

API Methods - FGO integration

v.3.2

Contents

OLDER VERSIONS	2
INTRODUCTION.....	2
REQUESTS.....	4
KEY VALUE LISTS.....	4
INVOICE	4
ISSUE	4
PRINT.....	7
DELETE.....	7
CANCEL.....	7
GETSTATUS.....	8
ADD RECEIPT / CASHING	8
DELETE RECEIPT.....	9
REVERSING INVOICE.....	9
ITEMS (PRODUCTS/SERVICES).....	10
LIST	10
CLIENTS	11
LIST	11
HOW TO CALCULATE HASH?	11

OLDER VERSIONS

Version	Details	Data
3.2	Document issue: - Added ContBancar field (Bank Account) – you can save the bank account for each client; AWB: - Added AWB method – saves an AWB number for an invoice in order to simplify identification and processing couriers' slips.	07.08.2019
3.1	Document issue: - Added the possibility of creating an order (TipFactura "Comanda"); these will be saved in "Comenzi" (orders) page, in your FGO account.	04.06.2019
3.0	Invoice issue: Added IdExtern parameter – order ID (from your online store) for which you issue the invoice;	23.05.2019
2.9	Added GetStatus method – returns the total amount of an invoice and the paid amount	22.05.2019
2.8	- Client[Tip] – required field; - Automatically mark client as VAT payer (if CodUnic contains RO or other initials of other countries); - client method - list - displays existing clients in FGO;	03.04.2019
2.7	Added the possibility to issue invoice with TipFactura "Scutit cu deducere" – S (exempt with deduction)	11.02.2019
2.6	Data type for Continut[i][NrProduce] changed from int to decimal	01.02.2019
2.5	- Added parameter DataScadenta for invoice issue; - Bug fixes;	09.01.2019
2.4	- Added method StergereIncasare for receipt deletion; - Added method /articol/list for items list;	19.12.2018

INTRODUCTION

Before API requests, you must follow the steps below:

1. Create FGO account:

- Test environment: <http://testapp.fgo.ro/inregistrare>
- Production environment: <https://www.fgo.ro/inregistrare>

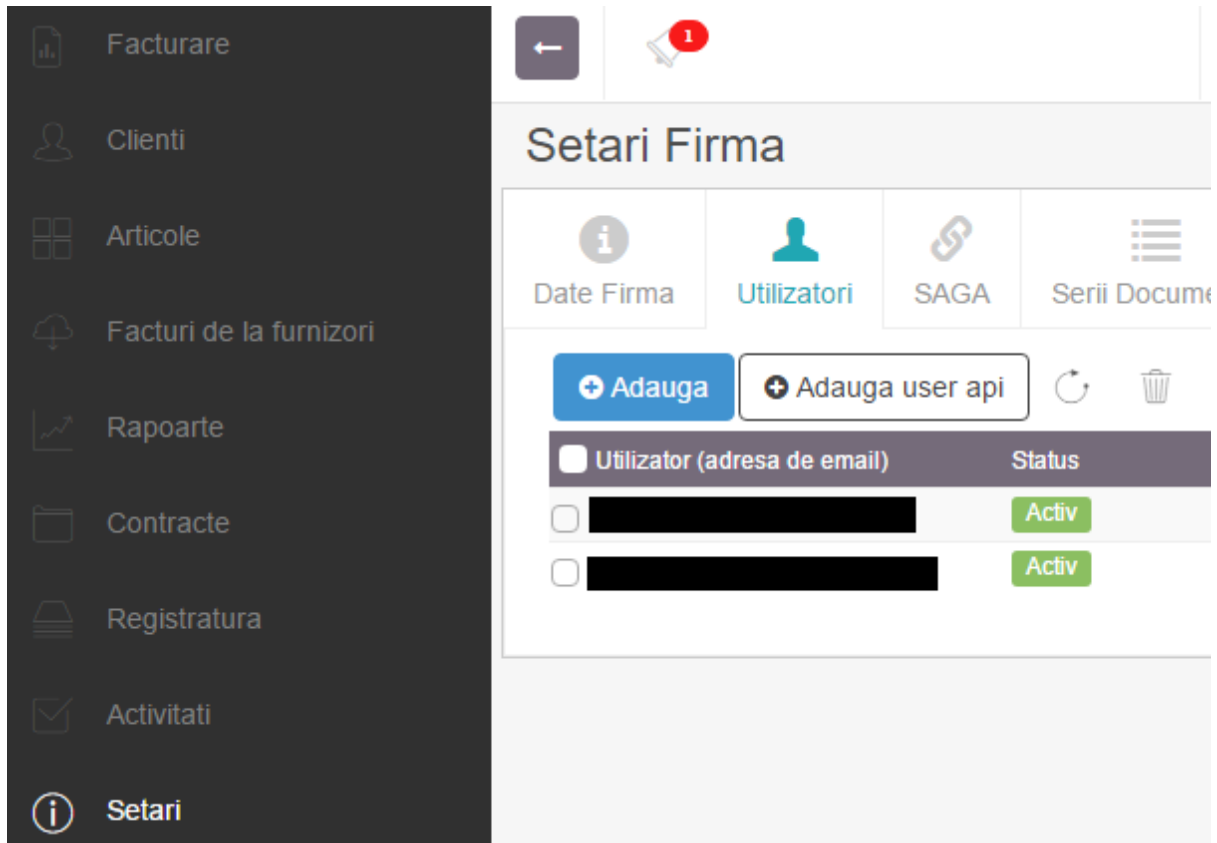
Notes:

- The two environments are independent and do not synchronize with each other. Therefore, you will have to create an account for each environment in order to switch from testing to production.
- For the TEST environment, you do NOT need a paid subscription (START, PRO or PREMIUM), as it works with FGO Free. For production environment, you need Pro or Premium subscription.

2. Add API user:

Login to your FGO account and access Setari – Utilizatori (Settings – Users) from the side menu. Press „Aduaga user API” (Add API user) to activate API connection. Then press again to see the password (**Cheie Privata/Private key**).

For PRODUCTION environment, you need PRO or PREMIUM subscription.



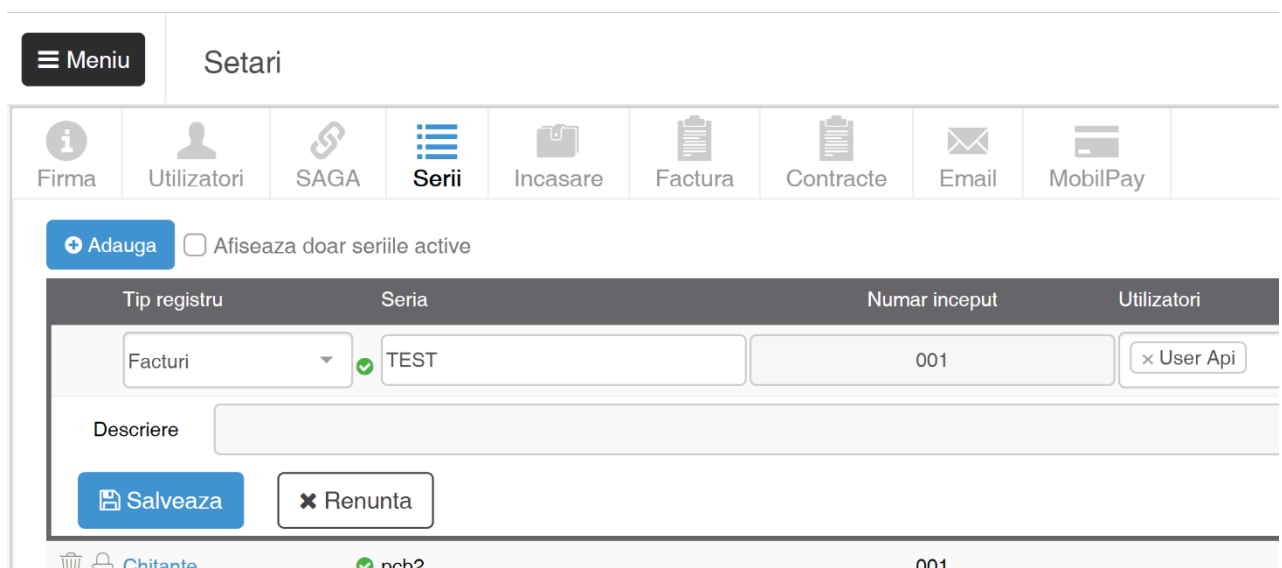
3. Define series:

In your FGO account, access Setari – Serii (Settings – Series) and define series for:

- Facturi (invoices)
- Proforme (orders)
- Chitante (receipts)

Give user API access to the series by selecting it in the „Utilizatori” field.

