

**РЕГЛАМЕНТ**  
программного доступа (API) к ресурсам ГосРеестра

**версия: 2.40**

СОДЕРЖАНИЕ

Введение.....	3
Authorization.....	4
Общее описание.....	4
Входные параметры.....	4
Возвращаемые данные.....	4
Структурный пример запроса / ответа.....	4
SearchService: метод GetStatements.....	5
Общее описание.....	5
Входные параметры.....	5
Особенности пакетной обработки.....	5
Возвращаемые данные.....	5
Описание состава и структуры файла.....	6
Структурный пример запроса / ответа.....	6
SearchService: метод GetGeneralInfoOnIP.....	8
Общее описание.....	8
Входные параметры.....	8
Особенности работы метода.....	8
Возвращаемые данные.....	8
Описание состава и структуры файла.....	9
Структурный пример запроса / ответа.....	9
SearchService: метод GetInvalidPassportDetailed.....	11
Общее описание.....	11
Входные параметры.....	11
Особенности работы метода.....	11
Возвращаемые данные.....	11
Описание состава и структуры файла.....	12
Структурный пример запроса / ответа.....	13

## ВВЕДЕНИЕ

Настоящий документ описывает программный интерфейс (API) для предоставления удаленного доступа к актуальным данным ГосРеестра в части ЕГРЮЛ и ЕГРИП: методы получения данных, их аргументы и возвращаемые структуры.

API доступен по адресу: <http://api.GosReestrRF.ru:7070/>

Работа через программный интерфейс осуществляется через сервисы:

### 1. **Authorization**

Данный сервис отвечает за процедуру авторизации.

Адрес: <http://api.GosReestrRF.ru:7070/Authorization>

### 2. **SearchService**

Сервис поиска информации в информационном ресурсе ГосРеестра. Данный сервис включает в себя следующие методы:

- **GetStatements** - применение данного метода позволяет посредством одной команды получать основные сведения и полные выписки из ГосРеестра в части ЕГРЮЛ и ЕГРИП в формате xml в отношении нескольких юридических лиц и индивидуальных предпринимателей, поиск по нескольким поисковым элементам: ОГРН и ИНН юридических лиц и индивидуальных предпринимателей.

Адрес: <http://api.GosReestrRF.ru:7070/SearchService/GetStatements>

- **GetGeneralInfoOnIP** – применение данного метода позволяет проверить физических лиц на их регистрацию в качестве индивидуального предпринимателя. Поиск осуществляется по точному совпадению всех поисковых элементов: Фамилия, Имя, Отчество и Дата рождения. Ответ содержит установочную информацию на физических лиц, являющихся индивидуальными предпринимателями, так и полные выписки из ГосРеестра в части ЕГРИП.

Адрес: <http://api.GosReestrRF.ru:7070/SearchService/GetGeneralInfoOnIP>

- **GetInvalidPassportDetailed** – применение данного метода позволяет проверить паспорта граждан Российской Федерации (серия и номер паспорта) на их наличие в списке недействительных [утраченных (похищенных), оформленных на утраченных (похищенных) бланках паспорта гражданина Российской Федерации, выданных в нарушение установленного порядка, а также признанных недействительными] паспортов граждан Российской Федерации, удостоверяющих личность гражданина Российской Федерации на территории Российской Федерации. Список формируется Главным Управлением по вопросам миграции МВД России посредством АС «Российский Паспорт». Ответ содержит xml-код, содержащий серию и номер паспорта и его статус как проходящего в списке недействительных паспортов.

Адрес: <http://api.GosReestrRF.ru:7070/SearchService/GetInvalidPassportDetailed>

## AUTHORIZATION

**Общее описание**

Сервис содержит метод авторизации пользователя.

