Addoro
Visit, Lumaparksvägen 9, 120 21 Stockholm
Post, Box 43, 121 25 Stockholm
Phone, +46 8 600 37 21
E-mail, info@addoro.com

Version 2.1, May 2020

Addoro API

Introduction

The Addoro API can be used to work with batches and documents. It is intended to be used by third party applications that wants to make a tight integration with Addoro. The third party application is referred to as the client in the rest of this document. The API is HTTPS based and consists of a number of URLs that can be invoked. Each call either returns a JSON structure or a document (most often a PDF document).

All data passed should be UTF8 encoded.

To handle security the client must first issue a login call. This call will return a session id. The client will then pass this session id with all calls. The session id should either be provided as a cockie named JSESSIONID or passed with the url as described below. Many http client libraries will automatically send all cockies retrieved with the following call. This means that the session handling will be automatically handled.

To test the API the Addoro test server, test.addoro.com can be used. Contact Addoro to get a login on the test server.

Addoro API Reference

To all API calls the parameter prettyResponse can be set to true if you want more readable format of the JSON returned by the API.

Example:

```
https://test.addoro.com/api/login?user=john&pass=secretpassword&prettyResponse=true
```

will return:

```
{
  "success": true,
  "ticket": <a ticket>
}
```

while

```
https://test.addoro.com/api/login?user=john&pass=secretpassword
```

will return:

```
{"success":true,"ticket":<a ticket>}
```

You may use an ordinary browser when you test API calls and then this parameter can be useful.

Login

Logs on to the Addoro system and establishes a session that can be used for executing subsequent commands. It returns a session id that must be provided in all other commands. The server might do a HTTP redirect and the client implementation should be able to handle HTTP redirects.

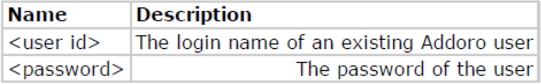
Syntax:

```
https://<host name>/api/login?user=<user id>&pass=<password>
```

Example:

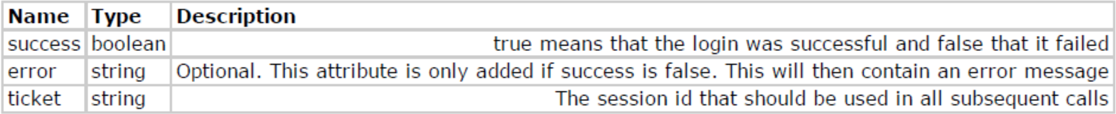
```
https://test.addoro.com/api/login?user=john&pass=secretpassword
```

Parameters:



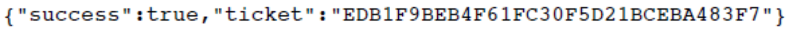
Name	Description
<user id>	The login name of an existing Addoro user
<password>	The password of the user

Returns a JSON with the following key/values:



Name	Type	Description
success	boolean	true means that the login was successful and false that it failed
error	string	Optional. This attribute is only added if success is false. This will then contain an error message
ticket	string	The session id that should be used in all subsequent calls

Example of returned JSON



```
{"success":true,"ticket":"EDB1F9BEB4F61FC30F5D21BCEBA483F7"}
```

initBatchUpload

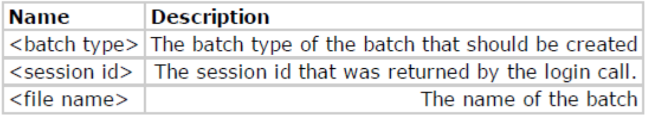
Uploading a batch is done in two steps. First a call to initBatchUpload should be made. This will create an empty batch with no data associated with it. initBatchUpload returns a JSON structure that contains the batch id that will uniquely identify the batch and will be used to refer to the batch in future calls.

