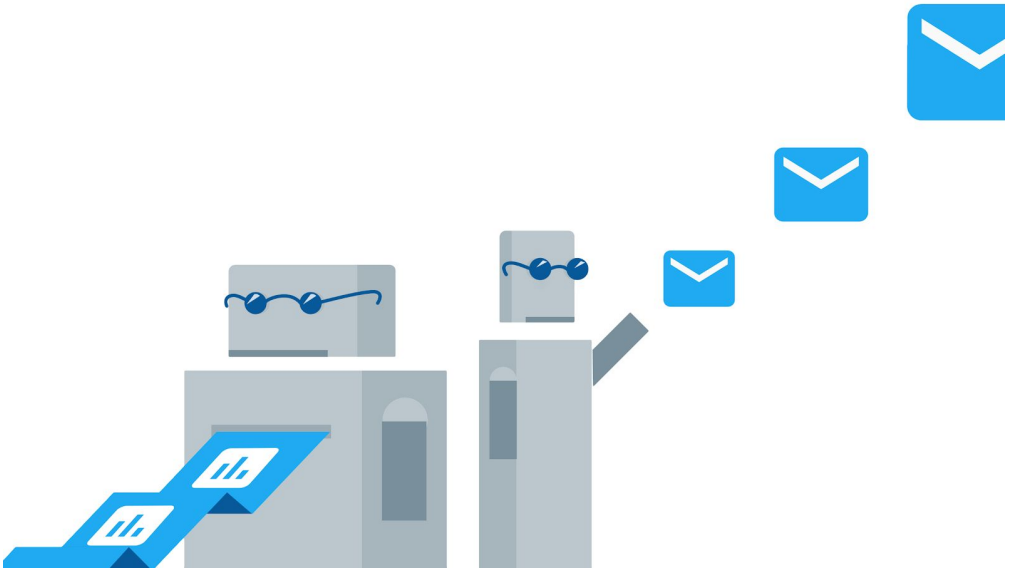
MiniCRM

MiniCRM REST API

2019. 05. 20.



Integrációs lehetőségek

A MiniCRM nyílt, a benne rögzített adatok MiniCRM API-n keresztül elérhetőek más programok számára, és azokat módosíthatod, bővítheted.

Különböző integrációs igények kielégítésére 3 API típus áll rendelkezésre:

- **XML szinkronizáció**

  Egy már meglévő más rendszerben lévő ügyfél/felhasználó adatbázis egyirányú szinkronizálása MiniCRM-be, hogy az ott kezelt folyamatokhoz szükséges alap információk rendelkezésre álljanak.

  Akkor érdemes használni ha nagy mennyiségű adatot kell átküldeni MiniCRM-be és nem feltétel, hogy az adatok valós időben átkerüljenek, néhány órás / 1 napos frissesség megfelelő.

  XML adatszinkronizáció során a külső rendszer előállít megfelelő formátumban egy XML adatállományt, amit a MiniCRM egy API hívásra reagálva letölt és feldolgoz. A benne található adatlapokat frissíti, illetve újakat hoz létre, ha szükséges. Egy-egy szinkronizáció során nem kell minden adatlapot/mezőt újraküldeni, részleges – csak módosult adatokat tartalmazó – XML-t is fel tud dolgozni a MiniCRM.

  Az XML alapú adatszinkronizálás akár több millió adatlap frissítésére is alkalmas.

- **REST API**

  Azonnali adat elérést és módosítást tesz lehetővé. Lehet keresni, lekérdezni és adatlapokat módosítani. Akkor érdemes használni, ha beazonosítási logika egyedi, a beküldendő adat függ a MiniCRM-ben tárolt aktuális adattól vagy a mozgatandó adatok száma relatíve alacsony.

- **CallLog API**

  VOIP szolgáltatók részére kialakított interface. Ezen keresztül lehet a MiniCRM-be beküldeni telefonhívások meta adatait, hogy azok a releváns ügyfelek adatlapjain az előzmények között megjelenhessenek.

Ez a dokumentáció a MiniCRM REST API működését írja le.

