**Спецификация**

**API ЕПГУ**

**версия 1.13**

Оглавление

[Термины и определения 3](#_Toc147132402)

[1. Технические условия: 5](#_Toc147132403)

[1.1. Требования к получению маркера доступа 5](#_Toc147132404)

[1.2. Реализация подключения по ГОСТ TLS 6](#_Toc147132405)

[1.2.1 Порядок проверки подключения по протоколу TLS с ГОСТ к тестовой среде 6](#_Toc147132406)

[1.3. Подключение через СМЭВ 4 7](#_Toc147132407)

[2. Описание процессов работы с заявлением 8](#_Toc147132408)

[2.1. Создание и подача заявления 11](#_Toc147132409)

[2.1.1 Описание целевого процесса 11](#_Toc147132410)

[2.1.2 Резервирование номера заявления 11](#_Toc147132411)

[2.1.3 Загрузка и отправка заявления после резервирования номера 12](#_Toc147132412)

[2.1.4 Загрузка и отправка заявления одним запросом 14](#_Toc147132413)

[2.2. Отмена заявления 16](#_Toc147132414)

[2.3. Получение статусов заявлений 17](#_Toc147132415)

[2.3.1 Получение статусов заявлений по переданному списку заявлений. 18](#_Toc147132416)

[2.3.2 Получение статусов всех заявлений с даты обновления статуса. 19](#_Toc147132417)

[2.4. Получение деталей по заявлению 21](#_Toc147132418)

[3. Получение справочных данных 29](#_Toc147132419)

[4. Скачивание файла 32](#_Toc147132420)

[5. Примеры запросов и ответов 32](#_Toc147132421)

[Резервирование номера заявления 32](#_Toc147132422)

[Получение справочных данных 33](#_Toc147132423)

[5.1. Загрузка и отправка заявления одним запросом 34](#_Toc147132424)

[5.2. Загрузка и отправка заявления после резервирования номера 34](#_Toc147132425)

[5.3. Отмена заявления 36](#_Toc147132426)

[5.4. Получение статусов заявлений 36](#_Toc147132427)

[5.5. Получение деталей по заявлению 38](#_Toc147132428)

[Приложение 1. Статусы заявления в процессе обработки в API (сервисе gu-smev) 41](#_Toc147132429)

[Приложение 2. Требования к предоставлению информации об инцидентах в области API ЕПГУ 44](#_Toc147132430)

[Приложение 3. Ограничения, применяемые при взаимодействии ВИС с API ЕПГУ 45](#_Toc147132431)

[Приложение 4. Ошибки, возвращаемые при запросах к API ЕПГУ 46](#_Toc147132432)

[Приложение 5. Структура файла sign\_config.xml 52](#_Toc147132433)

[История изменений документа 54](#_Toc147132434)

Термины и определения

| Термин | Описание |
| --- | --- |
| ВИС | Внешняя информационная система |
| ВС | Вид сведений |
| ГОСТ TLS | Защищённое соединение по протоколу безопасности транспортного уровня TLS (transport layer security), с использованием российских криптографических алгоритмов, соответствующих межгосударственным стандартам (ГОСТ) |
| ЕПГУ | Федеральная государственная информационная система «Единый портал государственных и муниципальных услуг (функций)» (http://www.gosuslugi.ru/). |
| ЕСИА | Федеральная государственная информационная система «Единая система идентификации и аутентификации в инфраструктуре, обеспечивающей информационно-технологическое взаимодей­ствие информационных систем, используемых для предоставления государственных и муниципальных услуг в электронной форме» |
| ИС | Информационная система |
| Криптонабор (Cipher Suite) | Набор криптографических алгоритмов и их параметров, определяющий работу протокола TLS в рамках соответствующей данному криптонабору сессии |
| ЛК | Личный кабинет пользователя ЕПГУ |
| ЕЛК | Единый личный кабинет госуслуг предоставляет сервис для синхронизации информации о заявлениях на получение госуслуг и стадии их исполнения между внешними информационными системами и Единым порталом Госуслуг. Сервис доступен федеральным, региональным, муниципальным органам власти и МФЦ. |
| Скоуп | Область доступа |
| ПОДД СМЭВ (СМЭВ4) | Подсистема обеспечения доступа к данным федеральной государственной информационной системы «Единая система межведомственного электронного взаимодействия» – Часть транспортной подсистемы СМЭВ, обеспечивающая доступ к данным, размещённым на Витринах данных |
| ФРГУ | Федеральный реестр государственных услуг |
| ЭП | Электронная подпись |
| УКЭП | Усиленная квалифицированная электронная подпись |
| Чанк | Фрагмент файла. Разделение отправляемого файла (zip-архива) на чанки должно производиться по байтам[[1]](#footnote-1) |
| ФСТЭК России | Федеральная служба по техническому и экспортному контролю |
| ФГИС | Федеральная государственная информационная система |
| IANA | Internet Assigned Numbers Authority, администрация адресного пространства Интернет, регистрирующая в т.ч. параметры сетевых протоколов |
| CLI | Command Line Interface, интерфейс командной строки |
| OpenAPI | Спецификация, определяющая формализованный стандарт, который описывает интерфейс к REST API сервису и позволяет определять возможности REST-сервиса без доступа к его исходному коду или документации |

Настоящий документ (далее – Спецификация) является дополнением  
к документам (далее – Связанные документы), необходимым для подключения по прикладному программному интерфейсу (далее – API) Единого портала государственных и муниципальных услуг (функций) (далее – ЕПГУ), расположенных по ссылке <https://partners.gosuslugi.ru/catalog/api_for_gu>, а именно:

* Регламент подключения к API Госуслуг (далее – Регламент);
* Руководство пользователя для организации-вендора по формированию API-Key и получению маркера доступа (далее – Руководство вендора);
* Руководство пользователя для организации-потребителя по формированию API-Key и получению маркера доступа (далее – Руководство потребителя);

Заполнение данных для конкретной услуги описано в спецификации соответствующей услуги/сервиса.

1. Технические условия:

## Требования к получению маркера доступа

Требования к получению маркера доступа зависят от типа получателя услуги. Для получателей - физических лиц маркер доступа получается в соответствии с разделом «Приложение В. Сервисы ЕСИА, основанные на протоколе OAuth2.0 и OpenID Connect 1.0» (подпункт В.2) методических рекомендаций по использованию ЕСИА (документ выложен на странице <https://partners.gosuslugi.ru/catalog/esia>).

Для получателей юридических лиц и индивидуальных предпринимателей требования к получению маркера доступа зафиксированы в разделе «Приложение Б. Сервисы ЕСИА на базе подхода REST» пункте «Б.12 Сервис получения маркера доступа по идентификационному ключу» методических рекомендаций по использованию ЕСИА (документ выложен на странице <https://partners.gosuslugi.ru/catalog/esia>).

Формирование и получение идентификационного ключа API-Key осуществляется в соответствии с регламентами, размещенными на странице <https://partners.gosuslugi.ru/catalog/api_for_gu>:

* «Руководство пользователя для организации-вендора по формированию API-Key и получение маркера доступа»;
* «Руководство пользователя для организации-потребителя по формированию API-Key и получение маркера доступа».

ВИС необходимо реализовать механизм получения маркера доступа к следующим скоупам ЕПГУ:

Таблица

|  |  |
| --- | --- |
| **Скоуп** | **Назначение скоупа** |
| http://lk.gosuslugi.ru/api-order | Подача заявлений с использованием программного интерфейса портала госуслуг |

Предоставление доступа к скоупу осуществляется в соответствии с документом «Регламент информационного взаимодействия Участников с Оператором ЕСИА и Оператором эксплуатации инфраструктуры электронного правительства» (документ выложен на странице <https://partners.gosuslugi.ru/catalog/esia>). Наименование скоупа необходимо указывать в поле «Получение данных из регистров ЕСИА через программные интерфейсы OAuth 2.0 / REST-API» заявки.

При получении маркера доступа можно указывать несколько скоупов. В случае получения маркера доступа на сотрудника организации в нём обязательно должен присутствовать один из скоупов ЕСИА для получения данных этой организации. Например, получение ОГРН организации, “http://esia.gosuslugi.ru/org\_ogrn?org\_oid=<value>”, где <value> – внутренний идентификатор организации в ЕСИА (обязательный параметр).

## Реализация подключения по ГОСТ TLS

В целях реализации возможности установления безопасного соединения по ГОСТ TLS был создан защищенный контур ЕПГУ, обращения к которому осуществляется по следующим адресам:

Таблица 2

|  |  |
| --- | --- |
| **Среда ЕПГУ** | **URL** |
| Тестовая среда | https://svcdev-gostapi.test.gosuslugi.ru/ |
| Продуктивная среда | https://www.gosuslugi.ru/ |

Для обеспечения безопасности данных, обрабатываемых при информационном взаимодействии с ФГИС ЕПГУ необходимо использовать средство криптографической защиты информации класса КС1 и выше (в случае, если при взаимодействии не происходит передача персональных данных от ФГИС ЕПГУ) или КС3 и выше (в случае, если при взаимодействии происходит передача персональных данных от ФГИС ЕПГУ), прошедшее сертификацию на соответствие требованиям ФСБ России, а также средство антивирусной защиты, прошедшее сертификацию на соответствие требованиям к средствам антивирусной защиты, утвержденным приказом ФСТЭК России от 20 марта 2012 г. № 28 (Профиль защиты ИТ.САВЗ.Г2.ПЗ).

### Порядок проверки подключения по протоколу TLS с ГОСТ к тестовой среде

Клиент (ВИС) при первом соединении с сервером отправляет сообщение ClientHello[[2]](#footnote-2), в котором указывает поддерживаемые им криптонаборы в порядке уменьшения приоритетов. Имена и идентификаторы криптонаборов указываются в соответствии с Приложением B.4 спецификации TLS v1.3 или реестром IANA.

На стороне клиента должен использоваться криптонабор TLS\_CIPHER\_2012 с идентификатором 0xff85.

Для проверки подключения ИС с использованием протокола TLS с ГОСТ к Тестовой среде средствами CLI (ОС семейство Unix[[3]](#footnote-3)) можно воспользоваться командой «curl» с параметрами «-vvI»..

*Пример:* curl -vvI <https://www.gosuslugi.ru/>

*Пример ответа:*

\* Trying 172.16.100.100:443...

\* Connected to www.gosuslugi.ru (172.16.100.100) port 443 (#0)

\* ALPN: offers h2, http/1.1

\* TLSv1.3 (OUT), TLS handshake, Client hello (1):

\* CAfile: /etc/ssl/certs/ca-certificates.crt

\* CApath: none

\* TLSv1.3 (IN), TLS handshake, Server hello (2):

…

## Подключение через СМЭВ 4

В целях создания возможности установления безопасного соединения также реализован механизм взаимодействия ВИС с ЕПГУ через СМЭВ4 (ПОДД), основанный на регламентированных запросах типа REST, описанных в OpenAPI спецификации API ЕПГУ, обращения к которому осуществляется при помощи Агента ПОДД в соответствии с Методическими рекомендациями по работе с ПОДД СМЭВ (далее – МР СМЭВ4), размещенными в Единой системе контекстных справок (https://info.gosuslugi.ru/docs/).

Правила формирования и отправки REST-запросов к API ЕПГУ в данном случае осуществляется в соответствии с параграфом 3.4 раздела 3 МР СМЭВ4.

