

Számla API

Az eVIR rendszer lehetőséget biztosít külső rendszer számára számla készítésére.

Az általános [API](#) leírásban szerepelnek az alapok, ezért itt már csak a specifikus részek szerepelnek.

A kommunikáció JSON alapokon zajlik, a következő módon:

```
https://evir.hu/cegnev/cgi-bin/index.cgi?json={"token":"jMuPFcKJdyKYqMZ7iGiLpxF/32k", "dok":"szamla_api_felvesz",.....}
```

A json form mező egy hash, melynek felépítése:

```
{  
  
  token: a klienst azonosító eVIR felhasználó tokenje  
  
  dok: "szamla_api_felvesz" stringet tartalmazó kötelező elem.  
  button: "b_felvesz" stringet tartalmazó kötelező elem.  
  muvelet: "0","M1","P1","M1P2","P1M2","P1M0" hogyan jöjjön létre a számla (lásd lentebb)  
  
  partner_id: partner_id  
  
  partner_tipus: "cég" vagy "személy" értékeket tartalmazó string  
  partner_nev: partner neve  
  partner_cim_ország: ország kétbetűs kódja, (pontosan 2 karakter)  
  partner_cim_ország_nev: ország megnevezése. Megegyezik az eVIR törzsadatai szerinti névvel  
  partner_cim_varos: település neve (maximum 40 karakter)  
  partner_cim_cim: közterület, házszám stb. (maximum 60 karakter)  
  partner_cim_irszam: irányítószám (maximum 10 karakter)  
  partner_adoszam: adószám (maximum 30 karakter)  
  
  email_cim: email cím (maximum 100 karakter), "valami@example.com" formátumban.  
  
  arkat: árkategória (ha nincs megadva, akkor a partnerhez rendelt, vagy alapértelmezett árkategória lesz)  
  fizmod: fizetési mód, (beállítás szerinti hatással a teljdat, fizhat mezőkre) ha nincs megadva, akkor a rendszer alapértelmezett fizetési módja lesz)  
  teljdat: teljesítési dátum (ha nincs megadva, fizetési mód szerint lesz)  
  fizhat: fizetési határidő (ha nincs megadva, fizetési mód szerint lesz)  
  megjegyzes: Opcionális megjegyzés a számlára  
  
  nyelv: a bizonylat nyelve (ha nincs megadva, HU)  
  
  tetel_mezok: [ ... ] egy tomb, ami a tételsorokat tartalmazza. egy tételsor egy hash  
  
}  
  
egy tétel adatai: {  
  tetel_tipusa: "termék" vagy "szolgáltatás" (kötelező)  
  tetel_forrasraktar: annak a raktárnak a kódja, ahonnan a terméket eladja.
```

```
szolgáltatás esetén ignorálva.
tetel_cikkszam: a termék vagy szolgáltatás cikkszáma (kötelező)
tetel_menny: numerikus mező, az eladott mennyiség (pozitív szám)
tetel_netto: nettó eladási egységár numerikus mező.
tetel_megj: tételhez kapcsolódó megjegyzés szöveg (opcionális)
}
```

token, dok, button, muvelet: mindig kötelező megadni

- vagy meg kell adnia a partner_id adatot, de ekkor ne adja meg a többi partner adatot
- vagy meg kell adni az összes többi partner adatot, de ekkor a partner_id adatot ne adja meg.
 - Név és cím pontos egyezés esetén, a létező partnerrel dolgozik, ha nem találja, új partnert rögzít és azzal dolgozik.

tetel_mezok: legalább egy tételt meg kell adni.

email_cim: Akkor és csak akkor kell megadni, ha ki kell küldeni számlát/másolatot/értesítést emailben. azaz a muvelet tartalmaz egy 'M' betűt is.

muvelet: lehetséges értékei

- „0” Az elkészült számla nem nyomtatódik ki. A nem nyomtatott számlák listájába kerül (sorszámát már megkapta) output: json
- „P1” A számlát kinyomtatja, output: pdf
- „M1” A számlát elküldi emailben, output: json
- „M1P2” A számlát elküldi emailben, számlamásolatot kinyomtatja, output: pdf
- „P1M2” A számlát kinyomtatja, a másolatot elküldi emailben, output: pdf
- „P1M0” A számlát kinyomtatja, a számláról értesítőt küld emailben, output: pdf

Ezen értékek NEM MINDEN ESETBEN adhatóak meg. A megadható értékek a beállítások függvényében változnak, az alábbi táblázat szerint. Rosszul megadott esetben vagy hibaüzenet, vagy a beállítások szerinti működés történik.

muvelet	email_ertesito modul installálva van	Első példány emailben	PDF csatolmány
0	mindegy	mindegy	mindegy
P1	mindegy	mindegy	mindegy
M1,M1P2	igen	igen	igen
P1M2	igen	nem	igen
P1M0	igen	mindegy	nem

nyelv: opcionális. A bizonylat ezen a nyelven fog létrejönni (ha ez engedélyezve van a konfigurációban)