MiniCRM alapfogalmak

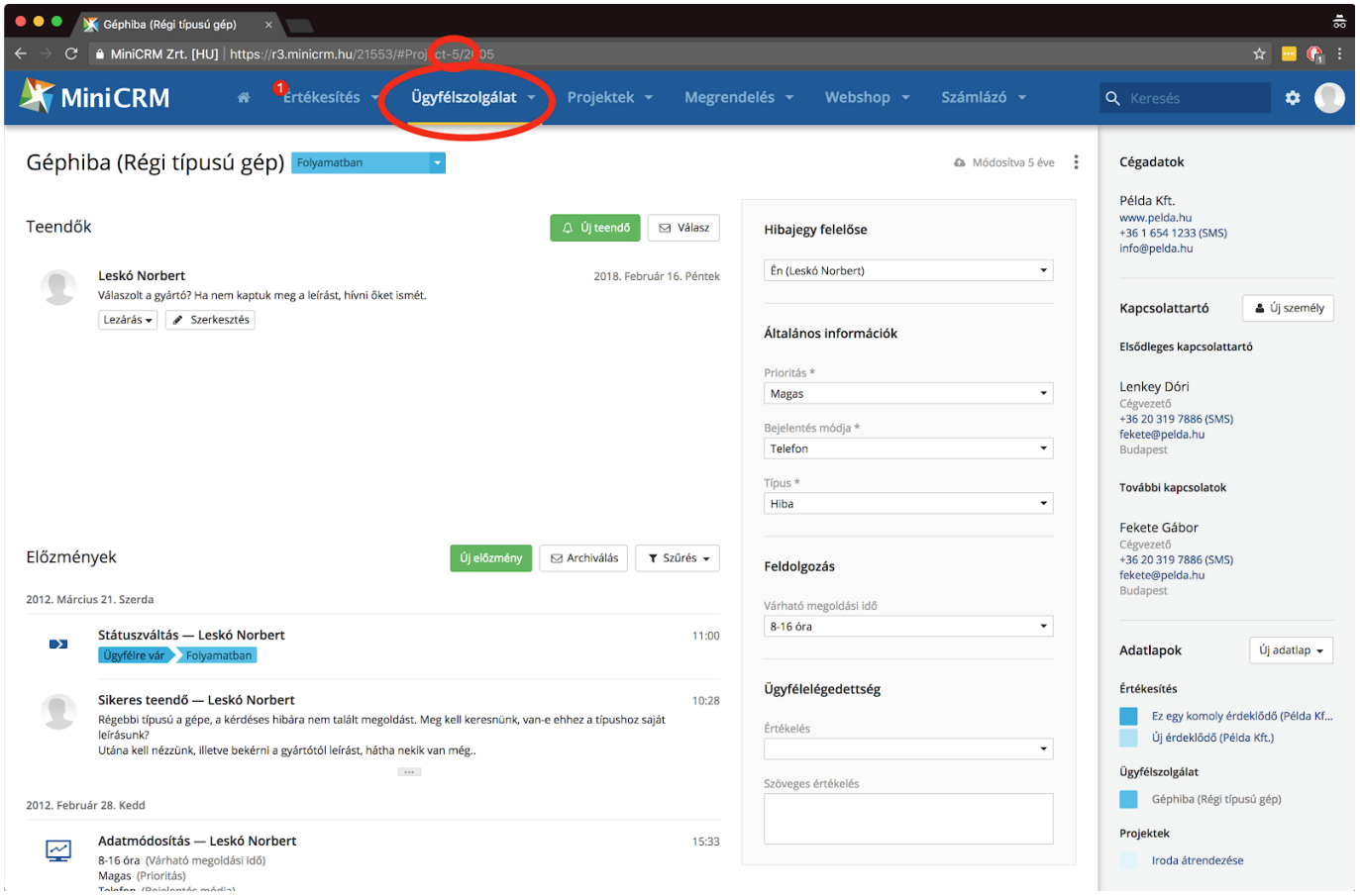
Avagy a felületen látható dolgok API-n keresztül hol és milyen néven érhetőek el?

