

FactuurSturen.nl API v1

FactuurSturen.nl has an API so external programs can reach the FactuurSturen.nl account of a user. To get access to our API, you'll need to generate an unique key for your account.

About this document

This document is for **version 1** of the API.

Create API key

- Log in in your FactuurSturen.nl account.
- Go to your settings (left in the menu)
- Click on the option "Connections" (verbindingen).
- Click on the button "Create API key" (maak API sleutel)
- A popup appears. Fill a name so you can recognize your key.
- Make sure the option "Active" is set
- Click the green "Add" button

You'll find your key on your screen right now. With this key you have access to your account.

Revoke access

To revoke access to your account with the API, you can delete the key. The application using this key now has no access. If you want to temporarily disable a key, you can click the title of the key and remove the checkbox with the label Active. Your key will not be deleted, but the application now as no access.

How to use the API

The API of FactuurSturen.nl enables you to access the information inside a FactuurSturen.nl account. The API is RESTful and uses XML or JSON for the data representation. Default is XML.

PHP class library

To help you on the way and to make communicating with our API easier, we have a PHP-class to help you getting started and to help you with the API calls.

You can download the PHP-class here: <https://www.factuursturen.nl/docs/fsnl-api.zip>

In this document, we'll use the PHP-class for the examples.

Create a connection the the API

To start, we'll have to create a connection to our API with the PHP Class. The API is available on <https://www.factuursturen.nl/api/v1>. Be sure you are using HTTPS. The API isn't available for HTTP connections. Lets include the class and setup the API object.

Example

First, include the class in your script.

```
// include the api class at the top of your script
require_once("fsnl_api.class.php");
```

Example

Lets define the API url and method.

```
// setup the API call with the URL and the method
$request = new fsnl_api('https://www.factuursturen.nl/api/v1/clients',
    'GET');
```

The `$request` object is now created. We can use this object for further communication.

Authentication

For accessing the API, we use **HTTP Basic Authentication**. Each request you make must contain your **username** and **api-key**.

You can add your username and api-key to the `$request` object.

Example

```
// set the username of the account where you want to connect to
$request->setUsername('myuser');
// enter the api-key as password
$request->setPassword('I3NimRMhrzPkmXWai9KdnUN42n81CkyVhY8mndEp');
```

Choose JSON or XML

JSON is the **default format** how we return data. If you would like to use **XML**, you can add this command to your script.

```
$request->setAcceptType('application/xml');
```

Building the POST body for POST and PUT methods

When using the method POST and PUT, you'll need to send extra information to our API. Like product information, invoice lines, etc. To add the POST body to your request, add the following lines of code.

Example

```
// build the post body we are going to submit
$request->buildPostBody(array(
    'code' => 'Soundboard-50S',
    'name' => 'Flash Soundboard 50 Samples',
    'price' => 39.95,
    'taxes' => 20
));
```

Use JSON

You can also JSON to send your extra information to our server. Send your JSON POST or PUT request with this extra header:

```
Content-Type: application/json
```

You can now provide your data as a JSON:

```
{
  "code": "Soundboard-50S",
  "name": "Flash Soundboard 50 Samples",
  "price": 39.95,
  "taxes": 20
}
```

Executing the command

If you want to execute a command, we have the right line of code for you.

```
$request->execute();
```

Read the response

After you have executed the command, we'll send you a response. To read the response, use the following command.

```
$request->getResponseBody();
```

You can save the results in a variable of something else.