**Входные параметры**

В качестве входных параметров, передаются следующие аргументы:

- grant\_type – тип авторизации, всегда будет равен = password, так как авторизация происходит по имени пользователя и паролю
- userName – имя пользователя
- password – пароль пользователя

Параметр	Тип	Описание
grant_type	string	Тип авторизации
userName	string	Имя пользователя
Password	string	Пароль пользователя

**Возвращаемые данные**

Метод возвращает json, в котором содержатся три параметра:

- access\_token - идентификатор сессии (токен), который необходимо использовать в дальнейшей работе с сервисом. Токен выдается на одни сутки, с момента запроса;
- token\_type – тип токена
- expires\_in – время срока действия токена, в секундах

Параметр	Тип	Описание
Access_token	string	Идентификатор сессии (токен)
token_type	string	Тип возвращаемого токена
expires_in	string	Время окончания срока действия данного токена

**Структурный пример запроса / ответа**

Пример request

```
POST /Authorization HTTP/1.1
Host: api.GosReestrRF.ru:7070
Content-Type: application/x-www-form-urlencoded
Cache-Control: no-cache
```

```
grant_type=password&userName=API@GosReestrRF.ru&password=!Test123456
```

```
http://api.GosReestrRF.ru:7070/Authorization
```

Пример response

```
{
  "access_token": "kmj_aCTRceAovIG0L5LoSM5TXTztfNevitxy2RqnCXwGY17-cjxh-
oFuVa094B1bV1qjq5rr3-XZPy72gn4nwb5K3_2xjSokzD94wHJ4LLaBoXo-
At2QM7n6mQfvsQpCWNObCzsA02gOWm7fZQGpEEmsyiVtpam10yVt6YYSbu8gFQMF7p_9KEXskwoAM
_ZgpBUrgMBpvxsZowSL7Ih1SdC0_hPu01nfnAGNxNi0Z1AbuzRVXpPxA71YTK4PVE7E",
  "token_type": "bearer",
  "expires_in": 86399
}
```

## SEARCHSERVICE: МЕТОД GETSTATEMENTS

### Общее описание

Применение данного метода позволяет посредством одной команды получать основные сведения и полные актуальные выписки из ГосРеестра в части ЕГРЮЛ и ЕГРИП в формате xml в отношении нескольких юридических лиц и индивидуальных предпринимателей, поиск по нескольким поисковым элементам: ОГРН и ИНН юридических лиц и индивидуальных предпринимателей.

Метод GetStatements возвращает массив json-элементов, каждый из которых содержит:

- Id запроса - некий уникальный идентификатор запроса, формируемый на стороне клиента;
- XML код который содержит актуальную выписку из ГосРеестра в части ЕГРЮЛ для юридического лица и ЕГРИП для индивидуального предпринимателя;
- XML код который содержит основные сведения об юридическом лице или индивидуальном предпринимателе.

### Входные параметры

Параметр	Тип	Описание
requestId	string	Уникальный идентификатор запроса
regNumbers	string	Список необходимых ОГРН/ОГРНИП/ИНН

### Особенности пакетной обработки

- Максимальное количество ОГРН / ОГРНИП / ИНН в рамках одного запроса не более: 10 000 идентификаторов.
- С целью бесперебойности процесса и достижения максимальной скорости обработки рекомендуется отправлять пакеты последовательно, то есть формирование нового пакетного запроса осуществляется после получения ответа на предыдущий пакетный запрос.
- Сжатие строки с XML кодом, происходит при помощи стандартной библиотеки .NET Framework System.IO.Compression и класса GZipStream.

### Возвращаемые данные

Возвращается массив json-элементов, содержащий найденные данные по входным критериям.

Параметр	Тип	Описание
requestId	string	Уникальный идентификатор запроса, сформированный клиентом.
xml_allInfo	string	XML код, для найденного индивидуального предпринимателя или юридического лица, в сжатом виде.
xml_generalInfo	string	XML код в котором содержатся основные сведения об индивидуальном предпринимателе или юридическом лице.

