|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| ЕПВВ |
|  |
|  |
| Передача ЭС посредством универсального REST v.1.2.3 |
|  |
|  |
| Количество листов 27 |

**Оглавление**

[Аннотация 3](#_Toc536696551)

[Глоссарий 4](#_Toc536696552)

[1 Общие сведения 5](#_Toc536696553)

[2 СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ УНИВЕРСАЛЬНОГО REST-СЕРВИСА 6](#_Toc536696554)

[2.1 Определение задачи для отправки 6](#_Toc536696555)

[2.2 Отправка сообщения 6](#_Toc536696556)

[2.3 Поиск сообщения 7](#_Toc536696557)

[2.4 Получение актуального статуса сообщения 7](#_Toc536696558)

[3 ТЕХНИЧЕСКОЕ ОПИСАНИЕ ВЗАИМОДЕЙСТВИЯ С УНИВЕРСАЛЬНЫМ REST-СЕРВИСОМ 9](#_Toc536696559)

[3.1 Параметры подключения к внешнему порталу с использованием универсального REST-сервиса 9](#_Toc536696560)

[3.2 Формат пакета с сообщением 9](#_Toc536696561)

[3.3 Авторизация 9](#_Toc536696562)

[3.4 Общие правила оформления сообщений, передаваемых по протоколу HTTP 1.1 9](#_Toc536696563)

[3.5 Передача файлов 9](#_Toc536696564)

[3.5.1 Для создания нового сообщения используется метод POST: 10](#_Toc536696565)

[3.5.2 Для создания сессии отправки HTTP используется метод POST: 13](#_Toc536696566)

[3.5.3 Для отправки файла по HTTP используется метод PUT: 14](#_Toc536696567)

[3.5.4 Для отправки сообщения используется метод POST: 16](#_Toc536696568)

[3.6 Получение УФР сообщений, файлов и информации 16](#_Toc536696569)

[3.6.1 Для получения всех сообщений с учетом необязательного фильтра (не более 100 сообщений) используется метод GET: 16](#_Toc536696570)

[3.6.2 Для получения данных о конкретном сообщении используется метод GET: 20](#_Toc536696571)

[3.6.3 Для скачивания конкретного сообщения используется метод GET: 23](#_Toc536696572)

[3.6.4 Для получения данных о конкретном файле используется метод GET: 23](#_Toc536696573)

[3.6.5 Для скачивания конкретного файла из конкретного сообщения используется метод GET: 24](#_Toc536696574)

[3.7 Удаление сообщений 25](#_Toc536696575)

[3.7.1 Для удаления конкретного сообщения используется метод DELETE: 25](#_Toc536696576)

[3.7.2 Для удаления конкретного файла или отмены сессии отправки используется метод DELETE: 26](#_Toc536696577)

[3.8 Получение справочной информации 26](#_Toc536696578)

[3.8.1 Для получения справочника задач используется метод GET: 26](#_Toc536696579)

[3.8.2 Для получения информации о своём профиле используется метод GET: 26](#_Toc536696580)

[3.8.3 Для получения информации о технических оповещениях используется метод GET: 27](#_Toc536696581)

[3.8.4 Для получения списка справочников используется метод GET: 28](#_Toc536696582)

[3.8.5 Для скачивания конкретного справочника используется метод GET: 28](#_Toc536696583)

Аннотация

Настоящий документ предназначен для описания технических аспектов информационного взаимодействия между ЕПВВ и системами участников информационного обмена с Банком России.

В документе описываются способы передачи информации, протоколы взаимодействия, структура входящих и исходящих сообщений, примеры сообщений, используемых при обмене.

Глоссарий

|  |  |
| --- | --- |
| Термин (определение) | Трактовка термина (определения) |
| HTTP | Hyper Text Transfer Protocol – Протокол передачи гипертекста |
| REST | Representational State Transfer – Передача состояния представления |
| БД | База данных |
| БР | Банк России |
| Внешний абонент | Пользователи ЕПВВ – участники финансового рынка |
| Внутренний абонент | Пользователи ЕПВВ – сотрудники Банка России, а также сотрудники региональных подразделений |
| ЕПВВ | Единая платформа внешнего взаимодействия |
| КО | Кредитная организация |
| КГРКО | Книга государственной регистрации кредитных организаций |
| ЛК | Личный кабинет |
| УФР | Участники финансового рынка |
| ЭС | Электронное сообщение |

1. Общие сведения

Полное наименование автоматизированной системы - Единая платформа внешнего взаимодействия Банка России.