OpenAPI спецификация API ЕПГУ размещена в Личном кабинете участника взаимодействия (далее – ЛК УВ) под префиксом в URL «gu-smev-api».

Для доступа внешних ИС к данной спецификации необходимо запросить и получить согласование Минцифры РФ:

* для проведения тестирования – к соответствующим регламентированным запросам типа REST-сервис в системе с мнемоникой MNSV05 (стенд SVCDEV);
* для эксплуатации в продуктивной среде – к соответствующим регламентированным запросам типа REST-сервис в системе с мнемоникой MNSV03.

1. Описание процессов работы с заявлением

В рамках взаимодействия ВИС с ЕПГУ посредством отправки запросов к API ЕПГУ при создании и обработке заявлений на получение государственных или иных услуг предусмотрен ряд отдельных, но взаимосвязанных процессов:

1. Создание заявления (включая его отправку);
2. Получение информации о ходе обработки заявления (деталей заявления);
3. Отмена заявления.

Каждый процесс сопровождается методом или набором методов API ЕПГУ.

Ниже в таблице 3 указаны относительные пути вызова методов. Для формирования полных URL необходимо к относительному пути добавить адрес соответствующей среды ЕПГУ, указанный в Регламенте[[4]](#footnote-4) в пп. «Подключение к тестовой среде» и «Подключение к продуктивной среде». Например, POST [URL\_среды\_ЕПГУ]**/**api/gusmev/order.

Таблица 3

| **Метод** | **Назначение** |
| --- | --- |
| POST /api/gusmev/order | Резервирование номера заявления (используется в случае, если требуется вносить номер заявления в XML запроса или если размер zip-архива более 50 000 000 байт) |
| POST /api/gusmev/push | Загрузка и отправка заявления одним запросом (для случая, если не требуется вносить номер заявления ЕПГУ в XML запроса и размер zip архива не более 50 000 000 байт) |
| POST /api/gusmev/push/chunked | Загрузка и отправка заявления после резервирования номера (в т.ч. загрузка архива по частям) (для случая, если требуется вносить номер заявления ЕПГУ в XML запроса или размер zip архива более 50 000 000 байт) |
| POST /api/gusmev/order/{orderId} | Получение данных заявления (информацию по статусам и другие детали заявления) |
| POST /api/gusmev/order/{orderId}/cancel | Отмена заявления (поддержка данного метода регламентируется спецификацией услуги и может быть доступна не для всех услуг/сервисов) |
| GET /api/gusmev/order/getUpdatedAfter/?pageNum={n}&pageSize={m}&updatedAfter={timestamp} | Получение статусов по всем заявлениям с даты обновления статуса |
| GET /api/gusmev/order/getOrdersStatus/?pageNum={n}&pageSize={m} | Получение статусов заявлений по переданному списку заявлений |
| POST /api/nsi/v1/dictionary/{code} | Получение справочных данных, см. п.3 Спецификации (поддержка данного метода регламентируется спецификацией услуги[[5]](#footnote-5) и может быть доступна не для всех услуг/сервисов) |
| GET /api/gusmev/files/download/{objectId}/{objectType}?mnemonic={mnemonic}&eserviceCode={eserviceCode} | Скачать вложение из ответа ведомства, см. п. 4 Спецификации (поддержка данного метода регламентируется спецификацией услуги и может быть доступна не для всех услуг/сервисов) |

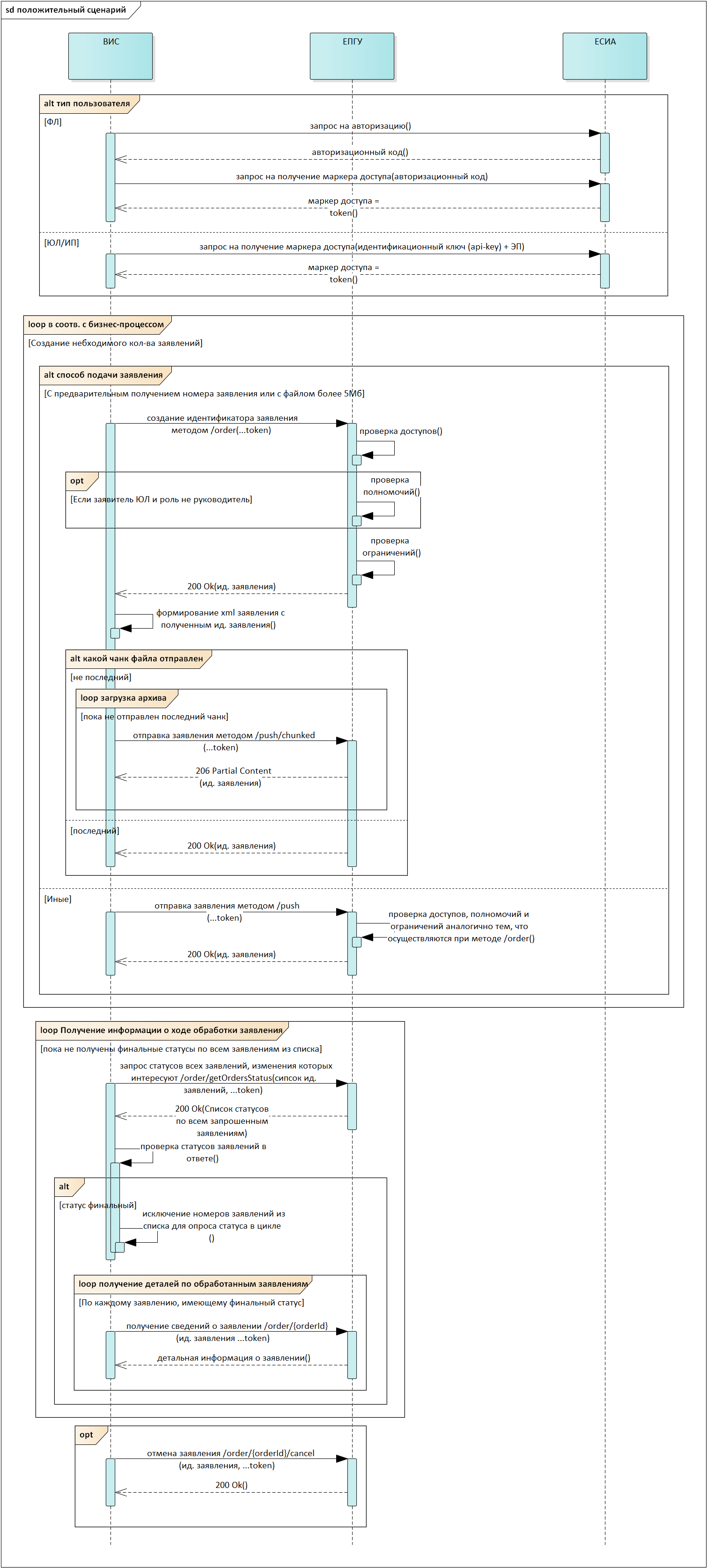


Рисунок . Диаграмма общего процесса взаимодействия через API ЕПГУ

## Создание и подача заявления

### Описание целевого процесса

Процесс подачи заявления по услугам с подписанием идентичен процессу подачи заявления по услугам без подписания.

Процесс предполагает два возможных варианта:

1. Если в соответствии со спецификацией услуги предполагается предварительное указание номера заявления в xml-файле, а также, если размер загружаемого zip архива более 50 000 000 байт, то отправка разбивается на этапы:
   1. Резервирование номера заявления через вызов метода /order;
   2. Загрузка и отправка заявления после резервирования номера (загрузка архива с файлами по заявлению – в т.ч. по частям, если это требуется) через вызов метода /push/chunked.
2. Если в соответствии со спецификацией услуги не предусмотрено внесение данных о номере заявления ЕПГУ и размер загружаемого zip архива менее 50 000 000 байт, то отправка происходит в один этап:
   1. Загрузка и отправка заявления одним запросом через вызов метода /push.

Архив должен содержать все необходимые файлы, согласно спецификации услуги. Если услуга предполагает подписание файлов заявления УКЭП, то файлы с отсоединенной подписью так же надо добавить в архив. Спецификациями к конкретным услугам могут быть установлены и иные ограничения к загружаемым файлам.

### Резервирование номера заявления[[6]](#footnote-6)

Этот метод необходимо использовать только в случае, если по спецификацией услуги предусмотрено внесение данных о номере заявления ЕПГУ в XML запроса.

Для резервирования номера заявления необходимо вызвать следующий метод:

Таблица 4

|  |  |
| --- | --- |
| **Метод** | **URL** |
| POST | /api/gusmev/order |

В заголовках запроса должны передаваться следующие данные:

Таблица 5

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Authorization | маркер доступа (тип Bearer) | Да |
| Content-Type | значение **«**application/json» | Да |

В теле запроса должен передаваться JSON объект со следующими параметрами:

Таблица 6

| **Параметр** | **Описание** | **Тип, значения** | **Обязательность** |
| --- | --- | --- | --- |
| ServiceCode | Код интерактивной формы на ЕПГУ | Строка | Да |
| TargetCode | Код цели обращения услуги в ФРГУ | Строка | Да |
| Region | Код ОКАТО местоположения пользователя (можно передавать код ОКАТО региона, если невозможно определить точнее) | Строка | Да |

В случае положительного ответа (http-код = 200) в теле возвращается JSON объект с номером, зарезервированным для заявления.

Таблица 7

| **Параметр** | **Описание** | **Тип** |
| --- | --- | --- |
| orderId | Номер заявления ЕПГУ | Число |

HTTP-коды возможных ошибок, их описание и рекомендации по действиям, которые необходимо предпринять на стороне ВИС при их получении, приведены в Приложении 4.

### Загрузка и отправка заявления после резервирования номера4

Этот метод необходимо использовать, только если выполняется хотя бы одно из двух условий:

* в соответствии со спецификацией услуги в XML запроса должен быть указан номер заявления на ЕПГУ;
* размер загружаемого zip-архива более 50 000 000 байт.

Для загрузки архива с файлами по заявлению необходимо вызвать следующий метод:

Таблица 8

|  |  |
| --- | --- |
| **Метод** | **URL** |
| POST | /api/gusmev/push/chunked |

В архив необходимо добавить все файлы заявления, сформированные в соответствии с видом сведений по услуге, по которой вы подаете заявление. Файлы необходимо называть в соответствии со спецификацией по услуге.

Архив не должен содержать вложенных архивов и папок с передаваемыми файлами, если это прямо не указано в спецификации услуги.

В случае подачи заявления с подписанием УКЭП, к каждому файлу необходимо добавить файл с отсоединённой подписью. Наименование файла с отсоединённой подписью должно совпадать с основным файлом. Например, файл с подписью для файла req.xml должен называться req.xml.sig.

Если в составе файлов по заявлению должны присутствовать документы, подписанные несколькими подписями, то перечень таких документов и подписей к ним должны быть переданы в составе специального файла sign\_config.xml. Структура этого файла и xsd-схема приведены в Приложении 5.

В заголовках запроса должны передаваться следующие данные:

Таблица 9

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Authorization | маркер доступа (тип Bearer) | Да |
| Content-Type | значение **«multipart/form-data»** | Да |

В теле запроса должны передаваться параметры:

Таблица 10

| **Параметр** | **Описание** | **Тип, значения** | **Обязатель­ность** |
| --- | --- | --- | --- |
| meta | Метаинформация в формате Content-Type = application/json {"region":"36000000000", "serviceCode":"10000092161", "targetCode":"-1959599598"}  Описание параметров данной структуры приведено в таблице 6 Спецификации. | Строка | Да |
| chunk | Номер чанка | Число | Нет[[7]](#footnote-7) |
| chunks | Кол-во чанков | Число | Нет5 |
| orderId | Номер заявления | Число | Да |
| file | Данные загружаемого архива | Двоичные данные | Да |

Размер каждого непоследнего чанка должен быть не менее 5 000 000 байт и не более 50 000 000 байт. При этом размер последнего чанка может быть менее 5 000 000 байт, но также не должен превышать 50 000 000 байт.