## Описание состава и структуры файла

Описание состава и структуры файла (XML), содержащего сведения из ГосРеестра, определяются нормативными документами, описывающих требования к XML файлам передачи данных и представлены в соответствующих схемах описания структуры XML документа:

- **Требования к составу и структуре файла, содержащего сведения из ГосРеестра в части ЕГРЮЛ** (в электронной версии документ представлен в файле "*ГосРеестр в части ЕГРЮЛ - Требования к составу и структуре файла*") и файл описания структуры XML документа "**GR\_UL\_XML\_Scheme\_03.xsd**";
- **Требования к составу и структуре файла, содержащего сведения из ГосРеестра в части ЕГРИП** (в электронной версии документ представлен в файле "*ГосРеестр в части ЕГРИП - Требования к составу и структуре файла*") и файл описания структуры XML документа "**GR\_IP\_XML\_Scheme\_03.xsd**"

## Структурный пример запроса / ответа

### Пример request

```
POST /SearchService/GetStatements HTTP/1.1
Host: api.GosReestrRF.ru:7070
Content-Type: application/json;charset=UTF-8
Authorization: bearer 1PveD8vQQFa3016x5L-G544_-zjsYDfVbFO1SGQLhidZJ-
BycUnsArEzGf8QulmMLodCTnLU9bAHZZ97Wom7sO1EFKKTm7T12DacdrG6vWgMoDhLzB7Iu8sg2nP
037qyceEiC9mM3rV41jSWQBNokabyzo3ZdqjfyOCAyJfa8qvO6WG9uMZL0wa58qllp0CaN5xEiGs4
0o2J_ZrnB47coGsbz1VqiCvghesF79LR_1ZtfAgsNaeAfeQvIkrlNGLa8wb98-
2TiQz2Vn43Hk9isvo6GrNbMhEfaTvOkSwF5o8ICxn8g5SFqZQH19_Sqr0d
Cache-Control: no-cache
```

### Body:

```
{
  "requestId": "123-456-789",
  "regNumbers": ["1117423000287", "4401184680", "231516752681", "317237500417340"]
}
```

http://api.GosReestrRF.ru:7070/SearchService/GetStatements

### Пример response

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: length
[
  {
    "requestId": "123-456-789",
    "xml_allInfo": "G1cQfq/U/3Dkp0Ysda45w4eas56d45646as5d463qwerAAdsdfd98",
    "xml_generalInfo": "<?xml version='1.0' encoding='windows-1251'?>
<EGRIP xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xmlns:xsd='http://www.w3.org/2001/XMLSchema'>
<СВИП ОГРНИП='317237500417837' ИННФЛ='231516752681' Статус='ДЕЙСТВУЮЩИЙ'>
<СвФЛ><ФИОРус Фамилия='СЛАЩЕВ' Имя='ДМИТРИЙ' Отчество='АНАТОЛЬЕВИЧ' /></СвФЛ>
<СвРожд ДатаРожд='1983-03-26' МестоРожд='ГОР. НОВОРОССИЙСК КРАСНОДАРСКОГО
КРАЯ' /></СВИП></EGRIP>"
  },
  {
    "requestId": "123-456-789",
    "xml_allInfo": "I0cQfY+Uf2j52dgz4xuWgH2KorxF+YXsKkJKQMPk2TxyWTYPoMUx1",
    "xml_generalInfo": "<?xml version='1.0' encoding='windows-1251'?>
<EGRIP xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xmlns:xsd='http://www.w3.org/2001/XMLSchema'>
<СВИП ОГРНИП='317237500417340' ИННФЛ='231129915410' Статус='ДЕЙСТВУЮЩИЙ'>
<СвФЛ><ФИОРус Фамилия='ТИТОВ' Имя='МАКСИМ' Отчество='ОЛЕГОВИЧ' /></СвФЛ>
<СвРожд ДатаРожд='1996-02-28' МестоРожд='ГОР. КРАСНОДАР' /></СВИП></EGRIP>"
  },
]
```

# Г О С Р Е Е С Т Р