Краткое наименование автоматизированной системы – ЕПВВ.

1. СЦЕНАРИИ ИСПОЛЬЗОВАНИЯ УНИВЕРСАЛЬНОГО REST-СЕРВИСА
	1. Определение задачи для отправки

Для определения списка доступных задач можно использовать метод, описанный в разделе 3.8.1. Из указанного списка необходимо выбрать задачу на основе:

* анализа описания;
* требований Банка России;
* руководства Пользователя ЕПВВ.

В случае, если необходимая задача отсутствует в списке, рекомендуются 2 действия:

* Вызвать метод из раздела 3.8.2, убедиться, что в списке видов деятельностей есть необходимый (тот, для которого реализуется взаимодействие с Банком России).
* Зайти в Личный Кабинет через web-интерфейс и проверить наличие задачи в разделе «Представление отчётности».
	1. Отправка сообщения

Любая отправка сообщения происходит из 4 этапов:



* На этапе создания сообщения вызывается метод POST /messages (3.5.1), в котором указывается базовая информация о сообщении и список файлов (не сами файлы). В ответ универсальный REST создаёт новое сообщение (в web-интерфейсе оно отображается как «Черновик»), возвращает зарезервированные ID для сообщения и для всех файлов.
* На этапе создания сессии отправки файла вызывается метод POST /messages/{msgId}/files/{fileId}/createUploadSession (3.5.2), где msgId и fileId – сгенерированы универсальным REST и переданы клиенту на предыдущем этапе. В ответ универсальный REST возвращает полную ссылку, по которой необходимо осуществлять отправку.
* На этапе отправки клиент методом PUT /messages/{msgId}/files/{fileId} (3.5.3) отправляет части (чанки) сообщений. Передаваемый диапазон байт регулируется самим клиентом (позволяет передать одной частью или раздробить на несколько частей).
* После того как все файлы сообщения успешно загружены на сервер, клиент вызывает метод POST /messages/{msgId} (3.5.4), которым финализирует отправку. Сообщение переходит в статус «Отправлено» и передаётся на обработку в Банк России.
	1. Поиск сообщения

Для получения информации о сообщении используется метод GET /messages (3.6.1). Есть возможность передачи в качестве параметров следующей фильтрующей информации:

* Фильтр по дате сообщения (с какой даты времени и по какую дату и время было отправлено сообщение)
* Фильтр по размеру сообщения (это может потребоваться для удаления «самых больших сообщений» для очистки квоты)
* Фильтр по типу сообщения (входящие, исходящие, новые, отвеченные)

Так же возможна реализация с поиском сообщения на стороне клиента. В этом случае по очереди с сервера запрашиваются все сообщения (без указания дополнительных фильтров), которые сохраняются у клиента дальнейший поиск информации осуществляется уже на локальном сервере/АРМе. Этот способ рекомендуется для клиентов, имеющих значительный объём взаимодействия с ЦБ (получение/передача более *30(значение уточняется)* сообщений в день), чтобы предотвратить излишнюю загрузку серверов ЕПВВ обработкой и фильтрацией запросов на стороне сервера.

* 1. Получение актуального статуса сообщения

После того, как нужное сообщение было найдено, статус сообщения можно определить двумя способами:

1. Вызвать метод GET /messages/msgId (3.6.2), определять по значению поля Status самого сообщения.

|  |  |  |
| --- | --- | --- |
| Статус | Статус в ЛК | Описание |
| draft | Черновик | Сообщение с данным статусом создано, но ещё не отправлено. |
| sent | Отправлено/Отправлен ответ | Сообщение получено сервером |
| delivered | Загружено | Сообщение прошло первоначальную проверку |
| processing | Принято в обработку | Сообщение передано во внутреннюю систему ЦБ |
| error | Ошибка | При обработке сообщения возникла ошибка |
| registered | Зарегистрировано | Сообщение зарегистрировано |
| rejected | Отклонено | Сообщение успешно дошло до получателя, но было отклонено |
| new | Новое | Только для входящих сообщений. Сообщение в данном статусе ещё не почтено Пользователем УФР. |
| read | Прочитано | Только для входящих сообщений. Сообщение в данном статусе почтено Пользователем УФР. |
| replied | Отклонено | Только для входящих сообщений. На сообщение в данном статусе направлен ответ. |

