamCard и т.п)
Status	string	Статус операции: Created, Accepted, Confirmed, Completed, Failed, Cancelled, Rejected.
StatusChangedTime	date	Дата и время последнего изменения статуса операции
StatusMessage	string	Подсказка для кассира в связи с переходом операции в текущий статус (например если операция подтверждена кассиру будет рекомендовано отпустить клиента и перейти к слдующей операции)
Title	string	Тип операции для отображения на UI
Legend	string	Описание операции для отображения на UI
DataErrors	Коллекция элементов типа Object	Ошибки. Словарь в формате JSON. Имя поля совпадает с именем поля в Data, значение - текст ошибки
SignDocument	URI	Документ, содержащий байты, электронную подпись которых необходимо передать при подтверждении операции
Documents	Коллекция элементов типа Uri	Документы, которые необходимо распечатать и передать на подпись клиенту (в большинстве случаев - заявление)
ClearingContext	Информация по клирингу	Информация по клирингу (сумма, комиссия и т.п.)
IsNeedSmsConfirmation	boolean	Нужно ли окно ввода sms отображать
AcceptedTime	date	Дата и время обработки операции

Подтверждение операции

В качестве подтверждения передается массив байтов в формате base64, сформированный в результате подписи документа, ссылка на который указана в поле Operation.SignDocument,формируется на статусе операции - Accepted

```
POST https://{hostname}/v2/operations/{id}/confirm
```

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Описание запроса

Данные для подтверждения операции типа Confirmation

Параметры строки запроса

Имя	Тип данных	Описание
Confirmation	string	Подпись. Байты в формате base64

Пример запроса на подтверждение операции сформированный в результате подписи документа :

[illegible]

Типовые операции

В параметрах создаваемой операции присутствует массив документов клиента. Структура каждого из передаваемых документов определяется соответствующей JSON-схемой, а список необходимых документов задается схемой идентификации.

Отправка перевода

Тип операции

Transfer

Описание

Отправка перевода по международной системе денежных переводов Юнистрим

Для отправки перевода необходимо указать данные о клиенте - GUID клиента и документ с серией (если есть) и номером. В блоке data указываем куда отправляется перевод, кому отправляется, валюту, сумму выдачи/отправки.

Чтобы узнать доступные страны для отправки безадресных переводов, необходимо воспользоваться методом:

`https://{sandbox}/v2/definitions/Transfer/CountryData/lookup?curPosId={}&curCountryId={}&curFlags={}&idLevel=Full&lang=en`, данные для запроса необходимо получить методом - `https://{sandbox}/v1/po`
`scontext`.

После того как сформируете правильный запрос, Вам вернутся страны, некоторые страны будут отмечены обозначением (-1) значит в данную страну безадресные переводы недоступны.

ВСЕ ВОЗМОЖНЫЕ ВХОДНЫЕ ДАННЫЕ В БЛОК DATA ПО ОПЕРАЦИИ "ОТПРАВКА ПЕРЕВОДА" - TRANSFER

Все возможные входные данные в блок data по операции "Отправка перевода" - Transfer

```
"Country": "ARM",
"CountryData": "30;2;ARM",
"AddressTransferFlag": "",
"Region": "35",
"Bank": "321230",
"CountryHint": "",
"BeneficiaryLastName": "Иванов",
"BeneficiaryFirstName": "Сергей",
"BeneficiaryMiddleName": "Петрович",
"Phone": "79091112233",
"BeneficiaryRussiaResidence": "",
"Postcode": "",
"State": "",
"City": "",
"Street": "",
"House": "",
"Building": "",
"Apartment": "",
"BirthDate": "",
"AcceptedCurrency": "RUB",
"AmountType": "WithdrawAmount",
"Amount": "10000.19",
"WithdrawCurrency": "AMD",
"FundsSource": "",
"FundsSourceOther": "",
"OperationPurpose": "",
"OperationAim": ""
```

Имя	Название	Описание	Дополнительно
Country	Страна получения	Трехбуквенный ISO код страны. Обязательное поле, если поле CountryData не заполнено	Обязательно
CountryData	Страна получения	Данные о стране. Обязательное поле,если поле Country не заполнено	Не обязательно
AddressTransferFlag	Тип перевода	Укажите тип перевода	Не обязательно
Region	Регион получения	Укажите регион получения	Не обязательно
Bank	Банк получения	Укажите банк получения	Не обязательно
CountryHint	Показать информацию для получателя		Не обязательно
BeneficiaryLastName	Фамилия получателя		Обязательно
BeneficiaryFirstName	Имя получателя		Обязательно
BeneficiaryMiddleName	Отчество получателя		Не обязательно
Phone	Телефон получателя	7 929 972 76 23	Не обязательно
BeneficiaryRussiaResidence	Статус получателя	Укажите, является ли получатель резидентом РФ	Не обязательно
Postcode	Индекс		Не обязательно
State	Обл./штат		Не обязательно
City	Город		Не обязательно
Street	Улица		Не обязательно
House	Дом		Не обязательно
Building	Строение		Не обязательно
Apartment	Квартира		Не обязательно
BirthDate	Дата рождения получателя	Введите дату рождения или выберите из календаря	Не обязательно
AcceptedCurrency	Валюта внесения	Трехбуквенный ISO код валюты приёма перевода	Обязательно
AmountType	Тип суммы	Если это значение не передавать в запросе, то по умолчанию сумма будет к выдаче. Если в значение "AmountType" указываем параметр "AcceptedAmount", то сумма будет к отправке.	Не обязательно
Amount	Сумма	Сумма к выдаче (указывается в валюте выдачи)	Обязательно
WithdrawCurrency	Валюта выдачи	Трехбуквенный ISO код валюты выдачи перевода	Обязательно
FundsSource	Источник происхождения денежных средств		Не обязательно
FundsSourceOther	Уточнить источник происхождения денежных средств		Не обязательно
OperationPurpose	Сведения о целях установления и предполагаемом характере деловых отношений с кредитной организацией		Не обязательно
OperationAim	Цель проведения операции		Не обязательно

1) Пример запроса на создание операции "Отправка перевода" в Республику Беларусь

```
{
  "clientContext": {
    "clientId": "77777777-7777-7777-7777-7777777777",
    "documents": [
      "Passport.RUS.6891321898"
    ]
  },
  "data": {
    "Country": "BLR",
    "BeneficiaryLastName": "Иванов",
    "BeneficiaryFirstName": "Алексей",
    "BeneficiaryMiddleName": "Петрович",
    "AcceptedCurrency": "RUB",
    "Amount": 100,
    "WithdrawCurrency": "USD",
    "AmountType": "WithdrawAmount"
  }
}
```

РАССМОТРИМ РАБОЧИЙ ПРИМЕР ОТПРАВКИ БЕЗАДРЕСНОГО ПЕРЕВОДА (ПО ШАГАМ), В ПРИМЕРЕ ПОДРАЗУМЕВАЕТСЯ, ЧТО КЛИЕНТ УЖЕ СОЗДАН.


Как создать или найти клиента описано в разделе клиенты.

1) Используем POST запрос - `POST https://{hostname}/v2/operations/transfer/6edaba8c-ad5a-4ced-a4f2-bc9401efe59e` и тело запроса

УКАЗАННЫЕ ПАРАМЕТРЫ В ТЕЛЕ ЗАПРОСА

Имя	Описание	Дополнительно
clientId	Идентификатор клиента (GUID)	обязательно

Имя	Описание	Дополнительно
documents	Тип и номер документа	обязательно
Country	Трехбуквенный ISO код страны. Обязательное поле	Указываете ISO codeAlpha 3 страны куда хотите отправить перевод
BeneficiaryLastName	Фамилия получателя	обязательно
BeneficiaryFirstName	Имя получателя	обязательно
BeneficiaryMiddleName	Отчество получателя указывается, если есть	не обязательно
AcceptedCurrency	Указываете ISO codeAlpha 3 валюты отправки	обязательно
Amount	Сумма	обязательно
WithdrawCurrency	Указываете ISO codeAlpha 3 валюты выдачи	обязательно
AmountType	По умолчанию WithdrawAmount - т.е. указываемая сумма будет к выдаче, если указать AcceptedAmount, то указываемая сумма будет к внесению.	не обязательно

 Если запрос на создание пройдет успешно, вернется код ответа из диапазона 2xx - означает успешный запрос, но не операцию в целом.

