**Спецификация**

**API ЕПГУ**

**версия 1.11**

Оглавление

[Термины и определения 3](#_Toc123309373)

[1. Требования к получению маркера доступа 4](#_Toc123309374)

[2. Получение справочных данных 5](#_Toc123309375)

[3. Подача заявления 7](#_Toc123309376)

[3.1. Описание процесса подачи заявления 7](#_Toc123309377)

[3.2. Реализация подключения по ГОСТ TLS 10](#_Toc123309378)

[3.3. Создание заявления 11](#_Toc123309379)

[3.4. Загрузка архива 12](#_Toc123309380)

[3.5. Загрузка архива по частям 14](#_Toc123309381)

[4. Отмена заявления 16](#_Toc123309382)

[5. Получение деталей по заявлению 17](#_Toc123309383)

[6. Скачивание файла 24](#_Toc123309384)

[7. Получение статусов заявлений 25](#_Toc123309385)

[7.1. Получение статусов всех заявлений с даты обновления статуса. 25](#_Toc123309386)

[7.2. Получение статусов заявлений по переданному списку заявлений. 26](#_Toc123309387)

[8. Примеры запросов и ответов 28](#_Toc123309388)

[8.1. Создание заявления 28](#_Toc123309389)

[8.2. Получение справочных данных 28](#_Toc123309390)

[8.3. Загрузка архива 29](#_Toc123309391)

[8.4. Загрузка архива по частям 30](#_Toc123309392)

[8.5. Отмена заявления 31](#_Toc123309393)

[8.6. Получение деталей по заявлению 31](#_Toc123309394)

[8.7. Получение статусов заявлений 34](#_Toc123309395)

[История изменений документа 37](#_Toc123309396)

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 криптоалгоритмов, 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 |
| 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 |
| ФРГУ | Федеральный реестр государственных услуг |
| ЭП | Электронная подпись |
| КЭП | Квалифицированная электронная подпись |
| ФСТЭК России | Федеральная служба по техническому и экспортному контролю |
| ФГИС | Федеральная государственная информационная система |

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 услуге/сервису.

1. 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).

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 справочников) необходимы в заявлении определены в спецификации конкретной услуги.

Для получения справочных данных необходимо вызвать следующий метод:

Таблица

|  |  |
| --- | --- |
| **Среда ЕПГУ** | **URL** |
| Тестовая среда SVCDEV | [https://svcdev-beta.test.gosuslugi.ru/api/nsi/v1/dictionary/{code}](https://pgu-uat-fed.test.gosuslugi.ru/api/nsi/v1/dictionary/%7Bcode%7D) |
| Продуктивная среда | [https://www.gosuslugi.ru/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 запросе необходимо передавать Content-Type = application/json.

В теле запроса должен передаваться JSON объект со следующими параметрами:

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| **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

|  |  |  |  |
| --- | --- | --- | --- |
| **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. Подача заявления

## Описание процесса подачи заявления

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 5 Мб, то отправка разбивается на этапы:
	1. Создание заявления через вызов метода /order;
	2. Отправка заявления и загрузка архива с файлами по заявлению 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 5Мб, то отправка происходит в один этап:
	1. Отправка заявления и загрузка архива с файлами по заявлению через вызов метода /push.

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



Рисунок . Общая схема подачи заявления



Рисунок 2. Схема работы с заявлениями

Для работы с заявлением доступны следующие методы:

Таблица

|  |  |
| --- | --- |
| **Метод** | **Назначение** |
| POST /api/gusmev/order | Создание заявления (используется в случае, если нам нужно вносить номер заявления в XML запроса или если размер zip-архива более 5 Мб) |
| POST /api/gusmev/push | Загрузка архива с файлами заявления (для случая, если нам не требуется вносить номер заявления ЕПГУ в XML запроса и размер zip архива не более 5Мб) |
| POST /api/gusmev/push/chunked | Загрузка архива с файлами заявления по частям (для случая, если нам требуется вносить номер заявления ЕПГУ в XML запроса или размер zip архива более 5Мб) |
| POST /api/gusmev/order/{orderId} | Получение данных заявления (информацию по статусам и другие детали заявления)  |
| POST /api/gusmev/order/{orderId}/cancel | Отмена заявления |
| GET /api/gusmev/files/{objectId}/{objectType}/download?mnemonic={mnemonic} | Скачать вложение из ответа ведомства |
| POST /api/gusmev/order/getUpdatedAfter/?pageNum={n}&pageSize={m}&updatedAfter={timestamp} | Получение статусов по всем заявлениям с даты обновления статуса |
| POST /api/gusmev/order/getOrdersStatus/?pageNum={n}&pageSize={m} | Получение статусов заявлений по переданному списку заявлений |

## 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

|  |  |
| --- | --- |
| **Среда ЕПГУ** | **URL** |
| Тестовая среда SVCDEV | 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).

## Создание заявления

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

Для создания заявления необходимо вызвать следующий метод:

Таблица 7

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

В заголовке Authorization запроса (тип Bearer) должен передаваться маркер доступа со скоупом «http://lk.gosuslugi.ru/api-order».

В заголовке запроса необходимо передавать Content-Type = application/json.

В теле запроса должен передаваться JSON объект со следующими параметрами:

Таблица 8

| **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 параметрами созданного заявления.

Таблица 9

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

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

Таблица 10

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

В случае ответа с http-код = 401 необходимо получить новый маркер доступа и повторить запрос.

## Загрузка архива

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 менее 5Мб.

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

Таблица 11

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

В заголовке Authorization запроса (тип Bearer) должен передаваться маркер доступа со скоупом «http://lk.gosuslugi.ru/api-order». В заголовке Content-Type необходимо указывать multipart/form-data.

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 КЭП, 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.

В теле запроса должны передаваться параметры:

Таблица 12

| **Параметр** | **Описание** | **Тип, значения** | **Обязательность** |
| --- | --- | --- | --- |
| meta | Метаинформация в формате Content-Type = application/json {"region":"36", "serviceCode":"10000092161", "targetCode":"-1959599598"} | 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 13

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

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

Таблица 14

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

В случае ответа с http-код = 401 необходимо получить новый маркер доступа и повторить запрос.

Список возможных ошибок:

Таблица 15

| **Текст** | **Условие** |
| --- | --- |
| access\_denied\_system | Доступ запрещен для системы |
| access\_denied\_service | Доступ запрещен для сервиса |
| access\_denied\_user | Доступ запрещен для данного типа пользователя |
| service\_not\_found | Услуга не найдена |
| order\_access | У пользователя нет прав для работы с текущим заявлением |

## Загрузка архива по частям

Этот метод необходимо использовать только в случае, если по ВС предусмотрено внесение данных о номере заявления ЕПГУ в XML запроса или размер загружаемого zip архива более 5Мб.

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

Таблица 16

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

В заголовке Authorization запроса (тип Bearer) должен передаваться маркер доступа со скоупом «http://lk.gosuslugi.ru/api-order». В заголовке Content-Type необходимо указывать multipart/form-data.

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 КЭП, 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.

В теле запроса должны передаваться параметры:

Таблица 17

| **Параметр** | **Описание** | **Тип, значения** | **Обязательность** |
| --- | --- | --- | --- |
| meta | Метаинформация в формате Content-Type = application/json {"region":"36", "serviceCode":"10000092161", "targetCode":"-1959599598"} | Строка | Да |
| chunk | Номер чанка | Число | Нет |
| chunks | Кол-во чанков | Число | Нет |
| orderId | Номер заявления | Число | Да |
| file | Данные загружаемого архива | Двоичные данные | Да |

Каждый чанк, кроме последнего, не может быть менее 1 Мб и не должен превышать ограничения в 5 Мб (включительно). Размер последнего чанка может быть менее 1 Мб.

Чанки можно отправлять в несколько потоков, при этом важна очередность отправки для первого и последнего чанка. Первым отправляется первый чанк, далее могут быть отправлены чанки без очередности, последним должен быть отправлен последний чанк.