1. Вызвать метод GET /messages/msgId (3.6.2), определять по наличию/отсутствию нужных квитанций в блоке Receipts. Тип квитанции определять по значению поле Status внутри массива значений Receipts. Возможные значения поля status:

|  |  |  |
| --- | --- | --- |
| Статус | Статус в ЛК | Описание |
| draft | Черновик | Сообщение с данным статусом создано, но ещё не отправлено. |
| sent | Отправлено/Отправлен ответ | Сообщение получено сервером |
| delivered | Загружено | Сообщение прошло первоначальную проверкуВ рамках 4600-У, квитанцией о загрузке считается эта квитанция. |
| processing | Принято в обработку | Сообщение передано во внутреннюю систему ЦБ |
| error | Ошибка | При обработке сообщения возникла ошибка |
| registered | Зарегистрировано | Сообщение зарегистрированоВ рамках 4600-У, квитанцией о регистрации считается эта квитанция. |
| rejected | Отклонено | Сообщение успешно дошло до получателя, но было отклонено |

1. ТЕХНИЧЕСКОЕ ОПИСАНИЕ ВЗАИМОДЕЙСТВИЯ С УНИВЕРСАЛЬНЫМ REST-СЕРВИСОМ
	1. Параметры подключения к внешнему порталу с использованием универсального REST-сервиса

Для подключения к Порталу "Биврёст" посредством универсального REST-сервиса необходимо в Клиентском ПО отправителя указать следующую ссылку: https://portal5.cbr.ru/back/rapi2/\*

Вместо звездочки необходимо указать методы, описанные в п. 2.1.5, 2.1.6 и 2.1.7.

* 1. Формат пакета с сообщением

Формат отправляемого пакета определяется в документе «ЕПВВ. Технические условия внешнего обмена».

* 1. Авторизация

Для осуществления информационного взаимодействия прежде всего необходимо пройти активацию Пользователя в ЛК ЕПВВ. После прохождения активации и изменении логина и пароля, их можно использовать для авторизации в универсальный REST. Авторизация осуществляется с помощью передачи HTTP заголовка «Authorization». Тип авторизации – «Basic».

В качестве логина и пароля передаются учетные записи, созданные во время активации ЛК.

* 1. Общие правила оформления сообщений, передаваемых по протоколу HTTP 1.1

СЛЕДУЕТ задавать абсолютный URL в параметрах методов POST и GET.

Поле Accept СЛЕДУЕТ заполнять значениями «application/json».

Поле User-Agent СЛЕДУЕТ заполнять строковым значением, идентифицирующим ПО, которое используется для взаимодействия с Внешним порталом.

Слово «СЛЕДУЕТ» используется для указания того, что данное требование спецификации должно быть обеспечено, если этому не препятствуют серьезные причины. Данное правило соответствует требованию к реализации протокола, определенное в RFC 2119.

* 1. Передача файлов

Отправка сообщений на стороне ЕПВВ осуществляется с использованием универсального REST-сервиса. Для сообщений потока предоставления информации операционного дня инициативным действием является отправка сообщения с помощью протокола FASP, предварительно создавать сообщение через универсальный REST запрещено.

* + 1. Для создания нового сообщения используется метод POST:

POST: \*/messages

В котором передается следующие параметры:

REQUEST

{

"Task": "string"

"CorrelationId": "string"

 "GroupId": "string"

 "Title": "string"

 "Text": "string"

 "Files": [

 {

 "Name": "string"

 "Encrypted": "boolean"

 "SignedFile": "string"

 "RepositoryType": "string"

 "Size": "integer"

}

]

"Receivers": [

{

 "Inn" : "integer"

 "Ogrn" : "integer"

 "Bik" : "integer"

 "Email" : "integer"

 "RegNum" : "integer"

 "DivisionCode" : "integer"

 "Activity" : "string"

 }

]

}

Где:

Task – код задачи (по справочнику задач в формате «Zadacha\_\*»);

CorrelationId - идентификатор корреляции сообщения (необязательно, указывается для формирования ответного сообщения для потоков, поддерживаемых данную функциональность);

GroupId –идентификатор группы сообщений (необязательно, указывается для передачи информации о том, что сообщение является частью группы сообщений для потоков, поддерживаемых данную функциональность);

Title – название сообщения, отображается в интерфейсе;

Text – текст сообщения, отображается в интерфейсе;

Files – файлы включенные в сообщение:

 Name – имя файла;

 Encrypted – признак зашифрованности файла;

 SignedFile – имя файла, подписью для которого является данный файл (заполняется только для файлов подписи \*.sig);

 RepositoryType – всегда указывается http;

Size – размер отправляемого файла.