Чанки можно отправлять в несколько потоков, при этом важна очередность отправки для первого и последнего чанка.

К очередности отправки чанков предъявляются следующие требования:

* первым отправляется первый чанк;
* далее могут быть отправлены чанки без очередности кроме последнего;
* последним должен быть отправлен последний чанк.

Нумерация чанков осуществляется следующим образом:

* Chunks = n
* Chunk = i (0... n-1)

Все чанки должны быть отправлены в течение 5 минут. Начало отсчета времени отправки всех чанков начинается с момента отправки первого чанка.

После отправки всех чанков инициируется отправка заявления.

В случае положительного результата обработки запроса на загрузку части архива (чанка) возвращается http-код = 206. После успешной обработки запроса на загрузку последнего из указанных в параметрах чанка возвращается http-код = 200. В теле таких ответов возвращается JSON объект со следующими параметрами:

Таблица 11

| **Параметр** | **Описание** | **Тип** | **Обязательность** |
| --- | --- | --- | --- |
| orderId | Номер заявления | Число | Да |

HTTP-коды возможных ошибок, их описание и рекомендации по действиям, которые необходимо предпринять на стороне ВИС при их получении, приведены в Приложении 4.

### Загрузка и отправка заявления одним запросом4

Этот метод необходимо использовать только в случае, если по ВС не предусмотрено внесение данных о номере заявления ЕПГУ в XML запроса и размер загружаемого zip архива не более 50 000 000 байт.

При этом после отправки файлов заявления:

* резервируется и сообщается в ответе номер заявления;
* производятся проверки, по результатам которых в создании заявления может быть отказано по различным причинам (см. коды ошибок в Приложении 4);
* в случае успешного прохождения проверок создается и отправляется в ведомство само заявление.

Для загрузки архива с файлами по заявлению необходимо вызвать следующий метод:

Таблица 12

|  |  |
| --- | --- |
| **Метод** | **URL** |
| POST | /api/gusmev/push |

В архив необходимо добавить все файлы заявления, сформированные в соответствии с правилами, описанными в спецификации услуги, по которой вы подаете заявление. Файлы необходимо называть в соответствии с требованиями ВС или со спецификацией по услуге.

В случае подачи заявления с подписанием УКЭП, к каждому файлу необходимо добавить файл с отсоединённой подписью. Наименование файла с отсоединённой подписью должно совпадать с основным файлом. Например, файл с подписью для файла req.xml будет req.xml.sig.

Если в составе файлов по заявлению должны присутствовать документы, подписанные несколькими подписями, то перечень таких документов и подписей к ним должны быть переданы в составе специального файла sign\_config.xml. Структура этого файла и xsd-схема приведены в Приложении 5.

В заголовках запроса должны передаваться следующие данные:

Таблица 13

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Authorization | маркер доступа (тип Bearer) | Да |
| Content-Type | значение **«multipart/form-data»** | Да |

В теле запроса должны передаваться параметры:

Таблица 14

| **Параметр** | **Описание** | **Тип, значения** | **Обязательность** |
| --- | --- | --- | --- |
| meta | Метаинформация в формате Content-Type = application/json {"region":"36", "serviceCode":"10000092161", "targetCode":"-1959599598"}  Описание упомянутых выше параметров приведено в таблице 6 Спецификации. | Строка | Да |
| file | Данные загружаемого архива | Двоичные данные | Да |

В случае положительного ответа возвращается http-код = 200. в теле возвращается JSON объект со следующими параметрами:

Таблица 15

| **Параметр** | **Описание** | **Тип** | **Обязательность** |
| --- | --- | --- | --- |
| orderId | Номер заявления | Число | Да |

HTTP-коды возможных ошибок, их описание и рекомендации по действиям, которые необходимо предпринять на стороне ВИС при их получении, приведены в Приложении 4.

## Отмена заявления

Отмена заявления возможна только в случае наличия у заявления параметра cancelAllowed со значением true. Значение данного параметра можно получить в ответе на запрос деталей заявления (см. параграф 2.4 Спецификации).

Для отмены заявления требуется вызывать следующий метод:

Таблица 16

|  |  |
| --- | --- |
| **Метод** | **URL** |
| POST | /api/gusmev/order/{orderId}/cancel |

В заголовке запроса должны передаваться следующие данные:

Таблица 17

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Authorization | маркер доступа (тип Bearer) | Да |

Таблица 18

| **Параметр** | **Описание** | **Тип, значения** | **Обязательность** |
| --- | --- | --- | --- |
| orderId | Номер заявления | Число | Да |

В случае успешного вызова возвращается ответ с http-код = 200.

HTTP-коды возможных ошибок, их описание и рекомендации по действиям, которые необходимо предпринять на стороне ВИС при их получении, приведены в Приложении 4.

Пример ответа при ошибке:

|  |
| --- |
| {  "code": "order\_access",  "message": "У пользователя нет прав для работы с текущим заявлением"  } |

## Получение статусов заявлений

В рамках взаимодействия ВИС с ЕПГУ посредством API ЕПГУ создается множество заявлений, инициатором которых является определенная ВИС. В рамках API ЕПГУ реализована возможность получить информацию по изменениям состояния заявлений, созданных определенной ВИС.

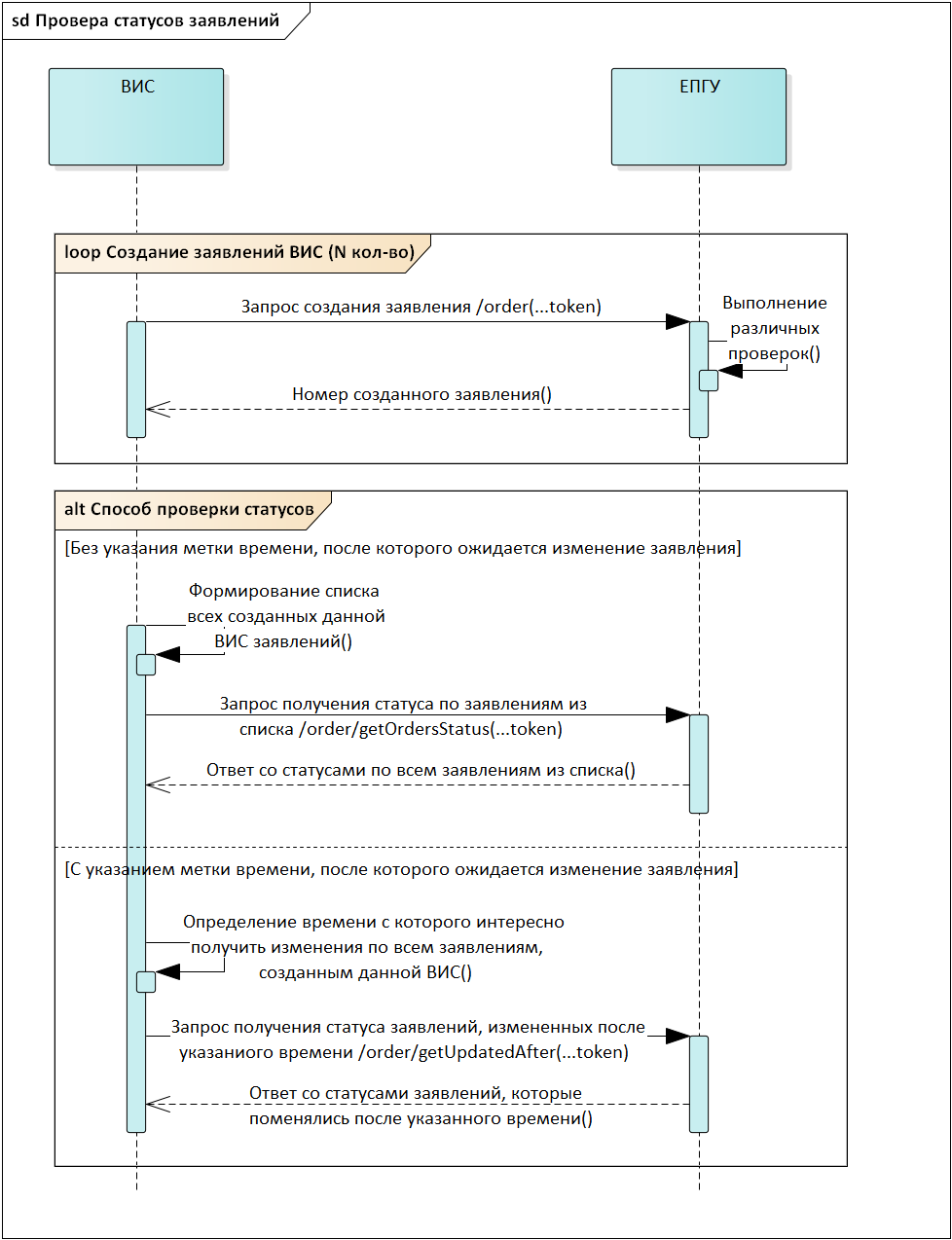


Рисунок 2. Схема работы со статусами заявлений, созданных одной ВИС

### Получение статусов заявлений по переданному списку заявлений.

Для получения статусов по списку заявлений необходимо вызвать следующий метод:

Таблица 19

|  |  |
| --- | --- |
| **Метод** | **URL** |
| GET | /api/gusmev/order/getOrdersStatus?pageNum={n}&pageSize={m}&orderIds={array[integer]} |

pageNum={n} – количество возвращаемых страниц.

pageSize={m} – количество записей на странице,

orderIds={array[integer]} – номера заявлений.

В заголовках запроса должны передаваться следующие данные:

Таблица 20

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Authorization | маркер доступа (тип Bearer) | Да |
| Content-Type | значение **«**application/json» | Да |

В случае положительного ответа (http-код = 200) в теле возвращается JSON объект с параметрами, которые предусмотрены для переданного типа значений:

Таблица 21

| **Параметр** | | | **Описание** | **Тип** | **Обязательность** |
| --- | --- | --- | --- | --- | --- |
| count | | | Количество записей, содержащихся в массиве Content | Число | Да |
| totalCount | | | Количество найденных записей, подходящих под условие | Число | Да |
| content | | | Массив записей, каждая из которых представлена объектом со структурой, описанной ниже | Массив объектов | Да |
|  | orderId | | Номер заявления | Число | Да |
| orderSearchStatus | | Признак нахождения статуса заявления. Может принимать значение FOUND/ NOT\_FOUND | Строка | Да |
| status | | Информация о текущем статусе заявления, представленная объектом со структурой описанной ниже | Объект | Нет |
|  | | statusId | Код статуса (код статуса из ЛК ЕПГУ, соответствует статусной модели услуг).  Данный статус соответствует параметру statusId последнего объекта массива statuses результата запроса деталей заявления (POST /api/gusmev/order/{orderId}) | Число | Да |
| statusName | Наименование текущего статуса | Строка | Да |
| updated | Дата и время обновления статуса | timestamp | Да |

HTTP-коды возможных ошибок, их описание и рекомендации по действиям, которые необходимо предпринять на стороне ВИС при их получении, приведены в Приложении 4.

