

УТВЕРЖДАЮ

Директор республиканского
унитарного предприятия
«Национальный центр электронных
услуг»



Р.А. Градусов
«_____» _____ 2021 г.

СОГЛАСОВАНО

Председатель Государственного
таможенного комитета
Республики Беларусь



В.Н. Орловский
«_____» _____ 2021 г.

**Технические условия
взаимодействия информационных систем заинтересованных лиц с
информационной системой таможенных органов через
общегосударственную автоматизированную информационную систему
(при передаче документов, указанных в таможенных документах либо
подтверждающие сведения, указанные в таможенных документах)**

Версия 1.0.

МИНСК
2021

СОДЕРЖАНИЕ

1. Сокращения.....	4
2. Описание веб-сервиса.....	4
3. Описание операций веб-сервисов.....	5
3.1. Отправка ЭПД в ОАИС	5
3.2. Получение списка запросов пользователя	7
3.2.1. По диапазону	7
3.2.2. По дате обновления.....	8
3.2.3. По диапазону дат обновления.....	9
3.2.4. По GUID файла.....	9
3.3. Получение информации о запросе пользователя	10
3.4. Получение списка сообщений, связанных с запросом.....	11
3.5. Получение содержимого сообщения	11
4. Описание кодов состояния HTTP	12
5. Описание порядка взаимодействия.....	13
5.1. Сценарий информационного обмена при передаче ЭПД.....	13
5.1.1. Формирование и подписание ЭПД в ИСЗЛ	13
5.1.2. Подача ЭПД в ОАИС	15
5.1.3. Передача и обработка ЭПД в ИСТО. Передача сообщений, сформированных ИСТО, в ОАИС	16
5.1.4. Получение ИСЗЛ таможенных сообщений из ОАИС	16
6. Правила использования ЭЦП при подписании ЭТД.....	18
6.1. Общие положения	18
6.2. Порядок формирования ЭЦП	18
6.3. Порядок проверки ЭЦП.....	20
6.4. Использование ЭЦП в прикладных электронных документах	21
6.5. Нормативные документы по ЭЦП.....	22
7. Требования при взаимодействии с ОАИС	22
Приложение А	23
Приложение Б	33
Приложение В.....	40
Приложение Г	42

Аннотация

Настоящие технические условия разработаны с целью обеспечения взаимодействия между информационными системами заинтересованных лиц и информационной системой таможенных органов посредством общегосударственной автоматизированной информационной системы и определяют порядок передачи информации, форматы и структуры передаваемых данных, а также смысловое содержание кодов информации.

Технические условия могут быть расширены в процессе внедрения и развития взаимодействия между информационными системами заинтересованных лиц и информационной системой таможенных органов посредством общегосударственной автоматизированной информационной системы .

1. Сокращения

Используемые сокращения приведены в таблице 1.1.

Таблица 1.1

Сокращение	Расшифровка
API	Application Programming Interface
GUID	Globally Unique Identifier - статистически уникальный 128-битный идентификатор
HTTP	HyperText Transfer Protocol – протокол передачи гипертекста
REST	Representational State Transfer – архитектурный стиль взаимодействия компонентов распределённого приложения в сети
XML	eXtensible Markup Language - расширяемый язык разметки
ЭЦП	Электронная цифровая подпись
ОАИС	Общегосударственная автоматизированная информационная система
ЭТД	Электронный таможенный документ
ИСЗЛ	Информационная система заинтересованных лиц
ИСТО	Информационная система таможенных органов
ЭПД	Электронные прилагаемые документы – документы указанные в таможенных документах, либо подтверждающие сведения, заявленные в таможенных документах

2. Описание веб-сервиса

Используется архитектурный стиль REST. В рамках REST взаимодействий транзакции происходят на сервере с помощью HTTP запросов и ответов.

Абсолютная часть указателя ресурса имеет вид:

https://<host>:<port>/ServiceISZL/ecd/v2 ,

где:

- https – сетевой протокол, используемый для обращения к ресурсу;
- host – полностью прописанное доменное имя хоста в системе DNS или IP-адрес хоста в формате четырех групп десятичных чисел, разделенных точками; числа – целые в интервале от 0 до 255;
- port – порт хоста для подключения;
- ServiceISZL – идентификатор приложения;
- ecd – наименование ресурса;
- v2 – постоянная часть указателя, определяющая версию API.

Абсолютная часть остается неизменной для ресурсов веб-сервиса. Используемая кодировка для запросов – UTF-8. В ответных сообщениях от API в заголовке Content-Language будет указан используемый язык (например, ru).

При вызове методов сервиса в заголовке запроса необходимо передавать:

- токен авторизации ИСЗЛ (выдается после успешного прохождения тестирования взаимодействия ИСЗЛ с ОАИС);
- идентификационный номер инициатора услуги (выдается каждому пользователю при заключении договора на оказание услуги).

Список передаваемых в заголовке запроса параметров приведен в таблице 2.1.

Таблица 2.1

Название параметра заголовка	Описание параметра	Значение	Обязательное
Authorization	Токен авторизации ИСЗЛ	Bearer <токен>	Да
UserId	Идентификационный номер инициатора услуги	строка	Да

При передаче неверного токена авторизации будет возвращен HTTP код 401, а тело ответа содержать следующее сообщение:

```
<ams:fault xmlns:ams="http://wso2.org/apimanager/security">
  <ams:code>900901</ams:code>
  <ams:message>Invalid Credentials</ams:message>
  <ams:description>Access failure for API: /ServiceISZL/ecd/v1, version: v1
status: (900901) - Invalid Credentials. Make sure you have given the correct
access token</ams:description>
</ams:fault>
```

3. Описание операций веб-сервисов

3.1. Отправка ЭПД в ОАИС

Относительная часть идентификатора ресурса имеет вид:

POST /request/{file_guid}?&doc_guid={doc_guid}&remark={remark}

При отправке запроса необходимо указать заголовок «Content-Type: application/xml». В теле запроса необходимо передавать содержимое ЭПД в формате XML:

```
<?xml version="1.0" encoding="utf-8"?>
< PD>
  <Declarant ID=" ... ">
    Тело ЭПД в соответствие со структурой XML-документа
  </Declarant>
  <Signature> ... </Signature>
```

</PD>

Тэг <Signature> является обязательным в случае использования ЭЦП при подписании ЭПД.

Сообщению, содержащему ЭПД, участником обмена присваивается GUID, представляющий собой шестнадцать байтный двоичный массив длиной 36 символов, структура которого приведена на рисунке 1.

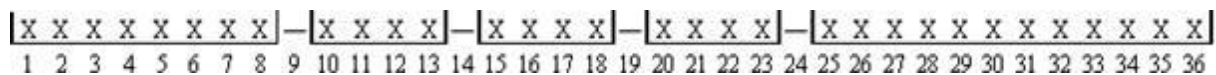


Рисунок 1 – Структура уникального идентификатора

Метод осуществляет загрузку ЭПД в ОАИС. При успешном сохранении ЭПД метод возвращает информацию о созданном объекте.

Входные параметры приведены в таблице 3.1.

Таблица 3.1

Имя	Описание	Значение	Обязательное
doc_guid	GUID таможенного документа, к которому привязаны ЭПД	GUID	Нет
file_guid	GUID отправляемого файла	GUID	Да
remark	Исходящий номер документа	строка	Нет

Структура возвращаемого сообщения:

```

{
  "request": [
    {
      "id": "...",
      "status_id": "...",
      "date_update": "...",
      "comment": "..."
    }
  ]
}
  
```

Описание полей возвращаемого сообщения приведено в таблице 3.2.

Таблица 3.2

Имя	Описание	Значение	Мн.
id	Идентификатор запроса в ОАИС	число	[1]

status_id	Код статуса запроса	число	[1]
date_update	Дата и время обновления записи	дата (YYYY-MM-DDThh:mm:ss)	[1]
comment	Замечания при принятии ЭПД в ОАИС	строка	[0..1]

Список кодов статусов запроса приведены в таблице 3.3.