Нумерация чанков осуществляется следующим образом:

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

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

После отправки всех чанков инициируется отправка заявления.

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

Таблица 18

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

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

Таблица 19

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

В случае ответа с http-код = 401 необходимо получить новый маркер доступа и повторить запрос.

Список возможных ошибок:

Таблица 20

| **Текст** | **Условие** |
| --- | --- |
| access\_denied\_system | Доступ запрещен для системы |
| access\_denied\_service | Доступ запрещен для сервиса |
| access\_denied\_user | Доступ запрещен для данного типа пользователя |
| service\_not\_found | Услуга не найдена |
| order\_access | У пользователя нет прав для работы с текущим заявлением |

1. Отмена заявления

Отмена заявления возможна только после получения от ведомства признака cancelAllowed со значением true. Данные признак будет указан в ответе метода получения деталей по заявлению (п.5).

Для отмены заявления требуется вызывать следующий метод:

Таблица 21

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

В заголовке Authorization запроса (тип Bearer) должен передаваться маркер доступа со скоупом «http://lk.gosuslugi.ru/api-order».

Таблица 22

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

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

http-код = 204 информирует, что заявление с указанным номером не найдено.

В случае ответа с http-код = 401 необходимо получить новый маркер доступа и повторить запрос.

http-код = 403 информирует о том, что у пользователя нет прав для работы с текущим заявлением.

В случае ответа с http-код = 400 необходимо проверить параметры запроса.

http-код = 409 информирует о невозможности обработать запрос.

Таблица 23

| **Текст** | **Условие** |
| --- | --- |
| order\_access | У пользователя нет прав для работы с текущим заявлением |
| cancel\_not\_allowed | Отмена заявления в текущем статусе невозможна |

Пример ответа при ошибке:

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

1. Получение деталей по заявлению

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

Таблица 24

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

В заголовке Authorization запроса (тип Bearer) должен передаваться маркер доступа со скоупом «http://lk.gosuslugi.ru/api-order».

{orderId} – номер заявления.

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

Таблица 25

|  |  |  |
| --- | --- | --- |
| **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 (для данных значений параметра embed массив будет пустой) | 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 (для данных значений параметра embed массив будет пустой) | 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 (при embed=STATUS\_HISTORY) | Массив |
| * date
 | Дата и время смены статуса | Строка дата и время |
| * cancelAllowed
 | Флаг наличия отмены | Логический |
| * unreadEvent
 | Признак прочтения | Логический |
| * sendMessageAllowed
 | Признак разрешения отправки сообщения | Логический |
| * deliveryCancelAllowed
 | Флаг наличия отмены доставки | Логический |
| * statusId
 | Код статуса | 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 | 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 (для данных значений параметра embed массив будет пустой) | 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 (для данных значений параметра embed массив будет пустой) | Массив объектов |
| serviceTargetId | Идентификатор цели | Строка |
| orderPayments | Информация о платежах (для данных значений параметра embed массив будет пустой) | Массив данных |
| unreadMessageCnt | Кол-во непрочитанных сообщений | Число |
| orderResponseFiles | Информация о файлах в ответе заявления (при embed=RESPONSE\_FILE) | 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 (при embed= ATTACHMENT\_FILE) | 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 КЭП | 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 (для данных значений параметра embed массив будет пустой) | 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-код = 204 информирует, что заявление с указанным номером не найдено.

В случае ответа с http-код = 401 необходимо получить новый маркер доступа и повторить запрос.

http-код = 403 информирует о том, что у пользователя нет прав для работы с текущим заявлением.

В случае ответа с http-код = 400 необходимо проверить параметры запроса.

http-код = 409 информирует о невозможности обработать запрос.

Пример ответа при ошибке:

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

1. Скачивание файла

Для скачивания файла необходимо вызвать следующий метод:

Таблица 26

|  |  |
| --- | --- |
| **Среда ЕПГУ** | **URL** |
| Тестовая среда | https://svcdev-beta.test.gosuslugi.ru/api/storage/v2/files/{objectId}/{objectType}/download?mnemonic={mnemonic} |
| Продуктивная среда | https://www.gosuslugi.ru/api/storage/v2/files/{objectId}/{objectType}/download?mnemonic={mnemonic} |

Метод запроса: GET

В заголовке Authorization запроса (тип Bearer) должен передаваться маркер доступа со скоупом «http://lk.gosuslugi.ru/api-order».