Receivers – получатели сообщения (необязательно, указывается для потоков адресной рассылки);

Inn – индивидуальный номер налогоплательщика получателя;

Ogrn – основной государственный регистрационный номер получателя;

Bik – банковский идентификационный код получателя;

Email – адрес электронной почты получателя;

RegNum – регистрационный номер КО – получателя по КГРКО;

DivisionCode – номер филиала КО – получателя по КГРКО;

Activity – краткое наименование вида деятельности.

RESPONSE

HTTP 200 – Ok

[

{

 "Id": "string"

 "CorrelationId": "string"

 "GroupId": "string"

 "Type": "string"

 "Title": "string"

 "Text": "string"

 "CreationDate": "string"

 "UpdatedDate": "string"

 "Status": "string"

 "TaskName": "string"

 "RegNumber": "string"

 "TotalSize": "integer"

 "Files": [

 {

 "Id": "string"

 "Name": "string"

 "Encrypted": "boolean"

 "SignedFile": "integer"

 "Size": "integer"

 "RepositoryInfo" : [

 {

 "Path": "string"

 "Host": "string"

 "Port": "integer"

 "RepositoryType": "string"

 }

 ]

 }

 ]

 "Receipts": [

 {

 "ReceiveTime": "string"

 "StatusTime": "string"

 "Status": "string"

 "Message": "string"

 "Files": [

 {

 "Id": "string"

 "Name": "string"

 "Encrypted": "boolean"

 "SignedFile": "integer"

 "Size": "integer"

 "RepositoryInfo": [

 {

 "Path": "string"

 "Host": "string"

 "Port": "integer"

 "RepositoryType": "string"

 }

 ]

 }

 ]

 }

 ]

}

]

Где:

Id – уникальный идентификатор сообщения;

CorrelationId - идентификатор корреляции сообщения;

GroupId – идентификатор группы сообщений;

Type – тип сообщения исходящее (значение: outbox) или входящее (значение: inbox);

Title – название сообщения;

Text – текст сообщения;

CreationDate – дата создания сообщения (ГОСТ ISO 8601-2001 по маске «yyyy-MM-dd’T’HH:mm:ss’Z’»);

UpdatedDate – дата последнего изменения статуса сообщения (ГОСТ ISO 8601-2001 по маске «yyyy-MM-dd’T’HH:mm:ss’Z’»);

Status – статус сообщения (возможные значения и их описание находится в п.2.4);

TaskName – наименование задачи;

RegNumber – регистрационный номер;

TotalSize – общий размер сообщения в байтах;

Files – файлы включенные в сообщение:

 Id – уникальный идентификатор файла;

 Name – имя файла;

 Encrypted – признак зашифрованности файла;

 SignedFile – идентификатор файла, подписью для которого является данный файл;

 Size - общий размер файла в байтах;

 RepositoryInfo – информация о репозиториях (описание репозитория в котором расположен файл. Данная информация используется как для загрузки файла, так и при его выгрузке):

 Path – путь к файлу в репозитории;

 Host – IP адрес или имя узла репозитория;

 Port – порт для обращения к репозиторию;

 RepositoryType – тип репозитория (значения: aspera, http);

receipts – квитанции, полученные в ответ на сообщение:

 ReceiveTime – время получения квитанции;

 StatusTime – время из самой квитанции;

 Status – состояние обработки сообщения (возможные значения и их описание находится в п.2.4);

Message – дополнительная информация из квитанции;

Files – файлы, включенные в квитанцию.

HTTP 405 – Invalid input.

* + 1. Для создания сессии отправки HTTP используется метод POST:

POST: \*/messages/{msgId}/files/{fileId}/createUploadSession

В котором передается следующие параметры:

REQUEST

{

"MsgId": "string"

"FileId": "string"

}

Где:

MsgId – уникальный идентификатор сообщения, полученный в качестве ответа при вызове метода из 3.5.1;

FileId – уникальный идентификатор файла, полученный в качестве ответа при вызове метода из 3.5.1.