Таблица 3.3

Код статуса	Описание статуса
0	документы переданы в ОАИС и ожидают отправки в ИСТО
1	документы переданы в ИСТО и находятся в обработке
2	отказ принятия документов в ИСТО
3	успешное принятие документов в ИСТО
9	ошибка обработки

3.2. Получение списка запросов пользователя

3.2.1. По диапазону

Относительная часть идентификатора ресурса имеет вид:

GET /requests?offset={offset}&limit={limit} .

Метод возвращает отсортированный по убыванию даты отправления список запросов данного пользователя с заданным смещением. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.4.

Таблица 3.4

Имя	Описание	Значение	Обязательно
offset	Смещение относительно последней отправленной записи	число, по умолчанию 0	Нет
limit	Ограничение на количество возвращаемых записей.	число, [0..100], по умолчанию 100	Нет

Структура возвращаемого сообщения:

<pre>{ "requests": [{</pre>

```

    "id": "...",
    "status_id": "...",
    "file_guid": "...",
    "ed_type": "...",
    "date_of": "...",
    "date_update": "...",
    "reg_no": "...",
    "app_no": "...",
    "date_reg": "...",
    "date_app": "...",
  },
]
}

```

Описание полей возвращаемого сообщения приведено в таблице 3.5.

Таблица 3.5

Имя	Описание	Значение	Мн .
id	Идентификатор запроса	число	[1]
status_id	Код статуса запроса	число	[1]
file_guid	GUID отправленного файла	GUID	[1]
ed_type	Тип ЭТД	строка «ЭПД»	[1]
date_of	Дата и время отправления ЭПД в ОАИС	дата (YYYY-MM-DDThh:mm:ss)	[1]
date_update	Дата и время обновления записи	дата (YYYY-MM-DDThh:mm:ss)	[1]

3.2.2. По дате обновления

Относительная часть идентификатора ресурса имеет вид:

GET /requests?date_update={date_update}&limit={limit} .

Метод возвращает список запросов данного пользователя, которые были обновлены после переданной даты. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.6.

Таблица 3.6

Имя	Описание	Значение	Обязательно
date_update	Дата и время обновления последнего полученного запроса	дата (YYYY-MM-DDThh:mm:ss)	Да
limit	Ограничение на количество	число, [0..100],	Нет

	возвращаемых записей.	по умолчанию 100	
--	-----------------------	------------------	--

Структура возвращаемого сообщения приведена в п. 3.2.1.

Описание полей возвращаемого сообщения приведено в таблице 3.5.

3.2.3. По диапазону дат обновления

Относительная часть идентификатора ресурса имеет вид:

GET /requests?date_from={date_from}&date_to={date_to} .

Метод возвращает список запросов данного пользователя, которые были обновлены в указанный период. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.7.

Таблица 3.7

Имя	Описание	Значение	Обязательно
date_from	Дата начала диапазона	дата (YYYY-MM-DDThh:mm:ss)	Да
date_to	Дата конца диапазона	дата (YYYY-MM-DDThh:mm:ss)	Да
limit	Ограничение на количество возвращаемых записей.	число, [0..100], по умолчанию 100	Нет

Структура возвращаемого сообщения приведена в п. 3.2.1.

Описание полей возвращаемого сообщения приведено в таблице 3.5.

3.2.4. По GUID файла

Относительная часть идентификатора ресурса имеет вид:

GET /requests?file_guid={file_guid}&limit={limit} .

Метод возвращает список запросов данного пользователя, содержащие заданный GUID файла. За один запрос возвращает не больше 100 записей.

Входные параметры приведены в таблице 3.8.

Таблица 3.8

Имя	Описание	Значение	Обязательно
file_guid	GUID отправленного файла	GUID	Да
limit	Ограничение на количество возвращаемых записей.	число, [0..100], по умолчанию 100	Нет

Структура возвращаемого сообщения приведена в п. 3.2.1.

Описание полей возвращаемого сообщения приведено в таблице 3.5.

3.3 Получение информации о запросе пользователя

Относительная часть идентификатора ресурса имеет вид:

GET /request/{rq_id} .

Метод возвращает информацию по заданному запросу пользователя.

Входные параметры приведены в таблице 3.9.

Таблица 3.9

Имя	Описание	Значение	Обязательное
rq_id	Идентификатор запроса	число	Да

Структура возвращаемого сообщения:

```
{
  "requests": {
    "id": "...",
    "status_id": "...",
    "file_guid": "...",
    "ed_type": "...",
    "date_of": "...",
    "date_update": "...",
    "reg_no": "...",
    "app_no": "...",
    "date_reg": "...",
    "date_app": "...",
  }
}
```

Описание полей возвращаемого сообщения приведено в таблице 3.10.

Таблица 3.10

Имя	Описание	Значение	Мн .
id	Идентификатор запроса	число	[1]
status_id	Код статуса запроса	число	[1]
file_guid	GUID отправленного файла	GUID	[1]
ed_type	Тип ЭТД	строка «ЭПД»	[1]
date_of	Дата и время отправления ЭПД в ОАИС	дата (YYYY-MM-DDThh:mm:ss)	[1]
date_update	Дата и время обновления	дата (YYYY-MM-	[1]

	записи	DDThh:mm:ss)	
--	--------	--------------	--

3.4. Получение списка сообщений, связанных с запросом

Относительная часть идентификатора ресурса имеет вид:

GET /files/{rq_id} .

Метод возвращает список объектов, содержащих дату создания, тип, идентификатор сообщений, относящихся к заданному запросу пользователя.

Входные параметры приведены в таблице 3.11.

Таблица 3.11

Имя	Описание	Значение	Обязательное
rq_id	Идентификатор запроса	число	Да

Структура возвращаемого сообщения:

<pre>{ "files": [{ "ln_id": "...", "date_of": "...", "ln_type": "..." },] }</pre>

Описание полей возвращаемого сообщения приведено в таблице 3.12.

Таблица 3.12

Имя	Описание	Значение	Мн .
ln_id	Идентификатор связанного сообщения	число	[1]
date_of	Дата и время создания сообщения	дата (YYYY-ММ-DDThh:mm:ss)	[1]
ln_type	Тип сообщения	число (см. таблицу 5.1)	[1]

3.5. Получение содержимого сообщения

Относительная часть идентификатора ресурса имеет вид:

GET /file/{ln_id} .

Метод возвращает содержимое xml-сообщения (декларантов, таможни, технологических), имеющего заданный идентификатор и относящегося к заданному пользователю.

Входные параметры приведены в таблице 3.13.

Таблица 3.13

Имя	Описание	Значение	Обязательное
-----	----------	----------	--------------

ln_id	Идентификатор сообщения	число	Да
-------	-------------------------	-------	----

В теле ответа будет возвращено содержимое запрашиваемого сообщения в формате XML и с Content-Type: application/xml.

4. Описание кодов состояния HTTP

При работе веб-сервисов используются следующие интерпретации кодов состояния HTTP.

200 OK – успешный запрос. Если клиентом были запрошены какие-либо данные, то они находятся в теле ответа.

400 Bad Request – ошибка запроса, неверные параметры запроса либо запрос не может быть выполнен.

401 Unauthorized – не удалось произвести авторизацию.

403 Forbidden – запрос производится к ресурсу, к которому нет доступа.

404 Not Found – запрос производится к ресурсу, который не найден.

429 Too Many Requests – производится слишком много запросов.

502 Bad Gateway – шлюз вернул некорректный ответ.

503 Service Unavailable – сервис недоступен.

504 Gateway Timeout - шлюз не отвечает.

500 Internal Server Error – при выполнении запроса произошла ошибка (описание данной ошибки будет отправлено в теле ответа). Ответ имеет следующую структуру:

```
{
  "errId": "...",
  "errDescr": "..."
}
```

где:

- errId - код результата обработки;
- errDescr - описание результата обработки.