Syntax:

```
https://<host name>/api/members/upload/initBatchUpload.action;jsessionid= <session
id>?batchTypeName=<batch type>&fileName=<file name>
```

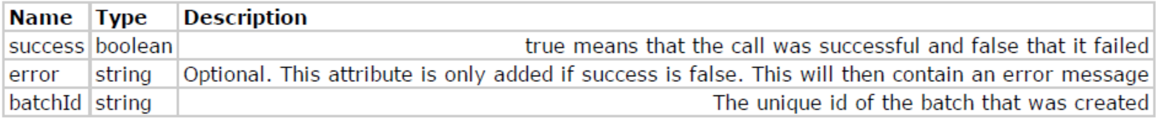
Example:

```
https://test.addoro.com/api/members/upload/initBatchUpload.action;jsessionid=EDB1F9BEB4F
61FC30F5D21BCEBA483F7?batchTypeName=3L&fileName=AugustInvoices.xml
```



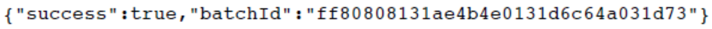
Name	Description
<batch type>	The batch type of the batch that should be created
<session id>	The session id that was returned by the login call.
<file name>	The name of the batch

Returns a JSON with the following key/values:



Name	Type	Description
success	boolean	true means that the call was successful and false that it failed
error	string	Optional. This attribute is only added if success is false. This will then contain an error message
batchId	string	The unique id of the batch that was created

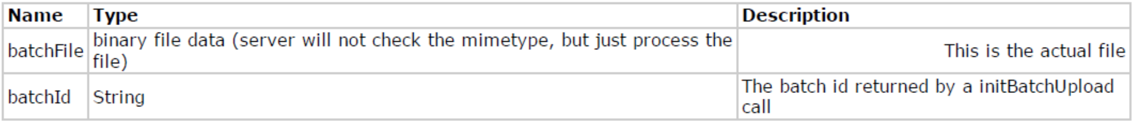
Example of returned JSON:



```
{"success":true,"batchId":"ff80808131ae4b4e0131d6c64a031d73"}
```

uploadBatch

Uploads data to a batch that was created with the initBatchUpload call. The url should be used with the HTTP Post command with a multipart request entity. This is similar to how a HTML form is posted. Parameters are here passed as separate parts of the multipart request. See Java example below for an example of how to create this request. The multipart should consist of the following parts:



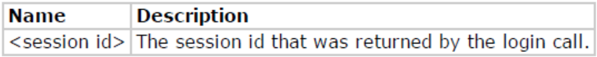
Name	Type	Description
batchFile	binary file data (server will not check the mimetype, but just process the file)	This is the actual file
batchId	String	The batch id returned by a initBatchUpload call

Syntax:

```
https://<host name>/api/members/upload/initBatchUpload.action;jsessionid=<session id>
```

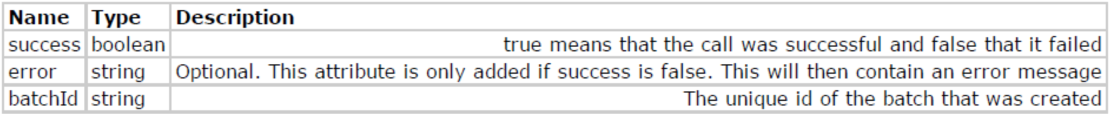
Example:

```
https://test.addoro.com/api/members/upload/uploadBatch.action;jsessionid=EDB1F9BEB4F61FC
30F5D21BCEBA483F7
```



Name	Description
<session id>	The session id that was returned by the login call.

Returns a JSON with the following key/values:



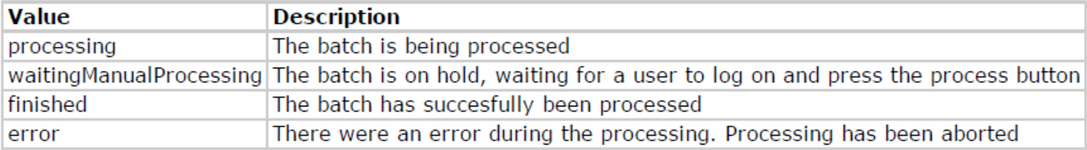
Name	Type	Description
success	boolean	true means that the call was successful and false that it failed
error	string	Optional. This attribute is only added if success is false. This will then contain an error message
batchId	string	The unique id of the batch that was created

Example of returned JSON:

```
{"success":true,"batchId":"ff80808131ae4b4e0131d6c64a031d73"}
```

getBatchStatus

returns the status of a Batch. The status can take the following values



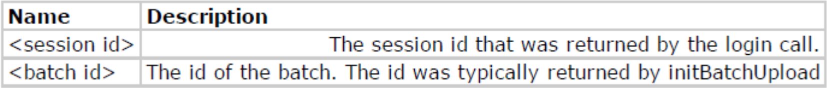
Value	Description
processing	The batch is being processed
waitingManualProcessing	The batch is on hold, waiting for a user to log on and press the process button
finished	The batch has succesfully been processed
error	There were an error during the processing. Processing has been aborted