### Получение статусов всех заявлений с даты обновления статуса.

В процессе взаимодействия ВИС с ЕПГУ может возникнуть потребность получить информацию об изменениях состояния заявления с определенного момента времени.

Таблица 22

|  |  |
| --- | --- |
| **Метод** | **URL** |
| GET | /api/gusmev/order/getUpdatedAfter?pageNum={n}&pageSize={m}&updatedAfter={timestamp} |

pageNum={n} – номер запрашиваемой страницы.

pageSize={m} – количество записей на странице.

updatedAfter={timestamp} – дата и время в формате «YYYY-MM-DD'T'HH:mm:ss.SSS[[8]](#footnote-8)», после которых были обновлены статусы.

В заголовке запроса должны передаваться следующие данные:

Таблица 23

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Authorization | маркер доступа (тип Bearer) | Да |

В случае положительного ответа (http-код = 200) в теле возвращается JSON объект с параметрами, которые предусмотрены для переданного типа значений:

Таблица 24

| **Параметр** | | | **Описание** | **Тип** | **Обязательность** |
| --- | --- | --- | --- | --- | --- |
| Count | | | Количество записей, содержащихся в массиве Content | Число | Да |
| totalCount | | | Количество найденных записей, подходящих под условие | Число | Да |
| Content | | | Массив записей, каждая из которых представлена объектом со структурой, описанной ниже | Массив объектов | Да |
|  | orderId | | Номер заявления | Число | Да |
| orderSearchStatus | | Признак нахождения статуса заявления. Может принимать значение FOUND/ NOT\_FOUND | Строка | Да |
| status | | Информация о текущем статусе заявления, представленная объектом со структурой описанной ниже | Объект | Нет |
|  | | statusId | Код статуса (код статуса из ЛК ЕПГУ, соответствует статусной модели услуг).  Данный статус соответствует параметру statusId последнего объекта массива statuses результата запроса деталей заявления (POST /api/gusmev/order/{orderId}) | Число | Да |
| statusName | Наименование текущего статуса | Строка | Да |
| updated | Дата и время обновления статуса | timestamp | Да |

HTTP-коды возможных ошибок, их описание и рекомендации по действиям, которые необходимо предпринять на стороне ВИС при их получении, приведены в Приложении 4.

## Получение деталей по заявлению

Для получения деталей по заявлению в т.ч. статуса обработки заявления необходимо вызвать следующий метод:

Таблица 25

|  |  |
| --- | --- |
| **Метод** | **URL** |
| POST | /api/gusmev/order/{orderId} |

{orderId} – номер заявления.

В заголовке запроса должны передаваться следующие данные:

Таблица 26

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Authorization | маркер доступа (тип Bearer) | Да |

В случае положительного ответа (http-код = 200) в теле возвращается JSON объект с параметрами, которые предусмотрены для переданного типа значений:

Таблица 27

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Описание** | **Тип** |
| Code | Код состояния заявления в соответствии с Приложением 1 Спецификации | Строка |
| Message | Текстовое сообщение, описывающее текущее состояние запроса на создание заявления | Строка |
| messageId | Идентификатор в СМЭВ исходящего запроса, созданного ЕПГУ на основании заявления, созданного методами API ЕПГУ.  Зная messageId, можно просмотреть данные о судьбе заявления в ЛК УВ (<https://lkuv.gosuslugi.ru/paip-portal/#/exchange-session>) | Строка |
| Order | В случае, если заявление уже создано на портале и отправлено в ведомство, параметр содержит строку в виде экранированного JSON-объекта, структура которого описана ниже | Строка |

Структура экранированного JSON-объекта, передаваемого в параметре order ответа на запрос деталей заявления представлена следующими параметрами (ключевые параметры для взаимодействия через API ЕПГУ выделены полужирным начертанием):

Таблица 28

| **Параметр** | **Описание** | **Тип** |
| --- | --- | --- |
| orderType | Тип заявления | Строка |
| hasNewStatus | Флаг нового статуса для заявления | Логический |
| smevTx | Код транзакции СМЭВ3 | Строка |
| stateOrgId | Код ведомства | Числовой |
| hasEmpowerment2021 | Флаг наличия делегирования | Логический |
| smevMessageId | Идентификатор СМЭВ-сообщения от ведомства, сменившего статус | Строка |
| notifySms | Флаг необходимости уведомления о смене статуса через СМС | Строка |
| notifyEmail | Флаг необходимости уведомления о смене статуса через сообщение на эл. почту | Строка |
| notifyPush | Флаг необходимости уведомления о смене статуса через push-сообщение | Строка |
| hasChildren | Флаг наличия дочерних заявлений | Логический |
| stateStructureName | Наименование ведомства | Строка |
| portalCode | Код портала | Строка |
| formVersion | Версия | Строка |
| stateOrgStatusCode | Код ведомственного статуса | Строка |
| stateOrgStatusName | Наименование ведомственного статуса | Строка |
| routingCode | Код маршрутизации СМЭВ-сообщения в ведомство | Строка |
| orderAttributeEvents | Атрибуты событий для заявления | Массив объектов |
| * Name | Наименование атрибута | Строка |
| * newValue | Новое значение | Строка |
| * oldValue | Старое значение | Строка |
| ownerId | Идентификатор пользователя | Число |
| eserviceId | Идентификатор формы заявления | Строка |
| currentStatusHistoryId | Идентификатор статуса заявления | Число |
| hasTimestamp | Флаг timestamp | Логический |
| orderStatusName | Наименование статуса заявления | Строка |
| portalName | Наименование портала | Строка |
| paymentRequired | Флаг наличия оплаты | Логический |
| paymentStatusEvents | Статус событий при оплате | Массив объектов |
| Payback | Служебный параметр | Логический |
| readyToPush | Служебный параметр | Логический |
| admLevelCode | Уровень услуги (региональный/федеральный) | Строка |
| **Id** | **Номер заявления** | **Число** |
| signCnt | Кол-во подписей, для заявлений от нескольких заявителей | Число |
| allFileSign | Флаг наличия ЭП для файлов | Логический |
| orgUserName | Наименование организации пользователя | Строка |
| noPaidPaymentCount | Количество неоплаченных платежей | Число |
| childrenSigned | Флаг подписи дочерних заявлений | Логический |
| creationMode | Режим создания | Строка |
| powerMnemonic | Мнемоника полномочия, с которым подается заявление | Строка |
| **orderStatusId** | **Код статуса заявления** | **Число** |
| withCustomResult | Признак необходимости отображения кнопки в Деталях заявления услуги | Логический |
| **Statuses** | **Статусы заявления** | **Массив объектов** |
| * **date** | **Дата и время смены статуса** | **Строка дата и время** |
| * **cancelAllowed** | **Флаг возможности отменить заявление** | **Логический** |
| * unreadEvent | Признак прочтения | Логический |
| * sendMessageAllowed | Признак разрешения отправки сообщения | Логический |
| * deliveryCancelAllowed | Флаг наличия отмены доставки | Логический |
| * **statusId** | **Код статуса[[9]](#footnote-9)** | **Число** |
| * **title** | **Наименование статуса** | **Строка** |
| * **comment** | **Комментарий к статусу** | **Строка** |
| * sender | Отправитель СМЭВ-сообщения о смене статуса | Строка |
| * mnemonic | Мнемоника ИС отправителя | Строка |
| * **orderId** | **Номер заявления** | **Число** |
| * editAllowed | Признак редактирования | Логический |
| * **hasResult** | **Флаг передачи файла в ответ на заявление** | **Строка** |
| * id | Идентификатор статуса | Число |
| * **finalStatus** | **Флаг финального статуса** | **Логический** |
| * **stateOrgStatusCode** | **Код ведомственного статуса[[10]](#footnote-10)** | **Строка** |
| * **stateOrgStatusDescr** | **Наименование ведомственного статуса** | **Строка** |
| orderDate | Дата и время создания заявления | Строка дата и время |
| **Updated** | **Дата и время обновления статуса заявления** | **Строка дата и время** |
| hasNoPaidPayment | Флаг наличия оплаченного платежа | Логический |
| servicePassportId | Идентификатор паспорта услуги | Строка |
| checkQueue | Флаг проверки очереди | Логический |
| withDelivery | Флаг доставки | Логический |
| Gisdo | Признак подключенности ведомства к ФГИС ДО | Логический |
| userSelectedRegion | Код ОКАТО местоположения пользователя | Строка |
| sourceSystem | Наименование системы от куда было подано заявление | Строка |
| serviceUrl | Ссылка на заявление | Строка |
| eQueueEvents | Массив объектов eQueueEvent | Массив объектов |
| hasActiveInviteToEqueue | Флаг записи на прием | Логический |
| multRegion | Признак регионозависимости | Логический |
| serviceEpguId | Идентификатор цели услуги ЕПГУ | Строка |
| extSystem | Признак, что создано внешней системой (через сервис ЕЛК) | Логический |
| useAsTemplate | Флаг черновика заявления | Логический |
| edsStatus | Идентификатор статуса проверки ЭП | Строка |
| allowToDelete | Флаг удаления заявки | Логический |
| Qrlink |  |  |
| * hasAltMimeType | Связанно с alternativeMimeTypes из сервиса тербайта | Логический |
| * fileSize | Размер файла | Число |
| * canSentToMFC | Флаг отправки в МФЦ | Логический |
| * hasDigitalSignature | Флаг наличия ЭП | Логический |
| requestDate | Метка даты и времени запроса | Строка |
| hasPreviewPdf | Флаг наличия пдф | Логический |
| hubFormVersion | Идентификатор регионозависимой формы старого конструктора форм | Число |
| stateOrgCode | Сокращенное наименование ведомства | Строка |
| testUser | Флаг тестового пользователя | Логический |
| personType | Тип пользователя | Строка |
| textMessages | Информация о сообщениях | Массив объектов |
| serviceTargetId | Идентификатор цели | Строка |
| orderPayments | Информация о платежах | Массив данных |
| unreadMessageCnt | Кол-во непрочитанных сообщений | Число |
| **orderResponseFiles** | **Информация о файлах в ответе заявления** | **Массив данных** |
| * hasAltMimeType | Флаг наличия альтернативного MIME-типа | Логический |
| * **fileName** | **​Наименование файла** | **Строка** |
| * edsStatus | ​Статус проверки ЭП в EDS | Строка |
| * fileSize | Размер файла | Число |
| * **link** | **​Ссылка на файл в TERRABYTE** | **Строка** |
| * **id** | **​Идентификатор файла** | **Строка** |
| * **mimeType** | **MIME-тип файла** | **Строка** |
| * **hasDigitalSignature** | **Флаг наличия ЭП к файлу** | **Логический** |
| **hasResult** | **Флаг передачи файла в ответ на заявление** | **Логический** |
| serviceName | Наименование цели | Строка |
| deprecatedService | Признак, что услуга больше не заказывается | Логический |
| hubForm | Признак, что форма-концентратор | Логический |
| userId | Идентификатор пользователя | Число |
| allowToEdit | Флаг редактирования заявления | Логический |
| **orderAttachmentFiles** | **Файлы заявления, отправленные пользователем** | **Массив данных** |
| * **fileName** | **Название файла** | **Строка** |
| * fileSize | Размер файла | Число |
| * **hasDigitalSignature** | **Наличие подписи** | **Логический** |
| * **id** | **Идентификатор файла** | **Строка** |
| * **link** | **Ссылка на файл в хранилище** | **Строка** |
| * **mimeType** | **MIME-тип** | **Строка** |
| * type | Тип | Строка |
| **closed** | **Флаг наличия финального статуса** | **Логический** |
| online | Признак, онлайн услуга или нет | Логический |
| readyToSign | Для ЮЛ, для подписания заявки, Маркер ожидания УКЭП | Логический |
| **currentStatusHistory** | **История статуса** | **Объект** |
| * **date** | **Дата и время смены статуса** | **Строка дата и время** |
| * **cancelAllowed** | **Флаг наличия отмены** | **Логический** |
| * unreadEvent | Признак прочтение события | Логический |
| * deliveryCancelAllowed | Флаг наличия отмены доставки | Логический |
| * finalStatus | Флаг финального статуса | Логический |
| * orderId | Номер заявления | Число |
| * **stateOrgStatusCode** | **Код ведомственного статуса** | **Строка** |
| * **stateOrgStatusDescr** | **Наименование ведомственного статуса** | **Строка** |
| * hasResult | Флаг передачи файла в ответ на заявление | Строка |
| * sendMessageAllowed | Признак разрешения отправки сообщения | Логический |
| * **statusId** | **Код статуса** | **Число** |
| * **title** | **Наименование статуса** | **Строка** |
| * editAllowed | Признак редактирования | Логический |
| * sender | Наименование ведомства | Строка |
| * **comment** | **Комментарий** | **Строка** |
| * id | Идентификатор статуса | Число |
| infoMessages | Информация о сообщениях | Массив данных |
| location | Код уровня услуги | Строка |
| paymentCount | Количество платежей | Число |
| draftHidden | Признак скрытия черновика | Логический |
| stateStructureId | Код ведомства | Строка |