2) Далее необходимо убедиться, что операция прошла все предварительные проверки и получила статус **Accepted** - готова к подтверждению, для этого нужно опросить операцию, используем **GET** запрос

GET https://{hostname}/v2/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe59e

2) Пример ответа при опросе операции

```
{
  "client": {
    "documents": [
      {
        "type": "Passport.RUS",
        "uniqueNumber": "Passport.RUS.1111111111",
        "legend": "Паспорт гражданина РФ серия номер 1111111111",
        "fields": {
          "Series": "1111",
          "Number": "111111",
          "Issuer": "ОМСКИЙ ОВД ОТДЕЛОМ № 2",
          "IssuerDepartmentCode": "550-005",
          "IssueDate": "2000-01-01T00:00:00+03:00"
        }
      }
    ],
    "confidants": [],
    "id": "11111111-1111-1111-1111-111111111111",
    "countryOfResidence": "RUS",
    "firstName": "Иван",
    "lastName": "Иванов",
    "middleName": "Сергеевич",
    "gender": "Male",
    "birthPlace": "г. Омск",
    "birthDate": "1988-01-23T00:00:00",
    "phoneNumber": "79298723311",
    "taxpayerIndividualIdentificationNumber": "1",
    "address": {
      "addressString": "1",
      "apartment": "1",
      "building": "1",
      "city": "г. Москва",
      "countryCode": "RUS",
      "house": "1",
      "postcode": "1",
      "state": "1",
      "street": "1"
    },
    "loyaltyCardNumber": null,
    "kazId": ""
  },
  "clientContext": {
    "clientId": "11111111-1111-1111-1111-111111111111",
    "documents": [
      "Passport.RUS.1111111111"
    ],
    "phone": null,
    "firstName": null,
    "lastName": null,
    "middleName": null,
    "confidantId": null,
    "loyaltyCardNumber": ""
  },
  "securityContext": {
    "agentId": 419635,
    "bankId": 420996,
    "cashierId": null,
    "cashWindow": null,
    "cashierUniqueId": null,
    "cashierLogin": "",
    "cashierInfo": null,
    "cashAccounts": [],
    "cashName": "",
    "absCashName": "",
    "absCashNumber": ""
  },
  "data": {
    "Output": {},
    "Closure": {},
    "Input": {
      "BeneficiaryFirstName": "Алексей",
      "BeneficiaryLastName": "Иванов",

```

```
"BeneficiaryMiddleName": "Петрович",
"BeneficiaryRussiaResidence": 2,
"Phone": "",
"AcceptedCurrency": "RUB",
"WithdrawCurrency": "RUB",
"Amount": 100,
"AmountType": "WithdrawAmount",
"Country": "BLR"
}
},
"Id": "6edaba0c-ad5a-4ced-a4f2-bc9401efe511",
"type": "Transfer",
"status": "Accepted",
"statusChangedTime": "2019-02-12T17:32:08.952",
"statusMessage": null,
"title": "Отправить перевод",
"legend": "257594864749 101.50 RUB Иванов Алексей Петрович",
"dataErrors": {},
"signDocument": "/v1/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe511/documents/06ca3f83-82f5-4384-8cbc-d7acb94ccb42",
"isNeedSmsConfirmation": false,
"documents": [
"/v1/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe511/documents/07c1dca6-230a-47f9-ab99-3bd4654b677d"
],
"clearingContext": {
"agentFee": 0.9,
"agentFeeCurrency": "RUB",
"amount": 100.0,
"clearingRef": "257594864749",
"currency": "RUB",
"totalFee": 1.5,
"totalFeeCurrency": "RUB"
},
"acceptedTime": null
}
```

В примере ответа содержится информация по операции, в данном случае "Отправка перевода", в ответе смотрим на поле статус "status", убеждаемся в успешном статусе, на данном шаге нас интересует статус **Accepted**, когда видим, что статус **Accepted**, распечатываем документ, документ содержится в блоке **"documents": ["/v1/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe511/documents/07c1dca6-230a-47f9-ab99-3bd4654b677d"]**

Скачиваем **PDF** документ по ссылке [https://\(hostname\)/v1/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe511/documents/07c1dca6-230a-47f9-ab99-3bd4654b677d](https://(hostname)/v1/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe511/documents/07c1dca6-230a-47f9-ab99-3bd4654b677d), документ идёт в двух экземплярах (для клиента и кассира) распечатываем и отдаём клиенту на подпись и проверку данных о переводе.

Пример документа для печати

ЗЯВЛЕНИЕ НА ОТПРАВКУ ПЕРЕВОДА

Контрольный номер перевода

257594864749

Дата

12.02.2019

Врем:

17:32

Прошу отправить денежный перевод по Международной платёжной системе денежных переводов «ЮНИСТРИМ»

Вносимая сумма	101,50 RUB	Комиссия	1,50 RUB
Сумма перевода	100,00 RUB	Курс обмена	
Назначение перевода	Частный перевод на текущие расходы	Дополнительная информация	
ОТПРАВИТЕЛЬ		ПОЛУЧАТЕЛЬ	
ФИО	Фёдоров Лев Сергеевич	Иванов Алексей Петрович	
Дата рождения	23.01.1988		
	Статус: РЕЗИДЕНТ	Телефон: 79298723311	Статус:
Адрес / ИНН	Россия, 1, 1, г.Москва, 1, дом 1, стр. 1, кв. 1	Беларусь	
Документ удостоверяющий личность	Паспорт гражданина РФ 1111 111111 выдан Омским ОВД ОТДЕЛОМ № 2 код подразд. 550-005 выд. 2000-01-01		
Пункт отправки	Ритейл, Держава, 8-й выход		

Подпись Отправителя (или его представителя)

Дополнительную информацию Вы можете получить:

8-800-333-22-44

www.unistream.ru

Да, я ознакомился с условиями Международной платёжной системы «ЮНИСТРИМ», в том числе с тем, что АО КБ «ЮНИСТРИМ» и Участники Системы имеют право хранить, обрабатывать и использовать указанные мною персональные данные. Я также согласен с получением от АО КБ «ЮНИСТРИМ» информации через любые каналы коммуникации в том числе по почте, SMS, электронной почте, телефону, системам обмена мгновенными сообщениями и иным средствам связи.

Подпись Отправителя (или его представителя)

После того как клиент подпишет документ переходим к подтверждению операции.

3) Для подтверждения нам потребуется сертификат (выдаётся центром сертификации "Юнистрим" ответственному лицу компании), алгоритм подписи описан в разделе "подтверждение операции".

После формирования массива байт в формате base64 сформированного в результате подписи документа, ссылка на который указана в поле `"signDocument": "/v1/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe511/documents/06ca3f83-82f5-4384-8cbc-d7acb94cc42"`

Передаётся **POST** запрос на подтверждение операции с массивом байт в формате base64

POST <https://{hostname}/v2/operations/926ac074-c2f7-4ac9-b08a-5380daa71068/confirm>

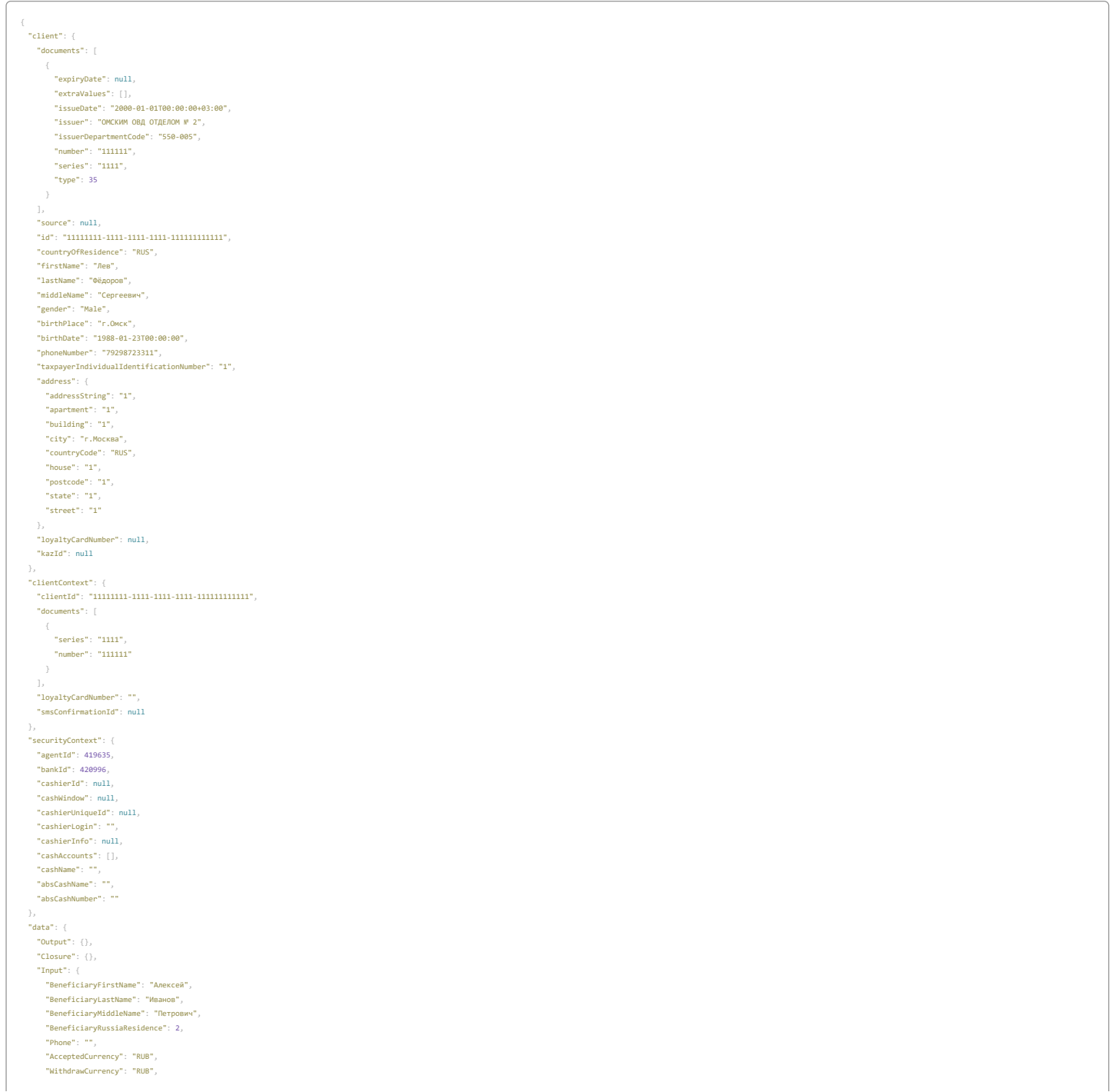
3) Пример получившегося base64 сформированного в результате подписи документа для подтверждения операции