Syntax:

```
https://<host name>/api/members/upload/getBatchStatus.action;jsessionid=<session
id>?batchId=<batch id>
```

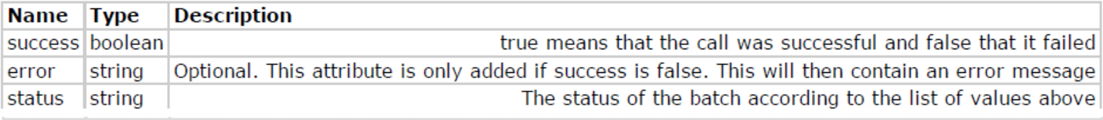
Example:

```
https://test.addoro.com/api/members/upload/getBatchStatus.action;jsessionid=<session
id>?batchId=ff80808131ae4b4e0131d6c64a031d73
```



Name	Description
<session id>	The session id that was returned by the login call.
<batch id>	The id of the batch. The id was typically returned by initBatchUpload

Returns a JSON with the following key/values:



Name	Type	Description
success	boolean	true means that the call was successful and false that it failed
error	string	Optional. This attribute is only added if success is false. This will then contain an error message
status	string	The status of the batch according to the list of values above

Example of returned JSON:

```
{"success":true,"status":"finnished"}
```

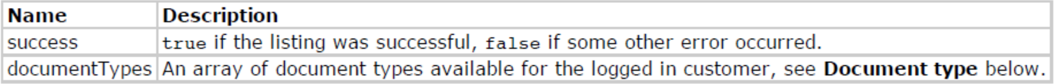
List document types

Syntax:

```
https://<host name>/api/members/documenType/list.action
```

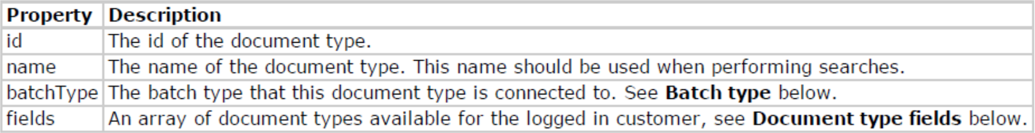
Lists the available document types and its metadata definitions available to the logged in customer.

Response



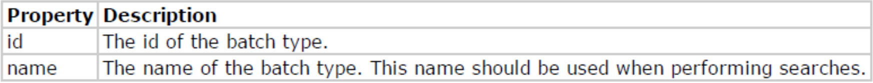
Name	Description
success	`true` if the listing was successful, `false` if some other error occurred.
documentTypes	An array of document types available for the logged in customer, see **Document type** below.