Get more info about the response

To get more info about the response, like the HTTP status code, transfer times etc, you can use the command `getResponseInfo()`;

```
$request->getResponseInfo();
```

You will receive an array holding more detailed information.

```
Array (
    [url] => https://www.factuursturen.nl/api/v1/clients
    [content_type] => text/html
    [http_code] => 200
    [header_size] => 342
    [request_size] => 168
    [filetime] => -1
    [ssl_verify_result] => 0
    [redirect_count] => 0
    [total_time] => 0.074751
    [namelookup_time] => 2.7E-5
    [connect_time] => 0.000119
    [pretransfer_time] => 0.018351
    [size_upload] => 0
    [size_download] => 807
    [speed_download] => 10795
    [speed_upload] => 0
    [download_content_length] => 807
    [upload_content_length] => 0
    [starttransfer_time] => 0.074744
    [redirect_time] => 0
    [certinfo] => Array ()
)
```

Rate limiting to 250 requests/hour

In order to provide our web users with the best service, we limit the number of API requests to our services to **250 requests/hour**. Please make sure your implementation respects these limits. The `X-RateLimit-Limit` and `X-RateLimit-Remaining` HTTP response headers contain information about the limit you are bound to. When you exceed the limit you will receive a 403 HTTP response code.

If your application needs more than 250 requests/hour, please contact us at support@factuursturen.nl

Defining the number of rows received

The Client API, Invoices API, Saved Invoices API, Products API, Repeating Invoices API and Search API have some extra options that can be set to control how many rows are returned. This way it's possible to build paging in your application to make sure the performance is at it's best.

There are two options possible:

Description	Method	URL
Define number of results	GET	?count={number}
Define start record	GET	?start={number}

Example

In this api-call we specify the number of results we want.

```
// setup the API call with the URL and the method
$request = new fsnl_api('https://www.factuursturen.nl/api/v1/invoices?
count=5', 'GET');
```

Example

In this api-call we'll also define a start record. Let's say we want to retrieve the records 15 till 25. We'll ask the api to start at record 15 and give us the next 10 records.

```
// setup the API call with the URL and the method
$request = new fsnl_api('https://www.factuursturen.nl/api/v1/invoices?
start=15&count=10', 'GET');
```

The Invoices API has even more exciting filter functions. Check the Invoices API chapter.

Clients API

To manage your clients, you can use the following commands.

Description	Method	URL
Get all contacts	GET	/api/v1/clients
Get one contact	GET	/api/v1/clients/{id}
Create new contact	POST	/api/v1/clients/
Update a contact	PUT	/api/v1/clients/{id}
Delete a contact	DELETE	/api/v1/clients/{id}

All attributes used for clients

Attribute	Type	Note
clientnr <small>required</small>	integer	
contact	string(50)	
showcontact	boolean	Show the contact name on the invoice
company	string(50)	
address	string(50)	
zipcode	string(7)	
city	string(50)	
country	integer or string	Country id. You can get a list of country id's with the function api/v1/countrylist . When creating or updating a client, you can supply a country id or a country name. We'll then try to find the id of the country you supplied.
phone	string(13)	
mobile	string(13)	
email	string(255)	Invoice is sent to this e-mail address, if the sendmethod is e-mail
bankcode	string(32)	The IBAN number of the client
biccode	string(32)	
taxnumber	string(18)	
tax_shifted	boolean	If the taxes on the invoice is shifted to the receiver

lastinvoice <i>read only</i>	date	When last invoice to this client was sent
sendmethod	string(11)	How to send the invoice to the receiver. 'mail' : print the invoices yourself. 'email' : send invoices through e-mail. 'printcenter' : send invoice through the printcenter.
paymentmethod	string(10)	How the invoice is going to be paid. 'bank' : normal bank account transfer 'autocollect' : the invoice will be collected (incasso)
top	integer	The term of payment in days. Defines when the invoice has to be paid by the recipient
stddiscount	integer	Standard discount percentage for this client. Every invoice defined for this client will automatically get this discount percentage.
mailintro	string(128)	The first line used in the e-mail to address the recipient. Example: "Dear sir/madam,"
reference	array	Three lines that will be printed on the invoice. Can be used for references to other documents or something else. In the array are three fields: 'line1' , 'line2' and 'line3' . All fieldstypes are strings.
notes	string	Notes saved for this client
notes_on_invoice	boolean	Print the field 'notes' on every invoice for the client
active	boolean	Non-active clients are hidden in the web application.
default_doclang	string(2)	In what language the invoice will be generated for this client. empty: default language 'nl' : dutch 'en' : english 'de' : german 'fr' : france 'es' : spanish
default_email <i>read only</i>	integer	ID of used e-mail text
currency	string(3)	Used currency in invoice. Like 'EUR' , 'USD' , etc.
mandate_id	string	The mandate identification
mandate_date	date	The date of the signature