```
{
  "requestId": "123-456-789",
  "xml_allInfo": "G8cVvi/QdljwqgUoav+4JAHJuSqK3hV9hSYoDLQ20ABNLyXZbuLak",
  "xml_generalInfo": "<?xml version='1.0' encoding='windows-1251'?>
<EGRIP xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xmlns:xsd='http://www.w3.org/2001/XMLSchema'>
<СвИП ОГРНИП='316231500053162' ИННФЛ='231516752681' Статус='Индивидуальный
предприниматель прекратил деятельность в связи с принятием им
соответствующего решения'>
<СвФЛ><ФИОРус Фамилия='СЛАЩЕВ' Имя='ДМИТРИЙ' Отчество='АНАТОЛЬЕВИЧ' /></СвФЛ>
<СвРожд ДатаРожд='1983-03-26' МестоРожд='ГОР. НОВОРОССИЙСК КРАСНОДАРСКОГО
КРАЯ' /></СвИП></EGRIP>"
},
{
  "requestId": "123-456-789",
  "xml_allInfo": "QWEQfaeKf5jSA0+yvLu62WA7TxQvPFHFF5AU1SqwCwKER0shCSmHW",
  "xml_generalInfo": "<?xml version='1.0' encoding='windows-1251'?>
<EGRUL xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xmlns:xsd='http://www.w3.org/2001/XMLSchema'>
<СвЮЛ ОГРН='1174401008880' ИНН='4401184680' Статус='ДЕЙСТВУЮЩЕЕ'>
<СвНаимЮЛ НаимЮЛПолн='ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ &quot;ВОЛЖСКИЙ
БЕРЕГ&quot;' НаимЮЛСокр='ООО &quot;ВОЛЖСКИЙ БЕРЕГ&quot;' /></СвЮЛ></EGRUL>"
},
{
  "requestId": "123-456-789",
  "xml_allInfo": "G0cUvq/Udlj5ImolL9ld73ptBchV1ZteVeoTNFEUKSVSEzW95Cckr",
  "xml_generalInfo": "<?xml version='1.0' encoding='windows-1251'?>
<EGRUL xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xmlns:xsd='http://www.w3.org/2001/XMLSchema'>
<СвЮЛ ОГРН='1117423000287' ИНН='7423009871' Статус='ДЕЙСТВУЮЩЕЕ'>
<СвНаимЮЛ НаимЮЛПолн='ПОТРЕБИТЕЛЬСКИЙ КООПЕРАТИВ ЖИВОТНОВОДЧЕСКОЕ
ТОВАРИЩЕСТВО № 2' НаимЮЛСокр='---' /></СвЮЛ></EGRUL>"
}
]
```

**SEARCHSERVICE: МЕТОД GETGENERALINFOONIP****Общее описание**

Применение данного метода позволяет проверить физических лиц на их регистрацию в качестве индивидуального предпринимателя. Поиск осуществляется по точному совпадению всех поисковых элементов: Фамилия, Имя, Отчество и Дата рождения. Ответ содержит установочную информацию на физических лиц, являющихся индивидуальными предпринимателями, так и полные выписки из ГосРеестра в части ЕГРИП.

Метод GetGeneralInfoOnIP возвращает массив json-элементов, каждый из которых содержит:

- Id запроса - некий уникальный идентификатор запроса, формируемый на стороне клиента;
- XML код который содержит актуальную выписку об индивидуальном предпринимателе;
- XML код который содержит основные сведения об индивидуальном предпринимателе.

**Входные параметры**

Параметр	Тип	Описание
requestId	string	Уникальный идентификатор запроса
lastName	string	Фамилия физического лица
firstName	string	Имя физического лица
patronymic	string	Отчество физического лица
birthDate	date	Дата рождения физического лица, в формате 'YYYY-MM-DD'

**Особенности работы метода**

- Необходимая кодировка запроса: UTF-8.
- requestId – это идентификатор запроса, который клиент устанавливает сам, для идентификации ответов. Формат идентификатора – произвольный.
- С целью бесперебойности процесса и достижения максимальной скорости обработки рекомендуется отправлять пакеты последовательно, то есть формирование нового пакетного запроса осуществляется после получения ответа на предыдущий пакетный запрос.
- Сжатие строки с XML кодом, происходит при помощи стандартной библиотеки .NET Framework System.IO.Compression и класса GZipStream

**Возвращаемые данные**

Возвращается массив json-элементов, содержащий найденные данные по входным критериям.

Параметр	Тип	Описание
requestId	string	Уникальный идентификатор запроса, сформированный клиентом.
xml_allInfo	string	XML код, для найденного индивидуального предпринимателя, в сжатом виде.



xml_generalInfo	string	XML код в котором содержатся основные сведения об физическом лице, являющимся индивидуальным предпринимателем.
-----------------	--------	--

## Описание состава и структуры файла

Описание состава и структуры файла (XML), содержащего сведения из ГосРеестра, определяются нормативными документами, описывающих требования к XML файлам передачи данных и представлены в соответствующих схемах описания структуры XML документа:

- **Требования к составу и структуре файла, содержащего сведения из ГосРеестра в части ЕГРЮЛ** (в электронной версии документ представлен в файле "*ГосРеестр в части ЕГРЮЛ - Требования к составу и структуре файла*") и файл описания структуры XML документа "GR\_UL\_XML\_Scheme\_03.xsd";
- **Требования к составу и структуре файла, содержащего сведения из ГосРеестра в части ЕГРИП** (в электронной версии документ представлен в файле "*ГосРеестр в части ЕГРИП - Требования к составу и структуре файла*") и файл описания структуры XML документа "GR\_IP\_XML\_Scheme\_03.xsd"

## Структурный пример запроса / ответа

### Пример request