Примеры значений параметров errId и errDescr приведены в таблице 4.1.

Таблица 4.1

Код результата обработки (errId)	Описание результата обработки (errDescr)
2	Неверный код типа сообщения.
3	Передача сообщений данного типа запрещена для текущего пользователя.
6	Пользователь заблокирован.

10	Сообщение с данным идентификатором файла передано в ОАИС ранее. Для повторной отправки создайте сообщение с новым идентификатором файла.
12	Сообщение не подписано.
31	Выявлено несоответствие лица, представляющего электронный документ, с лицом, указанным в сертификате открытого ключа. Электронный документ в таможенный орган не направлен. Представьте электронный документ с корректно заполненными сведениями.
100	Общая ошибка.
101	Ошибка авторизации (отсутствует заголовок UserId)
102	Отсутствует параметр запроса
103	Недопустимое значение параметра
104	Запись не найдена
105	Ошибка при разборе сообщения

Получение кода состояния HTTP отличного от 200 необходимо интерпретировать как неуспешный запрос.

5. Описание порядка взаимодействия

5.1. Сценарий информационного обмена при передаче ЭПД

Сценарий взаимодействия между ИСЗЛ и ИСТО через ОАИС при передаче ЭПД включает в себя следующие операции:

- подача ЭПД от ИСЗЛ в ОАИС;
- проверка при приеме ЭПД в ОАИС на обязательное заверение ЭПД ЭЦП.
- проверка при приеме в ОАИС ЭПД на полномочия лица на представление ЭПД.
- передача по запросу сообщений от ОАИС в ИСЗЛ (отказ в передаче ЭПД в ИСТО по причине не заверения ЭПД ЭЦП/по причине отсутствия полномочий на представление ЭПД);
- передача ЭПД от ОАИС в ИСТО;
- обработка ЭПД в ИСТО (результаты рассмотрения ЭПД; формирование сведений о результатах рассмотрения ЭПД).
- передача сообщений, сформированных ИСТО, в ОАИС;
- получение ИСЗЛ сообщений, сформированных ИСТО, из ОАИС.

5.1.1. Формирование и подписание ЭПД в ИСЗЛ

5.1.1.1. ИСЗЛ формирует ЭПД в формате XML и в соответствии со структурой.

Структура первичного ЭПД (элемент DocumentsArhiveRequest):

Элемент	Описание элемента	Тип	Мн.
---------	-------------------	-----	-----

1	RegNumber	Регистрационный номер таможенного документа, к которому относится ЭПД	xs:string	[1]
2	DocType	Тип таможенного документа, к которому относится ЭПД. Допустимые значения в Приложении Г.	xs:string	[1]
3	Sender	Информация об отправителе		[1]
	1 UNP	УНП	xs:string	[0..1]
	2 Name	Наименование	xs:string	[0..1]
4	Files	Загружаемые документы		[1]
	1 Document	Загружаемый документ		[1..n]
	1 Name	Имя файла	xs:string	[1]
	2 DocumentInfo	Информация о документе		[1]
	1 Column	Графа таможенного документа в которой указан файл	xs:string	[0..1]
	2 DocCode	Код вида документа: КЛАССИФИКАТОР ВИДОВ ДОКУМЕНТОВ И СВЕДЕНИЙ. Приложение N 8 к Решению Комиссии Таможенного союза от 20 сентября 2010 г. N 378	xs:string	[1]
	3 DocNumber	Номер документа	xs:string	[1]
	4 DocDate	Дата документа	xs:string	[1]
	5 DocType	Тип документа: 1 – указан в таможенном документе, к которому привязаны ЭПД 2 – не указан в таможенном документе, к которому привязаны ЭПД	xs:string	[1]
	3 ItemNumb	Номера товаров через запятую (без пробелов)	xs:string	[0..1]
	4 Extension	Расширение файла: pdf или tiff	xs:string	[1]
	5 Size	Размер файла: не должен превышать 15 Мб	xs:string	[1]
	6 MD5	Контрольная сумма по алгоритму хеширования MD5 (для проверки целостности файла)	xs:string	[1]

	7	ZIP	ZIP архив в кодировке BASE64	xs:string	[1]
	a	атрибут ID	уникальный идентификатор файла	xs:any	[0..1]

Структура ЭПД в ответ на требования (элемент DocumentsArchiveRequirement):

Элемент		Описание элемента	Тип	Мн.
1	Files	Загружаемые документы		[1]
	1 Document	Загружаемый документ		[1..n]
	1 Name	Имя файла	xs:string	[1]
	2 Extension	Расширение файла: pdf или tiff	xs:string	[1]
	3 Size	Размер файла: не должен превышать 15 Мб	xs:string	[1]
	4 MD5	Контрольная сумма по алгоритму хеширования MD5 (для проверки целостности файла)	xs:string	[1]
	5 ZIP	ZIP архив в кодировке BASE64	xs:string	[1]
	6 Requirement	Номер требования	xs:string	[1]
	a атрибут ID	уникальный идентификатор файла	xs:any	[0..1]

В ZIP архиве, поименованном в п. 7 вышеуказанной структуры могут содержаться только файлы формата pdf или tiff.

5.1.2. Подача ЭПД в ОАИС

5.1.2.1. ИСЗЛ отправляет сформированные ЭПД, вызвав метод POST /request описанный в разделе 3.1, указав при этом:

- параметры для идентификации пользователя согласно разделу 2;
- заголовок «Content-Type: application/xml»;
- GUID отправляемого файла, сформированный в ИСЗЛ;
- GUID таможенного документа, к которому привязаны ЭПД (при наличии).

5.1.2.2. При возникновении ошибки принятия передаваемых в ОАИС ЭПД будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием возникшей ошибки. Структура этого JSON приведена в разделе 4.

5.1.2.3. Перед сохранением ЭПД в ОАИС будет осуществляться обработка ЭПД в соответствии с пунктами 5.1.2.4 - 5.1.2.5.

5.1.2.4. ОАИС осуществляет проверку ЭПД на обязательное заверение ЭЦП.

5.1.2.5. В ОАИС осуществляется дополнительная проверка лица, представляющего ЭПД в таможенный орган, на полномочия по их представлению.

5.1.2.6. В случае не прохождения любого контроля из пунктов 5.1.2.4 - 5.1.2.6 будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием причины. Структура этого JSON приведена в разделе 4.

5.1.2.7. При успешном прохождении предусмотренных пунктами 5.1.2.4 - 5.1.2.6 контролей в HTTP ответе будет возвращен статус 200, а в теле ответа будет содержаться информация о принятом документе (запросе) в формате JSON (описание полей возвращаемого сообщения приведено в таблице 3.2). Код статуса запроса (status_id) для данных ЭПД будет изменен на 0.

5.1.3. Передача и обработка ЭПД в ИСТО. Передача сообщений, сформированных ИСТО, в ОАИС

5.1.3.1. ОАИС, при успешном принятии ЭПД от ИСЗЛ, передает данные ЭПД в ИСТО.

5.1.3.2. В случае неуспешного принятия ИСТО ЭПД, ОАИС изменяет код статуса запроса (status_id) для данных ЭПД на 9.

5.1.3.3. В случае успешного принятия ИСТО ЭПД, ОАИС изменяет код статуса запроса (status_id) для данных ЭПД на 1.5.1.3.4. ИСТО, в случае успешного приема ЭПД, передает в ОАИС документ «Извещение о принятии документа» (DocumentAcceptanceNotice), при этом ОАИС меняет код статуса запроса (status_id) на 3.

Сообщение DocumentAcceptanceNotice заполняется в соответствии с таблицами Б.13, Б.14 Приложения Б.

5.1.3.5. ИСТО, при наличии оснований для отказа в приеме ЭПД, передает в ОАИС документ «Извещение об отказе принятия документа» (DocumentRejectionNotice), при этом ОАИС меняет код статуса запроса (status_id) на 2.