collecttype	string(4)	The collection type: 'OOFF' : single direct debit 'FRST' : direct debit: first collection 'RCUR' : direct debit: recurring collection
timestamp <i>read only</i>	date	Date and time when record was updated
tax_type	string(5)	Will show if the products on the invoice for this client will be handled as excluding or including tax: 'intax' : products will be handled as including tax 'extax' : products will be handled as excluding tax

Get all contacts

GET /api/v1/clients

Response

Code: 200

Data: JSON or XML containing all attributes

Get one contact

GET /api/v1/clients/12345

Response

Code: 200

Data: JSON or XML containing all attributes

Create new contact

POST /api/v1/clients/

Create a POST with at least the required attributes.

Example

```
// build the post body we are going to submit
$request->buildPostBody(array(
    'contact' => 'John Doe',
    'showcontact' => true,
    'company' => 'Johnny Bravo Inc.',
    'address' => 'Sir John Road 100',
    'zipcode' => '1337 JB',
    'city' => 'Johnsville',
    'country' => 146,
    'phone' => '010 123 4567',
    'mobile' => '0612 34 56 78',
    'email' => 'johnny@bravo.com',
    'bankcode' => 'NL91INGB0001234567',
    'biccode' => 'INGBNL2A',
    'taxnumber' => 'NL001234567B01',
    'tax_shifted' => false,
    'sendmethod' => 'email',
    'paymentmethod' => 'bank',
    'top' => 3,
    'stddiscount' => 5.30,
    'mailintro' => 'Dear Johnny,',
    'reference' => array(
        'line1' => 'Your ref: ABC123',
        'line2' => 'Our ref: XZX0029/2932/001',
        'line3' => 'Thank you for your order'
    ),
    'notes' => 'This client is always late with his payments',
    'notes_on_invoice' => false,
    'active' => true,
    'default_doclang' => 'en',
    'currency' => 'USD',
    'tax_type' => 'intax'
));
```

Response

Code: 201

Data: clientnr

Update a contact

PUT /api/v1/clients/12345

Create a PUT with at least the required attributes and fields you want to update. See the example above for more help.

Example

Here we will update the e-mail address and phone number of a client.

```
// setup the API call with the URL and the method
$request = new fsnl_api('https://www.factuursturen.nl/api/v1/clients/12345', 'PUT');

// build the post body we are going to submit
$request->buildPostBody(array(
    'phone' => '010 123 4567',
    'email' => 'johnny@bravo.com'
));

// execute the request
$request->execute();
```

Response

Code: 200

Data: null

Delete a contact

DELETE /api/v1/clients/12345

If there are documents set for this client, like invoices, the client cannot be deleted. The API will return an error with HTTP code 403.

Response

Code: 200

Data: null

Products API

To manage your products, you can use the following commands.

Description	Method	URL
Get all products	GET	/api/v1/products
Get one product	GET	/api/v1/products/{id}
Create new product	POST	/api/v1/products/
Update a product	PUT	/api/v1/products/{id}
Delete a product	DELETE	/api/v1/products/{id}

All attributes used for products

Attribute	Type	Note
id <small>read only, required</small>	integer	Unique number for product
code	string(20)	An own defined product code (not displayed on invoices, but it's for your own reference, like an article code)
name	string	Name of the product shown on the invoice
price	float	The price of the product with max. 3 decimals
taxes	integer	Tax percentage. This must be a percentage that's also set in the user his account.
priceintax	float	The including tax price of the product with max. 3 decimals. When this field is empty, we'll calculate the priceintax automatically based on ' price ' and ' taxes '.

Get all products

GET /api/v1/products

Response

Code: 200

Data: JSON or XML containing all attributes

Get one product

GET /api/v1/products/12345

Response

Code: 200

Data: JSON or XML containing all attributes

Create new product

POST /api/v1/products/

Create a POST with at least the required attributes.