Modul / Category

MiniCRM-en belül az adatok és folyamatok modulokba rendeződnek. Egy modul egy fül/tab a felső menüben, pl.: Értékesítés, Ügyfélszolgálat, stb…

A modul azonosítója az url-ben itt látható: "#Project-**5**/2005".

API-n a modulok Category és azonosítójuk CategoryId néven érhetőek el.



Adatlap / Project

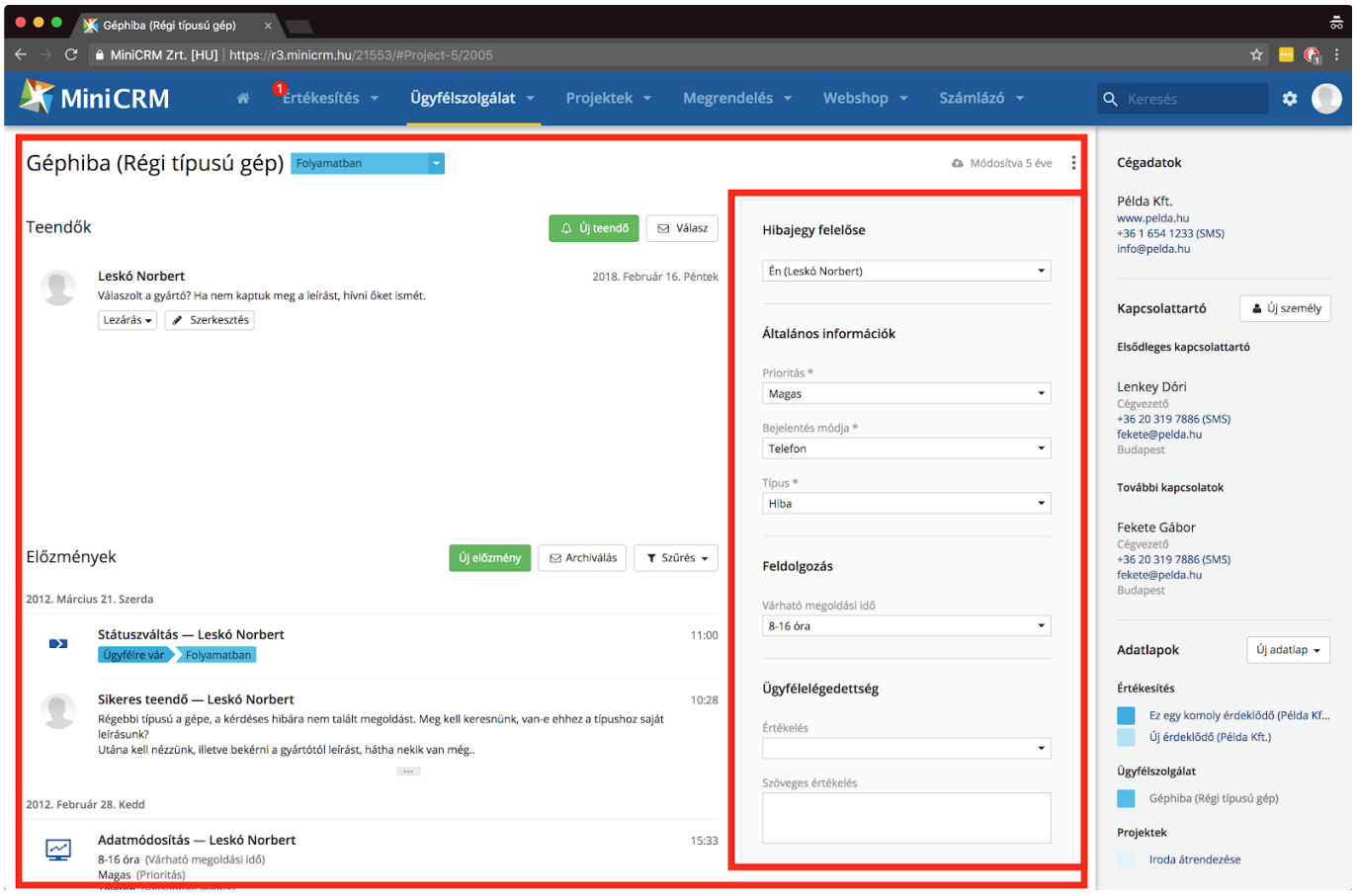
Modulokon belül adatlapokat listáz a MiniCRM. Egy adatlap struktúrája minden modulban egyforma. A konkrét mezőkben vannak különbségek és az adatlaphoz pluszban hozzárendelt adatokban.