4) После отправки запроса на подтверждение необходимо убедиться, что подпись сформировалась верно, операция подтвердилась и дошла до финального статуса **Completed**, для этого снова опрашиваем статус операции, используем **GET** запрос

GET <https://{hostname}/v2/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe59e>

4) Пример ответа при опросе операции, проверка финального статуса



```
    "Amount": 100,
    "AmountType": "WithdrawAmount",
    "Country": "BLR"
  }
},
"id": "6edaba0c-ad5a-4ced-a4f2-bc9401efe511",
"type": "Transfer",
"status": "Completed",
"statusChangedTime": "2019-02-12T18:06:19.225Z",
"statusMessage": "Перевод успешно создан, подпишите приходный кассовый ордер (только для Касс ЮНИСТРИМ), поставьте печать на заявление.",
"title": "Отправить перевод",
"legend": "257594864749 101.50 RUB Иванов Алексей Петрович",
"dataErrors": {},
"signDocument": "/v1/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe511/documents/06ca3f83-82f5-4384-8cbc-d7acb94ccb42",
"documents": [
  "/v1/operations/6edaba0c-ad5a-4ced-a4f2-bc9401efe511/documents/07c1dca6-230a-47f9-ab99-3bd4654b677d"
],
"clearingContext": {
  "agentFee": 0.9,
  "agentFeeCurrency": "RUB",
  "amount": 100.0,
  "clearingRef": "257594864749",
  "currency": "RUB",
  "totalFee": 1.5,
  "totalFeeCurrency": "RUB"
},
"isNeedSmsConfirmation": false,
"acceptedTime": "2019-02-12T18:06:16.914Z"
}
```

Исходя из ответа на опрос операции видим, что операция в статусе **Completed** - перевод успешно отправлен и готов к выдаче в указанной стране.

Изменить перевод

Тип операции

TransferNotice

Описание

Изменение данных получателя у отправленного перевода через систему денежных переводов Юнистрим

Для изменения перевода необходимо указывать информацию о клиенте - в данном случае это отправитель который хочет изменить ФИО получателя, ищем клиента в базе Юнистрим, если его нет, создаем.

Для изменение данных получателя у отправленного перевода, необходимо указывать код перевода в который вносим изменения, сумму и валюту к выдаче, и указать "нового" получателя.

РАССМОТРИМ РАБОЧИЙ ПРИМЕР ЗАПРОСА ДЛЯ ИЗМЕНЕНИЯ БЕЗАДРЕСНОГО ПЕРЕВОДА, В ПРИМЕРЕ ПОДРАЗУМЕВАЕТСЯ, ЧТО КЛИЕНТА МЫ УЖЕ СОЗДАЛИ И ИСПОЛЬЗУЕМ ЕГО ДАННЫЕ.

POST <https://{hostname}/v2/operations/TransferNotice/6edaba0c-ad5a-4ced-a4f2-bc9401efe59e>

Пример запроса на создание операции по изменению перевода - TransferNotice

```
{
  "clientContext": {
    "clientId": "df40ecab-c30b-4e6b-8f00-1721142c760d",
    "documents": [
      "Passport.RUS.4515341410"
    ]
  },
  "data": {
    "Code": "147331371988",
    "Amount": "2000",
    "Currency": "rub",
    "RecipientLastName": "Иванов",
    "RecipientFirstName": "Сепрей",
    "RecipientMiddleName": "Иванович"
  }
}
```

Имя	Описание	Дополнительно
clientId	Идентификатор клиента (GUID)	обязательно
documents	Тип и номер документа	обязательно
Code	Укажите номер перевода	обязательно
Amount	Укажите сумму перевода к выдаче	обязательно
Currency	Валюта перевода к выдаче	обязательно
RecipientLastName	Фамилия получателя	обязательно
RecipientFirstName	Имя получателя	обязательно
RecipientMiddleName	Отчество получателя указывается, если есть	не обязательно

Весь остальной алгоритм абсолютно схож с операцией "Отправка перевода", опрашиваем операцию, ожидаем статуса **Accepted**, скачиваем и распечатываем документ для клиента, после этого подтверждаем операцию и опрашиваем статус операции до получения статуса **Completed**

❗ При изменении перевода необходимо указывать валюту и сумму к выдаче, код перевода который хотим изменить и нового получателя.

Выдача перевода

Тип операции
PayOutTransfer

Описание
Выдача перевода отправленного через систему денежных переводов Юнистрим

Для выдачи перевода необходимо указывать информацию о клиенте, в данном случае это получатель, ищем клиента в базе Юнистрим, если его нет, создаем.

При формировании запроса на выдачу перевода, нам потребуется информация от клиента - номер перевода, валюта и сумма к выдаче.

РАССМОТРИМ РАБОЧИЙ ПРИМЕР ЗАПРОСА ДЛЯ ВЫДАЧИ БЕЗАДРЕСНОГО ПЕРЕВОДА, В ПРИМЕРЕ ПОДРАЗУМЕВАЕТСЯ, ЧТО КЛИЕНТА МЫ УЖЕ СОЗДАЛИ И ИСПОЛЬЗУЕМ ЕГО ДАННЫЕ.

POST https://{hostname}/v2/operations/PayOutTransfer/777aba0c-ad5a-4ced-7777-bc9401e7779e

Пример запроса на создание операции по выдаче перевода - PayOutTransfer

```
{
  "clientContext": {
    "clientId": "df40ecab-c30b-4e6b-8f00-1721142c760d",
    "documents": [
      "Passport.RUS.4515341410"
    ]
  },
  "data": {
    "Code": "147331371988",
    "Amount": "2000",
    "Currency": "rub"
  }
}
```

Имя	Описание	Дополнительно
clientId	Идентификатор клиента (GUID)	обязательно
documents	Тип и номер документа	обязательно
Code	Укажите номер перевода	обязательно
Amount	Укажите сумму перевода к выдаче	обязательно
Currency	Валюта перевода к выдаче	обязательно

Весь остальной алгоритм абсолютно схож с операцией "Отправка перевода", опрашиваем операцию, ожидаем статуса **Accepted**, скачиваем и распечатываем документ для клиента, после этого подтверждаем операцию и опрашиваем статус операции до получения статуса **Completed**

❗ Окончательную сумму к выдаче перевода получаете из clearingContext при опросе операции.

Аннулирование перевода

Тип операции
PayOutRevokedTransfer

Описание
"Аннулирование" происходит на основе выдачи перевода, только получатель это - отправитель, комиссия при использовании данного метода клиенту не возвращается..

При формировании запроса на "Аннулирование" перевода, нам потребуется информация от клиента - номер перевода, валюта и сумма к выдаче.

Для аннулирования перевода необходимо указывать информацию о клиенте - в данном случае это отправитель который хочет аннулировать перевод, ищем клиента в базе Юнистрим, если его нет, создаем.

РАССМОТРИМ РАБОЧИЙ ПРИМЕР ЗАПРОСА ДЛЯ АННУЛИРОВАНИЯ БЕЗАДРЕСНОГО ПЕРЕВОДА, В ПРИМЕРЕ ПОДРАЗУМЕВАЕТСЯ, ЧТО КЛИЕНТА МЫ УЖЕ СОЗДАЛИ И ИСПОЛЬЗУЕМ ЕГО ДАННЫЕ.