RESPONSE

HTTP 200 – Ok

{

 "UploadUrl": "string"

 "ExpirationDateTime": "string"

}

Где:

UploadUrl – путь для загрузки файла;

ExpirationDateTime – дата и время истечения сессии.

* + 1. Для отправки файла по HTTP используется метод PUT:

PUT: \*/messages/{msgId}/files/{fileId}

В котором передается следующий параметр:

REQUEST

{

"MsgId": "string"

"FileId": "string"

}

HEADER

Content-Length

Content-Range

BODY

Content

Где:

MsgId – уникальный идентификатор сообщения, полученный в качестве ответа при вызове метода из 3.5.1;

FileId – уникальный идентификатор файла, полученный в качестве ответа при вызове метода из 3.5.1;

Content-Length – размер чанка в байтах, передав размер в первый раз, для дальнейших отправок следует использовать тот же размер;

Content-Range - положение чанка и размер всего файла в байтах в виде ‘bytes 0-25/128’;

Сontent – массив байт.

RESPONSE

HTTP 201 – Created

{

"Id": "string"

 "Name": "string"

 "Encrypted": "boolean"

 "SignedFile": "integer"

 "Size": "integer"

 "RepositoryInfo": [

 {

 "Path": "string"

 "Host": "string"

 "Port": "integer"

 "RepositoryType": "string"

 }

 ]

}

Где:

Id – уникальный идентификатор файла;

Name – имя файла;

Encrypted – признак зашифрованности файла;

SignedFile – идентификатор файла, подписью для которого является данный файл;

Size - общий размер файла в байтах;

RepositoryInfo – информация о репозиториях (описание репозитория в котором расположен файл. Данная информация используется как для загрузки файла, так и при его выгрузке):

 Path – путь к файлу в репозитории;

 Host – IP адрес или имя узла репозитория;

 Port – порт для обращения к репозиторию;

 RepositoryType – тип репозитория (значения: aspera, http).

HTTP 202 – Accepted

{

"NextExpectedRange": "string"

}

Где:

nextExpectedRange – описывает следующий ожидаемый диапазон байтов.

* + 1. Для отправки сообщения используется метод POST:

POST: \*/messages/{msgId}

В котором передается следующие параметры:

REQUEST

{

"MsgId": "string"

}

Где:

MsgId – уникальный идентификатор сообщения.

RESPONSE

HTTP 200 – Ok;

HTTP 404 – Сообщение не найдено;

HTTP 406 – Сообщение не может быть отправлено (например, это полученное сообщение или сообщение уже было отправлено).

* 1. Получение УФР сообщений, файлов и информации

Получение сообщений и файлов на стороне УФР осуществляется с использованием универсального REST-сервиса.

* + 1. Для получения всех сообщений с учетом необязательного фильтра (не более 100 сообщений) используется метод GET:

GET: \*/messages

REQUEST

{

"Task": "string"

"MinDateTime": "string"

"MaxDateTime": "string"

"MinSize": "integer"

"MaxSize": "integer"

"Type": "string"

}

Где:

Task – наименование задачи (если параметр будет указан, то будут возвращены только сообщения полученные/отправленные в рамках указанной задачи);

MinDateTime – минимально возможная дата создания сообщения (если параметр будет указан, то будут возвращены только сообщения полученные/отправленные позднее указанной даты);

MaxDateTime – максимально возможная дата создания сообщения (если параметр будет указан, то будут возвращены только сообщения полученные/отправленные ранее указанной даты);

MinSize – минимально возможный размер сообщения в байтах (если параметр будет указан, то будут возвращены только сообщения больше указанного размера);

MaxSize – максимально возможный размер сообщения в байтах (если параметр будет указан, то будут возвращены только сообщения больше указанного размера);

Type – тип сообщения исходящее (значение: outbox), входящее (значение: inbox), новое не прочитанное (значение: new), прочитанное и отвеченное (значение: processed) (если параметр будет указан, то будут возвращены только сообщения соответствующего типа).