В случае успешного вызова возвращается ответ с http-код = 200.

HTTP-коды возможных ошибок, их описание и рекомендации по действиям, которые необходимо предпринять на стороне ВИС при их получении, приведены в Приложении 4.Пример ответа при ошибке:

|  |
| --- |
| {  "code": "order\_access",  "message": "У пользователя нет прав для работы с текущим заявлением"  } |

1. Получение справочных данных

В соответствии со спецификациями услуг, доступных в REST API ЕПГУ, для формирования заявления на получение услуги (файл в формате XML) могут потребоваться справочные сведения, хранящиеся в ЕСНСИ и/или ЕПГУ и доступные для получения по выделенному REST-сервису.

Информация о том, какие сведения из каких справочников (справочные значения и наименования и коды справочников) необходимы в заявлении определены в спецификации конкретной услуги.

Для получения справочных данных необходимо вызвать следующий метод:

Таблица 29

|  |  |
| --- | --- |
| **Среда ЕПГУ** | **URL** |
| Тестовая среда | [/api/nsi/v1/dictionary/{code}](https://pgu-uat-fed.test.gosuslugi.ru/api/nsi/v1/dictionary/%7bcode%7d) |
| Продуктивная среда | [/api/nsi/v1/dictionary/{code}](https://www.gosuslugi.ru/api/nsi/v1/dictionary/%7bcode%7d) |

Метод запроса: POST

Метод доступен без авторизации.

{code} – код справочника.

В заголовке запроса должны передаваться следующие данные:

Таблица 30

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Content-Type | значение **«**application/json» | Да |

В теле запроса должен передаваться JSON объект со следующими параметрами:

Таблица 31

|  |  |  |  |
| --- | --- | --- | --- |
| **Параметр** | **Описание** | **Тип, значения** | **Обязательность** |
| treeFiltering | Тип справочника (плоский / иерархический) | ONELEVEL / SUBTREE | Да |
| parentRefItemValue | Код родительского элемента | Строка | Нет |
| pageNum | Номер необходимой страницы | Число | Нет |
| pageSize | Количество записей на странице | Число | Нет |

В случае отсутствия в запросе параметров pageNum и pageSize возвращаются все элементы справочника.

В случае положительного ответа (http-код = 200) в теле возвращается JSON объект со следующими параметрами:

Таблица 32

|  |  |  |  |
| --- | --- | --- | --- |
| **Параметр** | **Описание** | **Тип** | **Обязательность** |
| error | Результат | Объект | Да |
| * code | Код результата | Число | Да |
| * message | Сообщение | Строка | Да |
| fieldErrors | Ошибки в полях запроса | Массив объектов | Нет |
| total | Общее количество найденных элементов | Число | Да, в случае error. code = 0 |
| items | Найденные элементы с учётом заданных pageNum и pageSize | Массив объектов | Нет |
| * value | Код элемента справочника | Строка | Да |
| * parentValue | Код родительского элемента | Строка | Нет |
| * title | Наименование элемента | Строка | Да |
| * isLeaf | Признак наличия подчинённых элементов | Логический | Нет |
| * children | Подчинённые элементы | Массив объектов | Нет |
| * attributes | Дополнительные атрибуты элемента справочника | Массив объектов | Нет |
| * attributeValues | Список значений дополнительных атрибутов элемента справочника | Объект | Нет |

1. Скачивание файла

Для скачивания файла необходимо вызвать следующий метод[[11]](#footnote-11):

Таблица 33

|  |  |
| --- | --- |
| **Среда ЕПГУ** | **URL** |
| Тестовая среда | /api/gusmev/files/download/{objectId}/{objectType}?mnemonic={mnemonic}&eserviceCode={eserviceCode} |
| Продуктивная среда | /api/gusmev/files/download/{objectId}/{objectType}?mnemonic={mnemonic}&eserviceCode={eserviceCode} |

Метод запроса: GET

{objectId}[[12]](#footnote-12) – currentStatusHistoryId заявления.

{objectType} – тип вложения (для файла с результатами по заявлению значение 3);

{mnemonic} – мнемоника файла

{eserviceCode} – код услуги.

Мнемонику и тип вложения можно найти в результате ответа от метода получения деталей по заявлению (п.5). Параметр link будет содержать строку вида «"terrabyte://.../…/ mnemonic/ objectType "».

В заголовке запроса должны передаваться следующие данные:

Таблица 34

| **Заголовок** | **Данные, передаваемые в заголовке** | **Обязательность** |
| --- | --- | --- |
| Authorization | маркер доступа (тип Bearer) | Да |

1. Примеры запросов и ответов

Получение справочных данных

**Пример запроса:**

|  |
| --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/nsi/v1/dictionary/BIC |

**Тело запроса:**

|  |
| --- |
| {  "treeFiltering": "ONELEVEL",  "pageNum": 1,  "pageSize": 2  } |

**Пример положительного ответа:**

|  |
| --- |
| {  "error": {  "code": 0,  "message": "operation completed"  },  "fieldErrors": [],  "total": 899,  "items": [  {  "value": "040001002",  "parentValue": null,  "title": "ПУ БАНКА РОССИИ N 43192",  "isLeaf": true,  "children": [],  "attributes": [],  "source": null,  "attributeValues": {}  },  {  "value": "040002002",  "parentValue": null,  "title": "ПУ БАНКА РОССИИ N 43197",  "isLeaf": true,  "children": [],  "attributes": [],  "source": null,  "attributeValues": {}  }  ]  } |

## Загрузка и отправка заявления одним запросом

**Пример запроса:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/push  Пример form-data запроса:   |  |  | | --- | --- | | **Параметр** | **Значение** | | meta | {"region":"36000000000", "serviceCode":"10000000113", "targetCode":"-10000000113"} | | file |  | |

**Пример положительного ответа:**

|  |
| --- |
| {  "orderId": 2058864134  } |

**Пример отрицательного ответа:**

|  |
| --- |
| {  "code": "access\_denied\_person\_permissions",  "message": "Пользователь не дал согласие Вашей системе на выполнение данной операции"  } |

## Загрузка и отправка заявления после резервирования номера

**Пример запроса отправки первого чанка:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/push/chunked  Пример form-data запроса:   |  |  | | --- | --- | | **Параметр** | **Значение** | | chunk | 1 | | chunks | 2 | | orderId | 764255248 | | file |  | |

**Пример ответа:**

|  |
| --- |
| {  "orderId": 764255248  }  Status: 206 |

**Пример запроса отправки второго и последующих чанков:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/push/chunked  Пример form-data запроса:   |  |  | | --- | --- | | **Параметр** | **Значение** | | orderId | 764255248 | | file |  | |

**Пример ответа:**

|  |
| --- |
| {  "orderId": 764255248  }  Status: 206 |

**Пример запроса последнего чанка:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/push/chunked  Пример form-data запроса:   |  |  | | --- | --- | | **Параметр** | **Значение** | | orderId | 764255248 | | file | C:\Users\oleg.efremov\Downloads\Telegram Desktop\image_2023-07-31_14-02-19.png | |

**Пример ответа:**

|  |
| --- |
| {  "orderId": 764255248  }  Status: 200 |

Резервирование номера заявления

**Пример запроса:**

|  |
| --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/order |

**Тело запроса:**

|  |
| --- |
| {  "region":"36000000000",  "serviceCode":"10000000113",  "targetCode":"-10000000113",  } |

**Пример положительного ответа:**

|  |
| --- |
| {  "orderId": 52353456  } |

**Пример отрицательного ответа:**

|  |
| --- |
| {  "code": "access\_denied\_system",  "message": "Доступ запрещен для системы"  } |

## Отмена заявления

**Пример запроса:**

|  |
| --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/order/73588067/cancel |

**Пример ответа:**

|  |
| --- |
| http-код = 200 - заявление отменено  http-код = 403 - у пользователя нет прав для этого действия  http-код = 204 - заявление не найдено  http-код = 405 - отмена невозможна  http-код = 401 - необходимо получить новый маркер доступа и повторить запрос |

## Получение статусов заявлений

**Пример запроса статусов всех заявлений с даты обновления статуса:**

|  |
| --- |
| GET https://svcdev-beta.test.gosuslugi.ru/api/gusmev/order/getUpdatedAfter/?pageNum=0&pageSize=5&updatedAfter=2022-12-10T12:31:42.000 |

**Пример ответа:**

|  |
| --- |
| {      "count": 5,      "totalCount": 40,      "content": [          {              "orderId": 764607248,              "orderSearchStatus": "FOUND",              "status": {                  "statusId": 24,                  "statusName": "Ошибка отправки заявления в ведомство",                  "updated": "2022-12-21T20:49:37.672"              }          },          {              "orderId": 764603930,              "orderSearchStatus": "FOUND",              "status": {                  "statusId": 15,                  "statusName": "Заявление требует исправления",                  "updated": "2022-12-21T19:51:16.852"              }          },          {              "orderId": 764602932,              "orderSearchStatus": "FOUND",              "status": {                  "statusId": 4,                  "statusName": "Истекло время подписания документов",                  "updated": "2022-12-16T15:31:42.083"              }          },          {              " orderId": 2354270898,              "orderSearchStatus": "NOT\_FOUND",              "status": **null**          },          {              "orderId": 2354271090,              "orderSearchStatus": "NOT\_FOUND",              "status": **null**          }      ]  } |

**Пример запроса статусов по списку заявлений:**

|  |
| --- |
| GET https://svcdev-beta.test.gosuslugi.ru/api/gusmev/order/getOrdersStatus/?orderIds=764607248,764603930,764602932,2312621299,2312620698&pageNum=0&pageSize=5 |

