

API «Регистратор»

Открытое API «Регистратор» предназначено для автоматического взаимодействия ФИПС с клиентами сервиса АРМ «Регистратор».

API располагается по следующему адресу:

<https://kpsrtz.fips.ru/api/v1>

Ключ авторизации

Для исключения несанкционированного доступа к заявкам в ЛК используется Ключ авторизации. Ключ авторизации предоставляется администратором системы АРМ «Регистратор» по запросу. Администратор использует специальный интерфейс для формирования Ключа авторизации. Данная функциональность доступна только пользователю с ролью «Администратор». Сформированный Ключ авторизации имеет срок действия по сроку действия закрытого ключа заявителя. В случае отзыва сертификата, к которому привязан Ключ авторизации, доступ по Ключу авторизации становится невозможным.

Для передачи Ключа авторизации при вызовах методов API используется HTTP-заголовок «Authorization».

Swagger



API «Регистратор»

Открытое REST API для автоматического взаимодействия ФИПС с клиентами сервиса [АРМ «Регистратор»](#).

Для доступа к API необходим **Ключ авторизации**. Ключ авторизации можно получить на странице [Настроек](#).

Для передачи Ключа авторизации при вызовах методов API используется HTTP-заголовок **Authorization**.

Created by ФИПС
[Contact the developer](#)

Applications

Show/Hide | List Operations | Expand Operations

Categories

Show/Hide | List Operations | Expand Operations

Documents

Show/Hide | List Operations | Expand Operations

DocumentTypes

Show/Hide | List Operations | Expand Operations

[BASE URL: /api , API VERSION: v1]

Для получения актуальной информации о параметрах вызова методов API и составе методов API используется пользовательский интерфейс Swagger. Адрес доступа к интерфейсу Swagger:

<https://kpsrtz.fips.ru/api/swagger>

С помощью интерфейса Swagger так же имеется возможность проверить функциональность любого метода. Для запуска метода необходимо заполнить все поля, помеченные как «required» и при необходимости загрузить файл. После заполнения всех полей по кнопке «Try it out!» пользователь получит результат либо в виде текста в формате JSON, либо получит ссылку для скачивания запрашиваемого файла.

Для предотвращения несанкционированного доступа к ЛК необходимо ввести Ключ авторизации в соответствующее поле «api_key». Значение этого поля передаётся в HTTP-заголовке «Authorization».

Общие HTTP-коды ответов

По результатам вызова методов API пользователь получает не только данные, но и HTTP-код результата. HTTP-код является трёхзначным числом. Следующие HTTP-коды общие для всех методов:

2xx	HTTP-коды, которые начинаются на «2» обозначают успешное выполнение.
207	Результат выполнения сразу нескольких независимых операций. HTTP-код и результат выполнения каждой операции размещаются в текстовом ответе в формате JSON.
4xx	HTTP-коды ошибок вызывающей стороны (клиента)
401	Не указан Ключ авторизации в HTTP-заголовке «Authorization». Если ключ передавался, значит срок его действия закончен.
5xx	HTTP-коды ошибок сервера

Если API вызывается не напрямую, а через Proxy сервер (зависит от сетевых настроек окружения пользователя), тогда возможны другие HTTP-коды ошибок и текст ошибки может отличаться от заявленного.

Библиотека заявок

Applications

Show/Hide | List Operations | Expand Operations

GET	/v1/Applications	Список заявок
POST	/v1/Applications	Создание заявки
DELETE	/v1/Applications/{id}	Удаление заявки
GET	/v1/Applications/{id}	Атрибуты заявки
PUT	/v1/Applications/{id}	Редактирование заявки
GET	/v1/Applications/{id}/Header	Пакет заявки для подписания
PUT	/v1/Applications/{id}/Send	Отправка заявки

Группа методов для получения информации по заявкам в текущем ЛК.