Az adatlapi mezők webes felületen keresztül szabadon testreszabhatóak.

Minden adatlaphoz tartoznak:

- Teendők
- Előzmények
- Adatlapi mezők

Az adatlapok API-n keresztül Project néven érhetőek el.



A megrendelés és számlázó modulban is adatlapokat (Project) listáz a MiniCRM és az egyes számlák és megrendelések 1-1 konkrét adatlaphoz vannak kövtve. Így az egyedi mezőket, teendőket és automatizálást lehet adatlapon intézni a rendelés/számla specifikus funkciók pedig külön tabon és API végponton üzemelnek.

Ügyfél, kapcsolat / Contact

Az adatlapok mellett jobb oldalon látható az ügyfél és kapcsolattartók akikhez az adatlap tartozik.

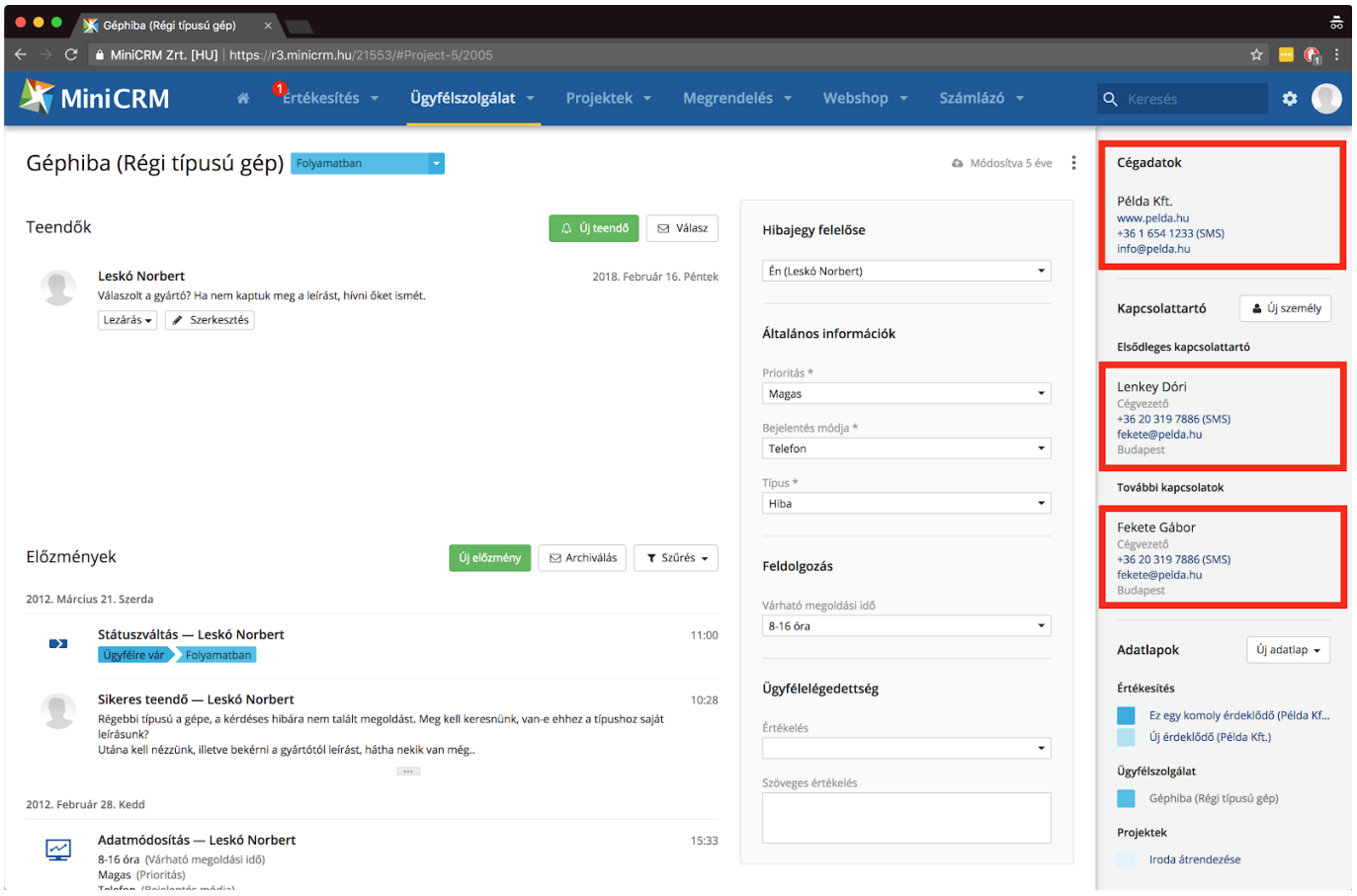
- Cég
- Elsődleges kapcsolattartó (ugyanazon cég különböző adatlapjainál lehet más-más elsődleges kapcsolattartó)
- További kapcsolatok

