

**ОПИСАНИЕ ИНТЕГРАЦИОННОГО СЕРВИСА ДЛЯ  
ВНЕШНИХ ВИС**

**2020**

Сервис доступен по адресу: IP/ExternalIntegrationService?wsdl.

Например: <http://localhost:8080/ExternalIntegrationService?wsdl>

## 1 МЕТОД ДЛЯ ОТДАЧИ ЗАЯВЛЕНИЙ ПО УСЛУГЕ + ВЕДОМСТВО GETORDERREQUEST

Взять для примера ведомство «Департамент архитектуры и градостроительства города Севастополя» с кодом ФРГУ 9200000000162958572.

И услугу «Выдача уведомления о решении комиссии по оспариванию кадастровой стоимости объектов недвижимости» с кодом ФРГУ 9200000000182015784.

Контрольный пример для запроса заявлений будет выглядеть следующим образом:

Ответ будет выглядеть следующим образом:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ext="http://smart-consulting.ru/ws/externalIntegration">
  <soapenv:Header/>
  <soapenv:Body>
    <ext:getOrderRequest>
      <ext:departmentId>9200000000162958572</ext:departmentId>
      <ext:serviceId>9200000000182015784</ext:serviceId>
      <!--Optional:-->
      <ext:useAtcfsFiles>true</ext:useAtcfsFiles>
    </ext:getOrderRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getOrderResponse xmlns="http://smart-consulting.ru/ws/externalIntegration">
      <objectId>b96a270d-4daf-1693-266f-46c99dddbdeb</objectId>
      <data>
        <CadastrNumber description="Кадастровый номер">121231234</CadastrNumber>
        <CadastrCost description="Кадастровая стоимость">650000</CadastrCost>
        <DateReport description="Дата отчета оценки рыночной стоимости">2019-08-07</DateReport>
        <RightAndObligations description="Затрагивающиеся права и обязанности заявителя">тест</RightAndObligations>
        <File1>1dddddcd1-9a2b-4d0b-b365-325d346b0ae0</File1>
        <File2>49b8be33-315b-43c7-a2e6-29f995ce82b8</File2>
        <File3>1638dcad-2a7b-466d-9d08-b41765a67f1b</File3>
        <File4>25d04943-3acd-447c-b28f-05fdf778f107</File4>
        <NumberReport description="Номер отчета оценки рыночной стоимости">3</NumberReport>
        <Check1 description="Соглашение">true</Check1>
```

```
<Check2 description="Достоверность">true</Check2>
<AddressObject description="Адрес объекта">
  <address>
    <cityType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">c</cityType>
    <districtCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">024</districtCode>
    <city xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Севастополь</city>
    <cityCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">055</cityCode>
    <districtType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">р-н</districtType>
    <okato xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">29246000049</okato>
    <regionCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">40</regionCode>
    <originalAddress xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">город Севастополь</originalAddress>
    <regionType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">обл</regionType>
    <localityType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">c</localityType>
    <level xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">VILLAGE</level>
    <localityCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">055</localityCode>
    <zipcode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">249370</zipcode>
    <oktmo xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">29646424111</oktmo>
    <parentGuid xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">A-862b8ddf-da38-417f-ade9-4d6db67de957</parentGuid>
    <district xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Хвастовичский</district>
    <kladr xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">4002400005500</kladr>
    <fullAddress xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Калужская область, р-н Хвастовичский, с. Севастополь</fullAddress>
    <guid xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">A-c71740fc-c2cd-49d9-8133-08aae5558fa5</guid>
    <region xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Калужская</region>
```

```
</address>
</AddressObject>
<OrderID description="Номер заявления в ВИС">1</OrderID>
  <RepresentativeCheck description="Являюсь представителем
заявителя">false</RepresentativeCheck>
  <CreateDate description="Дата и время подачи">2019-08-
12T15:27:00</CreateDate>
  <PlanDate>2019-10-04T00:00:00</PlanDate>
  <DateEnteringIntoSystem>2019-08-12T15:27:00</DateEnteringIntoSystem>