Document Type



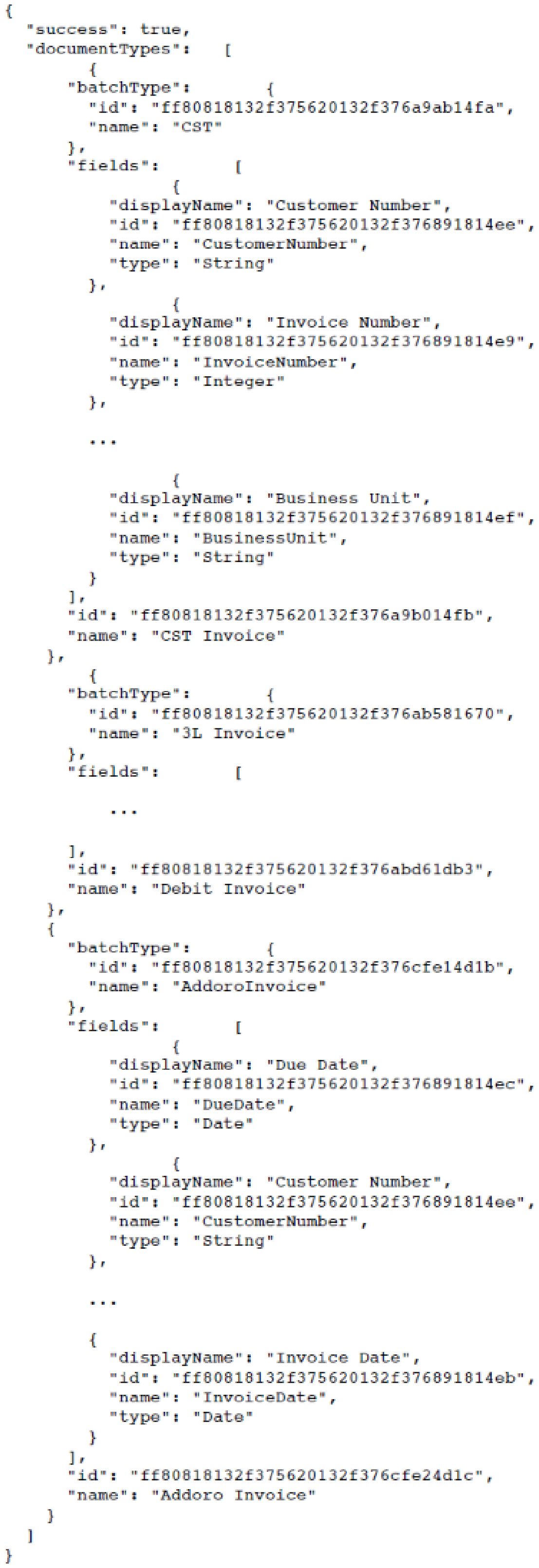
Property	Description
id	The id of the document type.
name	The name of the document type. This name should be used when performing searches.
batchType	The batch type that this document type is connected to. See **Batch type** below.
fields	An array of document types available for the logged in customer, see **Document type fields** below.

Batch Type



Property	Description
id	The id of the batch type.
name	The name of the batch type. This name should be used when performing searches.

Example: `https://test.addoro.com/api/members/documentType/list.action?prettyResponse=true`



```
{
  "success": true,
  "documentTypes":   [
        {
      "batchType":       {
        "id": "ff80818132f375620132f376a9ab14fa",
        "name": "CST"
      },
      "fields":       [
                {
          "displayName": "Customer Number",
          "id": "ff80818132f375620132f376891814ee",
          "name": "CustomerNumber",
          "type": "String"
        },
                {
          "displayName": "Invoice Number",
          "id": "ff80818132f375620132f376891814e9",
          "name": "InvoiceNumber",
          "type": "Integer"
        },

        ...

                {
          "displayName": "Business Unit",
          "id": "ff80818132f375620132f376891814ef",
          "name": "BusinessUnit",
          "type": "String"
        }
      ],
      "id": "ff80818132f375620132f376a9b014fb",
      "name": "CST Invoice"
    },
        {
      "batchType":       {
        "id": "ff80818132f375620132f376ab581670",
        "name": "3L Invoice"
      },
      "fields":       [

          ...

      ],
      "id": "ff80818132f375620132f376abd61db3",
      "name": "Debit Invoice"
    },
    {
      "batchType":       {
        "id": "ff80818132f375620132f376cfe14d1b",
        "name": "AddoroInvoice"
      },
      "fields":       [
                {
          "displayName": "Due Date",
          "id": "ff80818132f375620132f376891814ec",
          "name": "DueDate",
          "type": "Date"
        },
                {
          "displayName": "Customer Number",
          "id": "ff80818132f375620132f376891814ee",
          "name": "CustomerNumber",
          "type": "String"
        },

        ...

        {
          "displayName": "Invoice Date",
          "id": "ff80818132f375620132f376891814eb",
          "name": "InvoiceDate",
          "type": "Date"
        }
      ],
      "id": "ff80818132f375620132f376cfe24d1c",
      "name": "Addoro Invoice"
    }
  ]
}
```

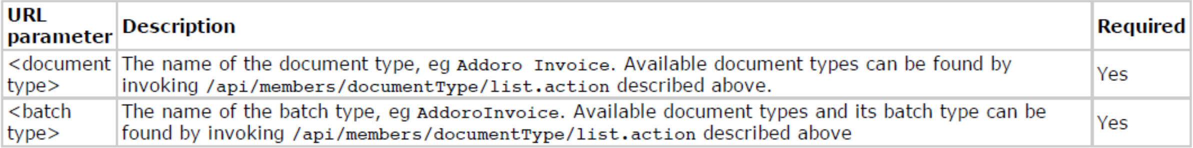
Document search

Search for documents in a given document type. The batch type and document type name are given by the caller in the URL. Session id should also be passed in the URL or by using cookies as described above.