Kapcsolatok API-n keresztül Contact néven érhetőek el. Cégek és személyek egységesen, egy "Type" nevű mező "Business" v "Person" értéke dönt arról, hogy személy vagy cég a kiválasztott kapcsolat.

Contact.Type mező értékét létező rekord esetén nem lehet módosítani.

Több kapcsolattartó a BusinessId nevű mezőn keresztül kapcsolódik össze. A lenti screenshoton látható cég, kapcsolatok és adatlapok mindegyikének ugyanaz a BusinessId-je.

BusinessId az ügyfél rekordot képviselő Contact rekord Id-je.



Technikai feltételek

A MiniCRM API egy egyszerű felület, ami használható bármilyen programozási nyelvből (Javascript, PHP, C#, C, Java, stb…). A használat feltétele, hogy a programot futtató számítógép rendelkezzen működő internetkapcsolattal, el tudja érni a MiniCRM API-t, valamint a program rendelkezzen érvényes MiniCRM rendszerrel, és hozzá tartozó API kulccsal.

Azonosítás szabványos http azonosítással “HTTP Auth” történik.

Az azonosításhoz szükséges

- **SystemId**: rendszer azonosító (a böngésző címsorában az https://r3.minicrm.hu/ után látható, maximum 5 jegyű szám).

- **ApiKey**: API kulcs, amit adminisztrátor jogosultságú felhasználóként tudsz igényelni a Beállítások > Rendszer képernyőn. Ehhez az Új API kulcs készítése gombra kell kattintani. Ha már generáltál API kulcsot, lehetőséged van újat generálni, vagy törölni a meglévőt. Egy rendszerben egyszerre csak egy API kulcs lehet érvényben, ha újat generálsz a régit töröljük.