<DifferentCreateAndEnteringDateCkeck>false</DifferentCreateAndEnteringDateCkeck>
  <ApplicantType description="Тип заявителя">
    <Name description="Тип заявителя">Физическое лицо</Name>
    <Code description="Код">FL</Code>
    <ShortName description="Краткое наименование">ФЛ</ShortName>
  </ApplicantType>
  <InputChannel description="Канал подачи">
    <Name description="Способ подачи">Личный прием</Name>
    <Code description="Код">rv</Code>
  </InputChannel>
  <ApplicantFLFact>
    <Surname description="Фамилия">Pasechnik</Surname>
    <Name description="Имя">Andre</Name>
    <BirthDate description="Дата рождения">2019-06-01</BirthDate>
    <PermitDocSerial description="Серия">1111</PermitDocSerial>
    <PermitDocNumber description="Номер">111111</PermitDocNumber>
    <PermitDocIssueDate description="Дата выдачи">2019-06-
01</PermitDocIssueDate>
    <PermitDocIssuer description="Кем выдан">ФМС</PermitDocIssuer>
    <RegAddress description="Адрес регистрации">
      <address>
        <cityType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">c</cityType>
        <districtCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">024</districtCode>
        <city xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Севастополь</city>
        <cityCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">055</cityCode>
```

```
        <districtType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">п-н</districtType>
        <okato xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">29246000049</okato>
        <regionCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">40</regionCode>
        <originalAddress xmlns:ns2="http://smart-
consulting.ru/ws/externalIntegration" xmlns="">Город Севастополь</originalAddress>
        <regionType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">обл</regionType>
        <localityType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">с</localityType>
        <level xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">VILLAGE</level>
        <localityCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">055</localityCode>
        <zipcode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">249370</zipcode>
        <oktmo xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">29646424111</oktmo>
        <district xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Хвастовичский</district>
        <parentGuid xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">A-862b8ddf-da38-417f-ade9-4d6db67de957</parentGuid>
        <fullAddress xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Калужская область, р-н Хвастовичский, с. Севастополь</fullAddress>
        <kladr xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">4002400005500</kladr>
        <guid xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">A-c71740fc-c2cd-49d9-8133-08aae5558fa5</guid>
        <region xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Калужская</region>
        </address>
        </RegAddress>
        <SameAddress description="Фактический адрес совпадает адресом
регистрации">true</SameAddress>
        <PostAddress description="Почтовый адрес">
        <address>
        <cityType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">с</cityType>
        <districtCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">024</districtCode>
```

```

        <city xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Севастополь</city>
        <cityCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">055</cityCode>
        <districtType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">р-н</districtType>
        <okato xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">29246000049</okato>
        <regionCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">40</regionCode>
        <originalAddress xmlns:ns2="http://smart-
consulting.ru/ws/externalIntegration" xmlns="">город Севастополь</originalAddress>
        <regionType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">обл</regionType>
        <localityType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">с</localityType>
        <level xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">VILLAGE</level>
        <localityCode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">055</localityCode>
        <zipcode xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">249370</zipcode>
        <oktmo xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">29646424111</oktmo>
        <district xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Хвастовичский</district>
        <parentGuid xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">A-862b8ddf-da38-417f-ade9-4d6db67de957</parentGuid>
        <fullAddress xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Калужская область, р-н Хвастовичский, с. Севастополь</fullAddress>
        <kladr xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">4002400005500</kladr>
        <guid xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">A-c71740fc-c2cd-49d9-8133-08aae5558fa5</guid>
        <region xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">Калужская</region>
        </address>
    </PostAddress>
    <FactAddress description="Фактический адрес">
        <address>
            <cityType xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"
xmlns="">с</cityType>

```

<districtCode xmlns="">024</districtCode> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<city xmlns="">Севастополь</city> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<cityCode xmlns="">055</cityCode> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<districtType xmlns="">р-н</districtType> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<okato xmlns="">29246000049</okato> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<regionCode xmlns="">40</regionCode> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<originalAddress xmlns="">город Севастополь</originalAddress> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<regionType xmlns="">обл</regionType> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<localityType xmlns="">с</localityType> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<level xmlns="">VILLAGE</level> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<localityCode xmlns="">055</localityCode> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<zipcode xmlns="">249370</zipcode> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<oktmo xmlns="">29646424111</oktmo> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<district xmlns="">Хвастовичский</district> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<parentGuid xmlns="">A-862b8ddf-da38-417f-ade9-4d6db67de957</parentGuid> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<fullAddress xmlns="">Калужская область, р-н Хвастовичский, с. Севастополь</fullAddress> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<kladr xmlns="">4002400005500</kladr> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<guid xmlns="">A-c71740fc-c2cd-49d9-8133-08aae5558fa5</guid> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

<region xmlns="">Калужская</region> xmlns:ns2="http://smart-consulting.ru/ws/externalIntegration"

</address>

</FactAddress>

<Phone description="Телефон">+ 7 978 111 11 11</Phone>

<Email description="Электронная почта">pasik@mail.gd</Email>