```
POST /SearchService/GetGeneralInfoOnIP HTTP/1.1
Host: api.GosReestrRF.ru:7070
Content-Type: application/json; charset=UTF-8
Authorization: bearer 1PveD8vQQFa3016x5L-G544_-zjsYDfVbFOlSGQLhidZJ-
BycUnsArEzGf8QulmMLodCTnLU9bAHZZ97Wom7sO1EFKKTm7T12DAcdrG6vWgMoDhLzB7Iu8sg2nP
037qyceEiC9mM3rV41jSWQBNokabyzo3ZdqjfyOCAyjfa8qvO6WG9uMZL0wa58qllp0CaN5xEiGs4
0o2J_ZrnB47coGsbz1VqiCvghesF79LR_1Zt fAgsNaeAfeQvIkr1NGLa8wb98-
2TiQz2Vn43Hk9isvo6GrNbMhEfaTvOkSwF5o8ICxn8g5SFqZQH19_Sqr0d
Cache-Control: no-cache
```

### Body:

```
[
  {
    "requestId": "123-456-789",
    "lastName": "БЕКТЯШКИНА",
    "firstName": "ГАЛИНА",
    "patronymic": "СЕРГЕЕВНА",
    "birthDate": "1949-10-19"
  },
  {
    "requestId": "123-456-789",
    "lastName": "УРСОК",
    "firstName": "РАШИД",
    "patronymic": "ТЕМИРОВИЧ",
    "birthDate": "1988-09-07"
  }
]
```

http://api.GosReestrRF.ru:7070/SearchService/GetGeneralInfoOnIP

### Пример response

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: length
[
  {
    "requestId": "123-456-789",
    "xml_allInfo": "V9tuE0kQ/ZXWPC2S7XTP+CrF4QmteEP8AqBVJEgkIBHEpDQah/WsT3",
```