You can use these series only for user API or you can use series that are also accessed by other users.



REQUESTS

Root links:

- For testing: <http://testapp.fgo.ro/publicws/>
- For production: <https://api.fgo.ro/v1>

You can access test forms at this address: <http://testapp.fgo.ro/publicws/testing.html>

Available API methods:

ROOT	Controller	Method
http://testapp.fgo.ro/publicws/ or https://api.fgo.ro/v1/	nomenclator/	tara (country)
	nomenclator/	judet (county)
	nomenclator/	tva (vat)
	nomenclator/	banca (bank)
	nomenclator/	tipincasare (cashing type)
	nomenclator/	tipfactura (invoice type)
	nomenclator/	tipclient (client type)
	factura/	emitere (issue)
	factura/	print (print)
	factura/	stergere (delete)
	factura/	anulare (cancel)
	factura/	incasare (cashing)

KEY VALUE LISTS

You can access key value lists lists through GET.

Example: <http://testapp.fgo.ro/publicws/nomenclator/tipfactura>

The answer will be a JSON similar to the one below:

```
{ "Success": true, "List": [{"Nume": "Valoare"}] }
```

*Nume = Name; Valoare = Value

INVOICE

ISSUE

Request

Link: ROOT/factura/emitere			
Parameter	Type	Required	Notes
CodUnic (Unique code)	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH”

Text	string(2000)		You can add information that will appear on the invoice (example: delegate's name)
Explicatii	string(2000)		Additional information that will appear on the invoice
Valuta (Currency)	string(3)	x	See „Valuta” (currency) key value list
TipFactura	string(50)	x	See „TipFactura” (invoice type) key value list
DataEmitere (issue date)	datetime		If you don't send any value, the default date will be the date and time of the request Format: yyyy-mm-dd
DataScadenta (due date)	datetime		Optional parameter; If you don't send any value, the default number of days will be the one set in your FGO account (Setari - Factura - Setari facturare -> Zile scadenta Settings – Invoice – Billing settings -> Days to maturity)
Numar (number)	string(50)		If you don't send any value, FGO will automatically generate the next available invoice number, according to your settings
Serie (series)	string(50)		If you don't send any value, FGO will use the default series according to your settings
TvaIncasare (VAT on collection)	bool		Accepts true or false
VerificareDuplicat (duplicate verification)	bool		Accepts true or false. If you don't send any value, the default parameter is considered true . Send false only if you DO NOT want to check the existence of the invoice in the system (there is a risk of issuing two invoices containing the same items)
Client[Denumire] (Client Name)	string(255)	x	Client's name
Client[CodUnic] (Client Unique code)	string(128)		Client's Unique Identification Code If your client is a person, not a company, and you don't have this code or is incorrect, make sure the address is completed following the format: Loc. [city name], Str. [street name], Nr. [street number] and, if necessary: Bl. [building], Sc. [staircase number] Et. [floor], Ap. [apartment]
Client[NrRegCom] (Client Trade Register Number)	string(128)		Client's trade register number (Registrul Comertului)
Client[Email]	string(100)		Client's email address
Client[Telefon]	string(100)		Client's phone number
Client[Tara] (country)	string(128)		See „Tara” (Country) key value list
Client[Judet] (county)	string(100)		See „Judet” (County) key value list (validation is made only when Client[Tara] contains ROMANIA or is empty)
Client[Adresa] (address)	string(500)		Make sure the address is completed following the format: Loc. [city name], Str. [street name], Nr. [street number] and, if necessary: Bl. [building], Sc. [staircase number] Et. [floor], Ap. [apartment]
Client[Tip] (type)	String(2)	x	See „TipClient” (client type) key value list (company/individual)
Client[IdExtern]	int		It allows sending an ID from a third party application, in order to correctly identify and save a client (useful especially for individual clients for whom you do not have the unique code)

