

Документация на SOAP API-то на Fakturirane.bg v 1.0 beta

Внимание! Преди да почнете да четете тази документация, е добре да се запознаете със следните понятия, SOAP, XML, също така е добре да сте запознати с езика за програмиране PHP, защото ще бъде позван в примерите.

1. Какво позволява API-то на Fakturirane.bg
2. Използвани библиотеки
3. Аутентизиране
4. Създаване на документ

Какво позволява API-то на Fakturirane.bg

В моята API-то на Fakturirane.bg е в начален етап, и има ограничени възможности. То ви позволява да създадете **проформа фактура** и **фактура** и тяхното изпращане до клиента, през трета страна.

Каква може да е тази трета страна?

Тази трета страна може да е вашият онлайн магазин или сайт, през който Вие вършите всекидневни операции, като обработка на поръчки и др. С тази своя възможност, ние Ви спестяваме едно влизане в Fakturirane.bg и едно преналиване на информацията, която трябва да се съдържа във Вашият документ, с което Вие спестявате време и извършване на една операция по няколко пъти.

API-то ви позволява да работите с Вашите потребители, като ползвате email-ът и паролата им за удостоверяване към API-то, като по този начин, може да вържете няколко човека, които да издават документи през техните акаунти.

Използвани библиотеки

С цел максимално Ви улесняване в разработката, Ви препоръчваме да използвате следните библиотеки.

Библиотека	Описание	URL
NuSoap	Целта на тази библиотека е да улесни връзването и аутентизирането към SOAP сървъра, както и улесняването на аутентизирането и изпращането на заявки към него.	http://sourceforge.net/projects/nusoap/
Array2XML	Целта на тази библиотека е да улесни създаването на XML-ът, съдържащ информация за документа с всичките му атрибути, от хора не дотолкова запознати с генериране на валидни XML документи.	http://www.lalit.org/lab/convert-php-array-to-xml-with-attributes/

Аутентикиране

За аутентикиране се използват същите email и парола, както и в системата на Fakturirane.bg

```
1. require_once 'soap/nusoap.php'; // Зареждаме основната библиотека
2. $client = new nusoap_client('http://fakturirane_bg/api'); // Връзваме се апито
3. $param = new SoapVar(array('username' => 'test@example.com', 'password' => '*****
**'), SOAP_ENC_OBJECT); // Изпращаме нашата логин информация
4. $header = new SoapHeader('http://fakturirane_bg/api', 'AuthenticationInfo', $param, false); // Запазва го в хедъра, който ще бъде изпратен за обработка
5. $client->setHeaders(array($header));
6. // Проверява за грешки при вход
7. $err = $client->getError();
8. if ($err) {
9.     // Ако имаме нека ги видим
10.    throw new Exception($err);
11.    die();
12.    // Нека спрем до тук щом няма успешна връзка
13. }
14.
```

Възможни грешки

Код	Върнатата грешка	Причина
403	Authentication failed!	Грешно изпратени email и парола.
401	User is disabled!	Този акаун не е активен

Създаване на документи

В момента системата поддържа само създаване на фактури и проформа фактури.

След, като се аутентикирате успешно, трябва да извикате функцията, която създава документи, като изпратите XML файл, които описва атрибутите на факурата.

