

# **A Számla Agent**

**php osztály használati dokumentációja**

# Tartalomjegyzék

1.0. Bevezető .....	2. oldal
2.0. Alapok .....	2. oldal
2.1. Karakterkódolás .....	2. oldal
2.2. Inicializálás .....	2. oldal
2.3. Egyedi konfiguráció .....	2. oldal
2.4. Alapbeállítások .....	3. oldal
3.0. Számlagenerálás.....	5. oldal
3.1. Többszintű xml adatok átadása.....	7. oldal
4.0. Sztornószámla készítése .....	7. oldal
5.0. Jóváírás rögzítése .....	7. oldal
6.0. Számla pdf lekérése.....	8. oldal

## Függelék:

F.1.0. A <code>_generateinvoice()</code> funkció elvárt, és opcionális bemenetei .....	9. oldal
F.2.0. A <code>_reverseInvoice()</code> funkció elvárt, és opcionális bemenetei .....	13. oldal
F.3.0. A <code>_payInvoice()</code> funkció elvárt, és opcionális bemenetei .....	14. oldal
F.4.0. A <code>_requestInvoicePDF()</code> funkció elvárt, és opcionális bemenetei .....	15. oldal

## 1.0. Bevezető

A php class feladata a Számla Agent rendszerével történő adatkommunikáció egyszerűbbé tétele a végfelhasználók számára.

Nem szükséges a felhasználó számára az xml generálási metódusok, és a php alapú távoli adatkapcsolati funkciók használatának ismerete, php osztályunk elvégzi ezt a munkát egy egyszerű eljáráshívási metodika mögé bújtatva.

## 2.0. Alapok

A class meghívása egy php kódhalmazon belül csupán egy alkalommal szükséges, abszolút, vagy akár relatív elérési útvonal megadásával, a felhasználó igénye szerint.

```
require_once("/a/fajl/eleresi/utvonala/invoiceAgent.class.php");
```

### 2.1. Karakterkódolás

A class használatához magyar nyelvterületen optimálisan használandó karakterkódolás az utf-8, ám természetesen a felhasználási körülmények függvényében szabadon változtatható.

```
header("charset=utf-8");
```

### 2.2. Inicializálás

Az inicializálás metódusa:

```
$szamlazz = new invoiceAgent();
```

A fentebbi metódus alkalmazásával a számlaAgent kommunikációját biztosító php osztály az invoiceAgent.ini néven tárolt konfigurációs fájlban elhelyezett alapbeállításokkal töltődik be.

### 2.3. Egyedi konfiguráció

Igény esetén az invoiceAgent egyedi konfigurációs fájl megadásával is inicializálható, mely megoldás metódusa az alábbi:

```
$szamlazz = new invoiceAgent("/fajl/eleresi/utvonala/egyedikonfig.ini");
```

Az Agent konfigurációs dokumentuma standard ini formátum szerint történt felépítésre, ennek irányelveit követve módosítható, bővíthető.

A [defaults] szekció az Agent működéséhez szükséges alapbeállításokat tartalmazza, nem elhagyható, ám tartalma szabadon módosítható. Az ini beállítófájlok beolvasása többdimenziós tömbként történik. Az egyes szekciók a \$cfg tömb alegelemeiként kerülnek tárolásra az alábbiak szerint:

```
$cfg['defaults']
$cfg['defaults']['log_level']
$cfg['defaults']['log_file']
...
$cfg['beallitasok']
    $cfg['beallitasok']['username']
    $cfg['beallitasok']['password']
...
```

A kötelező „default” beállítások mellett a végfelhasználónak lehetősége van egyéb beállítási tömböket definiálni, mely tömbökben beállított értékek az xml kérés fájlok generálásakor alapértelmezésként kerülnek beállításra (ha az eljáráshívás nem bírálja felül azt), így nincs szükség például a bejelentkezési adatok állandó megadására, a konfigurációs fájlban alapértelmezettként eltárolhatjuk azokat.