```

    <PermitDocType description="Вид документа, удостоверяющего личность">
      <Name description="Наименование">Паспорт гражданина РФ</Name>
      <CodePGU description="Код ЕПГУ">1.0</CodePGU>
      <CodeSmartroute description="Код Smart-
route">008001001000</CodeSmartroute>
      <CodeSmartrouteFns description="Код Smart-route
ФНС">21</CodeSmartrouteFns>
      <Code description="Код">21</Code>
    </PermitDocType>
  </ApplicantFLFact>
</data>
</getOrderResponse>
</soap:Body>
</soap:Envelope>

```

**В случае, если у каких-либо полей не хватает description, то требуется задать перевод на схеме данных. Если требуется задать перевод для служебных пакетов (Main), то требуется обратиться к Смарт Консалтинг.**

После того, как внешняя ВИС забрала заявление — это можно наблюдать в интерфейсе по заявлениям [/#/vis/c5a6295c-e6dc-34bb-21c9-17e0c2ca33f3/02eedb49-9c28-7bc3-a0e6-b5763021aba4/](#)

Сменится атрибут «Внешняя ВИС забрала заявление», сменится статус на В работе, и добавится запись в журнал действий по заявлению.

Файлы отдаются в виде guid файла из файлового хранилища. Получение содержимого описано в Приложении 1 к данному документу.



## 2 МЕТОД ДЛЯ СМЕНЫ СТАТУСА ПО ЗАЯВЛЕНИЮ UPDATESTATUSREQUEST

Контрольный пример выглядит следующим образом:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ext="http://smart-consulting.ru/ws/externalIntegration">
  <soapenv:Header/>
  <soapenv:Body>
    <ext:updateStatusRequest>
      <ext:objectId>3d121fa9-4019-e257-cab2-c103935f7a94</ext:objectId>
      <ext:eventDate>1561463818879</ext:eventDate>
      <ext:eventCode>2</ext:eventCode>
      <ext:eventComment>2</ext:eventComment>
      <ext:eventFile filename="text.txt">U29tZSB0ZXh0IGluIG15IGZpbGU=</ext:eventFile>
    </ext:updateStatusRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Ответ:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <updateStatusResponse xmlns="http://smart-consulting.ru/ws/externalIntegration">
      <code>OK</code>
    </updateStatusResponse>
  </soap:Body>
</soap:Envelope>
```

Бизнес процесс, который применяет изменения к заявлению в платформе: [ВсеВИС][ExternalIntegrationService] Изменение статуса по заявлению eventCode может принимать следующие значения:

- 3 (Принято решение о предоставлении – финальный статус)
- 4 (Принято решение об отказе – финальный статус)
- 6 (Взято в работу)
- 7 (Промежуточный результат по заявлению)

При других значениях статуса – не будут применены изменения к заявлению. Также данный статус направляется на РПГУ/МФЦ при изменениях внешней ВИС.

### 3 ЗАГРУЗКА ДОКУМЕНТОВ

Тип элемента формы «FormFilefield» указывает на необходимость загрузки файла(-ов) пользователем:

```
<field>
  <elementName>020496b3-8394-b092-18dd-5f50f0ddeb49</elementName>
  <isMultiple>false</isMultiple>
  <formElement>
    <objectType>FormFilefield</objectType>
    <properties>
      <name>020496b3-8394-b092-18dd-5f50f0ddeb49</name>
      <bindPropertyName>testFile</bindPropertyName>
      <defaultVal/>
      <help/>
      <isMultiple>false</isMultiple>
      <isRequired>false</isRequired>
      <label>файловое поле</label>
      <labelWidth>255</labelWidth>
      <multipleMax/>
      <multipleMin/>
      <placeholder/>
      <position>
        <x>0</x>
        <y>0</y>
      </position>
      <readonly>false</readonly>
      <signApplet>false</signApplet>
      <info/>
      <allowScan>false</allowScan>
    </properties>
  </formElement>
  <propertyName>testFile</propertyName>
  <objectId>f760164b-8731-fc6d-bda8-fe11210762b4</objectId>
</field>
```