POST https://{hostname}/v2/operations/PayOutRevokedTransfer/777aba0c-ad5a-4ced-7777-bc9401e7779e

Пример запроса на создание операции по аннулированию перевода - PayOutRevokedTransfer

```
{
  "clientContext": {
    "clientId": "df40ecab-c30b-4e6b-8f00-1721142c760d",
    "documents": [
      "Passport.RUS.4515341410"
    ]
  },
  "data": {
    "Code": "147331371988",
    "Amount": "2000",
    "Currency": "rub"
  }
}
```

Имя	Описание	Дополнительно
clientId	Идентификатор клиента (GUID)	обязательно
documents	Тип и номер документа	обязательно
Code	Укажите номер перевода	обязательно

Имя	Описание	Дополнительно
Amount	Укажите сумму перевода к выдаче	обязательно
Currency	Валюта перевода к выдаче	обязательно

Весь остальной алгоритм абсолютно схож с операцией "Отправка перевода", опрашиваем операцию, ожидаем статуса **Accepted**, скачиваем и распечатываем документ для клиента, после этого подтверждаем операцию и опрашиваем статус операции до получения статуса **Completed**

❗ При аннулировании перевода сумму необходимо указывать к выдаче.

Платежи

Платежи, имеющие статус "Confirm" и "Completed" не подлежат изменению и отмене через API.

Пополнение карты Юнистрим

Тип операции
CashToUnistreamCard

Описание
Пополнение карты ЮНИСТРИМ

POST <https://{hostname}/v2/operations/{operationType}/{id}>

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример запроса на создание операции "Пополнение карты Юнистрим"

```
{
  "clientContext": {
    "clientId": "680e6ec6-4ed0-4d62-8f76-6f204fdf46f9",
    "documents": ["Passport.BLR.MP3651151"]
  },
  "data": {
    "CardNumber": "4117270300010601",
    "AcceptedCurrency": "RUB",
    "Amount": "1000",
    "ThirdParty": "true",
    "ThirdPartyLastName": "Викторов",
    "ThirdPartyFirstName": "Мван",
    "ThirdPartyMiddleName": "Серебрен",
    "FundsSource": "",
    "OperationPurpose": "",
    "OperationAim": ""
  }
}
```

Входные данные

Имя	Название	Описание	Дополнительно
CardNumber	Номер карты	Укажите номер карты Юниситрим для пополнения	Обязательно
Amount	Сумма	Укажите сумму	Обязательно
AcceptedCurrency	Валюта внесения	Трехбуквенный ISO код валюты внесения	Обязательно
ThirdParty	Оплата за третье лицо	Обязательно укажите true , если клиент вносит деньги не на свою карту	Не обязательно если клиент вносит деньги на свою карту
ThirdPartyLastName	Фамилия владельца карты		Обязательно если пополняется карта третьего лица
ThirdPartyFirstName	Имя владельца карты		Обязательно если пополняется карта третьего лица
ThirdPartyMiddleName	Отчество владельца карты		Не обязательно
FundsSource	Источник происхождения денежных средств		Не обязательно
OperationPurpose	Сведения о целях установления и предполагаемом характере деловых отношений с кредитной организацией		Не обязательно
OperationAim	Цель проведения операции		Не обязательно

Пополнение в Тинькофф Банк

Тип операции
Tcs

Описание
Пополнение счета в Тинькофф Банк по номеру договора или номеру банковской карты

POST `https://{hostname}/v2/operations/{operationType}/{id}`

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример запроса на создание операции пополнение счета в Тинькофф Банк по номеру карты

```
{
  "clientContext": {
    "clientId": "680e6ec6-4ed8-4d62-8f76-6f204fd46f9",
    "documents": ["Passport.BLR.MP3651151"]
  },
  "data": {
    "FundsTarget": "Card",
    "CardNumber": "5213243848029532",
    "AcceptedCurrency": "RUB",
    "Amount": "3000.0",
    "RecipientLastName": "Викторов",
    "RecipientFirstName": "Иван",
    "RecipientMiddleName": "Серебренч",
    "FundsSource": "",
    "FundsSourceOther": "",
    "OperationPurpose": "",
    "OperationAim": ""
  }
}
```

Пример запроса на создание операции пополнение счета в Тинькофф Банк по номеру договора

```
{
  "clientContext": {
    "clientId": "680e6ec6-4ed8-4d62-8f76-6f204fd46f9",
    "documents": ["Passport.BLR.MP3651151"]
  },
  "data": {
    "FundsTarget": "Contract",
    "ContractNumber": "5095648407",
    "AcceptedCurrency": "RUB",
    "Amount": "5000.0",
    "RecipientLastName": "Викторов",
    "RecipientFirstName": "Иван",
    "RecipientMiddleName": "Серебренч",
    "FundsSource": "",
    "OperationPurpose": "",
    "OperationAim": ""
  }
}
```

Входные данные

Имя	Название	Описание	Дополнительно
FundsTarget	Указать тип пополнения	<code>Card</code> если пополняем по номеру карты и <code>Contract</code> если пополняем по номеру договора	Обязательно
CardNumber	Номер карты	Укажите номер карты	Обязательно если пополняем счет по номеру карты
ContractNumber	Номер договора	Укажите номер договора	Обязательно если пополняем счет по номеру договора
AcceptedCurrency	Валюта внесения	В этой валюте будет зачисление на карту RUB/USD/EUR. По номеру договора только RUB.	Обязательно
Amount	Сумма зачисления	Укажите сумму зачисления, не менее 100 RUB и не более 900000 RUB , не менее 5 USD/EUR и не более 15000 USD/EUR	Обязательно
RecipientLastName	Фамилия получателя		Зависит от значений других полей
RecipientFirstName	Имя получателя		Зависит от значения других полей
RecipientMiddleName	Отчество получателя		Не обязательно
FundsSource	Источник происхождения денежных средств		Не обязательно
OperationPurpose	Сведения о целях установления и предполагаемом характере деловых отношений с кредитной организацией		Не обязательно

Имя	Название	Описание	Дополнительно
OperationAim	Цель проведения операции	Укажите сумму перевода	Не обязательно

Пополнение карт VISA

Тип операции

Compass

Описание

Пополнение карт VISA

POST https://{hostname}/v2/operations/{operationType}/{id}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример запроса на создание операции пополнение карт VISA

```
{
  "clientContext": {
    "clientId": "680e6ec6-4ed0-4d62-8f76-6f204fdf46f9",
    "documents": ["Passport.BLR.MP3651151"]
  },
  "data": {
    "CardNumber": "4276740013643663",
    "RecipientLastName": "Викторов",
    "RecipientFirstName": "Иван",
    "RecipientMiddleName": "Сергеевич",
    "AcceptedCurrency": "RUB",
    "Amount": "500",
    "WithdrawCurrency": "RUB",
    "FundsSource": "",
    "OperationPurpose": "",
    "OperationAim": ""
  }
}
```

Входные данные

Имя	Название	Описание	Дополнительно
CardNumber	Номер карты	Укажите номер карты	Обязательно
RecipientLastName	Фамилия получателя		Обязательно
RecipientFirstName	Имя получателя		Обязательно
RecipientMiddleName	Отчество получателя		Не обязательно
AcceptedCurrency	Валюта внесения	Валюта внесённая клиентом	Обязательно
Amount	Сумма зачисления	Укажите сумму зачисления	Обязательно
WithdrawCurrency	Валюта зачисления	Валюта зачисления на карту	Обязательно
FundsSource	Источник происхождения денежных средств		Не обязательно
OperationPurpose	Сведения о целях установления и предполагаемом характере деловых отношений с кредитной организацией		Не обязательно
OperationAim	Цель проведения операции	Укажите сумму перевода	Не обязательно

Пополнение карт UZCARD

Тип операции

UZCARD

Описание

Пополнение карт UZCARD

POST https://{hostname}/v2/operations/{operationType}/{id}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример запроса на создание операции пополнение карт UZCARD

```
{
  "clientContext": {
    "clientId": "680e6ec6-4ed8-4d62-8f76-6f204fdf46f9",
    "documents": ["Passport.BLR.MP3651151"]
  },
  "data": {
    "CardNumber": "8600310435774278",
    "RecipientLastName": "Викторов",
    "RecipientFirstName": "Иван",
    "RecipientMiddleName": "Сергеевич",
    "AcceptedCurrency": "RUB",
    "Amount": "100000",
    "WithdrawCurrency": "UZS",
    "FundsSource": "",
    "OperationPurpose": "",
    "OperationAim": ""
  }
}
```

Входные данные

Имя	Название	Описание	Дополнительно
CardNumber	Номер карты	Укажите номер карты UZCARD	Обязательно
RecipientLastName	Фамилия получателя		Обязательно
RecipientFirstName	Имя получателя		Обязательно
RecipientMiddleName	Отчество получателя		Не обязательно
AcceptedCurrency	Валюта внесения	Валюта внесённая клиентом	Обязательно
Amount	Сумма зачисления	Укажите сумму зачисления минимум 100000, максимум 4000000 UZS	Обязательно
WithdrawCurrency	Валюта зачисления	Валюта зачисления на карту только UZS	Обязательно
FundsSource	Источник происхождения денежных средств		Не обязательно
OperationPurpose	Сведения о целях установления и предполагаемом характере деловых отношений с кредитной организацией		Не обязательно
OperationAim	Цель проведения операции	Укажите сумму перевода	Не обязательно