Сообщение DocumentRejectionNotice заполняется в соответствии с таблицами Б.11, Б.12 Приложения Б.

5.1.3.6. В случае необходимости исполнения требований ИСТО направляет в ОАИС документ «Извещение о необходимости исполнения требования» (DocumentRequirementNotice), при этом ОАИС меняет код статуса запроса (status_id) на 6.

Сообщение DocumentRequirementNotice заполняется в соответствии с таблицами Б.27, Б.28 Приложения Б.

5.1.4. Получение ИСЗЛ таможенных сообщений из ОАИС

5.1.4.1. Получение идентификатора запроса.

5.1.4.1.1. Для получения списка таможенных сообщений, переданных от ИСТО в ОАИС и привязанных к конкретному запросу (переданному от ИСЗЛ в ОАИС), необходим идентификатор этого запроса. Данный идентификатор запроса можно получить следующими способами:

- из поля «id» в ответном JSON при успешном сохранении ЭПД в ОАИС;

- из списка запросов пользователя.

5.1.4.1.2. Для получения списка запросов пользователя ИСЗЛ необходимо вызвать метод GET /requests согласно разделу 3.2 (с параметрами для идентификации пользователя согласно разделу 2).

5.1.4.1.3. При успешном вызове метода в HTTP ответе будет возвращен статус 200, а в теле ответа будет содержаться список запросов пользователя в формате JSON. Описание полей возвращаемого сообщения приведено в таблице 3.5. Идентификатор запроса будет содержаться в поле «id».

5.1.4.1.4. При возникновении ошибки будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием возникшей ошибки. Структура этого JSON приведена в разделе 4.

5.1.4.1.5. При получении статуса ответа отличного от 200 операцию считать неуспешной.

5.1.4.2. Получение списка сообщений, связанных с запросом.

5.1.4.2.1. ИСЗЛ, путем вызова метода GET /files, описанного в разделе 3.4, получает список сообщений, связанных с запросом. При вызове метода указываются:

- параметры для идентификации пользователя согласно разделу 2;
- идентификатор запроса, полученный в п. 5.1.4.1.

5.1.4.2.2. При успешном вызове метода в HTTP ответе будет возвращен статус 200, а в теле ответа будет содержаться в формате JSON список сообщений, связанных с запросом. Описание полей возвращаемого сообщения приведено в таблице 3.12. Идентификатор сообщения содержится в поле «ln_id», тип документа содержится в поле «ln_type». Список значений поля «ln_type» приведен в таблице 5.1.

Таблица 5.1

Тип сообщения	Наименование сообщения
0	Исходный документ
2	Извещение об отказе в принятии документа в ИСТО
3	Извещение о принятии документа в ИСТО
6	Извещение о необходимости исполнения требования

5.1.4.2.3. При возникновении ошибки будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием возникшей ошибки. Структура этого JSON приведена в разделе 4.

5.1.4.2.4. При получении статуса ответа отличного от 200 операцию считать неуспешной.

5.1.4.3. Получение сообщения, связанного с запросом.

5.1.4.3.1. ИСЗЛ, путем вызова метода GET /file, описанного в разделе 3.5, получает содержимое сообщения, связанного с запросом. При вызове метода указываются:

- параметры для идентификации пользователя согласно разделу 2;
- идентификатор сообщения, полученный в п. 5.1.4.2.2.

5.1.4.3.2. При успешном вызове метода в HTTP ответе будет возвращен статус 200, в заголовке указан «Content-Type: application/xml», а в теле ответа будет содержимое запрашиваемого таможенного сообщения в формате XML.

5.1.4.3.3. При возникновении ошибки будет возвращен HTTP ответ со статусом 500, в теле которого будет содержаться JSON с описанием возникшей ошибки. Структура этого JSON приведена в разделе 4.

5.1.4.3.4. При получении статуса ответа отличного от 200 операцию считать неуспешной.

6. Правила использования ЭЦП при подписании ЭТД

6.1 Общие положения

Для формирования ЭЦП используется алгоритмы и средства криптографической защиты информации (СКЗИ), разрешенные к применению в таможенных органах Республики Беларусь.

Используемые алгоритмы и СКЗИ должны соответствовать следующим документам:

- СТБ 34.101.45-2013 «Информационная технология и безопасность. Алгоритмы электронной цифровой подписи и транспорта ключа на основе эллиптических кривых»;
- СТБ 34.101.31-2011 «Информационные технологии. Защита информации. Криптографические алгоритмы шифрования и контроля целостности»;

Порядок формирования и использования ЭЦП для обеспечения достоверности электронного документа определяется в соответствии со спецификацией W3C «XML-Signature Syntax and Processing» (подпись XML) и нормативными документами Республиканского удостоверяющего центра ГосСУОК.

6.2 Порядок формирования ЭЦП

6.2.1 Создание элемента Signature, который содержит четыре дочерних элемента: SignedInfo, SignatureValue, KeyInfo и Object. Этот элемент является единственным обертывающим элементом для других тегов цифровой подписи XML. Атрибут ID данного элемента должен иметь значение атрибута ID первого из подписываемых элементов, с добавлением перед ним символов «SID-».

6.2.2 Создание дочерних элементов элемента Object.

Элемент предназначен для указания дополнительных данных, которые должны быть подписаны. Данным документом определяется использование данного элемента для создания метки времени.

Атрибут ID элемента Object должен содержать значение атрибута ID элемента Signature с добавлением перед ним символа «Т».

Элемент Object должен содержать элемент SignatureProperties, который в свою очередь должен содержать элемент SignatureProperty, атрибут Target которого должен быть равен значению атрибута ID элемента Signature.

Элемент SignatureProperty должен содержать элемент SigningTime типа xs:datetime из пространства имен «http://lab119.net/STBCrypt», значение которого должно быть равно времени формирования ЭЦП в UTC.

6.2.3 Создание дочерних элементов элемента SignedInfo.

Элемент CanonicalizationMethod – обязательный элемент, который идентифицирует алгоритм канонизации, применяемой к элементу SignedInfo до создания подписи. Атрибут Algorithm данного элемента должен иметь значение «http://www.w3.org/TR/2001/REC-xml-c14n-20010315».

Элемент SignatureMethod – указывает алгоритм, используемый для создания подписи. Атрибут Algorithm данного элемента должен иметь значение «http://www.w3.org/2001/04/xmldsig-more#STB34101312011-STB34101452013».

Элементы Reference – определяет данные, которые будут подписаны, а также преобразования, которые необходимо применить к этим данным перед вычислением значения подписи.

Каждый элемент Reference должен содержать следующую информацию:

- атрибут URI – ссылка на подписываемый элемент внутри данного XML документа. Значение атрибута должно быть равно значению атрибута ID подписываемого элемента с добавлением перед ним символа «#»;
- элемент Transforms – определяет преобразования данных до их подписания. Должен содержать единственное преобразование (элемент Transform) – приведение к каноническому виду (атрибут Algorithm со значением «http://www.w3.org/TR/2001/REC-xml-c14n-20010315»);
- элемент DigestMethod – указывает алгоритм, используемый для создания дайджеста. Атрибут Algorithm данного элемента должен иметь значение «http://www.w3.org/2001/04/xmldsig-more#STB34101312011»;
- элемент DigestValue – значение дайджеста. Должен содержать значение функции хеширования СТБ 34.101.31-2011 в кодировке base64, входными данными для которой является содержимое элемента (включая открывающий и закрывающий тэги), ссылка на который указана в атрибуте URI элемента Reference, приведенное к каноническому виду.

Количество элементов Reference должно быть равно количеству подписываемых элементов. Дополнительно каждая подпись должна содержать элемент Reference со значением атрибута URI равным значению атрибута ID элемента Object с добавлением перед ним символа «#» (ссылка на метку времени).

6.2.4 Создание дочерних элементов элемента KeyInfo.