RESPONSE

HTTP 200 – Ok

[

{

 "Id": "string"

 "CorrelationId": "string"

 "GroupId": "string"

 "Type": "string"

 "Title": "string"

 "Text": "string"

 "CreationDate": "string"($DateTime)

 "UpdatedDate": "string"($DateTime)

 "Status": "string"

 "TaskName": "string"

 "RegNumber": "string"

 "TotalSize": "integer"

 "Sender": [

 {

 "Inn": "integer"

 "Ogrn": "integer"

 "Bik": "integer"

 "RegNum": "integer"

 "DevisonCode": "integer"

 }

 "Files": [

 {

 "Id": "string"

 "Name": "string"

 "Description": "string"

 "Encrypted": "boolean"

 "SignedFile": "integer"

 "Size": "integer"

 "RepositoryInfo": [

 {

 "Path": "string"

 "Host": "string"

 "Port": "integer"

 "RepositoryType": "string"

 }

 ]

 }

 ]

 "Receipts": [

 {

 "ReceiveTime": "string"

 "StatusTime": "string"

 "Status": "string"

 "Message": "string"

 "Files": [

 {

 "Id": "string"

 "Name": "string"

 "Encrypted": "boolean"

 "SignedFile": "integer"

 "Size": "integer"

 "RepositoryInfo": [

 {

 "Path": "string"

 "Host": "string"

 "Port": "integer"

 "RepositoryType": "string"

 }

 ]

 }

 ]

 }

 ]

}

]

Где:

Id – уникальный идентификатор сообщения;

CorrelationId - идентификатор корреляции сообщения;

GroupId –идентификатор группы сообщений;

Type – тип сообщения исходящее (значение: outbox) или входящее (значение: inbox);

Title – название сообщения;

Text – текст сообщения;

CreationDate – дата создания сообщения (ГОСТ ISO 8601-2001 по маске «yyyy-MM-dd’T’HH:mm:ss’Z’»);

UpdatedDate – дата последнего изменения статуса сообщения (ГОСТ ISO 8601-2001 по маске «yyyy-MM-dd’T’HH:mm:ss’Z’»);

Status – статус сообщения (возможные значения и их описание находится в п.2.4);

TaskName – наименование задачи;

RegNumber – регистрационный номер;

TotalSize – общий размер сообщения в байтах;

Sender – отправитель сообщения (необязательное поле, только для сообщений, отправляемых другими Пользователями):

 Inn – ИНН отправителя сообщения

 Ogrn – ОГРН отправителя сообщения

 Bik – БИК отправителя сообщения

 RegNum – рег. номер КО-отправителя сообщения по КГРКО

 DevisonCode – номер филиала КО-отправителя сообщения по КГРКО

Files – файлы включенные в сообщение:

 Id – уникальный идентификатор файла;

 Name – имя файла;

Description – описание файла (необязательное поле, для запросов и предписаний из Банка России содержит имя файла с расширением, однако может содержать запрещённые символы Windows);

 Encrypted – признак зашифрованности файла;

 SignedFile – идентификатор файла, подписью для которого является данный файл;

 Size - общий размер файла в байтах;

 RepositoryInfo – информация о репозиториях (описание репозитория в котором расположен файл. Данная информация используется как для загрузки файла, так и при его выгрузке):

 Path – путь к файлу в репозитории;

 Host – IP адрес или имя узла репозитория;

 Port – порт для обращения к репозиторию;

 RepositoryType – тип репозитория (значения: aspera, http);

Receipts – квитанции, полученные в ответ на сообщение:

 ReceiveTime – время получения квитанции;

 RtatusTime – время из самой квитанции;

 Status – состояние обработки сообщения (возможные значения и их описание находится в п.2.4);

Message – дополнительная информация из квитанции;

Files – файлы, включенные в квитанцию.

* + 1. Для получения данных о конкретном сообщении используется метод GET:

GET: \*/messages/{msgId}

REQUEST

{

"MsgId" : "string"

}

Где:

msgId – уникальный идентификатор сообщения.

RESPONSE

HTTP 200 – Ok

{

 "Id": "string"

 "CorrelationId": "string"

 "GroupId": "string"

 "Type": "string"

 "Title": "string"

 "Text": "string"

 "CreationDate": "string"

 "UpdatedDate": "string"

 "Status": "string"

 "TaskName": "string"

 "RegNumber": "string"

 "TotalSize": "integer"

 "Sender": [

 {

 "Inn": "integer"

 "Ogrn": "integer"

 "Bik": "integer"

 "RegNum": "integer"

 "DevisonCode": "integer"

 }

 "Files": [

 {

 "Id": "string"

 "Name": "string"

 "Description": "string"

 "Encrypted": "boolean"

 "SignedFile": "integer"

 "Size": "integer"

 "RepositoryInfo": [

 {

 "Path": "string"

 "Host": "string"

 "Port": "integer"

 "RepositoryType": "string"