Список заявок

GET /v1/Applications Список заявок

Response Class (Status 200)
OK

Model | Example Value

```
[
  {
    "ID": "00000000-0000-0000-0000-000000000000",
    "CreateDate": "2020-09-01T19:29:38.811Z",
    "RegNumber": "2020700000",
    "Number": "W20000000",
    "Applicant": "Заявитель",
    "Description": "Комментарий",
    "StatusCode": "Created",
    "StatusTitle": "Имеются непрочитанные документы",
  }
]
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
kind	<input type="text" value=""/>	Вид списка	query	string
sortField	<input type="text" value=""/>	Сортировка по полю	query	string
sortDescend	<input type="text" value=""/>	Порядок сортировки	query	boolean

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Неизвестное поле в параметре sortField		
401	Ключ авторизации просрочен		

Метод предназначен для получения списка заявок. Результат содержит основной набор атрибутов и идентификаторы каждой заявки. Список заявок можно отфильтровать по виду (kind):

- Все заявки (All)
- С непрочитанными документами (HasUnreadDoc)
- С незарегистрированными документами (HasUnregisteredDoc)
- На подписании (Trusted)

Порядок заявок определяется по полю (sortField) и направлению сортировки (sortDescend)

Создание заявки

POST /v1/Applications Создание заявки

Response Class (Status 201)
Заявка создана

Model | Example Value

```
{
  "IsTrusted": false,
  "Trustees": [
    {
      "Applicant": "Доверенное лицо",
      "ConfidentRegNumber": "string"
    }
  ],
  "ID": "00000000-0000-0000-0000-000000000000",
  "CreateDate": "2020-09-01T19:29:38.827Z",
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
type	<input type="text" value="TMApplication"/>	Код типа заявки	query	string
data	<input type="text" value="(required)"/> Parameter content type: <input type="text" value="application/json"/>	Заявка	body	Model Example Value

```
{
  "Name": "Мой учётный номер",
  "Category": "Переписка",
  "Description": "Комментарий"
}
```

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Неизвестная категория; При сохранении были обнаружены ошибки		
401	Недостаточно прав; Ключ авторизации просрочен		

С помощью данного метода создаётся новая заявка. Для вызова необходимо указать код типа заявки (type):

- Заявка на ТЗ (TMApplication)
- Заявка на НМПТ/ПНМПТ (AOApplication)
- Заявка на ГУ/ПГУ (GeoIndication)
- Заявка на изобретение (Invention)
- Заявка на полезную модель (UsefulModel)

В качестве названия категории должен передаваться её полный путь.

В качестве ответа при успешном создании заявки возвращается HTTP-код 201 и в заголовке «Location» приходит адрес новой заявки. Ответ так же содержит идентификатор созданной заявки и другие атрибуты.

Набор проверок аналогичен проверкам в АРМ «Регистратор».

Атрибуты заявки

GET /v1/Applications/{id} Атрибуты заявки

Response Class (Status 200)
OK

Model **Example Value**

```
{
  "IsTrusted": false,
  "Trustees": [
    {
      "Applicant": "Доверенное лицо",
      "ConfidentRegNumber": "string"
    }
  ],
  "ID": "00000000-0000-0000-0000-000000000000",
  "CreateDate": "2020-09-01T19:29:38.839Z",
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	<input type="text" value="(required)"/>	Идентификатор заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка не найдена		

Метод возвращает расширенный набор атрибутов заявки.

В качестве идентификатора заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Редактирование заявки

PUT /v1/Applications/{id} Редактирование заявки

Response Class (Status 200)
Заявка обновлена

Model | Example Value