Example

```
// build the post body we are going to submit
$request->buildPostBody(array(
    'code' => 'Soundboard-50S',
    'name' => 'Flash Soundboard 50 Samples',
    'price' => 39.95,
    'taxes' => 21
));
```

Response

Code: 201

Data: id

Update a product

PUT /api/v1/products/12345

Response code: 200

Create a POST with at least the required attributes and attributes of fields you want to update. Check the example above how to build the POST body.

Response

Code: 200

Data: null

Delete a product

DELETE /api/v1/products/12345

Response

Code: 200

Data: null

Invoices API

To manage your invoices, you can use the following commands.

Description	Method	URL
Get all invoices	GET	/api/v1/invoices/
Get one invoice	GET	/api/v1/invoices/{invoicnr}
Create new invoice	POST	/api/v1/invoices/
Delete an invoice	DELETE	/api/v1/invoices/{invoicnr}

All attributes used for invoices

Attribute	Type	Note
invoicnr	string	Invoice number including layout When posting an invoice, do not specify or define this attribute. If you really want to define the invoice number, please include the invoice number as an integer number without any prefix layout. So to create INVOICE00023, please send us 23 as integer. Use this with caution! When used wrong it can ruin your administration and invoice numbers. Only use this when absolutely necessary.
reference	array	Contains reference lines on the invoice. 'line1', 'line2', 'line3'. All are strings.
lines	array	All invoice lines on the invoice Per line: 'amount' 'amount_desc' 'description' 'tax_rate' 'price' 'discount_pct' (discount percentage) 'tax_country' (boolean, input only)* 'linetotal' (read only) * if you want to use a foreign tax for this line, set this to true. More info: https://www.factuursturen.nl/help/69/ To use text lines on an invoice, only supply 'description' and do not use the other fields. Check the example below for more info.

profile	id	The ID of the used profile. Default is default profile.
discounttype	string(10)	The type of discount. 'amount' or 'percentage'
discount	float	If 'discounttype' is amount, then this is the amount of discount set on the invoice. If 'discounttype' is set to 'percentage' , this is the discount percentage set.
paymentcondition	string	The payment condition set on the invoice. Default is the payment condition set in the application.
paymentperiod	integer	Term of payment in days. Default is the payment period set with the client.
collection <i>read only</i>	boolean	If invoice is an automatic collection
taxes <i>read only</i>	array	All taxes defined in the invoice Per taxes: 'rate' : tax rate 'sum' : taxes on the invoice for this rate 'sum_of' : amount where this taxes is relates to
tax <i>read only</i>	float	Total of all taxes on this invoice
totalintax <i>read only</i>	float	Invoice total including taxes
clientnr	integer	Client number
contact <i>read only</i>	string(50)	
company <i>read only</i>	string(50)	
address <i>read only</i>	string(50)	
zipcode <i>read only</i>	string(7)	
city <i>read only</i>	string(50)	
country <i>read only</i>	integer or string	Country id. You can get a list of country id's with the function api/v1/countrylist . When creating or updating a client, you can supply a country id or a country name. We'll then try to find the id of the country you supplied.
phone <i>read only</i>	string(13)	
mobile <i>read only</i>	string(13)	
invoicenote <i>read only</i>	string	A note that is saved with the invoice

sent <i>read only</i>	date	The date the invoice is sent
uncollectible <i>read only</i>	date	The date when the invoice is marked as uncollectible
lastreminder <i>read only</i>	date	The date when the last reminder was sent
open <i>read only</i>	float	The amount that is still open on the invoice. If this amount is 0, then the invoice is paid. If it is negative, there is paid more than the invoice amount.
paiddate	date	The date of the last received payment
payment_url <i>read only</i>	string	The complete URL where the invoice can be paid
duedate <i>read only</i>	date	The duedate of the invoice. This is the sent-date + the payment period in days.