Példák felépítése

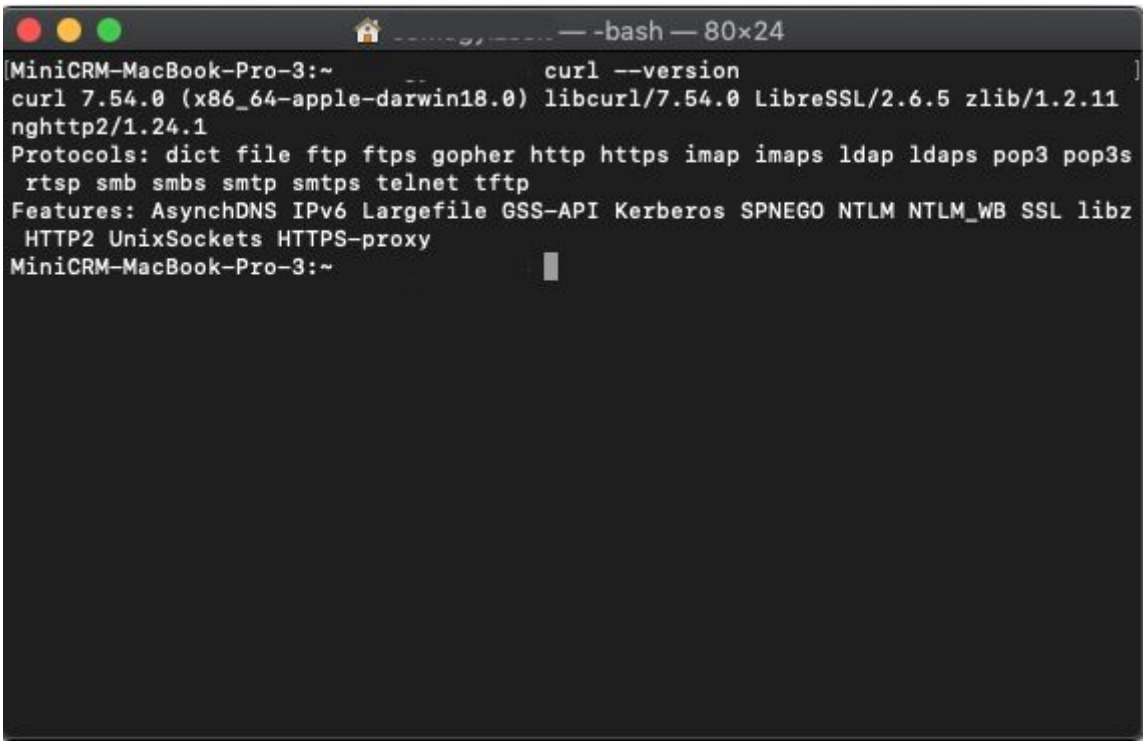
A dokumentációban szereplő példákat próbáld ki Te is saját számítógépeden. A konkrét példákat Mac OSX-en készítettük, de ugyanúgy működnek Linux és Windows alatt is.

Példáinkban a curl nevű programot használjuk.Ezzel a programmal terminálból lehetséges többek között HTTP REST kéréseket küldeni a szerver felé (GET, PUT, PATCH, DELETE)

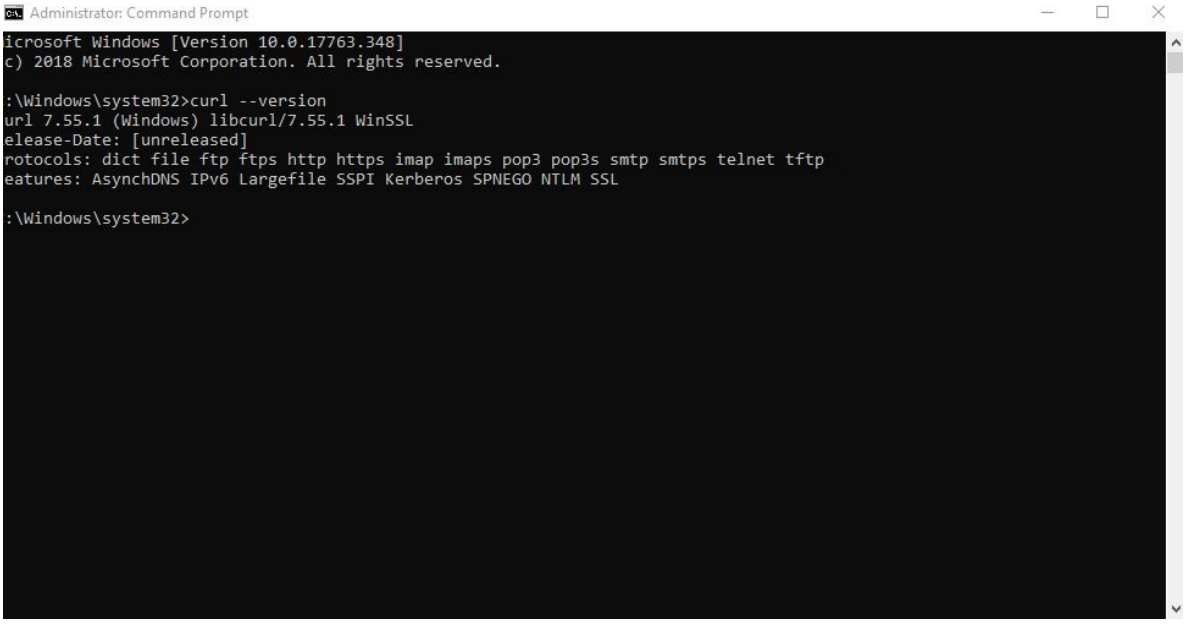
Mielőtt elkezdenéd az integrációs folyamatot, tanácsoljuk, hogy példáinkat próbáld ki, hogy működésüket könnyen megértsd.