Все файлы должны передаваться в виде .zip-архива в поле «BinaryData» блока «AppDocument». Архив должен удовлетворять следующим требованиям:

- все элементы архива должны находиться в корне, т.е. архив не должен содержать директорий;

- в архиве должен находиться xml-файл с описанием архива – «description.xml».

XSD-схема файла с описанием архива вложений (файл «description.xml»):

```
<rev:AppliedDocuments xmlns:rev="http://smev.gosuslugi.ru/rev120315">
  <rev:AppliedDocument>
    <rev:CodeDocument>propertyName элемента, в который загружается файл</rev:CodeDocument>
    <rev:Name>Наименование файла при загрузке пользователем</rev:Name>
    <rev:Number>Порядковый номер файла в архиве</rev:Number>
    <rev:URL>Наименование файла в архиве</rev:URL>
    <rev:Type>MIME тип файла</rev:Type>
  </rev:AppliedDocument>
</rev:AppliedDocuments>
```

При отправке нескольких документов блок AppliedDocument внутри AppliedDocuments формируется для каждого документа.

Пример description.xml:

```
<rev:AppliedDocuments xmlns:rev="http://smev.gosuslugi.ru/rev120315">
  <rev:AppliedDocument>
    <rev:CodeDocument>ApplicantFL.PermitDoc.File</rev:CodeDocument>
    <rev:Name>Тестовый документ 1.docx</rev:Name>
    <rev:Number>1</rev:Number>
    <rev:URL>file_1</rev:URL>
    <rev:Type>application/vnd.openxmlformats-officedocument.wordprocessingml.document</rev:Type>
  </rev:AppliedDocument>
  <rev:AppliedDocument>
    <rev:CodeDocument>RepresentDocs$0.File</rev:CodeDocument>
    <rev:Name>Тестовый документ 2.docx</rev:Name>
    <rev:Number>2</rev:Number>
    <rev:URL>file_2</rev:URL>
    <rev:Type>application/vnd.openxmlformats-officedocument.wordprocessingml.document</rev:Type>
  </rev:AppliedDocument>
  <rev:AppliedDocument>
    <rev:CodeDocument>Representative.PermitDoc.File</rev:CodeDocument>
    <rev:Name>Тестовый документ 3.docx</rev:Name>
    <rev:Number>3</rev:Number>
    <rev:URL>file_3</rev:URL>
    <rev:Type>application/vnd.openxmlformats-officedocument.wordprocessingml.document</rev:Type>
  </rev:AppliedDocument>
</rev:AppliedDocuments>
```

Пример готового архива с вложениями:



ФЛ + представитель.zip

В случае, если свойство элемента isMultiple имеет значение true:

```

<field>
  <elementName>f7b48a51-8715-8665-af2f-c8546dcc4e23</elementName>
  <isMultiple>false</isMultiple>
  <formElement>
    <objectType>FormFilefield</objectType>
    <properties>
      <name>f7b48a51-8715-8665-af2f-c8546dcc4e23</name>
      <bindPropertyName>testListFile</bindPropertyName>
      <defaultVal/>
      <help/>
      <isMultiple>true</isMultiple>
      <isRequired>false</isRequired>
      <label>файловое поле</label>
      <labelWidth>255</labelWidth>
      <multipleMax/>
      <multipleMin/>
      <placeholder/>
      <position>
        <x>0</x>
        <y>0</y>
      </position>
      <readonly>false</readonly>
      <signApplet>false</signApplet>
      <info/>
      <allowScan>false</allowScan>
    </properties>
  </formElement>
  <propertyName>testListFile$</propertyName>
  <objectId>668620f3-d135-1d70-206d-4e3526950adc</objectId>
</field>