```
{
  "IsTrusted": false,
  "Trustees": [
    {
      "Applicant": "Доверенное лицо",
      "ConfidentRegNumber": "string"
    }
  ],
  "ID": "00000000-0000-0000-0000-000000000000",
  "CreateDate": "2020-09-01T19:29:38.849Z",
}
```

Response Content Type: application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	(required)	Идентификатор заявки	path	string
data	(required)	Заявка	body	Model Example Value

Parameter content type: application/json

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Неизвестная категория; При сохранении были обнаружены ошибки		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка не найдена		

[Try it out!](#)

С помощью данного метода изменяются атрибуты заявки.

В качестве названия категории должен передаваться её полный путь.

В качестве идентификатора заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Набор проверок аналогичен проверкам в АРМ «Регистратор».

Удаление заявки

DELETE /v1/Applications/{id} Удаление заявки

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	<input type="text" value="(required)"/>	Идентификатор заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
204	Заявка удалена		
400	Документы находятся в состоянии, запрещающем удаление		
401	Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка не найдена		

Данный метод предназначен для удаления заявки.

В качестве идентификатора заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403. В случае успешной операции метод не возвращает ничего, кроме HTTP-кода 204.

Пакет заявки для подписания

GET /v1/Applications/{id}/Header Пакет заявки для подписания

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	<input type="text" value="(required)"/>	Идентификатор заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	ОК		
400	Не все документы подписаны		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка не найдена		

Перед отправкой заявки на регистрацию необходимо сформировать пакет заявки. Метод возвращает XML-документ в бинарном виде, который нужно подписать ЭП.

Формирование ЭП производится на стороне приложения, которое вызывает API «Регистратор».

Отправка заявки на регистрацию

Parameter	Value	Description	Parameter Type	Data Type
id	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
data	<input type="button" value="Выбор файла"/> Не выбран ни один файл	Файл подписи пакета заявки	formData	file

HTTP Status Code	Reason	Response Model	Headers
204	Заявка отправлена		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка не найдена		

Для отправки заявки на регистрацию необходимо сформировать ЭП пакета заявки. Поддерживаются следующие алгоритмы подписи:

- ГОСТ Р 34.10-2001
- ГОСТ Р 34.10-2012

Документы заявки

GET	/v1/Applications/{application}/Documents	Список документов заявки
POST	/v1/Applications/{application}/Documents	Создание документа заявки
DELETE	/v1/Applications/{application}/Documents/{id}	Удаление документа заявки
GET	/v1/Applications/{application}/Documents/{id}	Документ заявки
PUT	/v1/Applications/{application}/Documents/{id}	Редактирование документа заявки
DELETE	/v1/Applications/{application}/Documents/{id}/Content	Удаление файла документа заявки
GET	/v1/Applications/{application}/Documents/{id}/Content	Файл документа заявки
PUT	/v1/Applications/{application}/Documents/{id}/Content	Редактирование файла документа заявки
GET	/v1/Applications/{application}/Documents/{id}/XML	Файл заявления
PUT	/v1/Applications/{application}/Documents/{id}/XML	Редактирование файла заявления
GET	/v1/Applications/{application}/Documents/{id}/Signatures	Подписи документа
PUT	/v1/Applications/{application}/Documents/{id}/Signatures	Добавление подписи документа

Группа методов для получения информации по документам заявки

Список документов заявки

GET /v1/Applications/{application}/Documents Список документов заявки

Response Class (Status 200)
OK

Model Example Value

```
[
  {
    "ID": "00000000-0000-0000-0000-000000000000",
    "ParentID": "00000000-0000-0000-0000-000000000000",
    "CreateDate": "2020-09-01T19:29:38.866Z",
    "TransferDate": "2020-09-01T19:29:38.866Z",
    "Number": "W20000000",
    "Name": "Заявление на ТЗ \Мой учётный номер\"",
    "RegDate": "2020-09-01T19:29:38.866Z",
    "TypeCode": "TradeMark",
  }
]
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	<input type="text" value="(required)"/>	Идентификатор заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка не найдена		

Метод возвращает список документов указанной заявки. Результат содержит атрибуты и идентификаторы документов заявки.

В качестве идентификатора заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

```
},
{
  "ID": "9ffdf961-6a92-4692-9a5f-13b3ffde6b65",
  "ParentID": "7ec4e4ae-c924-4347-8304-c3c328e4d675",
  "CreateDate": "2024-08-28T12:15:20",
  "TransferDate": null,
  "Number": null,
  "Name": "Машиночитаемая доверенность",
  "RegDate": null,
  "DocumentType": {
    "Name": "Машиночитаемая доверенность",
    "Code": "PowerOfAttorney"
  }
},
]
```