Client[Strain] (foreign = not romanian)	bool		If the value is true , you can send any VAT rate for Continut[i][CotaTVA]
Client[ContBancar] (bank account)	string(100)		Bank Account
Continut[i][Denumire] (content/item name)	string(250)	x	Item's name (product/service). An article with a different name is a new article and will automatically receive a new identifier
Continut[i][CodArticol] (content item code)	string(128)		<p>1. item's code is generated in SAGA software, not FGO, at the time the item is imported in SAGA (the code thus generated will not return automatically in FGO, it must be specifically imported).</p> <p>2. In FGO, items' codes are displayed only for the list of items that were sent from SAGA to FGO (this a 2 steps user-explicit process – read no.6 below)</p> <p>3. If CodArticol (item's code) contains a value, the parameter Continut[i][Denumire] will be ignored</p> <p>5. There is no update on the items' name that will propagate between FGO and SAGA or between your API-platform and FGO.</p> <p>6. In order to obtain SAGA items' codes in FGO, follow these two steps: i. export data from SAGA software ii. import data in FGO (Setari - SAGA -> Incarca datele venite din SAGA. Settings – SAGA -> Upload data from SAGA)</p>
Continut[i][Descriere] (content item description)	string(1000)		Item's description
Continut[i][PretUnitar] (unit price)	decimal		<p>Calculation direction: normal (from unit price to total price) Unit price format: XXXX.XX Important: for normal calculation direction, don't send a value [PretTotal] (total price), as total price will be automatically calculated.</p>
Continut[i][PretTotal] (total price)	decimal		<p>Calculation direction: reverse (from total price to unit price) Unit price format: XXXX.XX Important: for reverse direction of calculation, don't send a value for [PretUnitar] (unit price), as unit price will be automatically calculated.</p>
Continut[i][UM]	string(50)	x	Measure unit
Continut[i][NrProduce] (quantity)	decimal	x	Item's quantity format: XXXX.XXX
Continut[i][CotaTVA] (vat rate)	decimal	x	See TVA (VAT) key value list

Response

```
{ Success = true, Message = "", Factura = { Numar = "001", Serie = "BV", Link = "" } }
```

or

{ Success = false, Message = "Mesaj eroare" }

PRINT

Request

Link: ROOT/factura/print			
Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
Numar	string(50)	x	
Serie	string(50)	x	

Response

{ Success = true, Message = "" , Factura = { Numar = "001", Serie = "BV", Link = "" } }

or

{ Success = false, Message = "Mesaj eroare" }

DELETE

Request

Link: ROOT/factura/stergere			
Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
Numar	string(50)	x	
Serie	string(50)	x	

Response

{ Success = true, Message = " Factura a fost stearsa." }

**Message="The invoice has been deleted"*

or

{ Success = false, Message = "Mesaj eroare" }

**Message="Error"*

CANCEL

Request

Link: ROOT/factura/anulare

Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
Numar	string(50)	x	
Serie	string(50)	x	

Response

```
{ Success = true, Message = "Factura a fost anulata." }
```

```
*Message="The invoice has been canceled"
```

or

```
{ Success = false, Message = "Mesaj eroare" }
```

```
*Message="Error"
```

GETSTATUS

returns the total amount of an invoice and the paid amount

Request

<u>Link:</u> ROOT/factura/getstatus			
Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
Numar	string(50)	x	
Serie	string(50)	x	

Response

```
{ "Success": true, "Factura": { "Numar": "1", "Serie": "X", "Valoare": "167.69", "ValoareAchitata": "167.69" } }
```

```
*"Success"=true,"Invoice": { „Number”:"1", „Series”:"X", „Amount”:"167.69", „PaidAmount”:"167.69" }
```

or

```
{ Success = false, Message = "Mesaj eroare" }
```

```
*Message="Error"
```

ADD RECEIPT / CASHING

Request

<u>Link:</u> ROOT/factura/incasare			
Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
NumarFactura (invoice number)	string(50)	x	Invoice number for which you add cashing/receipt

SerieFactura (invoice series)	string(50)	x	Invoice series for which you add cashing/receipt
TipIncasare (cashing type)	string(50)	x	See TipIncasare (cashing type) key value list
SumaIncasata (ammount cashed in)	decimal	x	Ammount cashed in format: XXXX.XX
DataIncasare (cashing date)	datetime	x	Cashing date in format: yyyy-mm-dd hh:mm:ss

Response

```
{ Success = true, Message = "", Incasare = { Numar = "001", Serie = "BV" } }
```

```
*Cashing= { Number ="001", Series = „BV” }
```

or

```
{ Success = false, Message = "Mesaj eroare" }
```

```
*Message="Error"
```