**Пример ответа:**

|  |
| --- |
| {      "count": 5,      "totalCount": 5,      "content": [          {              "orderId": 764607248,              "orderSearchStatus": "FOUND",              "status": {                  "statusId": 24,                  "statusName": "Ошибка отправки заявления в ведомство",                  "updated": "2022-12-21T20:49:37.672"              }          },          {              "orderId": 764603930,              "orderSearchStatus": "FOUND",              "status": {                  "statusId": 15,                  "statusName": "Заявление требует исправления",                  "updated": "2022-12-21T19:51:16.852"              }          },          {              "orderId": 764602932,              "orderSearchStatus": "FOUND",              "status": {                  "statusId": 4,                  "statusName": "Истекло время подписания документов",                  "updated": "2022-12-16T15:31:42.083"              }          },          {              "orderId": 2312621299,              "orderSearchStatus": "NOT\_FOUND",              "status": **null**          },          {              "orderId": 2312620698,              "orderSearchStatus": "NOT\_FOUND",              "status": **null**          }      ]  } |

## Получение деталей по заявлению

**Пример запроса:**

|  |
| --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/order/2058864134 |

**Пример ответа:**

|  |
| --- |
| {  "code": "OK",  "message": null,  "message\_id": "71e35026-ade1-11e9-9931-00155d040494",  "order": "{\"orderType\":\"ORDER\",\"smevTx\":\"b0ba0139-613d-4d4d-bc28-4240efc9a6ec\",\"hasEmpowerment2021\":false,\"smevMessageId\":\"WAIT\_RESPONSE\",\"formVersion\":\"1\",\"ownerId\":1078941357,\"notifySms\":\"\",\"hasTimestamp\":false,\"orderStatusName\":\"Ошибка обработки заявления\",\"portalName\":\"Портал гос услуг v1\",\"paymentRequired\":false,\"id\":2058864134,\"signCnt\":0,\"childrenSigned\":false,\"stateOrgStatusCode\":\"\",\"inviteToEqueueUrl\":\"\",\"statusColorCode\":\"\",\"orderStatusId\":24,\"withCustomResult\":true,\"hasNoPaidPayment\":false,\"servicePassportId\":\"600113\",\"routingCode\":\"\",\"sourceSystem\":\"PGU\",\"stateOrgStatusName\":\"\",\"serviceUrl\":\"\",\"serviceEpguId\":\"1\",\"edsStatus\":\"EDS\_NOT\_SUPPORTED\",\"allowToDelete\":true,\"requestDate\":\"2022-07-25T13:01:47.438+0300\",\"complexMode\":\"\",\"stateOrgCode\":\"pfr\",\"personType\":\"PERSON\",\"serviceTargetId\":\"-10000000113\",\"orderPayments\":[],\"orderResponseFiles\":[],\"hasResult\":false,\"userId\":1078941357,\"allowToEdit\":false,\"closed\":true,\"readyToSign\":false,\"currentStatusHistory\":{\"date\":\"2022-07-25T13:01:52.121+0300\",\"cancelAllowed\":false,\"unreadEvent\":true,\"deliveryCancelAllowed\":false,\"finalStatus\":true,\"orderId\":2058864134,\"stateOrgStatusCode\":\"\",\"author\":\"\",\"hasResult\":\"N\",\"stateOrgStatusDescr\":\"\",\"title\":\"Ошибка обработки заявления\",\"mfcFinalStatus\":false,\"sendMessageAllowed\":false,\"statusId\":24,\"editAllowed\":false,\"sender\":\"Пенсионный фонд Российской Федерации\",\"mnemonic\":\"\",\"comment\":\"\",\"id\":6268157772},\"infoMessages\":[],\"notifyEmail\":\"\",\"paymentCount\":0,\"draftHidden\":false,\"stateStructureId\":\"10000002796\",\"hasNewStatus\":true,\"stateOrgId\":266,\"redirectUrl\":\"\",\"hasChildren\":false,\"stateStructureName\":\"ПФР\",\"portalCode\":\"PGU\",\"esepOperationId\":\"\",\"orderAttributeEvents\":[],\"notifyPush\":\"\",\"eserviceId\":\"10000000113\",\"currentStatusHistoryId\":6268157772,\"linkToOrderForm\":\"\",\"extOrderUrl\":\"\",\"paymentStatusEvents\":[],\"payback\":false,\"readyToPush\":false,\"admLevelCode\":\"FEDERAL\",\"formPrefilling\":false,\"allFileSign\":false,\"orgUserName\":\"\",\"noPaidPaymentCount\":-1,\"creationMode\":\"direct\",\"steps\":[],\"powerMnemonic\":\"\",\"statuses\":[{\"date\":\"2022-07-25T13:01:46.413+0300\",\"cancelAllowed\":false,\"unreadEvent\":true,\"deliveryCancelAllowed\":false,\"finalStatus\":false,\"orderId\":2058864134,\"stateOrgStatusCode\":\"\",\"author\":\"\",\"hasResult\":\"N\",\"stateOrgStatusDescr\":\"\",\"title\":\"Черновик заявления\",\"mfcFinalStatus\":false,\"sendMessageAllowed\":false,\"statusId\":0,\"editAllowed\":false,\"sender\":\"\",\"mnemonic\":\"\",\"comment\":\"\",\"id\":6268157768},{\"date\":\"2022-07-25T13:01:47.426+0300\",\"cancelAllowed\":false,\"unreadEvent\":true,\"deliveryCancelAllowed\":false,\"finalStatus\":false,\"orderId\":2058864134,\"stateOrgStatusCode\":\"\",\"author\":\"\",\"hasResult\":\"N\",\"stateOrgStatusDescr\":\"\",\"title\":\"Зарегистрировано на портале\",\"mfcFinalStatus\":false,\"sendMessageAllowed\":false,\"statusId\":17,\"editAllowed\":false,\"sender\":\"Пенсионный фонд Российской Федерации\",\"mnemonic\":\"\",\"comment\":\"\",\"id\":6268157769},{\"date\":\"2022-07-25T13:01:48.520+0300\",\"cancelAllowed\":false,\"unreadEvent\":true,\"deliveryCancelAllowed\":false,\"finalStatus\":false,\"orderId\":2058864134,\"stateOrgStatusCode\":\"\",\"author\":\"\",\"hasResult\":\"N\",\"stateOrgStatusDescr\":\"\",\"title\":\"Заявление отправлено в ведомство\",\"mfcFinalStatus\":false,\"sendMessageAllowed\":false,\"statusId\":21,\"editAllowed\":false,\"sender\":\"Пенсионный фонд Российской Федерации\",\"mnemonic\":\"\",\"comment\":\"\",\"id\":6268157770},{\"date\":\"2022-07-25T13:01:51.810+0300\",\"cancelAllowed\":false,\"unreadEvent\":true,\"deliveryCancelAllowed\":false,\"finalStatus\":false,\"orderId\":2058864134,\"stateOrgStatusCode\":\"\",\"author\":\"\",\"hasResult\":\"N\",\"stateOrgStatusDescr\":\"\",\"title\":\"Заявление получено ведомством\",\"mfcFinalStatus\":false,\"sendMessageAllowed\":false,\"statusId\":2,\"editAllowed\":false,\"sender\":\"Пенсионный фонд Российской Федерации\",\"mnemonic\":\"\",\"comment\":\"Сообщение доставлено\",\"id\":6268157771},{\"date\":\"2022-07-25T13:01:52.121+0300\",\"cancelAllowed\":false,\"unreadEvent\":true,\"deliveryCancelAllowed\":false,\"finalStatus\":true,\"orderId\":2058864134,\"stateOrgStatusCode\":\"\",\"author\":\"\",\"hasResult\":\"N\",\"stateOrgStatusDescr\":\"\",\"title\":\"Ошибка обработки заявления\",\"mfcFinalStatus\":false,\"sendMessageAllowed\":false,\"statusId\":24,\"editAllowed\":false,\"sender\":\"Пенсионный фонд Российской Федерации\",\"mnemonic\":\"\",\"comment\":\"\",\"id\":6268157772}],\"orderDate\":\"2022-07-25T13:01:46.000+0300\",\"updated\":\"2022-07-25T13:01:52.129+0300\",\"checkQueue\":false,\"withDelivery\":false,\"gisdo\":false,\"userSelectedRegion\":\"00000000000\",\"description\":\"\",\"eQueueEvents\":[],\"hasActiveInviteToEqueue\":false,\"multRegion\":false,\"extSystem\":false,\"useAsTemplate\":false,\"qrlink\":{\"hasAltMimeType\":false,\"fileName\":\"\",\"edsStatus\":\"\",\"fileSize\":0,\"canSentToMFC\":false,\"link\":\"\",\"id\":\"\",\"mimeType\":\"\",\"hasDigitalSignature\":false,\"additionalName\":\"\"},\"hasPreviewPdf\":false,\"testUser\":false,\"textMessages\":[],\"unreadMessageCnt\":0,\"parentOrderStateStructureName\":\"\",\"cr\_uin\":\"\",\"infSysCode\":\"\",\"serviceName\":\"Выдача справки о размере пенсии и иных социальных выплат на дату\",\"deprecatedService\":false,\"hubForm\":false,\"extOrderId\":\"\",\"orderAttachmentFiles\":[{\"fileName\":\"req.xml\",\"fileSize\":653,\"link\":\"terrabyte://00/2058864134/req.xml/2\",\"id\":\"2058864134/files/cmVxLnhtbA\",\"mimeType\":\"application/xml\",\"hasDigitalSignature\":false,\"type\":\"REQUEST\"}],\"online\":false,\"location\":\"36\"}"  } |

Приложение 1. Статусы заявления в процессе обработки в API (сервисе gu-smev)

| **Статус** | **Тип** | **Описание** |
| --- | --- | --- |
| NEW | Промежуточный | Данный статус проставляет по факту регистрации запроса на создание заявления |
| FILES\_VERIFICATION | Промежуточный | Данный статус проставляется по факту начала процесса проверки подписей переданных вложений, если такая проверка предусмотрена конфигурацией услуги |
| FILES\_VERIFICATION\_SUCCESS | Промежуточный | Данный статус проставляется по факту успешного прохождения процесса проверки подписей переданных вложений, если такая проверка проводилась |
| DONE | Финальный | Данный статус проставляется по факту передачи сторонним компонентам ЕПГУ проверенного заявления на дальнейшую отправку в ведомство.  В части сервиса gu-smev (API ЕПГУ) считается, что данный статус финальный положительный. |
| LIMITATION\_EXCEPTION | Финальный | Данный статус проставляется по факту проверки ограничений на создание заявления / отправку запросов для конкретной ВИС (в случае превышения ограничений, указанных в Приложении 3 к Спецификации) |
| INVALID\_FILES\_STRUCTURE | Финальный | Данный статус проставляется по факту проверки файловой структуры переданного ВИС архива, если структура не соответствует требованиям спецификации (в данном случае, если имеются вложенные папки, т.е. структура не плоская) |
| VALIDATION\_ERROR | Финальный | Данный статус проставляется по факту проверки переданных вложений в валидаторах, соответствующих конфигурации конкретной услуги (если вложения не соответствуют требованиям) |
| REQ\_NOT\_FOUND | Финальный | Данный статус проставляется по факту проверки наличия во вложениях транспортного xml-файла (файл отсутствует) |
| MPC\_NOT\_FOUND | Финальный | Данный статус проставляется по факту проверки наличия во вложениях xml-файла, содержащего бизнес-данные в соответствии со спецификацией соответствующей услуги (файл отсутствует) |
| REQ\_VERIFY\_FAILED | Финальный | Данный статус проставляется по факту проверки соответствия переданных в xml-файлах данных, данным получателя услуги (если данные не соответствуют) |
| FILES\_VERIFICATION\_FAILED | Финальный | Данный статус проставляется по факту проведения проверки подписей переданных вложений, если такая проверка предусмотрена конфигурацией услуги, если подписи отсутствуют или не прошли валидацию в ГУЦ |
| RETRY | Промежуточный | Данный статус проставляется в случае временного прерывания процесса обработки запроса на создание заявления из-за каких-либо технических причин на стороне ЕПГУ.  Переход из данного статуса предусмотрен в тот статус, который был до установки рассматриваемого, на иной статус соответствующий этапу процесса обработки, а также статус, соответствующий завершению процесса обработки с ошибкой |
| INTERNAL\_ERROR | Финальный | Данный статус проставляется в случае возникновения не предусмотренных процессом обработки запроса на создание заявления технических ошибок |
| CREATE\_ORDER\_FORBIDDEN | Финальный | Данный статус проставляется по факту проверки в сторонних сервисах ЕПГУ возможности создания заявления, связанной с отсутствием полномочий или каких-либо настроек услуги. При получении такой ошибки необходимо направить обращение в СЦ с хештегом #createorderforbidden |

Приложение 2. Требования к предоставлению информации об инцидентах в области API ЕПГУ

В случае возникновения инцидента в области API ЕПГУ, решение которого невозможно со стороны представителей ВИС, следует сформировать запрос в адрес [sd@sc.digital.gov.ru](mailto:sd@sc.digital.gov.ru) (в ФГИС СЦ).