## 2.4. Alapbeállítások

A „defaults” konfigurációs tömb elemei és bemutatásuk:

- **log\_level** - az alkalmazott logolás „szintjét” határozza meg. 0, azaz nullaként megadott érték inaktívvá teszi a logot, 1-es érték esetén csakis error típusú eseményt rögzít (illetve igény szerint emailt küld erről, lásd. lentebb), 2-es érték esetén rögzítésre kerülnek a figyelmeztetések is, azon kivételekről melyek nem befolyásolják feltétlenül az üzemszerű működést. A 3-as érték a debug mód jelzésére szolgál, ezesetben a részfunkciók üzemszerű lefutásáról is folyamatos tájékoztatást kapunk.
- **log\_file** - meghatározhatjuk, hogy mely fájlban tárolódjanak a logolt információk. Alapértelmezés szerint ez a fájl az Agent osztály mappájában elhelyezett invoiceagent.log állomány, mely elérési útvonala szabadon módosítható. A hivatkozott sor „kikommentezése” esetén a rendszerszintű log kerül használatra, és elérhetővé válik a futtató szerver logrotálási megoldása is.
- **log\_email** - amennyiben kitöltjük, a honlap adminisztrátora e-mailt kap az error szintű eseményekről az itt megadott e-mail címére. Kikommentezés esetén a funkció inaktív.

- **return\_pdf\_as\_result** - az egyes eljáráshívásokat követően elvárt válaszunkat adhatjuk meg. „true” érték esetén az eljáráshívások válaszául (ahol alkalmazható) pdf dokumentumot kapunk vissza, illetve az aktuálisan generált számla számát.
- **return\_invoice\_number\_as\_result** - a kérések végrehajtása után elvárt Agent válasz az aktuálisan generált számla sorszáma.

*Amennyiben a return\_pdf\_as\_result és return\_invoice\_number\_as\_result beállítások értéke egyaránt false, az eljáráshívások egységesen a hívás sikerességének függvényében true vagy false logikai változót adnak vissza. (Kivéve a célzott pdf számla lekérés esetében, ekkor természetesen pdf dokumentum érkezik válaszként. Lásd. később)*

- **pdf\_file\_save\_path** - a válaszként generált pdf dokumentumok helyi mentési útvonalát adhatjuk meg az alkalmazásnak. Az elmentett pdf fájl minden esetben a generált számla sorszáma szerint kerül elnevezésre. A sor kikommentezése esetén a válaszként kapott pdf dokumentumok nem kerülnek automatikusan tárolásra.
- **xml\_file\_save\_path** - a távoli eljáráshívásokhoz generált xml fájlok is mentésre kerülhetnek a későbbi visszaellenőrizhetőség, hibakeresés megkönnyítésének érdekében. Útvonaluk az előző paraméterhez hasonlóan megadható, a sor kikommentezésével a szolgáltatás deaktiválható.
- **cookiejar** - biztonsági, és erőforrás takarékosági szempontból egyaránt fontos, hogy egy-egy eljáráshívás során a kliens egy felhasználónak látszódjon a számlaAgent távoli rendszere felől. A hívások során kapott session id az itt meghatározott rejtett fájlban kerül tárolásra, ahonnan a későbbiekben bármikor visszaolvasható.
- **agent\_url** - az eljáráshívások során használatos távoli elérési útvonal, módosítása nem szükséges, nem indokolt.
- **root\_cert\_file** - egyes szerver konfigurációk nem rendelkeznek a távoli https alapú adatkapcsolatok létesítéséhez, és a biztonsági ellenőrzéshez szükséges tanúsítvány csomaggal. Ezen problémát kiküszöbölendő szükséges egy root certificate egyedi megadása. (Lásd. részletesebben a következő pontban)
- **call\_method** - az Agent API elsődlegesen a legmodernebb curl metódus használatával kommunikál. Bizonyos szerver konfigurációkon előfordul, hogy a curl telepítés hiányos, nem kerültek településre a szükséges tanúsítvány csomagok, így ezen esetekben az eljáráshívás bármely visszajelzéstől mentes hibába ütközhet. Ezt kiküszöbölendő integráltuk a fentebbi root certificate fájl automatikus importálását, ám bizonyos esetekben

ez a megoldás is problémába ütközhet, például rendszerszintű tiltás esetén. Ilyen esetben az adatkommunikáció egy régebbi, `file_get_contents` típusú híváson bonyolítható le.