{objectId} – currentStatusHistoryId заявления.

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

{mnemonic} – мнемоника файла.

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 "».

1. Получение статусов заявлений

Получение статусов всех заявлений с даты обновления статуса.

Для получения списка статусов заявлений с даты их последнего обновления, необходимо вызвать следующий метод:

Таблица 27

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

В заголовке Authorization запроса (тип Bearer) должен передаваться маркер доступа со скоупом «http://lk.gosuslugi.ru/api-order».

pageNum={n} – номер запрашиваемой страницы.

pageSize={m} – количество записей на странице

updatedAfter={timestamp} – дата и время, после которых были обновлены статусы

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

Таблица 28

| **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 | Массив | Да |
| orderId | Номер заявления | Число | Да |
| orderSearchStatus | Признак нахождения статуса заявления. Может принимать значение FOUND/ NOT\_FOUND | Строка | Да |
| status | Информация о последнем статусе заявления | Массив | Нет |
| statusId | Код статуса | Число | Да |
| statusName | Наименование последнего статуса | Строка | Да |
| updated | Дата и время последнего обновления статуса | timestamp | Да |

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

Таблица 29

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

В случае ответа с http-код = 401 необходимо получить новый маркер доступа и повторить запрос.

Получение статусов заявлений по переданному списку заявлений.

Для получения статусов по списку заявлений, необходимо вызвать следующий метод:

Таблица 30

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

pageNum={n} – количество возвращаемых страниц.

pageSize={m} – количество записей на странице,

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

В заголовке Authorization запроса (тип Bearer) должен передаваться маркер доступа со скоупом «http://lk.gosuslugi.ru/api-order».

В заголовке запроса необходимо передавать Content-Type = application/json.

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

Таблица 31

| **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 | Массив | Да |
| orderId | Номер заявления | Число | Да |
| orderSearchStatus | Признак нахождения статуса заявления.Может принимать значение FOUND/ NOT\_FOUND | Строка | Да |
| status | Информация о последнем статусе заявления  | Массив | Нет |
| statusId | Код статуса | Число | Да |
| statusName | Наименование последнего статуса | Строка | Да |
| updated | Дата и время последнего обновления статуса | timestamp | Да |

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

Таблица 33

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

В случае ответа с http-код = 401 необходимо получить новый маркер доступа и повторить запрос.

1. Примеры запросов и ответов

Создание заявления

**Пример запроса:**

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

**Тело запроса:**

|  |
| --- |
| { "region":"36",  "serviceCode":"10000000113",  "targetCode":"-10000000113", } |

**Пример положительного ответа:**

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

**Пример отрицательного ответа:**

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

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 архива

**Пример запроса:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/pushПример form-data запроса:

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| meta | {"region":"36", "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 архива по частям

**Пример запроса отправки первого чанка:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| POST https://svcdev-beta.test.gosuslugi.ru/api/gusmev/push/chunkedПример form-data запроса:

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| 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/chunkedПример 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/chunkedПример form-data запроса:

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| orderId | 764255248 |
| file |  |

 |

**Пример ответа:**

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

## Отмена заявления

**Пример запроса:**

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

**Пример ответа:**

|  |
| --- |
| http-код = 200 - заявление отмененоhttp-код = 403 - у пользователя нет прав для этого действияhttp-код = 404 - заявление не найденоhttp-код = 405 - отмена невозможнаhttp-код = 401 - необходимо получить новый маркер доступа и повторить запрос |

## Получение деталей по заявлению

**Пример запроса:**

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

**Пример ответа:**

|  |
| --- |
| { "code": "OK", "message": null, "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\"}"} |

## 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**        }    ]} |

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 (таблица № 5);
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. |