Создание документа заявки

POST /v1/Applications/{application}/Documents Создание документа заявки

Response Class (Status 201)
Документ заявки создан

Model Example Value

```
{
  "ID": "00000000-0000-0000-0000-000000000000",
  "ParentID": "00000000-0000-0000-0000-000000000000",
  "CreateDate": "2020-09-01T19:29:38.870Z",
  "TransferDate": "2020-09-01T19:29:38.870Z",
  "Number": "W20000000",
  "Name": "Заявление на ТЗ \Мой учётный номер\"",
  "RegDate": "2020-09-01T19:29:38.870Z",
  "TypeCode": "TradeMark",
  "TypeTitle": "Заявление на ТЗ",
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
data	<input type="text" value="(required)"/>	Документ заявки	body	Model Example Value
type	<input type="text"/>	Код типа документа заявки, если имеется	query	string

Parameter content type:

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Неизвестный тип документа; При сохранении были обнаружены ошибки		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка не найдена		

С помощью данного метода создаётся новый пустой документ заявки. Для вызова необходимо указать либо код типа документа (type), либо название типа документа ("TypeTitle" в JSON).

Основные типы заявлений:

- Заявление на ТЗ (TradeMark)
- Заявление на НМПТ и ПНМПТ (AppellationOrigin)
- Заявление на ПНМПТ (AppellationOriginUsage)
- Заявление на ГУ и ПГУ (GeoIndication)
- Заявление на ПГУ (GeoIndicationUsage)
- Заявление на ИЗ (Invention)
- Заявление на ПМ (UsefulModel)

Полный список доступных типов документов с названиями и кодами можно получить с помощью методов для шаблонов (типов) документов.

В качестве ответа при успешном создании документа заявки возвращается HTTP-код 201 и в заголовке «Location» приходит адрес нового документа заявки. Ответ так же содержит идентификатор созданного документа и другие атрибуты.

Набор проверок аналогичен проверкам в АРМ «Регистратор».

Атрибуты документа заявки

GET /v1/Applications/{application}/Documents/{id} Документ заявки

Response Class (Status 200)
OK

Model Example Value

```
{
  "ID": "00000000-0000-0000-0000-000000000000",
  "ParentID": "00000000-0000-0000-0000-000000000000",
  "CreateDate": "2020-09-01T19:29:38.881Z",
  "TransferDate": "2020-09-01T19:29:38.881Z",
  "Number": "W20000000",
  "Name": "Заявление на ТЗ \"Мой учётный номер\"",
  "RegDate": "2020-09-01T19:29:38.881Z",
  "TypeCode": "TradeMark",
  "TypeTitle": "Заявление на ТЗ",
}
```

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	(required)	Идентификатор заявки	path	string
id	(required)	Идентификатор документа заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

Try it out!

Метод возвращает атрибуты одного документа заявки.

В качестве идентификатора заявки и документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Редактирование документа заявки

PUT /v1/Applications/{application}/Documents/{id} Редактирование документа заявки

Response Class (Status 200)
Документ заявки обновлён

Model Example Value

```
{
  "ID": "00000000-0000-0000-0000-000000000000",
  "ParentID": "00000000-0000-0000-0000-000000000000",
  "CreateDate": "2020-09-01T19:29:38.886Z",
  "TransferDate": "2020-09-01T19:29:38.886Z",
  "Number": "W20000000",
  "Name": "Заявление на ТЗ \"Мой учётный номер\"",
  "RegDate": "2020-09-01T19:29:38.886Z",
  "TypeCode": "TradeMark",
  "TypeTitle": "Заявление на ТЗ",
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
id	<input type="text" value="(required)"/>	Идентификатор документа заявки	path	string
data	<input type="text" value="(required)"/>	Документ заявки	body	Model Example Value

Parameter content type:

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	При сохранении были обнаружены ошибки		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

С помощью данного метода изменяются атрибуты документа заявки.

В качестве идентификатора заявки и документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Набор проверок аналогичен проверкам в АРМ «Регистратор».

Удаление документа заявки

DELETE /v1/Applications/{application}/Documents/{id} Удаление документа заявки

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
id	<input type="text" value="(required)"/>	Идентификатор документа заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
204	Документ заявки удалён		
400	Документ находится в состоянии, запрещающем удаление		
401	Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

Данный метод предназначен для удаления документа заявки.