**A call method változó lehetséges értékei: auto, normal, legacy.**

- **auto** beállítás esetén az osztály először curl hívás megkísérlésével próbálkozik, mely sikertelensége esetén az eredményt az adott munkafolyamatra eltárolva visszavált a `file_get_contents` lehetőségére.
- **normal** beállítás esetén a rendszer kizárólagosan curl metódus alkalmazásával kommunikál, hiba esetén nem próbálkozik mással, false értéket ad vissza.
- A **legacy** beállítás az esetben alkalmazandó, amikor tudjuk, hogy a curl lekérések hibába fognak ütközni, és nem várható ezen probléma megoldása. Ezen esetben a php class csakis `file_get_contents` hívásokkal kommunikál a távoli szerverrel.

***Vigyázat!** Számos informatikai fórumon tévesen azt állítják, hogy a hibába ütköző curl eljárás-hívások kijavíthatóak a távoli szerver ssl hitelesítésének deaktiválásával. Ez a megoldási javaslat semmilyen esetben sem követendő, hiszen biztonsági kockázatot hordoz magában. Az ilyen módon folytatott kommunikációért, és a kommunikáció során keletkezett esetleges károkért a számlaAgent fejlesztői és üzemeltetői nem vonhatóak felelősségre.*

### 3.0. Számlagenerálás

Számlák generálása során a php osztály `_generateInvoice()` funkciója használatos.

A funkció bemeneti értékeit tömbökben érdemes eltárolni az alábbiak mintájára.

(részletesebb példáért lásd. `teszt.php`)

```
$szamlazz = new invoiceAgent();
$settings = [
    "username" => "monostoria@gmail.com",
    "password" => "dr0g0n988"
    ...
];
$header = [ ... ];
$seller = [ ... ];
$buyer = [ ... ];
$waybill = [ ... ];
```

A számlán rögzítendő tételek az addItem() funkció használatával készíthetők elő:

```
$szamlazz->addItem([
    "name" => "Eladó izé",
    "quantity" => "1.0"
    ...
]);
$szamlazz->addItem([
    "name" => "Eladó izé 2"
    ...
]);
```

Az eljáráshívást a php osztály fentebb hivatkozott funkciójának meghívásával indíthatjuk el (fontos, hogy az adatokat tartalmazó tömböket helyes sorrendben adjuk meg), majd a szokásos módon kezelhetjük az eredményt a kapott adatok függvényében.

Példánkban logikai változót kapunk vissza, eszerint jelezzük a felhasználó számára a művelet sikerességét, vagy esetlegesen annak sikertelenségét:

```
if ($szamlazz->_generateInvoice($settings,$header,$seller,$buyer,$waybill))
{
    echo "sikerült";
} else {
    echo "nem sikerült";
}
unset($szamlazz);
```

Az elvégzett műveletek végén érdemes a létrehozott Agent object-et „elpusztítani” a szerver memória erőforrásai megtakarításának indokán.

Amennyiben Agentünket úgy paramétereztük (return\_invoice\_number\_as\_result = true), hogy válaszul a generált számlák számát adja vissza, sikeres eljáráshívás esetén a számlaszámot kapjuk vissza string formátumban, sikertelenség esetén pedig egy false logikai változót.

return\_pdf\_as\_result = true paraméter beállításakor, és siker esetén tömböt kapunk válaszul, melynek „invoice\_number” eleme a generált számla számát, a „pdf” elem pedig a pdf fájl kimenetét tartalmazza majd.

A kérés legenerálásának folyamatában a megadott tömb elemek a számlaAgent elvart sémájának megfelelően xml dokumentummá konvertálódnak. A kötelezőként elvart elemek, illetve a class közbenjárásával folytatott kommunikáció folyamán használt elnevezésük megismerését a jelen dokumentum utolsó oldalain található függelék hivatott támogatni.

### 3.1. Többszintű xml adatok átadása

Bizonyos esetekben szükségünk lehet többszintű xml elemek generálására, például főkönyvi tételek rögzítésekor, vagy a futárszolgálati adatok átadásához.

Ezesetben a kérést többdimenziós tömbbel szükséges kiegészíteni, az xml „gyermek” generálását ekkor szintúgy a php osztály végzi el:

```
$szamlazz->addItem([  
    ...  
    "product_accounting" => [  
        "transaction_event" => "asdf",  
        "income_account_no" => "654321"  
    ]  
]);
```

### 4.0. Sztornószámla készítése

Számla sztornózásakor a fentiek mintájára a `_reverseInvoice()` funkció használatos.