- „HU” magyar (default)
- „EN” angol
- „HU,EN” magyar-angol kétnyelvű

arkat: opcionális. A „Törzsadatok→Cikkszámok→Árkatégoriák→Árkatégoria-lista” menüből elérhető lista „Kategória” oszlopából választható érték, ahol a típus „értékesítési”. A kiválasztott árkatégoria pénznemében készül el a számla. Az árkatégoriának pénznemen felüli jelentősége akkor van, ha a tételknél nincs megadva „tetel_netto” egységár, mert ilyenkor az árképzés szabályai szerint az árkatégoriához tartozó áron fog szerepelni a tétel a bizonylaton.

fizmod: opcionális. A „Törzsadatok→Pénzügyi→Fizetési módok” menüben elérhető fizetési módok közül, a „Fizetési mód neve” oszlop értékei közül választhat azon feltételekkel, hogy: „Számlázáskor használható”. Az esetleges „Készpénzes” jelölésű fizetési módhoz ha kapcsolódik pénztárbizonylat készítés, az ezen az interface-en keresztül nem érhető el, ezért ilyen fizetési módok használata kerülendő.

fizhat, teljdat: dátum mező formátuma „YYYY-MM-DD” (aka ÉÉÉV-HÓ-NP) értéküket befolyásolhatja a fizetési módnál lévő beállítások, de használható vagy létrehozható olyan fizetési mód ami nem befolyásolja.

pelda JSON adat

```
{
  "token": "gnfdkjgjnknjfdshjfdk",
  "dok": "szamla_api_felvesz",
  "button": "b_felvesz",
  "nyelv": "HU",
  "muvelet": "0",

  "tetel_mezok": [
    {
      "tetel_netto": 2430,
      "tetel_megj": "XXL kék",
      "tetel_tipusa": "termék",
      "tetel_cikkszam": "c2",
      "tetel_menny": 3,
      "tetel_forrasraktar": "r1"
    }
  ],
  "partner_id": 4,
  "megjegyzes": "Példa számla",
  "arkat": "ar1",
  "fizmod": "átutalás",
}
```

Visszatérési értékek:

A rendszer a beállításoktól és a művelettől függően vagy PDF, vagy JSON választ készít. Sikeres számla rögzítés esetén, nyomtatás nélkül a lentihez hasonló.

Ha a visszatérési érték tartalmazza a 'status': 'OK' formában a status kulcshoz az OK értéket, akkor a „szamlaszam” kulcsban található a kiállított számla száma.

Más üzenetekkel együtt ezek megtalálhatóak a „message” listában is, ha az üzenethez tartozó „level” értéke „SysInfo” akkor ott a „message” üzenet „szamlaszam=SZ2015000110” formában az egyelőség jel után a létrehozott bizonylat (számla) számát fogja tartalmazni. Ha ilyen üzenet nincs, akkor feltételezhető, hogy nem jött létre a bizonylat.

```
[
  {
    "status": "OK",
    "szamlaszam": "SZ2015000110",
    "choice": {
      // nem dokumentált feature adatai
    },
    "framename": "main",
    "message": [
      {
        "level": "Info",

```

```

        "time":"2015.10.30 21:59:38.67",
        "message":"Számla rögzítve"
    },
    {
        "level":"SysInfo",
        "time":"2015.10.30 21:59:38.67",
        "message":"szamlaszam=SZ2015000110"
    },
    {
        "level":"SysInfo",
        "time":"2015.10.30 21:59:38.67",
        "message":"status=OK"
    },
    ],
    "result":null
}
]

```

Egy példa hibaüzenet formátuma:

A „message” struktúrája megegyezik a sikeres esettel. Azonban a „level” kulcs értékei sokfélék lehetnek.

- Hibának tekintett üzenetek: Fatal, Error
- Figyelmeztető üzenetek: Warning, Notice
- Tájékoztató üzenetek: Info, SysInfo

Hibás esetekben:

- a „result” kulcs alatt megjelenő adatszerkezetben a „name” kulcs alatt várható
- az „error” vagy „Kernel error” hibaüzenetről tájékoztató bejegyzés.
- A „result” ... „info” ... „errormessage” alatt pedig az a hibaüzenet, amit a rendszer visszaad.

```

[
  {
    "choice":{
      // nem dokumentált
    },
    "framename":"main",
    "message":[
      {
        "level":"Fatal",
        "time":"2015.10.30 22:11:52.31",
        "message":"Rendszerhiba: myexception (user): váratlan hibaüzenet at
        ../include/dok/szamla/szamla_api_felulet.pm line 346 captured by kernel"
      }
    ],
    "result":[
      {
        "info":{
          "errormessage":null
        },
        "name":"error"
      }
    ]
  }
]

```

Egyéb eredmények lehetnek ezenkívül még:

- nem 200 OK HTTP status: elég ritka, igyekezzük elkerülni.
- Content-type: application/x-pdf: szándékos működés eredménye. amikor a 'muvelet' mező 'P' betűt tartalmaz
- más Content-type: váratlan hibák esetén (pl: text/html)

Storno

Számla stornó gépi interface. A rendszerben kiállított számlákat a számlaszám ismeretében API híváson keresztül is lehet stornózni, nem csak a felületről. Ez a számla készítéshez nagyon hasonló módon történik:

```
http://...../cgi-bin/index.cgi?json={"dok":"szamla_storno_api_felvesz","sszlaszam":"SZ...","token":"...","muvelet":"P1"}
```

a JSON adat egy hash, melynek kulcsai az alábbi jelentésekkel bírnak

```
token: a klienst azonosító evir felhasználó tokenje (evir felhasználó jelszavának módosításakor kiíródik)
dok: "szamla_storno_api_felvesz" stringet tartalmazó kötelező elem.
sszlaszam: a stornózendó számla számát tartalmazó kötelező elem.
muvelet: "0","P1" hogyan jöjjön létre a számla (lásd fentebb a számla készítésnél)
```

A token, dok, sszlaszam, muvelet elemeket mindig kötelező megadni

A művelet hívására válaszul egy pdf dokumentum érkezik, ami a storno számlát tartalmazza. Hiba esetén a számla készítésnél megadott szerkezetű JSON hibaüzenet érkezik.

Még nem létező szolgáltatás rögzítése az API-n keresztül

Termék esetében kizárólag a cikktörzsben szereplő, készleten levő elemet lehet a bizonylatra tenni, de szolgáltatás esetében megoldható a rugalmasabb kezelés. Lehetőség van arra, hogy olyan szolgáltatás kerüljön az API-n keresztül kiszámlázásra, amelynek az adatai még nem szerepelnek a cikktörzsben. Ekkor a számlázandó tétel adatai között meg kell adni azokat a kötelező adatokat is, amelyek egyébként a cikktörzsből érkeznének. Az adatokat a tétel adatai között kell megadni a normál adatokon túlmenően:

```
tetel_cikk_rogzites: "szolg_fix" vagy "szolg_gen" a rögzítés módja szerint
tetel_cikk_megnevezes: A szolgáltatás megnevezése
tetel_cikk_afa: A szolgáltatáshoz tartozó áfa
tetel_cikk_megys: A szolgáltatás mennyiségi egysége (pl. darab, óra)
tetel_cikk_unit: A legkisebb egység, aminek a többszöröse a mennyiség lehet.
(általában pl 1)
```

A szolgáltatás rögzítéséhez ezenkívül felhasználja a tetel_cikkszam mezőt is, ami egyébként is kötelezően megadandó a számla készítéséhez.

A mezők értelmezése

- tetel_cikk_rogzites:
 - üres, vagy nincs megadva: Nem rögzít szolgáltatást, cikktörzsben szerepelnie kell a cikkszámnak
 - „szolg_fix”: Ha nem létezik a cikkszám, akkor ezzel a cikkszámmal rögzíteni fogja

- „szolg_gen”: Ha nem létezik a cikkszám, akkor levágja a végéről a számot, és egy új cikkszámot generál, aminek a végén lévő szám eggyel nagyobb a legutolsó ilyen kezdetű és számmal végződő cikkszámnál. Ez csak akkor működik, ha a beállításokban be van kapcsolva a „Következő cikkszám generálása”
- tetel_cikk_megnevezes: Ez a szolgáltatás megnevezése, a szöveg rákerül a bizonylatra
- tetel_cikk_afa: Az áfa mező lehetséges értékeinek illeszkedniük kell az ÁFA törzsadatban található értékekhez a következő módon
 - Ha áfakulcs tartozik hozzá, akkor az ott található számot kell megadni. pl. 27% áfa esetén „27.00”
 - Ha nem tartozik hozzá áfakulcs, akkor az „Adómentességi megjelölés” mező értékét kell pontosan megadni.
- tetel_cikk_megys: Ez a mennyiségi egység, pl. darab, óra
- tetel_cikk_unit: Ez a legkisebb mennyiségi egység. Ez leggyakrabban „1” szokott lenni.

Példa adat a tetel_mezok értékére.

```
...
"tetel_mezok": [
  {
    "tetel_netto":10000,
    "tetel_megj": "Kiszállással együtt",
    "tetel_tipusa": "szolgáltatás",
    "tetel_cikkszam": "c5",
    "tetel_menny": 2,
    "tetel_cikk_rogzites": "szolg_fix",
    "tetel_cikk_megnevezes": "Karbantartás",
    "tetel_cikk_afa": "27.00",
    "tetel_cikk_megys": "óra",
    "tetel_cikk_unit": 1,
  }
],
...
```

From:

<https://doc.evir.hu/> - eVIR tudásbázis

Permanent link:

https://doc.evir.hu/doku.php/evir:rendszer:api:api_szamla

Last update: **2020/06/26 11:07**