Пополнение карт Корти Милли

Тип операции
Fmfb_CortyMilli

Описание
Пополнение карт Корти Милли

POST https://{hostname}/v2/operations/{operationType}/{id}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример запроса на создание операции пополнение карт Корти Милли

```
{
  "clientContext": {
    "clientId": "680e6ec6-4ed8-4d62-8f76-6f204fdf46f9",
    "documents": ["Passport.BLR.MP3651151"]
  },
  "data": {
    "CardNumber": "5058270011100046",
    "RecipientLastName": "Викторов",
    "RecipientFirstName": "Иван",
    "RecipientMiddleName": "Сергеевич",
    "AcceptedCurrency": "RUB",
    "Amount": "10",
    "WithdrawCurrency": "RUB",
    "FundsSource": "",
    "OperationPurpose": "",
    "OperationAim": ""
  }
}
```

Входные данные

Имя	Название	Описание	Дополнительно
-----	----------	----------	---------------

Имя	Название	Описание	Дополнительно
CardNumber	Номер карты	Укажите номер карты	Обязательно
RecipientLastName	Фамилия получателя		Обязательно
RecipientFirstName	Имя получателя		Обязательно
RecipientMiddleName	Отчество получатля		Не обязательно
AcceptedCurrency	Валюта внесения	Валюта внесённая клиентом	Обязательно
Amount	Сумма зачисления	Укажите сумму зачисления	Обязательно
WithdrawCurrency	Валюта зачисления	Валюта зачисления на карту, допустимо только RUB	Обязательно
FundsSource	Источник происхождения денежных средств		Не обязательно
OperationPurpose	Сведения о целях установления и предполагаемом характере деловых отношений с кредитной организацией		Не обязательно
OperationAim	Цель проведения операции	Укажите сумму перевода	Не обязательно

Пополнение банковской карты VISA/MasterCard/Мир/UZcard/КортиМилли

Тип операции
CashToCard

Описание
Пополнение банковской карты VISA/MasterCard/Мир/UZcard/КортиМилли

POST https://{hostname}/v2/operations/{operationType}/{id}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример запроса на создание операции пополнение карт VISA

```
{
  "clientContext": {
    "clientId": "680e6ec6-4ed8-4d62-8f76-6f204fdf46f9",
    "documents": ["Passport.BLR.MP3651151"]
  },
  "data": {
    "CardNumber": "4276740813643663",
    "RecipientLastName": "Викторов",
    "RecipientFirstName": "Иван",
    "RecipientMiddleName": "Сергеевич",
    "AcceptedCurrency": "RUB",
    "Amount": "500",
    "WithdrawCurrency": "RUB",
    "FundsSource": "",
    "OperationPurpose": "",
    "OperationAim": ""
  }
}
```

Входные данные

Имя	Название	Описание	Дополнительно
CardNumber	Номер карты	Укажите номер карты	Обязательно
RecipientLastName	Фамилия получателя		Обязательно
RecipientFirstName	Имя получателя		Обязательно
RecipientMiddleName	Отчество получатля		Не обязательно
AcceptedCurrency	Валюта внесения	Валюта внесённая клиентом	Обязательно
Amount	Сумма зачисления	Укажите сумму зачисления	Обязательно
WithdrawCurrency	Валюта зачисления	Валюта зачисления на карту	Обязательно
FundsSource	Источник происхождения денежных средств		Не обязательно
OperationPurpose	Сведения о целях установления и предполагаемом характере деловых отношений с кредитной организацией		Не обязательно
OperationAim	Цель проведения операции	Укажите сумму перевода	Не обязательно

Оплата патента

Тип операции
Patent_Standart

Описание
Оплата налога на доходы физических лиц в виде фиксированного авансового платежа для иностранных граждан. Не требует поиска ИФНС.

POST https://{hostname}/v2/operations/{operationType}/{id}

Параметры строки запроса

Имя	Тип данных	Описание	Дополнительно
operationType	string	Тип операции	Обязательно для заполнения
id	globally unique identifier	Идентификатор операции	Обязательное для заполнения

Пример запроса на создание операции "Оплата патента"

```
{
  "clientContext": {
    "clientId": "9beff736-5109-4180-9488-5dc00b70b84d",
    "documents": [
      "Passport.TJK.1000010"
    ],
  },
  "data": {
    "fnsMode": "search",
    "thirdParty": "no",
    "thirdPartyLastName": "",
    "thirdPartyFirstName": "",
    "thirdPartyMiddleName": "",
    "Amount": "1000.00",
    "oktmo": "46623101",
    "innByPatentNumber": "manual",
    "ownerINN": "504716187777",
    "paymentBasis": "TP",
    "taxPeriod": ""
  }
}
```

Входные данные

Имя	Название	Описание	Дополнительно
FnsMode	Реквизиты ИФНС	в данном случае укажите <input type="text" value="search"/>	Обязательно
ThirdParty	За налогоплательщика платит третье лицо	Укажите <input type="text" value="yes"/> если оплачивается патент за третье лицо, в остальных случаях <input type="text" value="no"/>	Обязательно
ThirdPartyLastName	Фамилия владельца патента	Укажите фамилию владельца патента	Обязательно если оплачивается патент за третье лицо
ThirdPartyFirstName	Имя владельца патента	Укажите имя владельца патента	Обязательно если оплачивается патент за третье лицо
ThirdPartyMiddleName	Отчество владельца патента	Укажите отчество владельца патента	Не обязательно
Amount	Сумма	Укажите сумму	Обязательно
Oktmo	ОКТМО (105)	Введите 8 или 11 цифр ОКТМО органа ФНС, в который идет оплата	Обязательно
InnByPatentNumber	В данном случае укажите <input type="text" value="manual"/>		Обязательно
OwnerINN	ИНН владельца патента (60)	Введите 12 цифр ИНН владельца патента	Обязательно
PaymentBasis	Основание платежа (106)	Может принимать значения: <input type="text" value="0, TP, 3D, BF, TP, PC, OT, RT, PB, PR, AP, AR, IN, TL, ET"/> «ТП» - текущий год. Расшифровки остальных найдете в нормативных документах. Если клиент оплачивает за конкретный период, указанный в платежке (Оплата задолженности по требованию ФНС), ставится «ТР» (Срок, установленный в документе) и в TaxPeriod тогда уже ставится конкретная дата из платежки	Обязательно
TaxPeriod	Налоговый период (107)	Дата из реквизитов в формате: XX.XX.XXXX или "0"	Обязательно если в PaymentBasis указывается значение <input type="text" value="TP"/>

В InfoApi все методы носят справочный характер

URL для подключения:

```
https://info.api.unistream.com - production
https://slt-test.info.api.unistream.com - {sandbox}
```

Справочник стран

Список стран

Пример ответа:

```
{
  "page": 1,
  "totalPages": 5,
  "totalCount": 50,
  "items": [
    {
      "countryCode": "RUS",
      "name": "Russia",
      "_links": {
        "self": "/api/v1/countries/RUS"
      }
    },
    {
      "countryCode": "FRA",
      "name": "France",
      "_links": {
        "self": "/api/v1/countries/FRA"
      }
    },
    {
      "countryCode": "BLR",
      "name": "Belorussia",
      "_links": {
        "self": "/api/v1/countries/BLR"
      }
    }
  ],
  "_links": {
    "self": "/api/v1/countries"
  }
}
```

Получение списка стран с возможностью фильтрации и разбивкой на страницы

```
GET https://{hostname}/api/v1/countries?filter.term={filter.term}&page={page}&rows={rows}
```

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
filter.term	string	Произвольная строка для фильтрации списка	
page	integer	Номер страницы	Значение по умолчанию: 1
rows	integer	Количество элементов на странице	Значение по умолчанию: 50

Описание ответа

Запрошенная страница списка стран

Поле	Тип	Описание	Дополнительно
Page	integer	Номер страницы	
TotalPages	integer	Общее количество страниц	
TotalCount	integer	Общее количество элементов	
Items	Коллекция элементов типа CountryModel	Список элементов	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Страна по ISO-коду

Пример ответа:

```
{
  "countryCode": "RUS",
  "name": "Russia",
  "_links": {
    "self": "/api/v1/countries/RUS"
  }
}
```

```
}

```

Получение информации о стране по её ISO-коду `GET https://{hostname}/api/v1/countries/{isoCode}

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
isoCode	string	Трехбуквенный ISO-код страны	Обязательное поле

Описание ответа Объект с информацией о стране

Поле	Тип	Описание	Дополнительно
CountryCode	integer	ISO-код страны	
Name	string	Общее количество страниц	
TotalCount	integer	Локализованное наименование страны	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Список регионов по операции

Получение списка регионов для страны по типу операции с возможностью фильтрации и разбивкой на страницы