Элемент KeyInfo должен содержать элемент X509Data, содержимое которого содержит информацию о подписавшем. Элемент X509Data должен содержать следующие элементы:

- элемент X509IssuerSerial – информация об издателе и серийном номере сертификата подписавшего. Данное сочетание позволяет однозначно идентифицировать любой сертификат открытого ключа. Элемент X509IssuerSerial должен содержать элемент X509IssuerName, значение которого должно содержать имя издателя сертификата, и элемент X509SerialNumber, значение которого должно содержать серийный номер сертификата в десятичной системе счисления;
- элемент X509SKI – идентификатор открытого ключа субъекта. Должен содержать первые 20 байт идентификатора открытого ключа субъекта, вычисленного в соответствии с СТБ 34.101.45-2013, в кодировке base64;
- элемент X509Certificate – сертификат подписавшего. Должен содержать бинарное представление сертификата подписавшего в кодировке base64;
- элемент X509CRL (необязательный) – список отозванных сертификатов. Может содержать бинарное представление действующего на момент подписи списка отозванных сертификатов в кодировке base64.

6.2.5 Вычисление значения элемента SignatureValue.

Сформированный в п. 6.2.3 элемент SignedInfo приводится к каноническому виду, после чего обрабатывается функцией хеширования СТБ 34.101.31-2011. Полученное значение дайджеста передается на вход процедуре выработки ЭЦП СТБ 34.101.45-2013 с использованием личного ключа подписи, соответствующего открытому ключу подписи, содержащемуся в сертификате, указанному в элементе KeyInfo. Полученное значение ЭЦП помещается в элемент SignatureValue в кодировке base64.

6.3 Порядок проверки ЭЦП

6.3.1 Проверка значений дайджестов в элементах Reference.

Для каждого элемента Reference, содержащихся в элементе References элемента SignedInfo необходимо вычислить значение функции хеширования СТБ 34.101.31-2011, входными данными для которой является содержимое элемента (включая открывающий и закрывающий тэги), ссылка на который указана в атрибуте URI, приведенное к каноническому виду. Если вычисленное значение не совпадает с указанным в элементе DigestValue элемента Reference, принимается решение о недействительности ЭЦП.

6.3.2 Получение информации об открытом ключе подписи из элемента KeyInfo.

Сертификат открытого ключа подписи может быть получен:

- из элемента X509Certificate;

- из локального или сетевого хранилища сертификатов по информации об издателе и серийном номере сертификата;
- из локального или сетевого хранилища сертификатов по идентификатору открытого ключа субъекта.

Если вышеописанные способы не позволили получить сертификат подписавшего, принимается решение о невозможности проверки ЭЦП, и, следовательно, ее недействительности.

6.3.3 Проверка действительности сертификата на момент подписи.

Должны быть выполнены следующие проверки:

- построена цепочка сертификатов до сертификата корневого удостоверяющего центра ГосСУОК;
- проверен срок действия каждого из сертификатов в цепочке, при этом время подписи должно входить в срок действия каждого из сертификатов;
- проверена ЭЦП каждого сертификата с использованием открытого ключа подписи, указанного в сертификате издателя;
- проверен статус отзыва каждого из сертификатов в цепочке путем анализа соответствующих списков отозванных сертификатов на предмет нахождения в них проверяемых сертификатов, либо иными способами, определенными соответствующим УЦ (при этом может быть использован список отозванных сертификатов, указанный в элементе X509CRL).

При отрицательном результате любой из вышеуказанных проверок принимается решение о недействительности сертификата подписавшего на момент постановки подписи, и, следовательно, недействительности ЭЦП.

Время постановки подписи должно быть получено из элемента SigningTime. При невозможности определить время постановки подписи, в проверках используется текущее время.

6.3.4 Проверка значения ЭЦП указанного в элементе SignatureValue.

Элемент SignedInfo приводится к каноническому виду, после чего обрабатывается функцией хеширования СТБ 34.101.31-2011. Полученное значение дайджеста передается на вход процедуре проверки ЭЦП СТБ 34.101.45-2013 с использованием открытого ключа подписи, содержащегося в сертификате. При отрицательном результате процедуры проверки ЭЦП принимается решение о недействительности ЭЦП, при положительном – принимается решение о том, что ЭЦП действительна.

6.4 Использование ЭЦП в прикладных электронных документах

6.4.1 Общие положения.

ЭЦП на документ накладывает лицо, сформировавшее документ.

6.4.2 При формировании документа, лицо формирующее документ обязано подписать элемент Declarant. Сформированный элемент Signature должен быть помещен в корневой элемент PD ЭПД. Атрибут ID элемента Signature должен иметь значение равное значению атрибута ID элемента Declarant с добавлением перед ним символов «SID-».

6.4.3 При формировании электронного требования, сотрудник таможенного органа накладывает ЭЦП только на элемент NoticeInfo.

Сформированный элемент Signature должен быть помещен в корневой элемент PD ЭПД.

6.5 Нормативные документы по ЭЦП

Используются следующие нормативные документы по ЭЦП:

- СТБ 34.101.45-2013 «Информационная технология и безопасность. Алгоритмы электронной цифровой подписи и транспорта ключа на основе эллиптических кривых»;

- СТБ 34.101.31-2011 «Информационные технологии. Защита информации. Криптографические алгоритмы шифрования и контроля целостности»;

- СТБ 34.101.19-2012 «Информационные технологии и безопасность. Форматы сертификатов и списков отозванных сертификатов инфраструктуры открытых ключей»;

- СТБ 34.101.31-2011 «Информационные технологии. Защита информации. Криптографические алгоритмы шифрования и контроля целостности»;

- «Extensible Markup Language (XML) 1.0» <http://www.w3.org/TR/xml/>;

- «Namespaces in XML» <http://www.w3.org/TR/REC-xml-names>;

- «XML Schema Part 1:Structures» <http://www.w3.org/TR/xmlschema-1>;

- «XML Schema Part 2:Datatypes» <http://www.w3.org/TR/xmlschema-2>;

- «Web Services Description Language (WSDL) 1.1» <http://www.w3.org/TR/wsdl>;

- «Simple Object Access Protocol (SOAP) 1.2» <http://www.w3.org/TR/soap12-part0>, <http://www.w3.org/TR/soap12-part1>, <http://www.w3.org/TR/soap12-part2>;

- «Web Services Addressing (WS-Addressing) 1.0» <http://www.w3.org/Submission/ws-addressing/>;

- «XML Signature Syntax and Processing (Second Edition) » <http://www.w3.org/TR/xmldsig-core>;

- «Canonical XML 1.0» <http://www.w3.org/TR/xml-c14n>.

7. Требования при взаимодействии с ОАИС

Требования приведены в Порядке оказания электронных услуг ОАИС, размещенном на официальном сайте республиканского унитарного предприятия «Национальный центр электронных услуг» (https://nces.by/wp-content/uploads/poryadok_oais.pdf).