В *теме электронного пись*ма должны быть указаны следующие данные:

* + - * мнемоника ИС;
      * хеш-тэг #API-ЕПГУ;
      * краткое содержание ошибки.

Пример:

* + - * *«AAA1 #API-ЕПГУ #Ошибка 500 при получении orderId».*

*Собственно запрос* должен содержать следующую информацию:

* + - 1. Наименование и идентификатор (client\_id в токене доступа) ИС в ЕСИА;
      2. Информацию о получателе услуг:
         1. Наименование и идентификатор (org\_oid в токене доступа) Организации-получателя услуг, если получателем услуг является ЮЛ/ИП;
         2. Идентификатор пользователя и ФИО в ЕСИА получателя услуг, если он является ФЛ;
      3. Среда ЕПГУ, в которой осуществлялось взаимодействие, которое закончилось инцидентом (PROD – продуктивная или SVCDEV – тестовая);
      4. Дата и время запроса, а также дата и время получения токена доступа, использованного в запросе, по которому возник инцидент;
      5. Информация о запросе, выполнение которого завершилось инцидентом (для Продуктивной среды – обезличенные данные):
         1. http-метод (GET, POST и т.д.);
         2. host http-запроса;
         3. URL http-запроса;
         4. Список всех заголовков http-запроса и их значения;
         5. Параметры и (или) тело http-запроса (при наличии, в случае отсутствия указать на это);
         6. Дата и время отправки запроса с точностью до секунды.
      6. Номер заявления (orderId) на ЕПГУ (если был получен).
      7. Описание ситуации, при которой возник инцидент;
      8. Результаты анализа инцидента, проведенного представителем владельца ВИС, а также описание проведенных проверок в рамках инцидента в соответствии с рекомендациями, данными в настоящей Спецификации.

Все перечисленные выше данные по запросу необходимо передать в едином текстовом документе формата (txt или docx).

Приложение 3. Ограничения, применяемые при взаимодействии ВИС с API ЕПГУ

Процесс взаимодействия ВИС с API ЕПГУ предполагает наличие некоторых ограничений на количества входящих HTTP-запросов и на количество заявлений по конкретной услуге в единицу времени[[13]](#footnote-13). Детализация таких ограничений приведена в таблице ниже.

|  |  |
| --- | --- |
| **Ограничение** | **Значение по умолчанию** |
| Кол-во входящих HTTP-запросов от ВИС по всем сервисам и пользователям[[14]](#footnote-14): |  |
| в минуту | 2 000 |
| в час | 120 000 |
| в день | 2 880 000 |
| в месяц | 86 400 000 |
| в год | 1 036 800 000 |
| Кол-во заявлений по услуге[[15]](#footnote-15) для одного пользователя: |  |
| за 10 минут | 20 |

Приложение 4. Ошибки, возвращаемые при запросах к API ЕПГУ

Ниже приведены коды, описания, детализация и рекомендации в случае получения соответствующих ошибочных ситуаций.

Описание и рекомендации в случае получения ответов с кодами 204, 401, 429 и 502… 504 приведены в таблице ниже:

|  |  |  |
| --- | --- | --- |
| **Код ответа** | **Краткое описание** | **Рекомендации** |
| 204 | Заявление не найдено | Следует проверить корректность указанного в запросе orderId. Данный идентификатор должен соответствовать тому, что был возвращен в ответе при создании заявления |
| 401 | Отказ в доступе | Получить новый маркер доступа и повторить запрос |
| 429 | Слишком много запросов | Проверить текущие ограничения по количеству отправляемых запросов (см. Приложение 3). При необходимости подать заявку с предложением об увеличении ограничений (см. подпункт 4.3.4 Приложения 4.3 к Регламенту) |
| 502 | Некорректный шлюз | Повторить вызов соответствующего метода. Интервал и количество повторений задаются спецификацией услуги. Если в результате ошибки все еще имеют место – необходимо составить обращение об инциденте и отправить его на адрес [sd@sc.digital.gov.ru](mailto:sd@sc.digital.gov.ru) |
| 503 | Сервис недоступен | -"- |
| 504 | Шлюз не отвечает | -"- |

В случае ответа с http-кодами 400 (неверные параметры), 403 (доступ запрещен), 409 (невозможно обработать запрос) или 500 (внутренняя ошибка) в теле возвращается JSON объект со следующими параметрами:

| **Параметр** | **Описание** | **Тип** | **Обязательность** |
| --- | --- | --- | --- |
| code | Код ошибки | Строка | Нет |
| message | Текст ошибки | Строка | Нет |

Список возможных кодов ошибок приведен в таблице ниже:

| **Код** | **Описание** | **Рекомендации** |
| --- | --- | --- |
| access\_denied\_person\_permissions | Пользователь не дал согласие Вашей системе на выполнение данной операции.  Данная ошибка возникает при отсутствии в передаваемом токене доступа параметра permissions, содержащего в значении подстроку «APIPGU». | Необходимо скорректировать запрос на получение токена доступа в соответствии с разделом 4 Методических рекомендаций по интеграции с REST API Цифрового профиля. |
| access\_denied\_personal\_data | Доступ к файлам запрещен, так как в них имеются персональные данные | Если для целей инфор­мирования недостаточно только факта выполнения услуги, необходимо получить разрешение на работу с персональными данными, а также направить письмо с информацией о получении данной ошибки на адрес [api@digital.gov.ru](mailto:api@digital.gov.ru) |
| access\_denied\_service | Доступ ВИС к запрашиваемой услуге запрещен. | Необходимо проверить, что ранее направлялась заявка на получение доступа к данной услуге и такая заявка получила положительный результат рассмотрения (см. Приложение 4.3 к Регламенту подключения к API Единого портала государственных услуг, размещенному на странице https://partners.gosuslugi.ru/ catalog/api\_for\_gu). |
| access\_denied\_system | Доступ запрещен для ВИС, отправляющей запрос.  Запрет может быть связан с обязательностью для данной услуги взаимодействия по защищенному каналу связи (см. параграф 1.2 раздела 1 Спецификации).  Также запрет может быть связан с отсутствием доступа ВИС к любой из доступных в API ЕПГУ услуг. | В случае с возникновение ошибки из-за несоответствия контура взаимодействия между ВИС и ЕПГУ (необходимости ГОСТ TLS) следует убедиться, что устанавливаемое соединение соответствует ГОСТ TLS (см. раздел 1.2.1 Спецификации).  В случае ошибки из-за отсутствия доступа ВИС к конкретной услуге необходимо проверить, что ранее направлялась заявка на получение доступа к данной услуге и такая заявка получила положительный результат рассмотрения (см. Приложение 4.3 к Регламенту подключения к API Единого портала государственных услуг, размещенному на странице https://partners.gosuslugi.ru/ catalog/api\_for\_gu). |
| access\_denied\_user | Доступ запрещен для данного типа пользователя.  Ошибка возникает, когда для услуги, указанной в запросе (serviceCode), не предусмотрена подача заявлений от данного типа пользователя (ЮЛ/ИП/ФЛ). | По спецификации соответствующей услуги уточнить, какие типы пользователей могут быть получателями данной услуги. Затем следует убедиться, что маркер доступа, необходимый для создания заявления, получен на того пользователя, чей тип соответствует заявленному в соответствующей услуге спецификации. |
| access\_denied\_user\_legal | Данная ошибка возникает при попытке создать заявления посредством отправки запроса ВИС к API ЕПГУ для получения услуги, доступ к которой оформлялся заявкой со специальным параметром «Владелец ИС является организацией-потребителем», имеющим значение «Да».  Т.е. ошибка возникает при попытке создать заявление с использованием токена, полученного для организации, которая не является владельцем ВИС, отправляющей данный запрос. | Следует проверить значение в указанном в описании параметра заявки на доступ к услуге.  В случае ошибочного указания данного параметра необходимо сформировать новую скорректированную заявку (см. Приложение 4.3 к Регламенту подключения к API Единого портала государственных услуг, размещенному на странице https://partners.gosuslugi.ru/ catalog/api\_for\_gu). |
| bad\_delegation | Нет необходимых полномочий для создания заявления. | Следует проверить, что сотрудник, действующий от имени организации/ИП, имеет полномочия, необходимые для создания и отправки заявления (см. пп. 3.2 и 3.3 Регламента подключения к API Единого портала государственных услуг, размещенному на странице https://partners.gosuslugi.ru/ catalog/api\_for\_gu). |
| bad\_request | Ошибка в параметрах запроса | Следует проверить параметры запроса на соответствие спецификации метода.  Параметры, которые сформированы некорректно, как правило отражены в сообщении (параметр message ответа на запрос) |
| cancel\_not\_allowed | Отмена заявления в текущем статусе невозможна | Установление права на отмену заявление осуществляется ведомством в рамках процесса оказания услуг.  Необходимо либо дождаться смены статуса на тот, в котором возможна отмена заявления либо дождаться получения финального статуса по заявлению |
| config\_delegation | Полномочие для создания и подачи заявления по заданной услуги не существует. | Необходимо отправить заявку на адрес api@digital.gov.ru с предложением создать новое полномочие для заданной услуги. |
| internal\_error | Произошла ошибка в обработке заявления, причины которой можно выяснить при анализе инцидента | Необходимо сформировать и направить сообщение об инциденте в СЦ в соответствии с требованиями предоставления информации об инцидентах, описанными в Приложении 2 Спецификации |
| limitation\_exception | Превышение установленных ограничений, указанных в Приложении 3 к Спецификации. | Необходимо либо ожидать истечения срока действия текущих ограничений, либо инициировать увеличение ограничений (см. Приложение 3 к Спецификации). |
| not\_found | Заявление не найдено | Следует проверить корректность указанного в запросе orderId. Данный идентификатор должен соответствовать тому, что был возвращен в ответе при создании заявления |
| order\_access | У пользователя нет прав для работы с текущим заявлением | Необходимо убедиться, что для работы с заявлением используется корректный маркер доступа: выданный получателю данной услуги в соответствии с регламентом и у него не закончилось время жизни |
| push\_denied | Нет прав для отправки заявления. Отправить заявление может только руководитель организации или сотрудник с доверенностью | Необходимо убедиться, что маркер доступа выдан на сотрудника, имеющего полномочия на подачу заявлений либо выдан руководителю (см. пп. 3.2 и 3.3 Регламента подключения к API Единого портала государственных услуг, размещенному на странице https://partners.gosuslugi.ru/ catalog/api\_for\_gu). |
| service\_not\_found | Не найдена услуга, заданная кодом serviceCode в запросе. | Следует перепроверить параметр serviceCode в теле запроса на соответствие коду услуги, заданному в соответствующей спецификации услуги/сервиса. |

В случае ответа с иными (не описанными выше) ошибками необходимо составить обращение об инциденте и отправить его на адрес [sd@sc.digital.gov.ru](mailto:sd@sc.digital.gov.ru). Требования к информации, которая должна быть указана в обращении, приведены в Приложении 2 к Спецификации.