Пример ответа:

```
{
  "page": 1,
  "totalPages": 5,
  "totalCount": 50,
  "items": [
    {
      "id": 1,
      "countryCode": "RUS",
      "name": "g.Moscow",
      "currencies": null,
      "subway": true,
      "_links": {
        "self": "/api/v1/countries/RUS/regions/1"
      }
    },
    {
      "id": 2,
      "countryCode": "RUS",
      "name": "Moscow region",
      "currencies": null,
      "subway": false,
      "_links": {
        "self": "/api/v1/countries/RUS/regions/2"
      }
    },
    {
      "id": 3,
      "countryCode": "RUS",
      "name": "g.Helsinki",
      "currencies": null,
      "subway": false,
      "_links": {
        "self": "/api/v1/countries/RUS/regions/3"
      }
    }
  ],
  "_links": {
    "self": "/api/v1/countries/RUS/regions"
  }
}
```

Получение информации о стране по её ISO-коду

```
GET https://{hostname}/api/v1/countries/{countryIsoCode}/regions/{operation}?filter.term={filter.term}&page={page}&rows={rows}
```

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
countryIsoCode	string	Трехбуквенный ISO-код страны	Обязательное поле
operation	string	Тип операции, может принимать значения: "SendAddressTransfer", "SendAddresslessTransfer", "PayoutAddressTransfer", "PayoutAddresslessTransfer", "PayoutMobileTransfer"	Обязательное поле
filter.term	string	Произвольная строка для фильтрации списка	Обязательное поле
page	integer	Номер страницы	Значение по умолчанию: 1
rows	integer	Количество элементов на странице	Значение по умолчанию: 50

Описание ответа Запрошенная страница списка регионов

Поле	Тип	Описание	Дополнительно
Page	integer	Номер страницы	
TotalPages	integer	Общее количество страниц	
TotalCount	integer	Общее количество элементов	
Items	Коллекция элементов типа RegionModel	Список элементов	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Регион по идентификатору

Получение информации о регионе внутри страны, по идентификатору региона

Пример ответа:

```
{
  "id": 1,
  "countryCode": "RUS",
  "name": "г.Москва",
  "currencies": null,
  "subway": true,
  "_links": {
    "self": "/api/v1/countries/RUS/regions/1",
    "country": "/api/v1/countries/RUS"
  }
}
```

GET https://{hostname}/api/v1/countries/{countryIsoCode}/regions/{id}

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
countryIsoCode	string	Трехбуквенный ISO-код страны	Обязательное поле
id	integer	Идентификатор региона	Обязательное поле

Описание ответа Объект с информацией о регионе

Поле	Тип	Описание	Дополнительно
Id	integer	Идентификатор региона	
CountryCode	integer	ISO-код страны	
Name	string	Локализованное наименование региона	
Currencies	Коллекция элементов типа String	Допустимые валюты для региона	
Subway	boolean	Признак наличия метро в регионе	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Метро в регионе

Пример ответа:

```
{
  "stations": [
    "Динамо",
    "Belorusskaya",
    "Komsomolskaya"
  ],
  "_links": {
    "self": "/api/v1/countries/RUS/regions/1/subway",
    "region": "/api/v1/countries/RUS/regions/1"
  }
}
```

Получение информации о метро в регионе

GET https://{hostname}/api/v1/countries/{countryIsoCode}/regions/{regionId}/subway

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
countryIsoCode	string	Трехбуквенный ISO-код страны	Обязательное поле
regionId	integer	Идентификатор региона	Обязательное поле

Описание ответа Объект с информацией о метро в регионе

Поле	Тип	Описание	Дополнительно
Stations	Коллекция элементов типа String	Список станций метро	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Справочник точек

Список точек

Пример ответа:

```
{
  "page": 1,
  "totalPages": 5,
  "totalCount": 50,
  "items": [
    {
      "id": 1,
      "agentId": 2,
      "name": "ОАО КБ Unistream",
      "shortName": "Unistream",
      "address": "g.Moscow, ul.Verhniya Maslovka, k.20 str.2",
      "addressNotes": "Main office",
      "location": {
        "lat": 150.9876451,
        "lon": -45.2512365,
        "zoom": 5
      },
      "phone": "+7080123456789",
      "country": {
        "countryCode": "RUS",
        "name": "Russia",
        "_links": {
          "self": "/api/v1/countries/RUS"
        }
      },
      "region": {
        "id": 1,
        "countryCode": "RUS",
        "name": "g.Moscow",
        "currencies": null,
        "subway": false,
        "_links": {
          "self": "/api/v1/countries/RUS/regions/1",
          "country": "/api/v1/countries/RUS"
        }
      },
      "workingSchedule": {
        "openNow": true,
        "timeZoneId": "Europe/Moscow",
        "workingHours": [
          {
            "weekday": "Mon",
            "date": null,
            "open": "0830",
            "close": "1900"
          }
        ],
        "workingHoursNotes": ""
      },
      "metro": [
        {
          "id": 111,
          "name": "Domodedovskaya"
        }
      ],
      "currencies": [
        "RUR",
        "USD",
        "EUR"
      ],
      "operations": [
        {
          "type": "AcceptTransfer",
          "name": "Can accept non-address transfers"
        },
        {
          "type": "PayoutTransfer",
          "name": "Can withdraw non-address transfers"
        }
      ],
      "notes": "{}",
      "activity": null,
      "_links": {
        "self": "/api/v1/poses/1"
      }
    },
    {
      "id": 2,
      "agentId": 2,
      "name": "ОАО Testovoye",
      "shortName": "Test",
      "address": "g.Moscow",
      "addressNotes": "Opercash",
      "location": {
```

```
    "lat": 10.986901,
    "lon": -89.125,
    "zoom": 5
  },
  "phone": null,
  "country": {
    "countryCode": "RUS",
    "name": "Russia",
    "_links": {
      "self": "/api/v1/countries/RUS"
    }
  },
  "region": null,
  "workingSchedule": {
    "openNow": true,
    "timeZoneId": "Europe/Moscow",
    "workingHours": [
      {
        "weekday": "Mon",
        "date": null,
        "open": "0830",
        "close": "1900"
      }
    ],
    "workingHoursNotes": ""
  },
  "metro": [],
  "currencies": [
    "USD"
  ],
  "operations": [
    {
      "type": "SWIFT",
      "name": "SWIFT"
    }
  ],
  "notes": "-)",
  "activity": null,
  "_links": {
    "self": "/api/v1/poses/2"
  }
}
],
"_links": {
  "self": "/api/v1/poses?filter.agentId=2"
}
}
```

Получение списка точек с возможностью фильтрации и разбивкой на страницы

GET https://{hostname}/api/v1/poses?filter.term={filter.term}&filter.country={filter.country}&filter.region={filter.region}&filter.agent={filter.agent}&filter.operation={filter.operation}&filter.currency={filter.currency}&filter.activity={filter.activity}&page={page}&rows={rows}

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
filter.term	string	Произвольная строка для фильтрации списка	Обязательное поле
filter.country	string	Идентификатор или трехбуквенный ISO-код страны	Обязательное поле
filter.region	integer	Идентификатор региона	Обязательное поле
filter.agent	integer	Идентификатор агента	Обязательное поле
filter.operation	string	Список поддерживаемых операций через запятую	Обязательное поле
filter.currency	string	Список поддерживаемых валют через запятую	Обязательное поле
filter.activity	boolean	Активность точки в последние 7 дней(исключая переводы)	Обязательное поле
page	integer	Номер страницы	Значение по умолчанию: 1
rows	integer	Количество элементов на странице	Значение по умолчанию: 50

Описание ответа Запрошенная страница списка контрагентов

Поле	Тип	Описание	Дополнительно
Stations	Коллекция элементов типа String	Список станций метро	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Поиск точек

Поиск точек в GEO области заданного радиуса

Пример ответа:

```
{
  "totalCount": 0,
  "items": [
    {
      "id": 1,
      "agentId": 2,
      "name": "ОАО KB Unistream",
    }
  ]
}
```

```
{
  "shortName": "Unistream",
  "address": "g.Moscow, ul.Verhniya Maslovka, k.20 str.2",
  "addressNotes": "Main office",
  "location": {
    "lat": 150.9876451,
    "lon": -45.2512365,
    "zoom": 5
  },
  "phone": "+7080123456789",
  "country": {
    "countryCode": "RUS",
    "name": "Russia",
    "_links": {
      "self": "/api/v1/countries/RUS"
    }
  },
  "region": {
    "id": 1,
    "countryCode": "RUS",
    "name": "g.Moscow",
    "currencies": null,
    "subway": false,
    "_links": {
      "self": "/api/v1/countries/RUS/regions/1",
      "country": "/api/v1/countries/RUS"
    }
  },
  "workingSchedule": {
    "openNow": true,
    "timeZoneId": "Europe/Moscow",
    "workingHours": [
      {
        "weekday": "Mon",
        "date": null,
        "open": "0030",
        "close": "1900"
      },
      {
        "weekday": null,
        "date": "20141231",
        "open": "0030",
        "close": "1900"
      }
    ],
    "workingHoursNotes": ""
  },
  "metro": [
    {
      "id": 111,
      "name": "Domodedovskaya"
    }
  ],
  "currencies": [
    "RUR",
    "USD"
  ],
  "operations": [
    {
      "type": "CashToCard",
      "name": "Cash to card payment"
    },
    {
      "type": "SwiftTransfer",
      "name": "SWIFT"
    }
  ],
  "notes": ":",
  "activity": null,
  "_links": {
    "self": "/api/v1/poses/1"
  }
}