Search result can be paged using `query.startIndex` and `query.maxItems` parameters and sorted using parameters `query.sortColumn` and `query.ascending`. Each search also return the `totalSize` which tells how many documents in total that matched your query.

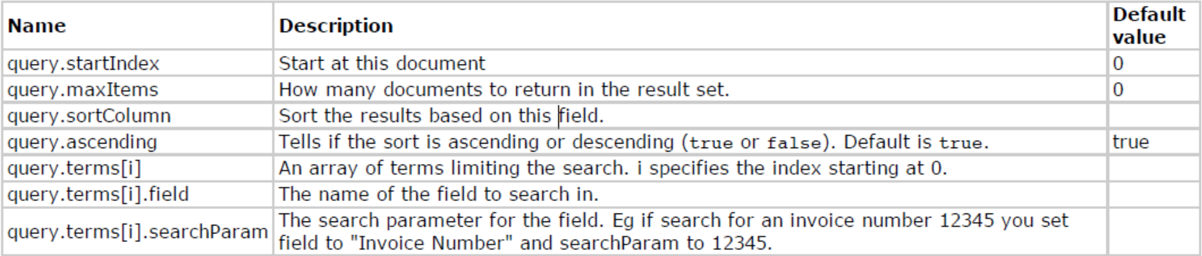
Syntax:

```
https://<host name>/api/members/search/<batch type>/<document type>/search.action
```



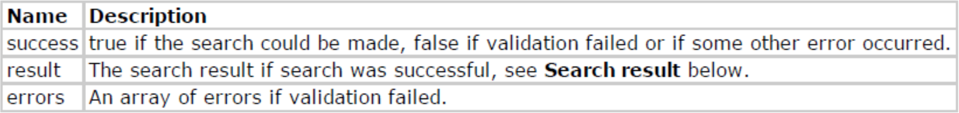
URL parameter	Description	Required
<document type>	The name of the document type, eg `Addoro Invoice`. Available document types can be found by invoking `/api/members/documentType/list.action` described above.	Yes
<batch type>	The name of the batch type, eg `AddoroInvoice`. Available document types and its batch type can be found by invoking `/api/members/documentType/list.action` described above	Yes

Parameters



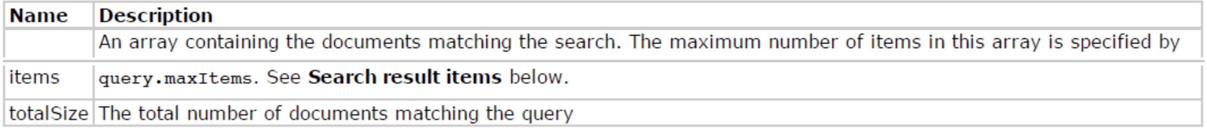
Name	Description	Default value
query.startIndex	Start at this document	0
query.maxItems	How many documents to return in the result set.	0
query.sortColumn	Sort the results based on this field.	
query.ascending	Tells if the sort is ascending or descending (`true` or `false`). Default is `true`.	true
query.terms[i]	An array of terms limiting the search. i specifies the index starting at 0.	
query.terms[i].field	The name of the field to search in.	
query.terms[i].searchParam	The search parameter for the field. Eg if search for an invoice number 12345 you set field to "Invoice Number" and searchParam to 12345.	

Response



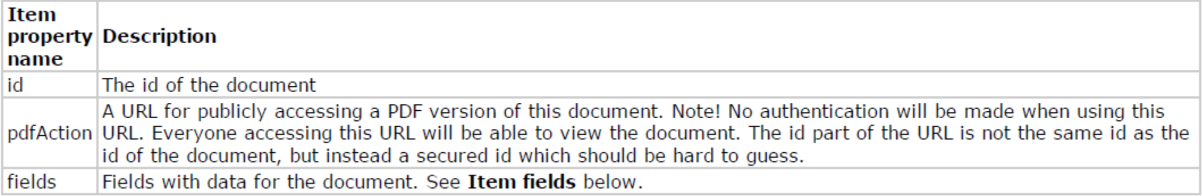
Name	Description
success	true if the search could be made, false if validation failed or if some other error occurred.
result	The search result if search was successful, see **Search result** below.
errors	An array of errors if validation failed.