 }

 ]

 }

 ]

 "Receipts": [

 {

 "ReceiveTime": "string"

 "StatusTime": "string"

 "Status": "string"

 "Message": "string"

 "Files": [

 {

 "Id": "string"

 "Name": "string"

 "Encrypted": "boolean"

 "SignedFile": "integer"

 "Size": "integer"

 "RepositoryInfo": [

 {

 "Path": "string"

 "Host": "string"

 "Port": "integer"

 "RepositoryType": "string"

 }

 ]

 }

 ]

 }

 ]

}

Где:

Id – уникальный идентификатор сообщения;

CorrelationId - идентификатор корреляции сообщения;

GroupId –идентификатор группы сообщений;

Type – тип сообщения исходящее (значение: outbox) или входящее (значение: inbox);

Title – название сообщения;

Text – текст сообщения;

CreationDate – дата создания сообщения (ГОСТ ISO 8601-2001 по маске «yyyy-MM-dd’T’HH:mm:ss’Z’»);

UpdatedDate – дата последнего изменения статуса сообщения (ГОСТ ISO 8601-2001 по маске «yyyy-MM-dd’T’HH:mm:ss’Z’»);

Status – статус сообщения (возможные значения и их описание находится в п.2.4);

TaskName – наименование задачи;

RegNumber – регистрационный номер;

TotalSize – общий размер сообщения в байтах;

Sender – отправитель сообщения (необязательное поле, только для сообщений, отправляемых другими Пользователями):

 Inn – ИНН отправителя сообщения

 Ogrn – ОГРН отправителя сообщения

 Bik – БИК отправителя сообщения

 RegNum – рег. номер КО-отправителя сообщения по КГРКО

 DevisonCode – номер филиала КО-отправителя сообщения по КГРКО

Files – файлы включенные в сообщение:

 Id – уникальный идентификатор файла;

 Name – имя файла;

 Description – описание файла (необязательное поле, для запросов и предписаний из Банка России содержит имя файла с расширением, однако может содержать запрещённые символы Windows);

 Encrypted – признак зашифрованности файла;

 SignedFile – идентификатор файла, подписью для которого является данный файл;

 Size - общий размер файла в байтах;

 RepositoryInfo – информация о репозиториях (описание репозитория в котором расположен файл. Данная информация используется как для загрузки файла, так и при его выгрузке):

 Path – путь к файлу в репозитории;

 Host – IP адрес или имя узла репозитория;

 Port – порт для обращения к репозиторию;

 RepositoryType – тип репозитория (значения: aspera, http);

Receipts – квитанции, полученные в ответ на сообщение:

 ReceiveTime – время размещения квитанции в Личном Кабинете;

 StatusTime – время из самой квитанции;

 Status – состояние обработки сообщения (возможные значения и их описание находится в п.2.4).;

Message – дополнительная информация из квитанции;

Files – файлы, включенные в квитанцию.

HTTP 404 – Сообщение не найдено

* + 1. Для скачивания конкретного сообщения используется метод GET:

GET: \*/ messages/{msgId}/download

REQUEST

{

"MsgId" : "string"

}

HEADER

"Range" : "string"

Где:

MsgId – уникальный идентификатор сообщения,

Range - Запрашиваемый диапазон байтов (необязательное поле).

RESPONSE

HTTP 200 – Ok; (для разового получения)

HTTP 206 – Partial content; (для получения определённого диапазона)

В BODY – запрашиваемое сообщение.

* + 1. Для получения данных о конкретном файле используется метод GET:

GET: \*/messages/{msgId}/files/{fileId}

REQUEST

{

"MsgId" : "string"

"FileId" : "string"

}

Где:

MsgId – уникальный идентификатор сообщения;

FileId – уникальный идентификатор файла.

RESPONSE

HTTP 200 – Ok

{

 "Id": "string"

 "Name": "string"

 "Encrypted": "boolean"

 "SignedFile": "string"

 "Size": "integer"

 "RepositoryInfo": [

 {

 "Path": "string"

 "Host": "string"

 "Port": "integer"

 "RepositoryType": "string"

 }

 ]

 }

 ]

{

Где:

Id – уникальный идентификатор файла;

Name – имя файла;

Encrypted – признак зашифрованности файла;

SignedFile – идентификатор файла, подписью для которого является данный файл;

Size - общий размер файла в байтах;

RepositoryInfo – информация о репозиториях (описание репозитория в котором расположен файл. Данная информация используется как для загрузки файла, так и при его выгрузке):

 Path – путь к файлу в репозитории;

 Host – IP адрес или имя узла репозитория;

 Port – порт для обращения к репозиторию;

 RepositoryType – тип репозитория (значения: aspera, http).