В качестве идентификатора заявки и документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403. В случае успешной операции метод не возвращает ничего, кроме HTTP-кода 204.

Файл документа заявки

GET /v1/Applications/{application}/Documents/{id}/Content Файл документа заявки

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
id	<input type="text" value="(required)"/>	Идентификатор документа заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	ОК		
401	Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

Метод возвращает содержимое документа заявки в бинарном виде. Содержимое документа необходимо для подписания ЭП документа.

Формирование ЭП производится на стороне приложения, которое вызывает API «Регистратор».

В качестве идентификатора заявки и документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Если запрашивается содержимое структурированного документа (например, заявления), тогда оно будет доступно только после успешного прохождения ФЛК.

Редактирование файла документа заявки

PUT /v1/Applications/{id}/Send Отправка заявки

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
data	<input type="button" value="Выбор файла"/> <input type="text" value="Не выбран ни один файл"/>	Файл подписи пакета заявки	formData	file

Response Messages

HTTP Status Code	Reason	Response Model	Headers
204	Заявка отправлена		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка не найдена		

С помощью данного метода изменяется содержимое документа заявки. Содержимое документа заявки передаётся в запросе в бинарном виде (MIME-тип «multipart/form-data»). Данный метод подходит только для неструктурированных документов.

В качестве идентификатора заявки и документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Набор проверок и требования к формату содержимого аналогичны проверкам и требованиям в АРМ «Регистратор».

Удаление содержимого документа заявки

DELETE /v1/Applications/{application}/Documents/{id}/Content Удаление файла документа заявки

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
id	<input type="text" value="(required)"/>	Идентификатор документа заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
204	Файл документа заявки удалён		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

Данный метод предназначен для очистки содержимого документа заявки. Сам документ заявки не удаляется.

В качестве идентификатора заявки и документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403. В случае успешной операции метод не возвращает ничего, кроме HTTP-кода 204.

Файл заявления

GET /v1/Applications/{application}/Documents/{id}/XML Файл заявления

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
id	<input type="text" value="(required)"/>	Идентификатор документа заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	OK		
401	Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

Метод возвращает содержимое структурированного документа заявки в виде XML-документа.

В качестве идентификатора заявки и документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Редактирование файла заявления

PUT /v1/Applications/{application}/Documents/{id}/XML Редактирование файла заявления

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	<input type="text" value="(required)"/>	Идентификатор заявки	path	string
id	<input type="text" value="(required)"/>	Идентификатор документа заявки	path	string
data	<input type="button" value="Выбор файла"/> <input type="button" value="Не выбран ни один файл"/>	Файл заявления	formData	file

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	Файл заявления обновлён		
204	No Content		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

С помощью данного метода изменяется содержимое структурированного документа заявки. Содержимое документа заявки передаётся в запросе как XML-документ в бинарном виде (MIME-тип «multipart/form-data»).

В качестве идентификатора заявки и документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Набор проверок (ФЛК) аналогичен проверкам в АРМ «Регистратор».

Подписи документа заявки

GET /api/v1/Applications/{application}/Documents/{id} Файл документа заявки

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	(required)	Идентификатор заявки	path	string
id	(required)	Идентификатор документа заявки	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	OK		
401	Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

[Try it out!](#)

Метод возвращает подписи документа заявки.

В качестве идентификатора заявки и идентификатора документа заявки может выступать как внутренний идентификатор, так и регистрационный номер (если имеется). Доступ к заявке есть либо у автора, либо у доверенного лица. В ином случае возвращается HTTP-код 403.

Добавление подписи документа

PUT /v1/Applications/{application}/Documents/{id}/Signatures Добавление подписи документа

Parameters

Parameter	Value	Description	Parameter Type	Data Type
application	(required)	Идентификатор заявки	path	string
id	(required)	Идентификатор документа заявки	path	string
data	<input type="file"/> Выбор файла Не выбран ни один файл	Файл подписи	formData	file

Response Messages

HTTP Status Code	Reason	Response Model	Headers
204	Файл подписи обновлён		
401	Недостаточно прав; Ключ авторизации просрочен		
403	Нет доступа к заявке		
404	Заявка либо документ заявки не найдены		

[Try it out!](#)

Для подписания документа заявки необходимо сформировать ЭП содержимого документа.