XML схема технологических электронных документов

```

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

<xs:schema xmlns:tns="http://gtk.gov.by/CustomsService"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://gtk.gov.by/CustomsService"
elementFormDefault="qualified">
  <xs:element name="DocumentRefusalNotice"
type="tns:DocumentRefusalNotice" nillable="true"/>
  <xs:complexType name="DocumentRefusalNotice">
    <xs:sequence>
      <xs:element name="NoticeInfo" type="tns:DocumentRefusalNoticeInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DocumentRefusalNoticeInfo">
    <xs:sequence>
      <xs:element name="DocumentID" type="xs:string"/>
      <xs:element name="RefusalReason" type="xs:string"/>
      <xs:element name="DateRefused" type="xs:dateTime"/>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:string"/>
  </xs:complexType>
  <xs:element name="DocumentPermissionNotice"
type="tns:DocumentPermissionNotice" nillable="true"/>
  <xs:complexType name="DocumentPermissionNotice">
    <xs:sequence>
      <xs:element name="NoticeInfo"
type="tns:DocumentPermissionNoticeInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DocumentPermissionNoticeInfo">
    <xs:sequence>
      <xs:element name="DocumentID" type="xs:string"/>
      <xs:element name="PermissionNumber" type="xs:string"/>
      <xs:element name="DatePermitted" type="xs:dateTime"/>
      <xs:element name="DestinationCustomsCode" minOccurs="0">
        <xs:simpleType>
          <xs:restriction base="xs:string">

```

```

        <xs:minLength value="5"/>
        <xs:maxLength value="8"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="DateLimit" type="xs:dateTime" minOccurs="0"/>
<xs:element name="Guarantee" minOccurs="0">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="GuaranteeNumber" type="xs:string"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentRejectionNotice"
type="tns:DocumentRejectionNotice" nillable="true"/>
<xs:complexType name="DocumentRejectionNotice">
    <xs:sequence>
        <xs:element name="NoticeInfo" type="tns:DocumentRejectionNoticeInfo"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentRejectionNoticeInfo">
    <xs:sequence>
        <xs:element name="DocumentID" type="xs:string"/>
        <xs:element name="DateRejected" type="xs:dateTime"/>
        <xs:element name="RejectionReason" type="tns:Reason"/>
        <xs:element name="ControlLog" type="tns:ControlLog" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:complexType name="Reason">
    <xs:sequence>
        <xs:element name="ReasonCode" type="xs:string"/>
        <xs:element name="Description" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ControlLog">
    <xs:sequence>
        <xs:element name="ControlDate" type="xs:dateTime"/>

```

```

    <xs:element name="EntryCount" type="xs:int"/>
    <xs:element name="Entries" type="tns:ArrayOfControlLogEntry"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfControlLogEntry">
  <xs:sequence>
    <xs:element name="Entry" type="tns:ControlLogEntry" nillable="true"
minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ControlLogEntry">
  <xs:sequence>
    <xs:element name="Type" type="xs:int"/>
    <xs:element name="Section" type="xs:string" minOccurs="0"/>
    <xs:element name="Field" type="xs:string" minOccurs="0"/>
    <xs:element name="Code" type="xs:string" minOccurs="0"/>
    <xs:element name="SubCode" type="xs:string" minOccurs="0"/>
    <xs:element name="Text" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="DocumentRegistrationNotice"
type="tns:DocumentRegistrationNotice" nillable="true"/>
<xs:complexType name="DocumentRegistrationNotice">
  <xs:sequence>
    <xs:element name="NoticeInfo"
type="tns:DocumentRegistrationNoticeInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentRegistrationNoticeInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateRegistered" type="xs:dateTime"/>
    <xs:element name="RegistrationNumber" type="xs:string"/>
    <xs:element name="ControlLog" type="tns:ControlLog" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentAcceptanceNotice"
type="tns:DocumentAcceptanceNotice" nillable="true"/>
<xs:complexType name="DocumentAcceptanceNotice">
  <xs:sequence>

```

```

    <xs:element name="NoticeInfo"
type="tns:DocumentAcceptanceNoticeInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentAcceptanceNoticeInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateAccepted" type="xs:dateTime"/>
    <xs:element name="AcceptanceNumber" type="xs:string" minOccurs="0"/>
    <xs:element name="ControlLog" type="tns:ControlLog" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentReturnNotice" type="tns:DocumentReturnNotice"
nillable="true"/>
<xs:complexType name="DocumentReturnNotice">
  <xs:sequence>
    <xs:element name="NoticeInfo" type="tns:DocumentReturnNoticeInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentReturnNoticeInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateReturned" type="xs:dateTime"/>
    <xs:element name="ReturnReason"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentArrivalGoods"
type="tns:DocumentArrivalGoods" nillable="true"/>
<xs:complexType name="DocumentArrivalGoods">
  <xs:sequence>
    <xs:element name="NoticeInfo" type="tns:DocumentArrivalGoodsInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentArrivalGoodsInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateRegistered" type="xs:dateTime"/>
    <xs:element name="DestinationCustomsCode">
      <xs:simpleType>
        <xs:restriction base="xs:string">

```

```

        <xs:minLength value="5"/>
        <xs:maxLength value="8"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
<xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentRefusalArrivalGoods"
type="tns:DocumentRefusalArrivalGoods" nillable="true"/>
<xs:complexType name="DocumentRefusalArrivalGoods">
    <xs:sequence>
        <xs:element name="NoticeInfo"
type="tns:DocumentRefusalArrivalGoodsInfo"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentRefusalArrivalGoodsInfo">
    <xs:sequence>
        <xs:element name="DocumentID" type="xs:string"/>
        <xs:element name="RefusalReason" type="xs:string"/>
        <xs:element name="DateRegistered" type="xs:dateTime"/>
        <xs:element name="DestinationCustomsCode">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:minLength value="5"/>
                    <xs:maxLength value="8"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentTemporaryStorage"
type="tns:DocumentTemporaryStorage" nillable="true"/>
<xs:complexType name="DocumentTemporaryStorage">
    <xs:sequence>
        <xs:element name="NoticeInfo"
type="tns:DocumentTemporaryStorageInfo"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentTemporaryStorageInfo">
    <xs:sequence>

```

```

    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateRegistered" type="xs:dateTime"/>
    <xs:element name="RegistrationNumber" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentRefusalTemporaryStorage"
type="tns:DocumentRefusalTemporaryStorage" nillable="true"/>
  <xs:complexType name="DocumentRefusalTemporaryStorage">
    <xs:sequence>
      <xs:element name="NoticeInfo"
type="tns:DocumentRefusalTemporaryStorageInfo"/>
    </xs:sequence>
  </xs:complexType>
<xs:complexType name="DocumentRefusalTemporaryStorageInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="RefusalReason" type="xs:string"/>
    <xs:element name="DateRegistered" type="xs:dateTime"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentVehicle" type="tns:DocumentVehicle"
nillable="true"/>
  <xs:complexType name="DocumentVehicle">
    <xs:sequence>
      <xs:element name="NoticeInfo" type="tns:DocumentVehicleInfo"/>
    </xs:sequence>
  </xs:complexType>
<xs:complexType name="DocumentVehicleInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateRegistered" type="xs:dateTime"/>
    <xs:element name="RegistrationNumber" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="DocumentRefusalVehicle"
type="tns:DocumentRefusalVehicle" nillable="true"/>
  <xs:complexType name="DocumentRefusalVehicle">
    <xs:sequence>
      <xs:element name="NoticeInfo" type="tns:DocumentRefusalVehicleInfo"/>

```

```

</xs:sequence>
</xs:complexType>
<xs:complexType name="DocumentRefusalVehicleInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="RefusalReason" type="xs:string"/>
    <xs:element name="DateRegistered" type="xs:dateTime"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="ReviewResultsNotice" type="tns:ReviewResultsNotice"
nillable="true"/>
<xs:complexType name="ReviewResultsNotice">
  <xs:sequence>
    <xs:element name="NoticeInfo" type="tns:ReviewResultsNoticeInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ReviewResultsNoticeInfo">
  <xs:sequence>
    <xs:element name="DocumentID" type="xs:string"/>
    <xs:element name="DateAccepted" type="xs:dateTime"/>
    <xs:element name="AcceptanceNumber" type="xs:string"/>
    <xs:element name="Vehicle" minOccurs="0" maxOccurs="unbounded">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="VehicleNumber" type="xs:string"/>
          <xs:element name="VehicleDecision" type="xs:string" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="ControlLog" type="tns:ControlLog" minOccurs="0"/>
    <xs:element name="NameAuthority" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
<xs:element name="Fault" type="tns:FaultType" nillable="true"/>
<xs:complexType name="FaultType">
  <xs:sequence>
    <xs:element name="Code">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="Value" type="xs:string"/>