Ennek elvárt bemenetei a settings, header, seller, és buyer tömbök. A kötelezően elvárt, illetve opcionálisan megadható adatok megismeréséhez lásd. függelékünket!

Az eljáráshívásra a `teszt2.php` fájl tartalmaz részletes példát.

### 5.0. Jóváírás rögzítése

Egy-egy számla befizettként történő megjelöléséhez, vagy részletfizetések hozzáadásához a `_payInvoice()` funkció alkalmazandó.

Az `addPayment` funció alkalmazásával készíthető elő egy-egy fizetési tranzakció:

```
$szamlazz->addPayment([  
    "date" => "2015-12-13",  
    "transaction_title" => "készpénz",  
    "amount" => "999",  
]);
```

A `_payInvoice()` funkció végrehajtáskori egyetlen elvárt bemenete a settings tömb, és annak tartalmi. Gyakorlati példaként lásd. a `teszt3.php` fájlt.



## 6.0. Számla pdf lekérése

Egy már legenerált számla pdf végállománya a `_requestInvoicePDF` funkció igénybevételevel kérhető le újra.

A funkció kizárólagosan elvárt bemeneti a kommunikáció lebonyolításához szükséges hitelesítő adatok, illetve az újragenerálni kívánt számla sorszáma.

```
$scope = [  
    "username" => "monostoria@gmail.com",  
    "password" => "dr0g0n988",  
    "invoice_num" => "E-MEGFI-2015-265",  
];  
$valasz = $szamlazz->_requestInvoicePDF($scope);
```

Az eljárás hívás minden esetben - bármely egyéb beállítástól függetlenül - pdf dokumentumot ad vissza válaszként. Példafájlként lásd. a `teszt4.php` állományt.

# Függelék

## F.1.0. A \_generateinvoice() funkció elvárt, és opcionális bemenetei

xml tag elnevezés      php class azonosító      adattípus      kötelező-e?      alapérték

### Szülőelem

### beallitasok

felhasznalo	username	string	kötelező	
jelszo	password	string	kötelező	
apiKey	api_key	string		
eszamla	e_invoice	logikai változó	kötelező	true
kulcstartojelszo	keychain	string		
szamlaLetoltes	download_invoice	logikai változó	kötelező	false
szamlaLetoltesPld	download_count	szám		
valaszVerzio	response_type	szám	kötelező	1
aggregator	aggregator	string		

### Szülőelem

### fejlec

keltDatum	invoice_date	dátum	kötelező	
teljesitesDatum	fulfillment	dátum	kötelező	
fizetesiHataridoDatum	payment_due	dátum	kötelező	
fizmod	payment_method	string	kötelező	
penznem	currency	string	kötelező	Ft
szamlaNyelve	language	string	kötelező	hu
megjegyzes	comment	string		
arfolyamBank	exchange_bank	string		
arfolyam	exchange_rate	szám		
rendelesSzam	order_no	string		
elolegszamla	is_deposit	logikai változó		
vegszamla	is_final	logikai változó		
helyesbitoszamla	is_corrective	logikai változó		
helyesbitettSzamlaszam	corrected_num	string		
dijbekero	is_proform	logikai változó		
logoExtra	logo_extra	string		
szamlaszamElotag	num_prefix	string		
fizetendoKorrekcio	correction_to_pay	szám		
fizetve	is_paid	logikai változó		
arresAfa	profit_vat	logikai változó		

## Szülőelem

## elado

bank  
bankszamlaszam  
emailReplyto  
emailTargy  
emailSzoveg  
alairoNeve

bank  
bank\_account  
email\_replyto  
email\_subject  
email\_content  
signatory

string  
string  
string  
string  
string  
string

## Szülőelem

## vevo

nev  
ország  
irsz  
telepules  
cim  
email  
sendEmail  
adoszam  
adoszamEU  
postazasiNev  
postazasiOrszag  
postazasiIrsz  
postazasiTelepules  
postazasiCim  
vevoFokonyv  
azonosito  
alairoNeve  
telefonszam  
megjegyzes

name  
country  
zip  
city  
address  
email  
send\_email  
tax\_no  
eu\_tax\_no  
postal\_name  
postal\_country  
postal\_zip  
postal\_city  
postal\_address  
buyer\_account  
buyer\_id  
signatory  
phone  
comment

string  
string  
string  
string  
string  
string  
logikai változó  
string  
string  
string  
string  
string  
string  
string  
string  
string  
string  
string