Поддерживаются следующие алгоритмы подписи:

- ГОСТ Р 34.10-2001
- ГОСТ Р 34.10-2012

Справочники

Группа методов для получения справочников системы.

Категории заявителя

GET /v1/Categories/Profile Категории заявителя

Response Class (Status 200)
Категории

Model Example Value

```
[
  {
    "Name": "Переписка/Категория"
  }
]
```

Response Content Type

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Ключ авторизации просрочен		

Метод возвращает категории заявителя. Каждая категория в названии содержит полный путь. В таком виде название категории передаётся при создании/редактировании атрибутов заявки.

Для передачи сведений о физическом лице используется новая схема для формирования XML. Изменения в схеме:

- В структуре XML добавлен элемент TYPE у элемента RPAPPLGR
- Для обратной совместимости новый элемент является необязательным; в случае отсутствия элемента TYPE тип заявителя устанавливается как юридическое лицо (ОПФ устанавливается как «ЮЛ» либо «ИН2»)
- Следующие значения элемента TYPE допустимы:
 - o 1 — физическое лицо (ОПФ либо «ФИЗ» либо «ИН1»)
 - o 2 — юридическое лицо (ОПФ либо «ЮЛ» либо «ИН2»)
 - o 4 — индивидуальный предприниматель (ОПФ равен «ИП»)

```
<xs:element name="TYPE" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Тип заявителя: физ. лицо, юр. лицо</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="1">
        <xs:annotation>
          <xs:documentation>1 - Физическое лицо</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="2">
        <xs:annotation>
          <xs:documentation>2 - Юридическое лицо</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="4">
        <xs:annotation>
          <xs:documentation>4 - Индивидуальный предприниматель</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

Шаблоны (типы) документов заявителя

GET /v1/DocumentTypes/Profile Шаблоны документов заявителя

Response Class (Status 200)
Шаблоны документов

Model Example Value

```
[
  {
    "Name": "Письмо",
    "Code": "TradeMark",
    "Description": "Для переписки"
  }
]
```

Response Content Type

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Ключ авторизации просрочен		

Метод возвращает шаблоны (типы) документов заявителя. Для каждого типа документа возвращается название и описание.

Общие шаблоны (типы) документов

GET /v1/DocumentTypes/Shared Общие шаблоны документов

Response Class (Status 200)
Шаблоны документов

Model Example Value

```
[
  {
    "Name": "Письмо",
    "Code": "TradeMark",
    "Description": "Для переписки"
  }
]
```

Response Content Type

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Ключ авторизации просрочен		

Метод возвращает общие шаблоны (типы) документов. Для каждого типа документа возвращается название, код типа (при наличии) и описание.

К общим шаблонам (типам) документов добавлена машиночитаемая доверенность (МЧД) для заявок на ТЗ/ИЗ/ПМ/НМПТ/ГУ. Формат МЧД – [версия 3](#). Для данного типа документов обязательна отсоединенная подпись. В заявке допускается размещение нескольких МЧД.

```
{
  "Name": "Машиночитаемая доверенность",
  "Code": "PowerOfAttorney",
  "Description": "Машиночитаемая доверенность (МЧД)"
},
```

Загруженные МЧД отображаются в интерфейсе АРМ Регистратор в виде HTML-страницы.

▼ Просмотр документа

Версия формата	EMCHD_1
Флаг обязательности элементов	Указаны основные элементы В2В доверенности: <ul style="list-style-type: none">• Элементы, необходимые для получения государственных и муниципальных услуг с использованием платформы полномочий Госуслуг (ЕСИА)
Идентификатор файла	ON_EMCHD_20240104_a51368af-8f9a-3b39-9308-427755f3d2dd
Внешний идентификатор объекта ЕСИА	0
Состав и структура документа	
Доверенность	
Сведения о доверенности	
Вид доверенности	«Обычная» доверенность (с возможностью отзыва)
Признак возможности оформления передоверия	Без права передоверия