```

```

    <xs:element name="Subcode" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="Value" type="xs:string"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Reason">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Text" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:complexType name="Requirement">
  <xs:sequence>
    <xs:element name="RequirementType" type="xs:string"/>
    <xs:element name="RequirementText" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="RequirementInfo">
  <xs:sequence>
    <xs:element name="SerialNumber">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:maxLength value="5"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="IndivNumber" type="xs:string"/>
    <xs:element name="Requirements" type="tns:Requirement"
maxOccurs="2"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="DocumentRequirementNotice"
type="tns:DocumentRequirementNotice" nillable="true"/>
<xs:complexType name="DocumentRequirementNotice">

```

```

    <xs:sequence>
      <xs:element name="NoticeInfo"
type="tns:DocumentRequirementNoticeInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DocumentRequirementNoticeInfo">
    <xs:sequence>
      <xs:element name="DocumentID" type="xs:string"/>
      <xs:element name="DateRequirement" type="xs:dateTime"/>
      <xs:element name="RequirementReason" type="tns:Reason"/>
      <xs:element name="RequirementsInfo" type="tns:RequirementInfo"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:string"/>
  </xs:complexType>
  <xs:element name="DocumentCompletionNotice"
type="tns:DocumentCompletionNotice" nillable="true"/>
  <xs:complexType name="DocumentCompletionNotice">
    <xs:sequence>
      <xs:element name="NoticeInfo"
type="tns:DocumentCompletionNoticeInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DocumentCompletionNoticeInfo">
    <xs:sequence>
      <xs:element name="DocumentID" type="xs:string"/>
      <xs:element name="PtdegNumber" type="xs:string"/>
      <xs:element name="DateCompletion" type="xs:dateTime"/>
      <xs:element name="IndivInfo" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="SerialNumber">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:maxLength value="5"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
            <xs:element name="IndivNumber" type="xs:string"/>
            <xs:element name="IndivDecision" type="xs:string"/>
            <xs:element name="RejectionReason" type="xs:string" minOccurs="0"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```

```
    </xs:complexType>
  </xs:element>
</xs:sequence>
  <xs:attribute name="Id" type="xs:string"/>
</xs:complexType>
</xs:schema>
```

Структуры электронных таможенных сообщений (документов)

Таблица Б.1. Структура типа данных Document.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DocumentType	Код вида документа согласно "Справочнику видов документов" (Решение Комиссии Таможенного союза от 20.09.2010 N 378)	xs:string	[1]
3	DocumentTypeVersion	Версия формата документа согласно "Справочнику видов документов" (Решение Комиссии Таможенного союза от 20.09.2010 N 378)	xs:string	[1]
4	DocumentBody	Тело документа	DocumentBody	[1]

Таблица Б.2. Структура типа данных DocumentBody.

Элемент		Описание элемента	Тип	Мн.
1		Любой XML элемент	xs:any	[1]

Таблица Б.3. Структура типа данных Reason.

Элемент		Описание элемента	Тип	Мн.
DocumentListRequest		Корневой элемент		[1]
1	ReasonCode	Код причины (ошибки)	xs:string	[1]
2	Description	Описание причины (ошибки)	xs:string	[1]

Таблица Б.4. Структура типа данных ControlLog.

Элемент		Описание элемента	Тип	Мн.
1	ControlDate	Дата проведения ФЛК	xs:datetime	[1]
2	EntryCount	Количество ошибок ФЛК	xs:integer	[1]

3	Entries	Перечень ошибок ФЛК	ArrayOfControlLogEntry	[1]
---	---------	---------------------	------------------------	-----

Таблица Б.5. Структура типа данных ArrayOfControlLogEntry.

Элемент		Описание элемента	Тип	Мн.
1	Error	Информация об ошибке	ControlLogEntry	[0..n]

Таблица Б.6. Структура типа данных ControlLogEntry.

Элемент		Описание элемента	Тип	Мн.
DocumentNonEntryNotice		Корневой элемент		[1]
1	Type	Тип ошибки (0 – ошибка, 1 –предупреждение, 2 – информация)	xs:int	[1]
2	Section	Графа, в которой допущена ошибка	xs:string	[0..1]
3	Field	Поле, в котором допущена ошибка	xs:string	[0..1]
4	Code	Код ошибки	xs:string	[0..1]
5	SubCode	Подкод ошибки	xs:string	[0..1]
6	Text	Текстовое описание ошибки	xs:string	[1]

Таблица Б.7. Структура элемента DocumentSubmissionRequest.

Элемент		Описание элемента	Тип	Мн.
1	RequestInfo	Данные запроса	DocumentSubmissionRequestInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.8. Структура типа данных DocumentSubmissionRequestInfo

Элемент		Описание элемента	Тип	Мн.
1	ReferenceDocumentID	Уникальный идентификатор (DocumentID) основного документа	xs:string	[0..1]
2	Document	Информация о документе	Document	[1]

Таблица Б.9. Структура элемента DocumentRevocationRequest

Элемент		Описание элемента	Тип	Мн.
1	RequestInfo	Данные запроса	DocumentRevocationRequestInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.10. Структура типа данных DocumentRevocationRequestInfo

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор отзыва документа	xs:string	[1]
2	Reason	Причина отзыва	xs:string	[1]

Таблица Б.11. Структура элемента DocumentRejectionNotice.

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentRejectionNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.12. Структура типа данных DocumentRejectionNoticeInfo.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateRejected	Дата отказа в принятии документа в ИСТО	xs:datetime	[1]
3	RejectionReason	Причина отказа	Reason	[1]
4	ControlLog	Протокол ФЛК (заполняется при наличии протокола ФЛК)	ControlLog	[0..1]

Таблица Б.13. Структура элемента DocumentAcceptanceNotice.

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentAcceptanceNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.14. Структура типа данных DocumentAcceptanceNoticeInfo.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateAccepted	Дата принятия документа в ИСТО	xs:datetime	[1]
3	AcceptanceNumber	Номер регистрации принятия документа	xs:string	[0..1]
4	ControlLog	Протокол ФЛК (заполняется при наличии протокола ФЛК)	ControlLog	[0..1]

Таблица Б.15. Структура элемента DocumentRegistrationNotice.

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentRegistrationNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.16. Структура типа данных DocumentRegistrationNoticeInfo.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateRegistered	Дата регистрации документа	xs:datetime	[1]
3	RegistrationNumber	Регистрационный номер документа	xs:string	[1]
4	ControlLog	Протокол ФЛК (заполняется при наличии протокола ФЛК)	ControlLog	[0..1]

Таблица Б.17. Структура элемента DocumentPermissionNotice.

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentPermissionNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.18. Структура типа данных DocumentPermissionNoticeInfo.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	PermissionNumber	Принятое решение	xs:string	[1]
3	DatePermitted	Дата выдачи принятия решения	xs:datetime	[1]
4	DestinationCustomsCode	Код таможенного органа назначения (8 знаков)	xs:string	[0..1]
5	DateLimit	Срок доставки	xs:datetime	[0..1]
6	Guarantee	Меры по обеспечению соблюдения таможенного транзита.		[0..1]
	6.1	GuaranteeNumber	Регистрационный номер сертификата, подтверждающего применение мер обеспечения соблюдения таможенного транзита	[1..n]

Таблица Б.19. Структура элемента DocumentRefusalNotice.

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentRefusalNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.20. Структура типа данных DocumentRefusalNoticeInfo.

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	RefusalReason	Причина отказа	xs:string	[1]
3	DateRefused	Дата отказа в выдаче разрешения/свидетельства	xs:datetime	[1]

Таблица Б.21. Структура элемента DocumentReturnNotice

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentReturnNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.22. Структура типа данных DocumentReturnNoticeInfo

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateReturned	Дата отказа в регистрации документа	xs:datetime	[1]
3	ReturnReason	Причина отказа	xs:string	[1]

Таблица Б.23. Структура элемента DocumentAbortNotice

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentAbortNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.24. Структура типа данных DocumentAbortNotice

Элемент		Описание элемента	Тип	Мн.
1	DocumentID	Уникальный идентификатор документа	xs:string	[1]
2	DateAborted	Дата прерывания оформления документа	xs:datetime	[1]
3	AbortReason	Причина прерывания оформления документа	xs:string	[1]

Таблица Б.25. Структура элемента DocumentCancellationNotice

Элемент		Описание элемента	Тип	Мн.
1	NoticeInfo	Данные извещения	DocumentCancellationNoticeInfo	[1]
2	Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.26. Структура типа данных
DocumentCancellationNoticeInfo

Элемент	Описание элемента	Тип	Мн.
1 DocumentID	Уникальный идентификатор документа	xs:string	[1]
2 DateCancelled	Дата фактического аннулирования документа	xs:datetime	[1]
3 CancelledNumber	Номер регистрации аннулированного документа	xs:string	[1]

Таблица Б.27. Структура элемента DocumentRequirementNotice.

Элемент	Описание элемента	Тип	Мн.
1 NoticeInfo	Данные извещения	DocumentRequirementNoticeInfo	[1]
2 Signature	Данные ЭЦП	xs:any	[0..1]

Таблица Б.28. Структура типа данных
DocumentRequirementNoticeInfo.

Элемент	Описание элемента	Тип	Мн.
1 DocumentID	Уникальный идентификатор документа	xs:string	[1]
2 RequirementID	Уникальный идентификатор требования	xs:string	[0..1]
3 DateIssued	Дата выставления требования	xs:datetime	[1]
4 ExpirationDate	Срок исполнения требования	xs:datetime	[1]
5 RequirementText	Текст требования	xs:string	[1]

XML схема ЭПД

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2011 rel. 2 (http://www.altova.com) by TeaM DJiNN (TeaM DJiNN) -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:element name="PD">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Declarant">
          <xs:complexType>
            <xs:sequence>
              <xs:choice>
                <xs:element name="DocumentsArhiveRequest">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="RegNumber" type="xs:string"/>
                      <xs:element name="Files">
                        <xs:complexType>
                          <xs:sequence>
                            <xs:element name="Document" maxOccurs="unbounded">
                              <xs:complexType>
                                <xs:sequence>
                                  <xs:element name="Name" type="xs:string"/>
                                  <xs:element name="DocumentInfo">
                                    <xs:complexType>
                                      <xs:sequence>
                                        <xs:element name="Column" type="xs:string"
minOccurs="0"/>
                                        <xs:element name="DocCode" type="xs:string"/>
                                        <xs:element name="DocNumber" type="xs:string"/>
                                        <xs:element name="DocDate" type="xs:string"/>
                                        <xs:element name="DocType" type="xs:string"/>
                                      </xs:sequence>
                                    </xs:complexType>
                                  </xs:element>
                                  <xs:element name="ItemNumb" type="xs:string"
minOccurs="0"/>
                                  <xs:element name="Extension" type="xs:string"/>
                                  <xs:element name="Size" type="xs:string"/>
                                  <xs:element name="MD5" type="xs:string"/>
                                  <xs:element name="ZIP" type="xs:string"/>
                                </xs:sequence>
                                <xs:attribute name="id" type="xs:string"/>
                              </xs:complexType>
                            </xs:element>
                          </xs:sequence>
                        </xs:complexType>
                      </xs:element>
                    <xs:element name="Sender">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element name="UNP" minOccurs="0"/>
                          <xs:element name="Name" minOccurs="0"/>
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                  <xs:element name="DocType" type="xs:string"/>
                </xs:choice>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:schema>