cUrl telepítési folyamat

Mac OSX: Mac eszközökön a curl telepítés nélkül elérhető a terminálból.



Windows: Windows 10 rendszerben 17063 verziójú, vagy annál frissebb termékben elérhető, telepítés nélkül.



Példáinkban feltételezzük, hogy a rendszer azonosító és api kulcs SystemId és ApiKey nevű környezeti változókban beállításra kerültek. Pl.:

```
$ export SystemId=12345
$ export ApiKey=NemTudom
$ curl --user $SystemId:$ApiKey -I -X GET "https://r3.minicrm.hu/Api/R3/Category"

HTTP/1.1 401 Unauthorized
```

A fenti példában a $ jelöli a parancssort. Az az utáni karaktereket terminal programba másolva futtathatóak a parancsok és a példával egyező eredményt/kimenetet produkálnak.

A curl egy kényelmes eszköz a MiniCRM API megismerésére és a neked fontos végpontok feltérképezésére. Gyorsan meg tudod tanulni milyen kérésekre milyen válaszokat ad a MiniCRM, amiket konkrét példák után könnyebb lesz implementálni saját rendszeretekben.

A MiniCRM REST API JSON választ ad, formázás nélkül. Ezt számítógépek könnyen tudják értelmezni, emberi fogyasztásra kevésbé alkalmas.

```
$ curl --user $SystemId:$ApiKey "https://r3.minicrm.hu/Api/R3/Category"
{"3":"\u00c9rt\u00e9kes\u00edt\u00e9s","5":"\u00dcgyf\u00e9lszolg\u00e1lat","6":"P
rojektek","7":"Megrendel\u00e9s","8":"Webshop","9":"Sz\u00e1ml\u00e1z\u00f3"}
```

Én a jq nevű programot használom, hogy az API válaszokat könnyebben tudjam értelmezni:

```
$ curl -s --user $SystemId:$ApiKey "https://r3.minicrm.hu/Api/R3/Category" | jq
{
  "10": "Értékesítés",
  "5": "Ügyfélszolgálat",
  "6": "Projektek",
  "7": "Megrendelés",
  "8": "Webshop",
  "9": "Számlázó",
  "3": "Partnerek"
}
```

Ha több paramétert adsz át API-nak (& jellel elválasztva) nagyon fontos, hogy idéző jelek közé – " – tedd a curl-nek adott Url-t, különben a & utáni rész "elveszik". Én az egyszerűség kedvéért mindig " jelek közé teszem az Url-t, mindegy épp mi van benne.

Kódlap

A MiniCRM API az Unicode kódlapot használja UTF-8 ábrázolási metódussal. Minden bejövő paramétert UTF-8 kódolásban vár és minden választ UTF-8 kódlapon ad.

Formátum

Bemeneti és kimeneti adatformátum minden esetben JSON serialized tömb.

Lapozás

Azok a végpontok amiken keresztül szűrni/keresni lehet a rendszerben egységes keret szerint adják vissza a találatokat.