Get all invoices

GET /api/v1/invoices

Response

Code: 200

Data: JSON or XML containing all attributes

Get one invoice

GET /api/v1/invoices/F012345

Response

Code: 200

Data: JSON or XML containing all attributes

Special filtering for invoices

The invoices have a special filter to quickly access different invoice types, like invoices that are overdue. Possible filter options:

Description	Method	URL
Get open invoice (all invoices that are not fully paid yet)	GET	?filter=open
Get overdue invoices	GET	?filter=overdue
Get sent invoices	GET	?filter=sent

Get party paid invoices	GET	?filter=partly
Get invoices with too much paid	GET	?filter=toomuch
Get paid invoices	GET	?filter=paid
Get invoices that couldn't be collected	GET	?filter=uncollectible

Example

Receive all invoices that are overdue

```
// setup the API call with the URL and the method
$request = new fsnl_api('https://www.factuursturen.nl/api/v1/invoices?
filter=overdue', 'GET');
```

Filter invoices on a date

You can supply a since and an until date to get invoices from a specific period.

Description	Method	URL
Since date	GET	?since=YYYY-MM-DD
Until date optional	GET	?until=YYYY-MM-DD

Example

Receive all invoices that are overdue

```
// setup the API call with the URL and the method
$request = new fsnl_api('https://www.factuursturen.nl/api/v1/invoices?
since=2016-12-15&until=2016-12-31', 'GET');
```

Create new invoice

POST /api/v1/invoices/

When posting a new invoice, there are several extra attributes needed besides the ones shown above.

Attribute	Type	Note
action required	string(6)	Define the action what we should do with your request. 'send' : send the invoice 'save' : save the invoice as a draft 'repeat' : plan a repeating invoice
sendmethod semi-required	string(11)	How to send the invoice to the receiver. Required when you use the action 'send' 'mail' : print the invoices yourself. We'll send you the invoice number so you can execute a command to retrieve the PDF if you need so. 'email' : send invoices through e-mail. It will be sent immediately. 'printcenter' : send invoice through the printcenter.
savename semi-required	string(20)	When the action is 'save' or 'repeat' you must supply a savename. We'll save the invoice under that name.
overwrite_if_exist	boolean	If a savename already exists, it will not be overwritten unless this attribute is set to 'true' . Default is 'false' .
convert_prices_to_euro	boolean	When this option is set to 'true' we will convert all the given prices on the invoices to euro, based on the currency set in the selected client and the invoice date (to retrieve the current exchange rate).
alreadypaid not possible for repeating invoices	float or string	Mark a part of the invoice as already paid. We'll place an extra line in the PDF that says that a part of the invoice is already paid. We'll also register that a part is paid in your account. Supply the amount in float . You can also supply 'full' as a value. We will mark the invoice total as amount.
alreadypaidmethod not possible for repeating invoices	string(32)	Enter a string of the payment method. It can be anything you want, for example: 'Pin', 'Cash', 'MasterCard', 'Gift from Santa', etc.

When planning a repeating invoice with the **action 'repeat'** there are several fields also needed for us to process the invoice

Attribute	Type	Note
initialdate required	date	Date when the first invoice must be sent. Please use YYYY-MM-DD
finalsenddate optional	date	Date when the last invoice must be sent. After this date the repeating invoice entry is deleted. Please use YYYY-MM-DD
frequency required	string(10)	The frequency when the invoice must be sent. Based on the initialdate. Possible options: 'weekly' , 'monthly' , 'quarterly' , 'halfyearly' , 'yearly' , 'biweekly' , 'bimonthly' , 'fourweekly'
repeattype required	string(6)	Set if the repeating invoice is automatically sent by our system. 'auto' : send automatically when due 'manual' : do not send automatically