За цел на примера, инфомрацията е показана като масив.

```
1. array(
2.     'company' => array(
3.         'name' => 'Име на контрагента', // Името на контрагента
4.         'bulstat' => '1111111111', // ЕИК на контрагента
5.         'vat_bulstat' => '', // ЕИК по ДДС
6.         'adress' => 'ж.к. Хаджи Димитър бл. 79 вх. Г ап. 71', // Адрес на фирмата
7.         'mol' => 'Милена Миткова', // МОЛ
8.         'phone' => '0898 506133', // Телефон
9.         'email' => 'dia2001@abv.bg', // Email
```

```

10.     ),
11.     'goods' => array( // Описание на редовете
12.         'good' => array(
13.             'title' => 'Абонамент за семинари 01.06.2014 - 31.08.2014', // Име на
стоката
14.             'type' => 0, // Тип на стоката
15.             'price' => '60.0000', // Цена до 4 знака
16.             'quantity' => 1, // Количество
17.             'vat_included' => false, // Цената е с или без ДДС
18.         )
19.     ),
20.     'total_no_vat' => '72', // Цена без ддс
21.     'invoice_tax_amount' => '72', // Данъчна основа
22.     'total' => '72', // Тотал
23.     'vat_rate' => 20, // % ДДС - по подразбиране е 20%
24.     'attend_date' => '20.05.2013',
25.     'return_value' => 'id' // Какво да върне след създаване. Може да е id, number
и both. При both ще се върне във формат id::number
26.     'payment_type' => 1, // Тип на плащане
27.     'paid_date' => '20.05.2013', // Падеж на плащането,
28.     'notes' => '', // Бележка към фактурата
29.     'mail' => array( // Използва се, ако искате фактурата да бъде изпратена по ем
айл
30.         'from' => '', // От кой емайл се изпраща. Ако е оставено празно ще се ипо
злва емайлт на фирмата от която се изпраща, а ако то е празно на логнатия потреби
тел
31.         'to' => 'fbyalkov@bulmar.com',
32.         'subject' => 'Фактура за закупен видео семинар',
33.         // Съобщението на емайлът. Виж таблицата надолу за допълнителна информаци
я за възможните променливи
34.         'message' => 'Здравейте,
35.
36.             Издадена е фактура с № {{invoice_number}} за закупени видео семинари.
37.
38.             За въпроси може да пишете на имейл support@fakturirane.bg.
39.
40.             Благодарим Ви за интереса към нашите семинари!
41.     Accounting Seminars',
42.         'attach_document' => true,
43.         'header' => false,
44.     )
45. );
46.

```

Внимание! Датата на фактурата, на този етап се задава автоматичо от апито (текущата дата.)

От следващата версия типът на информацията за създадения документ ще се премести, като параметър на функцията за създаване. В момента се праща като параметър на XML-ът - `return_value`

Променлива	Описание
<code>{{invoice_number}}</code>	Ще бъде заместен с номер на документа
<code>{{invoice_id}}</code>	Ще бъде заместен с служебният номер на документа
<code>{{invoice_date}}</code>	Ще бъде заместен с датата на документа
<code>{{customer_name}}</code>	Ще бъде заместен с името на клиента, на който се издава документа
<code>{{customer_eik}}</code>	Ще бъде заместен с ЕИК номера, на който се издава документа
<code>{{total}}</code>	Ще бъде заместен с тотала на документа
<code>{{total_no_vat}}</code>	Ще бъде заместен с тотала без ддс
<code>{{payment_type}}</code>	Ще бъде заместен с метода на плащане

Извикване на функцията за създаване на документ

```
1. $xmlResult = Array2XML::createXML('document', $xml); // Създава xml от горният ма
   сив
2. $response = $client->call('api.create_document', array('xml' => $xmlResult->saveX
   ML(), 'type' => 'invoice', 'eik' => '12321312312')); // Объръща се към апито
3.
```

Функцията приема 3 параметъра.

1. `xml` : XML файлът в който е описана фактурата
2. `type` : типът на създавания документ. В момента се поддържа **invoice** и **proforma**
3. `eik` : ЕИК номерът на фирмата от която се издава фактурата

Възможни грешки

Грешка	Описание
ERROR : Cannot create company!	Не може да създаде клиента. Възможна причина - не са изпратени всички полета

Грешка	Описание
ERROR : Cannot create customer office	Не може да създаде офиса на клиента. Възможна причина - не са изпратени всички полета
ERROR : Cannot find bank!	Не може да открие банката. Номера на банковата сметка е грешен
ERROR : Cannot create document rows!	Не може да запази атрибутите на документа. Не са изпратени всички полета
ERROR : Cannot create document!	Не може да запази документа. Не са избратени всички полета
ERROR : Invalid Payment Type!	Невалиден тип на плащане.

Номенклатури с типове

Тип на стоката

Тип	Описание
0	Стока
1	Услуга
2	Друго

Тип на плащането

Тип	Описание
1	В брой
2	Банков превод
3	Бартер / Насрещно прихващане
4	Наложен платеж
5	Плащане карта
6	Пощенски паричен превод
7	Виртуална валута
8	Насрещно прихващане

Когато подавате тип на плащане "Банков превод" не трябва да подавате стойност за [bank]