```

В описании архива вложений в CodeDocument для каждого файла указывается порядковый номер элемента в массиве после указания bindPropertyName элемента, например:

```

<rev:AppliedDocuments xmlns:rev="http://smev.gosuslugi.ru/rev120315">
  <rev:AppliedDocument>
    <rev:CodeDocument>SomeFile$0</rev:CodeDocument>
    <rev:Name>Тестовый документ 1.docx</rev:Name>
    <rev:Number>1</rev:Number>
    <rev:URL>file_1</rev:URL>
    <rev:Type>application/vnd.openxmlformats-officedocument.wordprocessingml.document</rev:Type>
  </rev:AppliedDocument>
  <rev:AppliedDocument>
    <rev:CodeDocument> SomeFile$1</rev:CodeDocument>
    <rev:Name>Тестовый документ 2.docx</rev:Name>
    <rev:Number>2</rev:Number>
    <rev:URL>file_2</rev:URL>
    <rev:Type>application/vnd.openxmlformats-officedocument.wordprocessingml.document</rev:Type>
  </rev:AppliedDocument>
</rev:AppliedDocuments>

```

## Описание

Большинство URL endpoints ожидают, что запрос будет содержать следующие HTTP заголовки и соответствующие значения (если не указано обратное):

- AtcFs-VisId – уникальный идентификатор, который выдается внешней системе для работы с файловым хранилищем;
- AtcFs-VisUser – имя пользователя, который инициировал запрос;
- AtcFs-RequestId – произвольный UUID;
- AtcFs-Sign – «ключ», который представляет собой md5 хеш от следующей строки: «AtcFs-VisId + \_ + AtcFs-VisUser + \_ + AtcFs-RequestId + \_ + visSecretKey», где visSecretKey представляет собой секретный ключ, который выдается внешней системе, для взаимодействия с файловым хранилищем.

В случае, если клиент не отправил определенный HTTP заголовок или отправил некорректные данные, то сервер ответит ошибкой авторизации (403), а также текстом самой ошибки, находящейся в теле HTTP ответа.

### Интерфейс работы с записями (Record)

Record - запись, которая хранит в себе единственное текстовое поле data. Записи применяются для хранения любой информации, которую также можно ассоциировать с одним или несколькими файлами.

#### 1. Создание записи

- Тип запроса: POST;
- URL: /records;
- Тело запроса - строка;
- Результат запроса: идентификатор созданной записи.

В таблице records будет создана новая запись.

#### 2. Чтение записи

- Тип запроса: GET;
- URL: /records/{recordId};
- Результат запроса: данные из записи с переданным в запросе идентификатором {recordId} в формате JSON.

Получение записи с идентификатором {recordId} из таблицы records. Если искомая запись не найдена, в результате будет возвращён статус 404 NOT\_FOUND.

#### 3. Просмотр информации о файлах, ассоциированных с записью

- Тип запроса: GET;
- URL: /records/{recordId}/files;
- Результат запроса: массив объектов - данных, описывающих файлы, в формате JSON.

Поиск записи с идентификатором {recordId} и возвращение информации о файлах, связанных с искомой записью. Если искомая запись не найдена, в результате будет возвращён статус 404 NOT\_FOUND.

Формат информации о файле:

```
{
  "id": "идентификатор файла",
  "fileName": "имя файла",
  "contentType": "тип файла",
  "size": размер файла,
  "signStatus": "статус подписи файла",
  "signInfo": "информация о подписи",
  "additionalInfo": "дополнительная информация",
  "signStatusMulti": [ массив данных о подписях ]
}
```

#### 4. Обновление записи

- Тип запроса: PUT;
- URL: /records/{recordId};
- Тело запроса - строка;
- Результат запроса: статус 200 ОК.

Обновление записи с идентификатором {recordId} из таблицы records. Если искомая запись не найдена, в результате будет возвращён статус 404 NOT\_FOUND.

#### 5. Удаление записи

- Тип запроса: DELETE;
- URL: /records/{recordId};
- Результат запроса: статус 200 ОК.

Удаление записи с идентификатором {recordId} из таблицы records. Если искомая запись не найдена, в результате будет возвращён статус 404 NOT\_FOUND.