Example - sending an invoice through the e-mail

In this example we'll send an invoice through the e-mail. When this command is executed, the e-mail is immediately sent.

```
$request->buildPostBody(array(
    'clientnr' => 218064,
    'reference' => array(
        'line1' => 'Soundboard Setup',
        'line2' => 'Thank you for the order',
        'line3' => 'Our reference is: 3993029/11BD'
    ),
    'lines' => array(
        array('amount' => 50,
            'amount_desc' => 'panels',
            'description' => 'Sound buttons',
            'tax_rate' => 21,
            'price' => 5.952
        ),
        array('amount' => 1,
            'amount_desc' => '',
            'description' => 'Wooden case',
            'tax_rate' => 21,
            'dicount_pct' => 25,
            'price' => 500
        ),
        array('amount' => 10,
            'amount_desc' => 'hours',
            'description' => 'Support',
            'tax_rate' => 6,
            'price' => 62.5
        ),
        array('description' => 'This is a textline')),
    'discount' => 10,
    'discounttype' => 'percentage',
    'action' => 'send',
    'sendmethod' => 'email'
));
```

Example - save a draft invoice

In this example, we'll create an invoice and then save it as a draft so we can edit it later on in the web application.

```
$request->buildPostBody(array(
    'clientnr' => 218064,
    'lines' => array(
        array('amount' => 50,
            'amount_desc' => 'panels',
            'description' => 'Sound buttons',
            'tax_rate' => 21,
            'price' => 5.952,
        )
    ),
    'discount' => 10,
    'discounttype' => 'percentage',
    'action' => 'save',
    'savename' => 'my_invoice'
));
```

Example - plan a repeating invoice

In this example, we'll create an invoice and then plan it as a repeating invoice.

```
$request->buildPostBody(array(
    'clientnr' => 218064,
    'lines' => array(
        array('amount' => 50,
            'amount_desc' => 'panels',
            'description' => 'Sound buttons',
            'tax_rate' => 21,
            'price' => 5.952,
        )
    ),
    'discount' => 10,
    'discounttype' => 'percentage',
    'action' => 'repeat',
    'savename' => 'planning_invoice',
    'initialdate' => '2014-03-07',
    'finalsenddate' => '2016-01-01',
    'frequency' => 'monthly',
    'repeattype' => 'auto'
));
```

Response

Code: 201

Data: Invoice number

Example - send an invoice and mark it immediately as paid

In this example, we'll send an invoice and mark it directly as paid. Please note: marking the invoice as paid doesn't change anything to the invoice email text that is sent. Make sure the email text contains information that the invoice is already paid or use the sendmethod 'mail'. This way, no email is sent. You can create an own email in your program and retrieve the invoice pdf file with our api.

```
$request->buildPostBody(array(
    'clientnr' => 218064,
    'lines' => array(
        array('amount' => 50,
            'amount_desc' => 'panels',
            'description' => 'Sound buttons',
            'tax_rate' => 21,
            'price' => 5.952
        )
    ),
    'action' => 'send',
    'sendmethod' => 'mail',
    'paiddatetime' => '2014-09-21'
));
```

Response

Code: 201

Data: Invoice number

Example - send an invoice and mark a part as paid

In this example, we'll send an invoice and mark a part of the invoice as paid (like when the customer already paid a part of the invoice up front). Please note: marking the invoice as paid doesn't change anything to the invoice email text that is sent. Make sure the email text contains information that the invoice is already paid or use the sendmethod 'mail'. This way, no email is sent. You can create an own email in your program and retrieve the invoice pdf file with our api.

```
$request->buildPostBody(array(
    'clientnr' => 218064,
    'lines' => array(
        array('amount' => 50,
            'amount_desc' => 'panels',
            'description' => 'Sound buttons',
            'tax_rate' => 21,
            'price' => 5.952
        ),
        'action' => 'send',
        'sendmethod' => 'mail',
        'alreadypaid' => 150, // use 'full' for the invoice total
        'alreadypaidmethod' => 'Pin',
    ));
```

Response

Code: 201
Data: Invoice number

Delete an invoice

DELETE /api/v1/invoices/F012345

Response

Code: 200
Data: null

Invoice Payment API

Description	Method	URL
Mark payment	POST	/api/v1/invoices_payment/{id or invoicent}

All attributes used for invoices payments

Attribute	Type	Note
date required	date	Date of the payment. Please use YYYY-MM-DD
amount required	string	The amount that is paid or use 'full' . When 'full' is used, the invoice total including tax is marked as a payment.