```

GET https://{hostname}/api/v1/poses/search?location={location}&radius={radius}&maxResults={maxResults}&filter.term={filter.term}&filter.country={filter.country}&filter.region={filter.region}&filter.agent={filter.agent}&filter.operation={filter.operation}&filter.currency={filter.currency}

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
location	string	Опорная точка для ГЕО-локации (центр области поиска) в формате "широта,долгота", например -45.7890,110.25	
radius	unsigned integer	Радиус поиска в метрах	Значение по умолчанию: 5000
maxResults	integer	Максимальное количество возвращаемых точек	Значение по умолчанию: 100
filter.term	string	Произвольная строка для фильтрации списка	
filter.country	string	Идентификатор или трехбуквенный ISO-код страны	
filter.region	integer	Идентификатор региона	
filter.agent	integer	Идентификатор агента	
filter.operation	string	Список поддерживаемых операций через запятую	
filter.currency	string	Список поддерживаемых валют через запятую	

Описание ответа Список найденных точек

Поле	Тип	Описание	Дополнительно
TotalCount	integer	Список станций метро	Общее количество точек, соответствующих условию запроса
Items	Коллекция элементов типа PointOfServiceModel	Список найденный точек	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Точка по идентификатору

Получение информации о точке по её идентификатору

Пример ответа:

```
{
  "id": 1,
  "agentId": 2,
  "name": "DAO KB Unistream",
  "shortName": "Unistream",
  "address": "g.Moscow, ul.Verhniya Maslovka, k.20 str.2",
  "addressNotes": "Main office",
  "location": {
    "lat": 150.9876451,
    "lon": -45.2512365,
    "zoom": 5
  },
  "phone": "+7000123456789",
  "country": {
    "countryCode": "RUS",
    "name": "Russia",
    "_links": {
      "self": "/api/v1/countries/RUS"
    }
  },
  "region": {
    "id": 1,
    "countryCode": "RUS",
    "name": "g.Moscow",
    "currencies": null,
    "subway": false,
    "_links": {
      "self": "/api/v1/countries/RUS/regions/1",
      "country": "/api/v1/countries/RUS"
    }
  },
  "workingSchedule": {
    "openNow": true,
    "timeZoneId": "Europe/Moscow",
    "workingHours": [
      {
        "weekday": "Mon",
        "date": null,
        "open": "0830",
        "close": "1900"
      },
      {
        "weekday": "Tue",
        "date": null,
        "open": "0830",
        "close": "1900"
      },
      {
        "weekday": "Wed",
        "date": null,
        "open": "0830",
        "close": "1900"
      },
      {
        "weekday": "Thu",
        "date": null,
        "open": "0830",
        "close": "1900"
      },
      {
        "weekday": "Fri",
        "date": null,
        "open": "0830",
        "close": "1900"
      },
      {
        "weekday": "Sat",
        "date": null,
        "open": "1000",
        "close": "1800"
      },
      {
        "weekday": "Sun",
        "date": null,
        "open": "0830",
        "close": "1900"
      },
      {
        "weekday": null,
        "date": "20141231",
        "open": "0830",
        "close": "1900"
      }
    ],
    "workingHoursNotes": ""
  },
  "metro": [
    {
      "id": 111,
```

```
    "name": "Domodedovskaya"
  },
  ],
  "currencies": [
    "RUR",
    "USD",
    "EUR"
  ],
  "operations": [
    {
      "type": "AcceptTransfer",
      "name": "Can accept non-address transfers"
    },
    {
      "type": "PayoutTransfer",
      "name": "Can withdraw non-address transfers"
    },
    {
      "type": "CashToCard",
      "name": "Cash to card payment"
    },
    {
      "type": "SwiftTransfer",
      "name": "SWIFT"
    }
  ],
  "notes": "{}",
  "activity": null,
  "_links": {
    "self": "/api/v1/poses/1"
  }
}
```

GET https://{hostname}/api/v1/poses/{id}

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
id	string	Идентификатор точки	Обязательное поле
radius	unsigned integer	Радиус поиска в метрах	Значение по умолчанию: 5000
maxResults	integer	Максимальное количество возвращаемых точек	Значение по умолчанию: 100
filter.term	string	Произвольная строка для фильтрации списка	
filter.country	string	Идентификатор или трехбуквенный ISO-код страны	
filter.region	integer	Идентификатор региона	
filter.agent	integer	Идентификатор агента	
filter.operation	string	Список поддерживаемых операций через запятую	
filter.currency	string	Список поддерживаемых валют через запятую	

Описание ответа

Объект с информацией о точке

Поле	Тип	Описание	Дополнительно
Id	integer	Идентификатор точки	
AgentId	integer	Идентификатор агента	
Name	string	Наименование точки	
ShortName	string	Короткое наименование точки	
Address	string	Адрес точки	
AddressNotes	string	Дополнительная информация по адресу	
Location	ГЕО координата	Расположение точки	
Phone	string	Телефон точки	
Country	Информация о стране	Страна точки	
Region	Информация о регионе	Регион точки	
WorkingSchedule	Информация о расписании работы точки	Расписание работы точки	
Metro	Коллекция элементов типа SubwayStationModel	Информация о ближайших станциях метро	
Currencies	Коллекция элементов типа String	Список поддерживаемых валют	
Operations	Коллекция элементов типа OperationModel	Список поддерживаемых операций	
Notes	string	Дополнительная информация по точке	
Activity	Информация об активности точки	Информация об активности точки	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Справочник контрагентов

Список контрагентов

Получение списка контрагентов с возможностью фильтрации и разбивкой на страницы

Пример ответа:

```
{
  "page": 1,
  "totalPages": 5,
  "totalCount": 50,
  "items": [
    {
      "id": 1,
      "name": "ZAO Unistream",
      "_links": {
        "self": "/api/v1/agents/1"
      }
    },
    {
      "id": 2,
      "name": "ООО Testovoe 1",
      "_links": {
        "self": "/api/v1/agents/2"
      }
    },
    {
      "id": 3,
      "name": "ООО Testovoe 2",
      "_links": {
        "self": "/api/v1/agents/3"
      }
    }
  ],
  "_links": {
    "self": "/api/v1/agents"
  }
}
```

GET https://{hostname}/api/v1/agents?filter.term={filter.term}&filter.country={filter.country}&page={page}&rows={rows}

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
filter.term	string	Произвольная строка для фильтрации списка	
filter.country	string	Идентификатор или трехбуквенный ISO-код страны	
page	integer	Номер страницы	Значение по умолчанию: 1
rows	integer	Количество элементов на странице	Значение по умолчанию: 50

Описание ответа

Запрошенная страница списка контрагентов

Поле	Тип	Описание	Дополнительно
Page	integer	Номер страницы	
TotalPages	integer	Общее количество страниц	
TotalCount	integer	Общее количество элементов	
Items	Коллекция элементов типа AgentModel	Список элементов	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Контрагент по идентификатору

Получение информации о контрагенте по его идентификатору

Пример ответа:

```
{
  "id": 1,
  "name": "ZAO Unistream",
  "_links": {
    "self": "api/v1/agents/1"
  }
}
```

GET https://{hostname}/api/v1/agents/{id}

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
id	integer	Идентификатор контрагента	Обязательное поле

Описание ответа Объект с информацией о контрагенте

Поле	Тип	Описание	Дополнительно
Id	integer	Идентификатор агента	
Name	string	Имя агента	
_links	Коллекция ссылок	Список ссылок, связанных с моделью	

Переводы

Получение информации о переводах

Информация о переводе

Получение статуса перевода по контрольному номеру перевода (КНП)

Пример ответа:

```
{
  "code": "140856100000",
  "status": "Accepted"
}
```

GET https://{hostname}/api/v1/transfer/{code}

Параметры строки запроса

Имя	Тип	Описание	Дополнительно
code	string	Код перевода	Обязательное поле
filter.country	string	Идентификатор или трехбуквенный ISO-код страны	
page	integer	Номер страницы	Значение по умолчанию: 1
rows	integer	Количество элементов на странице	Значение по умолчанию: 50

Описание ответа

Структура данных с информацией о переводе

Поле	Тип	Описание	Дополнительно
Code	string	Контрольный номер перевода (КНП)	
Status	Информация о статусе перевода	Статус перевода	

В зависимости от статуса перевода могут вернуться следующие значения в поле "Status":

- 1.Accepted (принят)
- 2.CalledBack (отозван)
- 3.Cancelled (аннулирован)
- 4.WithdrawConfirmed (выдан или аннулирован через выдачу методом PayOutRevokedTransfer)

Расчет комиссии

Производит расчет комиссии для заданных параметров перевода или платежа.