* + 1. Для скачивания конкретного файла из конкретного сообщения используется метод GET:

GET: \*/messages/{msgId}/files/{fileId}/download

REQUEST

{

"MsgId": "string"

"FileId": "string"

}

HEADER

"Range" : "string"

Где:

MsgId – уникальный идентификатор сообщения;

FileId – уникальный идентификатор файла;

Range - Запрашиваемый диапазон байтов (необязательное поле).

RESPONSE

HTTP 200 – Ok; (для разового получения)

HTTP 206 – Partial content; (для получения определённого диапазона)

В BODY – запрашиваемое сообщение.

* 1. Удаление сообщений

Удаление сообщений и файлов на стороне УФР осуществляется с использованием универсального REST-сервиса.

* + 1. Для удаления конкретного сообщения используется метод DELETE:

DELETE: \*/messages/{msgId}

REQUEST

{

"MsgId": "string"

}

Где:

MsgId – уникальный идентификатор сообщения.

RESPONSE

HTTP 200 – Ok;

HTTP 404 – Сообщение не найдено.

* + 1. Для удаления конкретного файла или отмены сессии отправки используется метод DELETE:

DELETE: \*/messages/{msgId}/files/{fileId}

REQUEST

{

"MsgId": "string"

"FileId": "string"

}

Где:

MsgId – уникальный идентификатор сообщения;

FileId – уникальный идентификатор файла.

RESPONSE

HTTP 200 – Ok.

* 1. Получение справочной информации
		1. Для получения справочника задач используется метод GET:

GET: \*/tasks

RESPONSE

HTTP 200 – Ok.

[

 {

 "Code": "string",

 "Description": "string"

 }

]

Где:

Code – код задачи в формате Zadacha\_\*\*, используется для идентификации задачи

Description – текстовое описание задачи, может быть не заполнено

* + 1. Для получения информации о своём профиле используется метод GET:

GET: \*/profile

RESPONSE

HTTP 200 – Ok.

{

 "ShortName": "string",

 "FullName": "string",

 "Activities": [

 {

 "FullName": "string",

 "ShortName": "string",

 "SupervisionDevision": {

 "Name": "string"

 }

 }

 ],

 "Inn": 0,

 "Ogrn": 0,

 "InternationalId": 0,

 "Opf": "string",

 "Email": "string",

 "Address": "string",

 "Phone": "string",

 "CreationDate": "string",

 "Status": "string"

}

Где:

ShortName – краткое наименование компании;

FullName – полное наименование компании;

Activities – список видов деятельностей компании;

Inn – ИНН компании;

Ogrn – ОГРН компании;

InternationalId – международный идентификатор (необязательное, зарезервированное поле);

Opf – организационно-правовая форма компании;

Email – электронный адрес компании;

Address – почтовый адрес компании;

Phone – контактный телефон компании;

CreationDate – дата создания ЛК компании;

Status – текущий статус ЛК компании.

* + 1. Для получения информации о технических оповещениях используется метод GET:

GET: \*/notifications

RESPONSE

HTTP 200 – Ok.

[

 {

 "Text": "string",

 "Date": "2018-11-15T11:13:23.104Z"

 }

]

Где для каждого элемента массива:

Text – текст технического оповещения;

Date – Дата и время технического оповещения.

* + 1. Для получения списка справочников используется метод GET:

GET: \*/dictionaries

RESPONSE

HTTP 200 – Ok.

[

 {

 "Id": "string",

 "Text": "string",

 "Date": "string "

 }

]

Где:

Id – уникальный идентификатор справочника, используется для идентификации задачи;

Text – текстовое наименование справочника;

Date – дата последнего обновления справочника.

* + 1. Для скачивания конкретного справочника используется метод GET:

GET: \*/dictionaries/{dictId}/

REQUEST

{

"dictId": "string"

}

Где:

dictId – уникальный идентификатор сообщения;

RESPONSE

HTTP 200 – Ok

В BODY – запрашиваемый справочник.

Запрещается запрашивать справочник, предварительно не убедившись в том, что он был обновлён (при запросе списка справочников Date изменилось на актуальную).