Mark payment

POST /api/v1/invoices_payment/F012345

Example

```
$request->buildPostBody(array(  
    'date' => '2013-01-30',  
    'amount' => 'full'  
));
```

Response

Code: 201

Data: null

Saved Invoices API

To manage your saved invoices, you can use the following commands.

Description	Method	URL
Get all saved invoices	GET	/api/v1/invoices_saved/
Get one saved invoice	GET	/api/v1/invoices_saved/{id}
Delete saved invoice	DELETE	/api/v1/invoices_saved/{id}

All attributes used for saved invoices

Attribute	Type	Note
id read only	integer	Unique invoice id
name read only	string(20)	Unique name for the invoice
reference read only	array	Contains reference lines on the invoice. 'line1', 'line2', 'line3'. All are strings.
lines read only	array	All invoice lines on the invoice Per line: 'amount' 'amount_desc' 'description' 'tax_rate' 'price' 'discount_pct' (discount percentage) 'tax_country' (boolean, input only)* 'linetotal' (read only) * if you want to use a foreign tax for this line, set this to true. More info: https://www.factuursturen.nl/help/69/
profile read only	id	The ID of the used invoice profile
discounttype read only	string(10)	The type of discount. 'amount' or 'percentage'
discount read only	float	If 'discounttype' is amount, then this is the amount of discount set on the invoice. If 'discounttype' is set to 'percentage', this is the discount percentage set.

paymentcondition read only	string	The payment condition set on the invoice
paymentperiod read only	integer	Term of payment in days
totaldiscount read only	float	The total amount of discount set
totalintax read only	float	Invoice total including taxes
alreadypaid read only	float	The amount that is already marked as paid on the invoice
alreadypaidmethod read only	string(32)	The payment method of the part that is already paid
clientnr read only	integer	Client number
contact read only	string(50)	
company read only	string(50)	
address read only	string(50)	
zipcode read only	string(7)	
city read only	string(50)	
country read only	integer or string	Country id. You can get a list of country id's with the function api/v1/countrylist . When creating or updating a client, you can supply a country id or a country name. We'll then try to find the id of the country you supplied.

Get all saved invoices

GET /api/v1/invoices_saved/

Response

Code: 200

Data: JSON or XML containing all attributes

Get one saved invoice

GET /api/v1/invoices_saved/123

Response

Code: 200

Data: JSON or XML containing all attributes

Delete saved invoice

DELETE /api/v1/invoices_saved/123

Response

Code: 200

Data: null

Repeating Invoices API

To manage your repeating invoices, you can use the following commands.

Description	Method	URL
Get all repeating invoices	GET	/api/v1/invoices_repeated/
Get one repeating invoice	GET	/api/v1/invoices_repeated/{id}
Delete repeating invoice	DELETE	/api/v1/invoices_repeated/{id}

All attributes used for repeating invoices

Attribute	Type	Note
id read only	integer	Unique invoice id
name read only	string	The unique name of the invoice
reference read only	array	Contains reference lines on the invoice. 'line1' , 'line2' , 'line3' . All are strings.
lines read only	array	All invoice lines on the invoice Per line: 'amount' 'amount_desc' 'description' 'tax_rate' 'price' 'discount_pct' (discount percentage) 'tax_country' (boolean, input only)* 'linetotal' (read only) * if you want to use a foreign tax for this line, set this to true. More info: https://www.factuursturen.nl/help/69/
profile read only	id	The ID of the used invoice profile
discounttype read only	string	The type of discount. 'amount' or 'percentage'