# Г О С Р Е Е С Т Р

```
"xml_generalInfo": "<?xml version="1.0" encoding="windows-1251"?><EGRIP
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><СВИП ОГРНИП="305631813700712"
ИННФЛ="631800326376" Статус="ФИЗИЧЕСКОЕ ЛИЦО УТРАТИЛО ГОСУДАРСТВЕННУЮ
РЕГИСТРАЦИЮ В КАЧЕСТВЕ ИНДИВИДУАЛЬНОГО ПРЕДПРИНИМАТЕЛЯ НА ОСНОВАНИИ СТ.3
ФЕДЕРАЛЬНОГО ЗАКОНА ОТ 23.06.2003 №76-ФЗ"><СвФЛ><ФИОРус Фамилия="БЕКТЯШКИНА"
Имя="ГАЛИНА" Отчество="СЕРГЕЕВНА" /></СвФЛ><СвРожд ДатаРожд="1949-10-19"
МестоРожд="РОССИЯ,,,,,,,,," /></СВИП></EGRIP>"
},
{
"requestId": "123-456-789",
"xml_allInfo": "vx8AAB+LCAAAAAAABAFs/Qf1joqQVkytKogxYz1NR9K3oLzRBEaB1gAZo",
"xml_generalInfo": "<?xml version="1.0" encoding="windows-1251"?><EGRIP
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><СВИП ОГРНИП="304636309800022"
ИННФЛ="636300975244" Статус="Индивидуальный предприниматель прекратил
деятельность в связи с принятием им соответствующего решения"><СвФЛ><ФИОРус
Фамилия="БЕКТЯШКИНА" Имя="ГАЛИНА" Отчество="СЕРГЕЕВНА" /></СвФЛ><СвРожд
ДатаРожд="1949-10-19" МестоРожд="гор.Самара" /></СВИП></EGRIP>"
},
{
"requestId": "123-456-789",
"xml_allInfo": "ABwAAB+LCAAAAAAABADtWvtPG0cUfq/U/7DyUyvZsLPrC5Yweaqqv1X9C01U",
"xml_generalInfo": "<?xml version="1.0" encoding="windows-1251"?><EGRIP
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><СВИП ОГРНИП="309637736300014"
ИННФЛ="636300975244" Статус="Индивидуальный предприниматель прекратил
деятельность в связи с принятием им соответствующего решения"><СвФЛ><ФИОРус
Фамилия="БЕКТЯШКИНА" Имя="ГАЛИНА" Отчество="СЕРГЕЕВНА" /></СвФЛ><СвРожд
ДатаРожд="1949-10-19" МестоРожд="гор.Самара" /></СВИП></EGRIP>"
},
{
"requestId": "123-456-789",
"xml_allInfo": "rRAAAB+LCAAAAAAABADlWNluG0UUfpXRXoFkJzP+baQ4vUKIO8Qr0ApFgkSi",
"xml_generalInfo": "<?xml version="1.0" encoding="windows-1251"?><EGRIP
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><СВИП ОГРНИП="307010725000017"
ИННФЛ="010701237131" Статус="ДЕЙСТВУЮЩИЙ"><СвФЛ><ФИОРус Фамилия="УРСОК"
Имя="РАШИД" Отчество="ТЕМИРОВИЧ" /></СвФЛ><СвРожд ДатаРожд="1988-09-07"
МестоРожд="ГОРОД ТЕУЧЕЖСК ААО КРАСНОДАРСКОГО КРАЯ" /></СВИП></EGRIP>"
}
]
```

## SEARCHSERVICE: МЕТОД GETINVALIDPASSPORTDETAILED

### Общее описание

Применение данного метода позволяет проверить паспорта граждан Российской Федерации (серия и номер паспорта) на их наличие в списке недействительных паспортов граждан Российской Федерации, удостоверяющих личность гражданина Российской Федерации на территории Российской Федерации.

Под недействительными паспортами понимаются следующие категории паспортов граждан Российской Федерации, удостоверяющих личность гражданина Российской Федерации на территории Российской Федерации:

- утраченные (похищенных);
- оформленные на утраченных (похищенных) бланках паспорта гражданина Российской Федерации;
- выданные в нарушение установленного порядка;
- признанных недействительными

Оригинальный список формируется Главным Управлением по вопросам миграции МВД России посредством АС «Российский Паспорт».

Ответ содержит xml-код, содержащий серию и номер паспорта и его статус как проходящего в списке недействительных паспортов.