Search result (result)



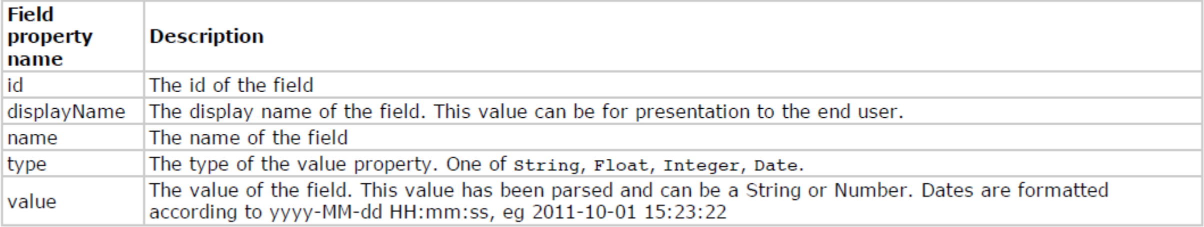
Name	Description
items	An array containing the documents matching the search. The maximum number of items in this array is specified by `query.maxItems`. See **Search result items** below.
totalSize	The total number of documents matching the query

Search result items (result.items[i])



Item property name	Description
id	The id of the document
pdfAction	A URL for publicly accessing a PDF version of this document. Note! No authentication will be made when using this URL. Everyone accessing this URL will be able to view the document. The id part of the URL is not the same id as the id of the document, but instead a secured id which should be hard to guess.
fields	Fields with data for the document. See **Item fields** below.

Item fields (result.items[i].fields[i])



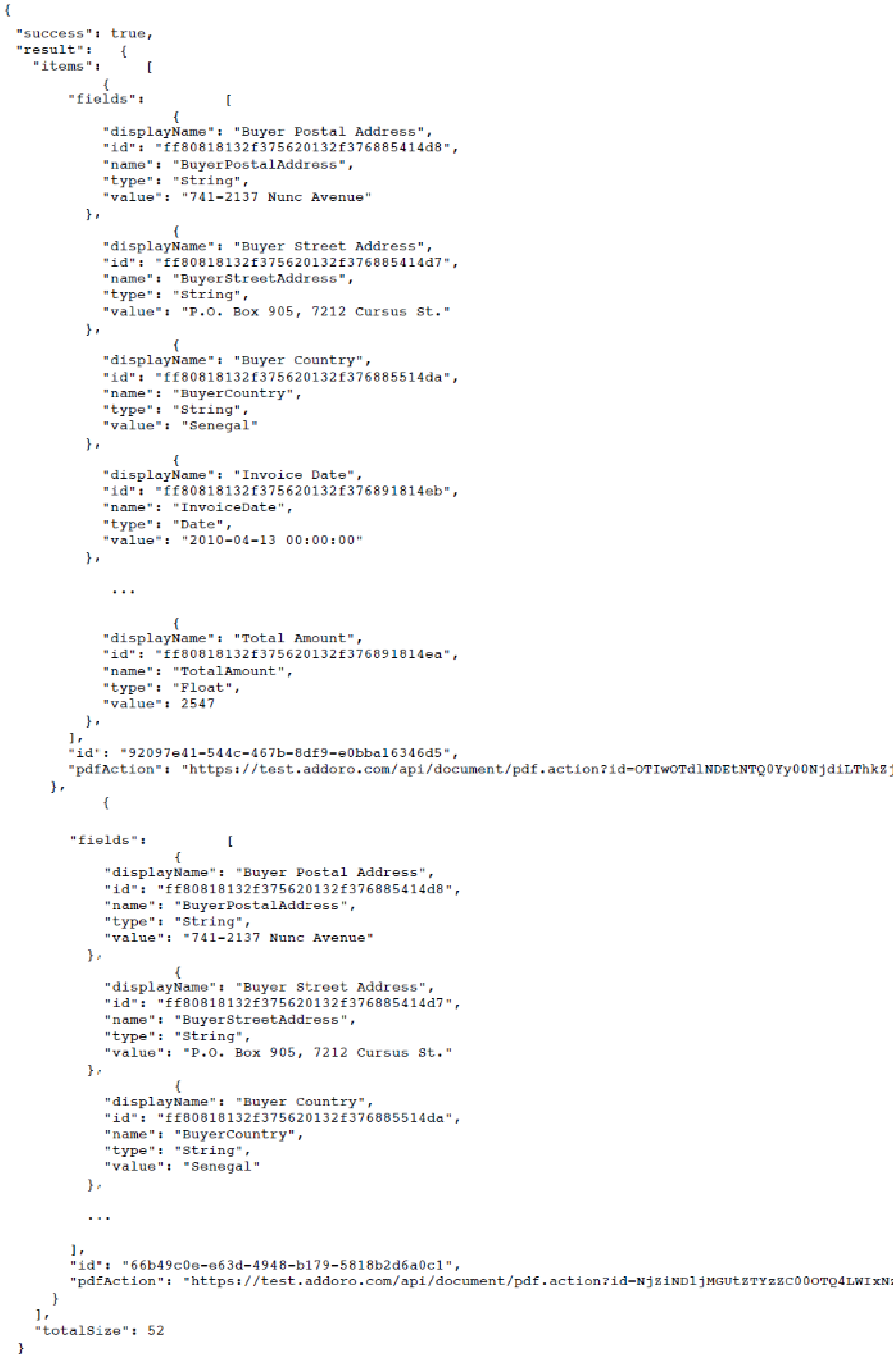
Field property name	**Description**
id	The id of the field
displayName	The display name of the field. This value can be for presentation to the end user.
name	The name of the field
type	The type of the value property. One of `String`, `Float`, `Integer`, `Date`.
value	The value of the field. This value has been parsed and can be a String or Number. Dates are formatted according to yyyy-MM-dd HH:mm:ss, eg 2011-10-01 15:23:22