Приложение 5. Структура файла sign\_config.xml

Документы и бизнес-подписи к ним в файле sign\_config.xml должны быть указаны следующим образом:

* в файле sign\_config.xml может быть несколько блоков <signedDocument> (подписанный документ или документы);
* в элементах <signData> (вложенных в элементы <signedDocument>) указываются подписи к данному документу;
* в элементах <documentDescription> и <signFileDescription> могут быть указаны соответственно описание подписанного документа и описание конкретной подписи к нему.

Ниже приведены пример файла sign\_config.xml и xsd-схема.

Пример файла sign\_config.xml.

<?xml version=**"1.0"** encoding=**"utf-8"**?>

<signedAttachments

xmlns:xsi=**"http://www.w3.org/2001/XMLSchema-instance"**>

<signedDocument>

<documentFileName>**edkpXml.xml**</documentFileName>

<documentDescription>**ЭДКП**</documentDescription>

<signData>

<signFileName>**edkpXmlSellerSign.sig**</signFileName>

<signFileDescription>**Подпись продавца**</signFileDescription>

</signData>

<signData>

<signFileName>**edkpXmlBuyerSign.sig**</signFileName>

<signFileDescription>**Подпись покупателя**</signFileDescription>

</signData>

</signedDocument>

</signedAttachments>

XSD-схема:

<?xml version=**"1.0"** encoding=**"utf-8"**?>

<xs:schema xmlns:xsi=**"http://www.w3.org/2001/XMLSchema-instance"** xmlns:xs=**"http://www.w3.org/2001/XMLSchema"**>

<xs:element name=**"signedAttachments"**>

<xs:complexType>

<xs:sequence>

<xs:element name=**"signedDocument"** minOccurs=**"1"**>

<xs:complexType>

<xs:sequence>

<xs:element name=**"documentFileName"** type=**"xs:string"** minOccurs=**"1"**/>

<xs:element name=**"documentDescription"** type=**"xs:string"** minOccurs=**"0"**/>

<xs:element name=**"signData"** minOccurs=**"1"** maxOccurs=**"unbounded"** >

<xs:complexType>

<xs:sequence>

<xs:element name=**"signFileName"** type=**"xs:string"** minOccurs=**"1"** />

<xs:element name=**"signFileDescription"** type=**"xs:string"** minOccurs=**"0"**/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

История изменений документа

| **Версия** | **Дата** | **Имя автора** | **Комментарий** |
| --- | --- | --- | --- |
| **1.0** | 01.10.2021 | Куделина А.В. | Первая версия документа |
| **1.1** | 15.12.2021 | Куделина А.В. | Дополнены разделы 3.1, 4, 5, 7, 8 |
| **1.2** | 29.12.2021 | Куделина А.В. | Отредактированы разделы 5, 8 |
| **1.3** | 27.04.2022 | Трубников А.Л. | Отредактированы разделы 3.2, 3.4, 6 |
| **1.4** | 27.06.2022 | Лютенков А.В. | Отредактированы разделы 2, 3, 4, 5, 6, 7. Удален раздел 6 в предыдущей редакции (в новой редакции разделы 6 и 7 соответствуют разделам 7 и 8 предыдущей редакции). |
| **1.5** | 20.07.2022 | Лютенков А.В. | Множественные правки всего документа, таблиц. Составление новой схемы процесса подачи заявления. |
| **1.6** | 01.09.2022 | Трубников А.Л. | Добавлено описание минимального размера чанка. Добавлено описание размера последнего чанка. |
| **1.7** | 23.09.2022 | Трубников А.Л. | В разделе 1 поправлены названия регламентов, размещенных на странице <https://partners.gosuslugi.ru/catalog/api_for_gu>.  Добавлено название указанного в примере метода ЕСИА. |
| **1.8** | 21.10.2022 | Лютенков А.В. | Исправление ошибок предыдущих правок:   1. в параграф 3.1. в части описания процесса подачи заявления; 2. в параграф 3.1. в части описания назначения методов /order и /push; 3. в параграф 3.4. в части описания метода /push/chunked (корректировка о том, что наименования файлов должно соответствовать правилам, указанным в спецификации по конкретной услуге); 4. в параграф 3.4. в части размеров чанков метода /push/chunked; 5. изменена структура, добавлен новый параграф 3.2 про взаимодействие по ГОСТ TLS, нумерация параграфов в 3 разделе, начиная с 3.2 предыдущей редакции смещена на 1. |
| **1.9** | 19.12.2022 | Трубников А.Л. | В разделе 3.2 убрано описание подключения без ГОСТ TLS. Добавлено описание средств криптографической защиты.  В разделе 3.5 добавлено описание:   * требований к архиву; * нумерации чанков.   Добавлено общее описание документа.  В раздел 7.4 добавлены примеры отправки архива несколькими чанками. |
| **1.10** | 26.12.2022 | Трубников А.Л. | Добавлен раздел 7 – «Получение статусов заявлений»  Добавлен раздел 8.7 – «Примеры получения статусов заявлений» |
| **1.11** | 10.03.2023 | Могилевская С.В. | Отредактированы разделы 7 и 8.7 для методов getUpdatedAfter и getOrdersStatus.  Методы были изменены с POST на GET |
| **1.12** | 31.07.2023 | Лютенков А.В.,  Ефремов О.В. | Исправление ошибок предыдущих правок:   1. Дополнен раздел «Термины и определения»; 2. Уточнен алгоритм подключения по протоколу TLS; 3. Уточнена диаграмма в п.2; 4. Добавлена диаграмма в п.2.3; 5. Указаны ссылки на соответствующие связанные пункты Регламента; 6. Переработано приложение 2; 7. Добавлены приложения 3 и 4 |
| **1.12.1** | 14.11.2023 | Ефремов О.В. | 1. Изменены требования к размерам чанков в пп. 2, 2.1.1, 2.1.3 и 2.1.4 (5 000 000 байт – минимальный, 50 000 000 байт - максимальный). 2. Исправлены опечатки в пп.2.3.1, 2.3.2 и 5.4 (удален символ «/» после getOrdersStatus и getUpdatedAfter в URL соответствующих методов) |
| **1.13** | 18.07.2024 | Ефремов О.В. | 1. Дополнен п.1.3 (указаны мнемоники ЕПГУ для доступа к спецификации OpenAPI в тестовой и продуктивной средах); 2. Указаны новые REST-методы для скачивания файлов в п.2 (таблица 3) и в п.4; 3. В п.2 описан принцип формирования полных URL при вызове методов API ЕПГУ; 4. Описана возможность передавать несколько подписей к документу в пп. 2.1.3 и 2.1.4; 5. Изменены (уточнены) названия методов в пп. 2.1.2, 2.1.3, 2.1.4 и по тексту; 6. В п.2.4 описан новый элемент message\_id; 7. В пп. 2, 2.1.1, 2.1.3 и 2.1.4 указаны новые значения минимального и максимального размеров чанков; 8. Изменено Приложение 1: удален статус VALIDATION, добавлен статус CREATE\_ORDER\_FORBIDDEN; 9. Добавлено приложение 5; 10. Редакционные правки |

1. При разделении на чанки необходимо использовать именно разделение по байтам, а не создавать из исходного файла многотомный архив. Подробнее о работе с чанками см. в п. 2.1.3 Спецификации. [↑](#footnote-ref-1)
2. См. п.4.1.2 официальной спецификации TLS v1.3, размещенной на странице https://datatracker.ietf.org/doc/html/rfc8446. [↑](#footnote-ref-2)
3. Данная команда должна использоваться именно в среде ОС семейства Unix, в среде ОС Windows она не даст результатов. [↑](#footnote-ref-3)
4. Регламент подключения к API ЕПГУ выложен на ресурсе <https://partners.gosuslugi.ru/catalog/api_for_gu>. [↑](#footnote-ref-4)
5. Спецификации услуг выложены на ресурсе <https://partners.gosuslugi.ru/catalog/api_for_gu> в разделе «Технические спецификации». [↑](#footnote-ref-5)
6. В спецификациях услуг на время переходного периода по уточнению и синхронизации терминологии можеи быть указано другое название данного метода. [↑](#footnote-ref-6)
7. Параметр необязателен в ситуации, когда отправляется только один чанк, в остальных случаях – обязателен. При этом требование к максимальному размеру чанка (не более 50 000 000 байт) – сохраняется. [↑](#footnote-ref-7)
8. Указывается для часового пояса GMT+3 (Москва). [↑](#footnote-ref-8)
9. Используемая статусная модель ЕПГУ и ее соотнесение со статусной моделью конкретной услуги (ведомственными статусами) содержатся в спецификации соответствующей услуги. [↑](#footnote-ref-9)
10. Статусные модели конкретных услуг закреплены в соответствующих спецификациях. Ведомственными именуются те статусы, которые используются конкретными ведомственными ИС при изменении статуса заявления на ЕПГУ при взаимодействии через СМЭВ в соответствии с ВС определенной услуги. [↑](#footnote-ref-10)
11. При вызове данного метода проверяется наличие у контрагента (т.е. для указанных услуги и типа пользователя) прав на работу с персональными данными. При наличии таких прав файлы по заявлению скачиваются, в противном случае возвращается ошибка 403 с кодом «access\_denied\_personal\_data». [↑](#footnote-ref-11)
12. Не путать с номером заявления. У результатов по заявлению objectId не равен номеру заявления, это именно currentStatusHistoryId, то есть идентификатор для данного статуса. [↑](#footnote-ref-12)
13. Данные ограничения изначально задаются при подаче заявки на подключение к API ЕПГУ в соответствии с Приложением 4.3 к Регламенту подключения к API Единого портала государственных услуг, размещенному на странице https://partners.gosuslugi.ru/  
    catalog/api\_for\_gu). При необходимости может быть подана заявка об увеличении текущих ограничений, ее необходимо направить на адрес [sd@sc.digital.gov.ru](mailto:sd@sc.digital.gov.ru) (см. подпункт 4.3.4 Приложения 4.3 к Регламенту подключения к API Единого портала государственных услуг). [↑](#footnote-ref-13)
14. Количества запросов по умолчанию определяются для указанных календарных (астрономических) промежутков времени. [↑](#footnote-ref-14)
15. В спецификации услуги могут быть указаны иные ограничения. [↑](#footnote-ref-15)