discount read only	float	If ' discounttype ' is amount, then this is the amount of discount set on the invoice. If ' discounttype ' is set to ' percentage ', this is the discount percentage set.
paymentcondition read only	string	The payment condition set on the invoice
paymentperiod read only	integer	Term of payment in days
datesaved read only	date	The date the invoice was saved
totalintax read only	float	Invoice total including taxes
initialdate read only	date	The date when the first invoice was sent (or is going to be sent)
nextsenddate read only	date	The next date when the next invoice is going to be sent.
finalsenddate read only	date	The date when the last invoice is sent. After this date, this invoice will be deleted.
frequency read only	string	The frequency of sending
repeattype read only	string	If this invoice is automatically sent.
clientnr read only	integer	Client number
contact read only	string	
company read only	string	
address read only	string	
zipcode read only	string	
city read only	string	
country read only	integer or string	Country id. You can get a list of country id's with the function api/v1/countrylist . When creating or updating a client, you can supply a country id or a country name. We'll then try to find the id of the country you supplied.

Get all repeating invoices

GET /api/v1/invoices_repeated/

Response

Code: 200

Data: JSON or XML containing all attributes

Get one repeating invoice

GET /api/v1/invoices_repeated/123

Response

Code: 200

Data: JSON or XML containing all attributes

Delete repeating invoice

DELETE /api/v1/invoices_repeated/123

Response

Code: 200

Data: null

Get Invoices PDF API

To download a PDF of a sent invoice, you can use the following commands.

Description	Method	URL
Get invoice PDF	GET	/api/v1/invoices_pdf/{id or invoicentr}

Get invoice PDF

GET /api/v1/invoices_pdf/F012345

Response

Code: 200

Data: PDF

Profiles API

To get a list of used profiles, you can use the following commands

Description	Method	URL
Get all profiles	GET	/api/v1/profiles/

All attributes used for profiles

Attribute	Type	Note
id read only	integer	the unique id of this profile
name read only	string	The name of the profile

Get all profiles

GET /api/v1/profiles/

Response

Code: 200

Data: JSON or XML containing all attributes

Balance API

To read the current balance of the FactuurSturen.nl account, you can use this command

Description	Method	URL
Get balance	GET	/api/v1/balance/

Get all profiles

GET /api/v1/balance/

Response

Code: 200

Data: Balance

Country list API

To read all the country id's used for FactuurSturen.nl you can request a list.

Description	Method	URL
Get country list	GET	/api/v1/countrylist/{language}

All attributes used for countries

Attribute	Type	Note
id read only	integer	the unique id of this country
name read only	string	The name of the country in the language you requested

Get country list

GET /api/v1/countrylist/en

Possible languages:

'nl': dutch

'en': english

'de': german

'fr': france

'es': spanish

Response

Code: 200

Data: JSON or XML containing all attributes

Taxes API

Get a list of all taxes defined by the user in his account

Description	Method	URL
Get all defined taxes	GET	/api/v1/taxes

All attributes used for taxes

Attribute	Type	Note
percentage read only	integer	The tax percentage. There is an important difference between 0 and -1: 0 means "0% taxes", -1 means "No taxes"
type read only	string	The tax type. This can be: 'H': High tax rate 'L': Low tax rate 'Z': Zero tax rate 'N': No tax
default read only	boolean	The percentage that is the default, is marked with true

Get all defined taxes

GET /api/v1/taxes

Response

Code: 200

Data: JSON or XML containing all attributes

Search API

Search within FactuurSturen.nl

Description	Method	URL
Execute a search	GET	/api/v1/{area}/{field}/{search_query}

All attributes used for search

Attribute	Type	Note
area	string	Define the search area where we should look. Possible options are: 'invoices' , 'clients' and 'products'
field	string	Define in what specific field you want to search. Use 'all' if you want to search in all fields.
search_query	string	What you want to search. We search in all the fields known in the search area.

Execute a search

Search of client "john" in all fields

GET /api/v1/search/clients/all/john

Search for a client with a specific email address in the email area

GET /api/v1/search/clients/email/pete@acme.com

Search of clients in Amsterdam, but only in the city field

GET /api/v1/search/clients/city/amsterdam

Search of a product with code 20554

GET /api/v1/search/products/code/20554

Response

Code: 200

Data: JSON or XML containing all attributes