Пример запроса на расчёт комиссии по безадресному переводу:

```
{
  "senderBankId": 359293,
  "recipientBankId": null,
  "countryCode": "UZB",
  "acceptedCurrency": "RUB",
  "withdrawCurrency": "USD",
  "amount": 1500.55,
  "operationType": null,
  "amountType": null,
  "values": null
}
```

Пример ответа:

```
{
  "message": null,
  "fees": [
    {
      "name": null,
      "acceptedAmount": 1500.0,
      "acceptedCurrency": "RUB",
      "withdrawAmount": 1234.33,
      "withdrawCurrency": "RUB",
      "rate": 1.0,
      "acceptedTotalFee": 265.67
    }
  ]
}
```

GET https://{hostname}/api/v1/transfer/calculate

Описание

Метод рассчитывает комиссию по переводу или платежу. Обратите внимание на зависимость полей, к примеру для расчета комиссии по адресному переводу необходимо заполнить поле RecipientBankId или CountryCode если перевод безадресный. Для расчета комиссии по платежам необходимо заполнить поле OperationType (указать тип платежа, например Patent) и оставить пустыми поля RecipientBankId и CountryCode.

Параметры запроса


Имя	Тип	Описание	Дополнительно
SenderBankId	integer	Идентификатор банка-отправителя	
RecipientBankId	integer	Идентификатор банка-получателя для адресных переводов	
CountryCode	string	ISO-код страны выдачи	Используется только при расчёте комиссий по переводам
AcceptedCurrency	string	ISO-код валюты приема	
WithdrawCurrency	string	ISO-код валюты выдачи	
Amount	decimal number	Сумма перевода (платежа)	
OperationType	string	Тип операции для расчета комиссий платежей	Используется только по расчётам комиссии по платежам
AmountType	string	Тип суммы. Доступно только для отправки перевода. Если в значение "AmountType" указываем параметр "AcceptedAmount", то сумма будет к отправке. Значение по-умолчанию "WithdrawAmount" - сумма к выдаче.	
Values	Коллекция элементов типа Object	Дополнительные значения для общего типа операций	

Описание ответа

Информация о комиссии

Поле	Тип	Описание	Дополнительно
Message	string	Дополнительная информация	
Fees	Коллекция элементов типа CalculateFee	Массив информации по рассчитанной комиссии	

FAQ

Часто задаваемые вопросы (раздел в режиме обновления) 

Общие вопросы

1.Как получить учетные данные и сертификат для подтверждения операции ?

- заполняется заявка на подключение, и направляется бизнес-куратору в отсканированном виде.

2.В API взаиморасчёты будут идти также, как и в Global2/ПОЮ ?

- да, если не будут внесены дополнительные изменения согласно договоренностям обеих сторон.

3.Где посмотреть код ошибки ?

- кодов ошибок нет, информация об ошибке возвращается при опросе операции в поле "statusMessage".

4.Мы закончили реализацию протокола, как начать работу на продуктивном стенде ?

- заполняете акт тестирования, отправляете скан и электронную версию документа Вашему менеджеру Юнистрим на проверку. При отправке данного акта тестирования, также приложите к письму архив со скриншотами каждого шага всех реализованных операций.

Безопасность, авторизация и подтверждение операции

1. Зачем нужен сертификат ?

- сертификат нужен для подтверждения операции, без сертификата операцию нельзя подтвердить.

2. Сколько сертификатов нужно для работы ?

- зависит от реализации, Вы можете установить центр сертификации, подчиненный корневому центру (АО КБ ЮНИСТРИМ) и выдавать сертификаты для каждого операциониста, либо установить один сертификат на сервер (выдаётся АО КБ ЮНИСТРИМ).

3. В тестовой среде и на продуктивном стенде, сертификаты для подтверждения идентичные ?

- сертификаты для теста и для продуктивного стенда выпускаются отдельно и не могут быть взаимозаменяемыми.

4. Какие авторизационные заголовки необходимо передавать для отправки запроса ?

- Пример и описание авторизации здесь, для отправки запросов в `infoApi` авторизационные заголовки не требуются.

5. Можно ли все операции создавать от одного POSID (офиса) ?

- нет, каждому филиалу (офису) будет назначен определённый POSID, который нужно передавать в авторизационных заголовках в зависимости от филиала.

6. Как подтверждать операцию ?

- в качестве подтверждения передается массив байт в формате base64, сформированный в результате подписи документа, ссылка на документ указана в поле "Operation.SignDocument", формирующаяся на статусе операции – `Accepted`.

7. Сколько времени операция ждёт подтверждения ?

- на тестовом стенде 30 минут, продуктивный стенд 60 минут.

8. Что будет, если операцию не подтверждать ?

- операция автоматически отменится, в сроки указанные выше и перейдет в статус `Canceled`.

Клиенты

1. Что нельзя менять в досье клиента ?

- самостоятельно у клиента нельзя менять тип документа, серию и номер документа (касается паспортов всех типов), ФИО, пол, дату рождения - эти данные можно изменить по заявлению от Клиента через заявку в SD Unistream .

2. Нужно ли заполнять номер телефона клиента при его создании ?

- Да, при создании клиента, будь то отправитель или получатель, необходимо указывать корректный номер телефона, с указанием кода страны. Например, для России формат будет 79096101726.

3. Можно ли удалять недействительные документы ?

- только через заявку в "SD Unistream" с прикрепленным заявлением от Клиента.

4. Можно ли добавлять клиенту несколько документов ?

- да.

Операции

1. Где взять GUID операции ?

- при создании новой операции, Партнер генерирует GUID на своей стороне.

2. Как узнать КНП (контрольный номер перевода), после создания операции «Отправка перевода» ?

- номер перевода возвращается при опросе операции в поле "clearingRef"; Клиент узнает КНП из распечатанного заявления, также клиенту придет СМС от Юнистрим (Ваш перевод №123456789012 отправлен/выдан)

3. Можно ли самостоятельно генерировать КНП ?

- нет.

4. Существует ли предварительный метод по поиску перевода, где можно узнать ФИО получателя, сумму и валюту ?

- предварительно можно узнать только статус перевода, клиентские данные возвращаются непосредственно при выдаче перевода

5. Если мы выдали перевод, его можно отменить ?

- на уровне API нет таких методов.

6. Как при выдаче узнать ФИО получателя, которые указал отправитель при отправке перевода ?

- при опросе операции в поле, `ExpectedRecipient` - предполагаемый получатель (как указал отправитель) и `RealRecipient` - фактический получатель, указанный Вами при запросе на выдачу перевода.

7. Какие документы необходимы клиенту для отправки перевода ?

- зависит от суммы, резидентства и страны. Узнать схему идентификации по стране и возможные документы для создания клиента можно в разделе идентификация.

8. При создании операции необходимо передавать паспортные данные получателя или отправителя ?

- отправителя, если это отправка перевода и получателя, если это выдача перевода.

9.Какой статус операции считается окончательно-успешным ?

- *completed.*

10.Как часто опрашивать операцию для получения необходимого статуса ?

- *операцию рекомендуется опрашивать каждые 3-4 сек.*

11.Ошибка Current client level is Unknown.Required level is Simple ?

- *не хватает уровня идентификации клиента для осуществления операции, отсутствуют необходимые документы. Узнать подробнее об идентификации клиента можно [здесь](#)*

12.Как будет происходить расчёт комиссий ?

- *комиссии настраиваются и рассчитываются на стороне АО КБ Юнистрим, если иное не предусмотрено договором.*

13.Где смотреть распределение комиссий ?

- *после создания операции (начиная со статуса Accepted), при ее опросе в ответе необходимо смотреть раздел «clearingContext»:*

agentFee: - Вознаграждение партнера за осуществление данной операции (выдача/отправка)

agentFeeCurrency": - валюта вознаграждения Партнера

amount: - сумма операции

clearingRef: - КНП (контрольный номер перевода)

currency: - валюты суммы операции

totalFee: - Суммарная клиентская комиссия (тариф)

totalFeeCurrency: валюта суммарной клиентской комиссии (тарифа)

Ошибки

i Для индикации результата выполнения запроса (успех или ошибка) API использует коды статуса HTTP протокола. В общем случае, коды из диапазона 2xx означают успешный запрос. Коды из диапазона 4xxx - ошибку запроса вызванную некорректными входными данными (например, неправильные параметры запроса). Коды из диапазона 5xx означают ошибку на стороне сервера. Дополнительная информация об ошибке может быть передана в теле запроса.

Код ошибки	Значение
400	Bad Request -- The request cannot be fulfilled due to bad syntax.
401	Unauthorized -- Your API key is wrong. Not authorized to access resource without authentication.
403	Forbidden -- Forbidden to access resource.
404	Not Found -- The specified kitten could not be found.
405	Method Not Allowed -- You tried to access a resource with an invalid method.
406	Not Acceptable -- You requested a format that isn't supported.
410	Gone -- The resource requested has been removed from the server.
429	Too Many Requests -- Incoming request rate is to high, slow down.
430	Identification Required -- Need client identification for this request.
500	Internal Server Error -- Some error occured on the server-side.
503	Service Unavailable -- We're temporarily offline for maintenance. Please try again later.