Метод GetInvalidPassportDetailed возвращает json-элемент, который содержит:

- Id запроса - некий уникальный идентификатор запроса, формируемый на стороне клиента;
- XML код содержащий информацию о паспортах;

### Входные параметры

Параметр	Тип	Описание
requestId	string	Уникальный идентификатор запроса
regNumbers	string	Список искомых паспортных данных

### Особенности работы метода

- Серию и номер паспортных данных можно указывать как слитно, так и отдельно. Все не цифровые символы из каждой строки будут игнорироваться.
- В случае если в присланном списке (перечне паспортов подлежащих проверке) паспорт не содержится в списке недействительных, то ему присваивается статус: В СПИСКЕ НЕДЕЙСТВИТЕЛЬНЫХ НЕ ЗНАЧИТСЯ.
- Для работы с xml используется xsd схема - GR\_Passport\_XML\_Scheme\_01.xsd

### Возвращаемые данные

Возвращается json-элемент, содержащий найденные данные по входным критериям.

Параметр	Тип	Описание
requestId	string	Уникальный идентификатор запроса, сформированный клиентом.
xml_allInfo	string	XML код, содержащий информацию о паспортах – серия и номер паспорта и статус.

## Описание состава и структуры файла

### 1. Перечень структурных элементов логической модели файла обмена

Наименование элемента	Сокращенное наименование (код) элемента	Признак типа элемента	Формат элемента	Дополнительная информация
Дата актуализации информации	ДатаАкт	А		Типовой элемент <xs:date>
Сведения о паспорте	СвПаспорт	С		Состав элемента представлен в табл. 1.2

Наименование элемента	Сокращенное наименование (код) элемента	Признак типа элемента	Формат элемента	Дополнительная информация
Серия и номер паспорта	СерияНомер	А		Элемент формируется в результате конкатенации серии и номера паспорта
Статус паспорта	Статус	А		Принимает одно из следующих значений: – НЕДЕЙСТВИТЕЛЬНЫЙ; – В СПИСКЕ НЕДЕЙСТВИТЕЛЬНЫХ НЕ ЗНАЧИТСЯ

### 2. язык описания структуры XML получаемого клиентом документа

Язык описания структуры XML получаемого клиентом документа представлен в отдельном файле "**GR\_Passport\_XML\_Scheme\_01.xsd**"

**Структурный пример запроса / ответа****Пример request**

```
POST /SearchService/GetInvalidPassportDetailed HTTP/1.1
Host: api.GosReestrRF.ru:7070
Content-Type: application/json;charset=UTF-8
Authorization: bearer 1PveD8vQQFa3016x5L-G544_-zjsYDfVbFOlSGQLhidZJ-
BycUnsArEzGf8QulmMLodCTnLU9bAHZZ97Wom7sO1EFKKTm7Tl2DAcdrG6vWgMoDhLzB7Iu8sg2nP
037qyceEiC9mM3rV41jSWQBNokabyzo3ZdqjfyOCAyjfa8qvO6WG9uMZL0wa58qllp0CaN5xEiGs4
0o2J_ZrnB47coGsbz1VqiCvghesF79LR_lZtfAgsNaeAfeQvIkr1NGLa8wb98-
2TiQz2Vn43Hk9isvo6GrNbMhEfaTvOkSwF5o8ICxn8g5SFqZQH19_Sqr0d
Cache-Control: no-cache
```

**Body:**

```
{
  "requestId" : "123-456-789",
  "regNumbers" : ["3514313213", "8100093251", "0500 271171"]
}
http://api.GosReestrRF.ru:7070/SearchService/GetInvalidPassportDetailed
```

**Пример response**

```
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: length
{
  "requestId": "123-456-789",
  "xml_allInfo": "<?xml version='1.0' encoding='windows-1251'?>
<ROSPASSPORT xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xmlns:xsd='http://www.w3.org/2001/XMLSchema' ДатаАкт='2017-12-05'>
  <СвПаспорт СерияНомер='3514313213' Статус='В СПИСКЕ НЕДЕЙСТВИТЕЛЬНЫХ
НЕ ЗНАЧИТСЯ' />
  <СвПаспорт СерияНомер='8100093251' Статус='НЕДЕЙСТВИТЕЛЬНЫЙ' />
  <СвПаспорт СерияНомер='0500271171' Статус='НЕДЕЙСТВИТЕЛЬНЫЙ' />
</ROSPASSPORT>"
}
```