DELETE RECEIPT

Request

<u>Link</u> : ROOT/factura/stergereincasare			
Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
NumarFactura (invoice number)	string(50)	x	Invoice number for which you add cashing/receipt
SerieFactura (invoice series)	string(50)	x	Invoice series for which you add cashing/receipt
SerieIncasare (cashing series)	string(50)	x	Cashing series of the receipt you want to delete
NumarIncasare (cashing number)	string(50)	x	Cashing number of the receipt you want to delete

Response

```
{ Success = true, Message = "Incasarea a fost stearsa" }
```

```
*Message="Cashing has been deleted"
```

or

```
{ Success = false, Message = "Mesaj eroare" }
```

```
*Message="Error"
```

REVERSING INVOICE

Request

Link: ROOT/factura/stornare			
Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
Numar (number)	string(50)	x	
Serie (series)	string(50)	x	

Response

```
{ Success = true, Message = "", Factura = { Numar = "001", Serie = "BV", Link = "" } }
```

```
*Invoice= { Number = „001”, Series = „BV”, Link = „” }
```

or

```
{ Success = false, Message = "Mesaj eroare" }
```

```
*Message="Error"
```

AWB

Request

Link: ROOT/factura/awb			
Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
Numar (number)	string(50)	x	
Serie (series)	string(50)	x	
AWB	string(128)	x	AWB number

Response

```
{ Success = true }
```

or

```
{ Success = false, Message = "Mesaj eroare" }
```

```
*Message="Error"
```

ITEMS (PRODUCTS/SERVICES)

LIST

Request

Link: ROOT/articol/list

Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
NrPagina (number of pages)	int	x	For the paging calculation, consider the parameter „Total” you receive in response.
NrArticole (number of items)	int	x	Represents the number of items displayed on request. The default value is 10. For the paging calculation, will be considered the parameter „Total” received in response.

Response

```
{ "Result": { "Total": 1335, "NrPagina": 1, "NrArticole": 1, "List": [ { "Nume": " storno partial factura 036 servicii juridice", "PretUnitar": 0, "UM": "", "CodConta": null, "Stoc": -4 } ] }, "Success": true, "Message": null }
```

or

```
{ Success = false, Message = "Mesaj eroare" }
```

CLIENTS

LIST

Request

Link: ROOT/client/list			
Parameter	Type	Required	Notes
CodUnic	string	x	Your company's unique identification code
Hash	string	x	Scroll down and read section „How to calculate HASH” (section b)
NrPagina	int	x	For the paging calculation, consider the parameter „Total” you receive in response.
NrArticole	int	x	Represents the number of items displayed on request. The default value is 10. For the paging calculation, will be considered the parameter „Total” received in response.

Response

```
{ "Result": { "Total": 1335, "NrPagina": 1, "NrClienti": 1, "List": [ { "Nume": "Client 1", "CodUnic": "123456", "NrRegCom": "J11/1111/2013", "Email": "", "Telefon": "", "Tara": "RO", "Judet": "CJ", "Adresa": "STR.Principala ,CLUJ-NAPOCA", "Banca": "", "ContBanca": "", "IdExtern": null } ] }, "Success": true, "Message": null }
```

or

```
{ Success = false, Message = "Mesaj eroare" }
```

HOW TO CALCULATE HASH?

For any request, there will be built a [SHA-1 Uppercase](#) HASH, formed of:

- Invoice issue

Hash=Cod Unic Furnizor (Your Unique Code)+Cheie Privata (Private Key)+Denumire Client (Client's name)

Ex:

- Cod Unic Furnizor: 2864518,
- Cheie Privata: 1234567890
- Denumire Client: Ionescu Popescu

Encryption is done -> SHA-1(28645181234567890Ionescu Popescu)

Hash=8C3A7726804C121C6933F7D68494B439463996E2

- b. Anulare (cancel), Stornare (reverse), Print (print), Incasare (cashing) or StergereIncasare (cashing deletion), instead of client's name you will send **invoice number**, exactly as returned by the issuing method (Ex. "001", "123") without series.

Ex:

- Cod Unic Furnizor (Your Unique Code): 2864518,
- Cheie Privata (Private Key): 1234567890
- Numar Factura (Invoice Number): 123

Encryption is done -> SHA-1(28645181234567890**123**)

- c. Items

Ex:

- Cod Unic Furnizor: **2864518**,
- Cheie Privata: 1234567890

Encryption is done -> SHA-1(**2864518**1234567890)