```

```

        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="DocumentsArhiveRequirement">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="Files">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Document" maxOccurs="unbounded">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="Name" type="xs:string"/>
                      <xs:element name="Extension" type="xs:string"/>
                      <xs:element name="Size" type="xs:string"/>
                      <xs:element name="MD5" type="xs:string"/>
                      <xs:element name="ZIP" type="xs:string"/>
                      <xs:element name="Requirement" type="xs:string"/>
                    </xs:sequence>
                    <xs:attribute name="id" type="xs:string"/>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:choice>
      <xs:sequence>
        <xs:attribute name="ID"/>
      </xs:sequence>
      <xs:element name="Signature">
        <xs:complexType>
          <xs:attribute name="ID"/>
        </xs:complexType>
      </xs:element>
    </xs:choice>
  </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Тип таможенного документа, к которому относится ЭПД:

- 1) DT - декларация на товары
- 2) KT - корректировка декларации на товары
- 3) TD - транзитная декларация
- 4) RS - документ, отражающий исчисление и уплату утилизационного сбора
- 5) DTEG - декларация на товары для экспресс-грузов
- 6) ZVT - заявление о выпуске товаров до подачи декларации на товары
- 7) UV - уведомление о размещении товаров в ЗТК
- 8) BT - отчет о товарах, реализованных в магазине беспошлинной торговли
- 9) OT - отчет об отходах, образовавшихся в результате переработки
- 10) UD - уведомление об отходах
- 11) ZP - заявление о завершении действия таможенной процедуры СТЗ
- 12) ZVSTZ - заявление о выдаче разрешения, допускающего вывоз товаров с территории СТЗ без завершения действия таможенной процедуры СТЗ в соответствии с пунктом 4 статьи 205 ТК ЕАЭС
- 13) RT - Отчет о реализации товаров на бортах
- 14) UZ - уведомление о наступлении обстоятельств, указанных в пунктах 12 и 13 статьи 207 Таможенного кодекса Евразийского экономического союза (далее - ТК ЕАЭС) в отношении товаров, помещенных под таможенную процедуру свободной таможенной зоны и являющихся оборудованием, введенным в эксплуатацию и используемым резидентом свободной (специальной, особой) экономической зоны (далее - СЭЗ) для реализации договора об условиях деятельности в СЭЗ, или товарами, использованными для создания объектов недвижимости на территории СЭЗ и являющимися составной частью таких объектов недвижимости, введенных в эксплуатацию (далее - уведомление об оборудовании и (или) объектах недвижимости на территории СЭЗ)
- 15) US - уведомление о наступлении обстоятельств, указанных в пункте 9 статьи 215 ТК ЕАЭС в отношении товаров, помещенных под таможенную процедуру свободного склада и являющихся оборудованием, введенным в эксплуатацию и используемым владельцем свободного склада, или товарами, использованными владельцем свободного склада для создания объектов недвижимости на территории свободного склада и являющимися составной частью таких объектов недвижимости, введенных в эксплуатацию (далее - уведомление об оборудовании и (или) объектах недвижимости на территории свободного склада)
- 16) UP - уведомление о потреблении товаров в соответствии с частью первой пункта 12 Положения о завершении таможенной процедуры (далее - уведомление о потреблении товаров)
- 17) KDTEG - Корректировка декларации на товары для экспресс-грузов
- 18) D4 - Декларация таможенной стоимости ДТС1
- 19) T3 - Декларация таможенной стоимости ДТС2
- 20) PTD - Пассажирская таможенная декларация

- 21) TDTS - Декларация на транспортное средство
- 22) TMPA - Заявление о совершении операций в отношении находящихся за пределами таможенной территории Евразийского экономического союза транспортных средств международной перевозки, являющихся товарами, помещенными под таможенную процедуру временного ввоза (допуска)
- 23) PTDEG - Пассажирская таможенная декларация для экспресс-грузов
- 24) KPTDEG – корректировка пассажирской таможенной декларации для экспресс-грузов
- 25) PIAT - предварительная информация о товарах, предполагаемых к ввозу на таможенную территорию Евразийского экономического союза автомобильным транспортом
- 26) PIRW - предварительная информация о товарах, предполагаемых к ввозу на таможенную территорию Евразийского экономического союза железнодорожным транспортом
- 27) PIAIR - предварительная информация о товарах, предполагаемых к ввозу на таможенную территорию Евразийского экономического союза воздушным транспортом