Example

URL:

```
/api/members/search/AddoroInvoice/Addoro%20Invoice/search.action?prettyResponse=true&que
ry.maxItems=1&query.terms[0].field=BuyerName&query.terms[0].searchParam=a*
```

Response:



```
{
  "success": true,
  "result":   {
    "items":     [
            {
        "fields":          [
                    {
            "displayName": "Buyer Postal Address",
            "id": "ff80818132f375620132f376885414d8",
            "name": "BuyerPostalAddress",
            "type": "String",
            "value": "741-2137 Nunc Avenue"
          },
                    {
            "displayName": "Buyer Street Address",
            "id": "ff80818132f375620132f376885414d7",
            "name": "BuyerStreetAddress",
            "type": "String",
            "value": "P.O. Box 905, 7212 Cursus St."
          },
                    {
            "displayName": "Buyer Country",
            "id": "ff80818132f375620132f376885514da",
            "name": "BuyerCountry",
            "type": "String",
            "value": "Senegal"
          },
                    {
            "displayName": "Invoice Date",
            "id": "ff80818132f375620132f376891814eb",
            "name": "InvoiceDate",
            "type": "Date",
            "value": "2010-04-13 00:00:00"
          },

            ...

                    {
            "displayName": "Total Amount",
            "id": "ff80818132f375620132f376891814ea",
            "name": "TotalAmount",
            "type": "Float",
            "value": 2547
          },
        ],
        "id": "92097e41-544c-467b-8df9-e0bba16346d5",
        "pdfAction": "https://test.addoro.com/api/document/pdf.action?id=OTIwOTdlNDEtNTQ0Yy00NjdiLThkZj
      },
            {

        "fields":          [
                    {
            "displayName": "Buyer Postal Address",
            "id": "ff80818132f375620132f376885414d8",
            "name": "BuyerPostalAddress",
            "type": "String",
            "value": "741-2137 Nunc Avenue"
          },
                    {
            "displayName": "Buyer Street Address",
            "id": "ff80818132f375620132f376885414d7",
            "name": "BuyerStreetAddress",
            "type": "String",
            "value": "P.O. Box 905, 7212 Cursus St."
          },
                    {
            "displayName": "Buyer Country",
            "id": "ff80818132f375620132f376885514da",
            "name": "BuyerCountry",
            "type": "String",
            "value": "Senegal"
          },

          ...

        ],
        "id": "66b49c0e-e63d-4948-b179-5818b2d6a0c1",
        "pdfAction": "https://test.addoro.com/api/document/pdf.action?id=NjZiNDljMGUtZTYzZC00OTQ4LWIxN
      }
    ],
    "totalSize": 52
  }
```

Code Examples

Contact Addoro for example code anders.hermansson@addoro.com.