kötelező  
  
kötelező  
kötelező  
kötelező

## Szülőelem

## vevo::vevoFokonyv

konyvelesDatum  
vevoAzonosito  
vevoFokonyviSzam

account\_date  
buyer\_id  
account\_id

dátum  
string  
string

### Szülőelem

### fuvarlevel

uticel  
futarSzolgalat  
vonalkod  
megjegyzes  
tof  
ppp  
sprinter

destination  
parcel  
barcode  
comment  
trans-o-flex  
pick-pack-point  
sprinter

string  
string  
string  
string

### Szülőelem

### fuvarlevel::tof

azonosito  
shipmentID  
csomagszam  
countryCode  
zip  
service

numeric\_id  
shipment\_id  
packet\_no  
country\_code  
zip  
service

string  
string  
szám  
string  
string  
string

### Szülőelem

### fuvarlevel::ppp

vonalkodPrefix  
vonalkodPostfix

barcode\_prefix  
barcode\_postfix

string  
string

### Szülőelem

### fuvarlevel::sprinter

azonosito  
feladokod  
iranykod  
csomagszam  
vonalkodPostfix  
szallitasildo

numeric\_id  
sender\_id  
shipment\_zip  
packet\_no  
barcode\_postfix  
shipping\_time

string  
string  
string  
szám  
string  
string

## Szülőelem

tetel			
megnevezes	name	string	kötelező
azonosito	id	string	
mennyiseg	quantity	szám	kötelező
mennyisegiEgyseg	quantity_unit	string	kötelező
nettoEgysegar	unit_price	szám	kötelező
afakulcs	vat	string	kötelező
arresAfaAlap	profit_vat	szám	
nettoErtek	net_price	szám	kötelező
afaErtek	vat_amount	szám	kötelező
bruttoErtek	gross_amount	szám	kötelező
megjegyzes	comment	string	
tetelFokonyv	product_accounting		

## Szülőelem

tetel::tetelFokonyv			
gazdasagiEsem	transaction_event	string	
gazdasagiEsemAfa	transaction_vat	string	
arbevetelFokonyviSzam	income_account_no	string	
afaFokonyviSzam	vat_account_no	string	

## F.2.0. A \_reverseInvoice() funkció elvárt, és opcionális bemenetei

xml tag elnevezés      php class azonosító      adattípus      kötelező-e?      alapérték

### Szülőelem

#### beallitasok

felhasznalo	username	string	kötelező	
jelszo	password	string	kötelező	
apiKey	api_key	string		
eszamla	e_invoice	logikai változó	kötelező	true
kulcstartojelszo	keychain	string		
szamlaLetoltes	download_invoice	logikai változó	kötelező	false
szamlaLetoltesPld	download_count	szám		
aggregator	aggregator	string		

### Szülőelem

#### fejlec

szamlaszam	invoice_num	string	kötelező	
keltDatum	invoice_date	dátum		
teljesitesDatum	fulfillment	dátum		
tipus	invoice_type	string		

### Szülőelem

#### elado

emailReplyto	email_replyto	string		
emailTargy	email_subject	string		
emailSzoveg	email_content	string		

### Szülőelem

#### vevo

email	email	string		
-------	-------	--------	--	--

### F.3.0. A \_payInvoice() funkció elvárt, és opcionális bemenetei

xml tag elnevezés	php class azonosító	adattípus	kötelező-e?	alapérték
-------------------	---------------------	-----------	-------------	-----------

#### Szülőelem

#### beallitasok

felhasznalo	username	string	kötelező	
jelszo	password	string	kötelező	
apiKey	api_key	string		
szaamlaszam	invoice_num	string	kötelező	
additiv	additive	logikai változó	kötelező	false
aggregator	aggregator	string		

#### Szülőelem

#### kifizetes

datum	date	dátum	kötelező	
jogcim	transaction_title	string	kötelező	
osszeg	amount	szám	kötelező	

#### F.4.0. A \_requestInvoicePDF() funkció elvárt, és opcionális bemenetei

xml tag elnevezés	php class azonosító	adattípus	kötelező-e?	alapérték
felhasznalo	username	string	kötelező	
jelszo	password	string	kötelező	
apiKey	api_key	string		
szaamlaszam	invoice_num	string	kötelező	
valaszVerzio	response_type	szám	kötelező	1