### Интерфейс работы с файлами

#### 1. Сохранение файла

- Тип запроса: POST;
- URL: /files?recordId={recordId}&fileName={fileName};
- Тело запроса - контент файла;
- Результат запроса: идентификатор созданного файла.

В таблице files будет создана новая запись с соответствующими параметрами. Если файл является zip-архивом, будет предпринята попытка найти sig-файл внутри. Если найден, он будет проверен и к информации о файле будет добавлена информация о подписи. Параметр fileName является обязательным. Если указан recordId, то файл будет

ассоциирован с соответствующей записью. Если запись с указанным идентификатором не будет найдена, будет возвращена ошибка 400 BAD\_REQUEST. Если запись найдена, но с ней уже ассоциирован файл с именем fileName, будет возвращена ошибка 400 BAD\_REQUEST.

## 2. Чтение файла

- Тип запроса: GET;
- URL: /files/{fileId};
- Результат запроса: контент файла.

Получение файла с идентификатором {fileId}. Если искомый файл не найден, в результате будет возвращён статус 404 NOT\_FOUND.

## 3. Получение информации о файле

- Тип запроса: GET;
- URL: /fileinfo/{fileId};
- Результат запроса: объект - информация о файле.

Получение информации о файле с идентификатором {fileId}. Если искомый файл не найден, в результате будет возвращён статус 404 NOT\_FOUND.

Формат информации о файле:

```
{
  "id":          "идентификатор      файла",
  "fileName":    "имя                файла",
  "contentType": "тип                  файла",
  "size":        размер              файла,
  "signStatus": "статус      подписи    файла",
  "signInfo":    "информация о      подписи",
  "additionalInfo": "дополнительная информация",
  "signStatusMulti": [ массив данных о подписях ]
}
```

## 4. Получение временной ссылки на скачивание файла

- Тип запроса: GET;
- URL: /tmpFileLink/{fileId};
- Результат запроса: строка-ссылка на скачивание файла.

Формирование ссылки на скачивание файла. Время жизни ссылки - 5 секунд. Если файл не найден, в результате будет возвращён статус 404 NOT\_FOUND.

## 5. Скачивание файла по временной ссылке

- Тип запроса: GET;
- URL: /tmpFiles/{tmpFileLink};
- Результат запроса: контент файла.

Получение файла по временной ссылке. Если ссылка не найдена, в результате будет возвращён статус 404 NOT\_FOUND.

## 6. Удаление файла

- Тип запроса: DELETE;
- URL: /files/{fileId};
- Результат запроса: статус 200 OK.

Удаление файла с идентификатором {fileId}. Если файл не найден, в результате будет возвращён статус 404 NOT\_FOUND.

## 7. Проверка и создания информации о подписи файла

- Тип запроса: POST;
- URL: /filesign/{fileId};
- Результат запроса: объект - информация о файле.

Возвращает информацию о файле, если информация о подписи уже существует. Если отсутствует, формируются новые данные. Если файл является zip-архивом, будет предпринята попытка найти sig-файл внутри. Если найден, он будет проверен и к информации о файле будет добавлена информация о подписи.

## 8. Обновление файла

- Тип запроса: POST;
- URL: /filesign?fileId={fileId}&fileName={fileName};
- Тело запроса - контент файла;
- Результат запроса: идентификатор созданного файла.

В таблице files будет обновлена запись с соответствующими параметрами. Параметры fileId и fileName является обязательным. Если файл является zip-архивом, будет предпринята попытка найти sig-файл внутри. Если найден, он будет проверен и к информации о файле будет добавлена информация о подписи. Если указан recordId, то файл будет ассоциирован с соответствующей записью. Если запись с указанным идентификатором не будет найдена, будет возвращена ошибка 400 BAD\_REQUEST. Если запись найдена, но с ней уже ассоциирован файл с именем fileName, будет возвращена ошибка 400 BAD\_REQUEST.

## Интерфейс мониторинга

### 1. Мониторинг

- Тип запроса: GET;
- URL: /monitoring;
- Результат запроса: html страница.

Получение данных о состоянии сервера.

### 2. Расширенный мониторинг



- Тип запроса: GET;
- URL: /monitoring/ext;
- Результат запроса: html страница.

Получение расширенных данных о состоянии сервера.