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Project?CategoryId=3"
{
  "Count": 42,
  "Results": {
    "2009": {
      "Id": 2009,
      "Name": "Ez egy komoly érdeklődő (Példa Kft.)",
      "Url": "https://r3.minicrm.hu/Api/R3/Project/2009",
      "ContactId": 7461,
      "StatusId": 2540,
      "UserId": 22404,
      "Deleted": 0,
      "BusinessId": 7453
    },
...
  }
}
```

- **Count**: A válaszban a találatok egy tömbben érkeznek, melynek Count kulcsán található a talált elemek száma.
- **Results**: A Results kulcs alatt egy-egy külön tömbben találhatóak a megtalált elemek. A listába csak a legfőbb adatok kerülnek be.
- **Url**: Az egyes elemekről a teljes elérhető adat lekérdezhető API-n keresztül az Url mezőben kapott címről.

Az API keresési oldalanként 100 találatot jelenít meg. A találatok lapozhatóak a Page paraméter segítségével. Fontos, hogy a lapozás az API esetében 0-ról indul, tehát a második oldalt a Page=1 paraméterrel lehet elérni! Pl.:

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Project?CategoryId=5&Page=1"
{
  "Count": 181,
  "Results": {
...
  }
}
```

Hibaüzenetek és hibakezelés

A MiniCRM API http státuszkódokkal válaszol az esetleges hibákra. Amennyiben 200 – OK kóddal tér vissza, a kérés sikeres volt. Ha hiba érkezik az API-tól, akkor a válasz formátuma nem JSON tömb, hanem plain text!

Hibakódok listája

- **400 – Bad Request**: A kérésben olyan paraméter szerepel, ami nem ismerhető fel vagy nem engedélyezett a rendszerben. Továbbá olyankor is ezt a hibát küldi a rendszer, ha az adott metódus paramétert vár, de nem kapott paramétert.
- **401 – Unauthorized**: Hozzáférés megtagadva. Ellenőrizd, hogy a rendszer azonosítót és API kulcsot valóban helyesen adtad-e meg? Díjcsomag váltás volt? Ha kissebb csomagra váltottál amiben már nincs benne a MiniCRM REST API, szintén ezt a hibaüzenetet kapod.
- **404 – Not Found**: A megadott URL nem található a rendszerben. Leggyakoribb esetben a hibás / elgépelt URL-ek esetében jöhet elő ez a hiba.
- **405 – Method Not Allowed**: Olyan esetben találkozhatunk ezzel a hibával, ha hibás vagy nem engedélyezett metódussal próbálunk meghívni egy URL-t.
- **500 – Internal Server Error**: Belső feldolgozási hiba a rendszerben. Leggyakrabban akkor találkozhatunk vele, ha hibás adatokat próbálunk feltölteni API-n keresztül a rendszerbe, vagy ha például egy adott mezőnek nem található az értéke, amit fel szeretnénk tölteni.

API Request rate limit

Rendszerünk stabilitásának védelmében a MiniCRM rendszerekre **180 request / perc / rendszer** rate limit érvényes.

A limitet meghaladó használat esetén a sztenderdeknek megfelelően **HTTP 429 “Too Many Requests”** válasz fog érkezni a szerver felől az érintett request-ekre és azok nem kerülnek feldolgozásra.

Éles és teszt környezet

MiniCRM API-n keresztül könnyen integrálható két rendszer. Egy ilyen fejlesztés sokszor jár próbálkozással, tesz adatok betöltésével illetve hibásan létrejött adatokkal.

Fejlesztéshez igényelhető "éles" környezetben ingyenes MiniCRM rendszer. Ennek hátránya, hogy nem tartalmazza az "éles-éles" rendszerben lévő adatokat, testre szabásokat. Ezért a MiniCRM 2 egymástól teljesen elkülönített, külön üzemeltetett környezetet (STAGE) biztosít ügyfeleinek.

- PROD / Éles környezet: https://r3.minicrm.hu/
- TEST / Teszt környezet: http://r3-**test**.minicrm.hu/

Igény esetén ügyfélszolgálatunk áttölti éles rendszerüket TEST környezetbe. Ilyenkor minden olyan felhasználó hozzáférhet TEST környezetben is a rendszerhez, akinek áttöltéskor hozzáférése volt a PROD rendszerhez.

Ha tehát fejlesztőjének hozzáférést szeretne adni TEST környezetben a rendszerhez, akkor őket felhasználónak előbb PROD környezetben kell felvenni. TEST környezetben biztonsági okokból új felhasználók felvételére nincs lehetőség.

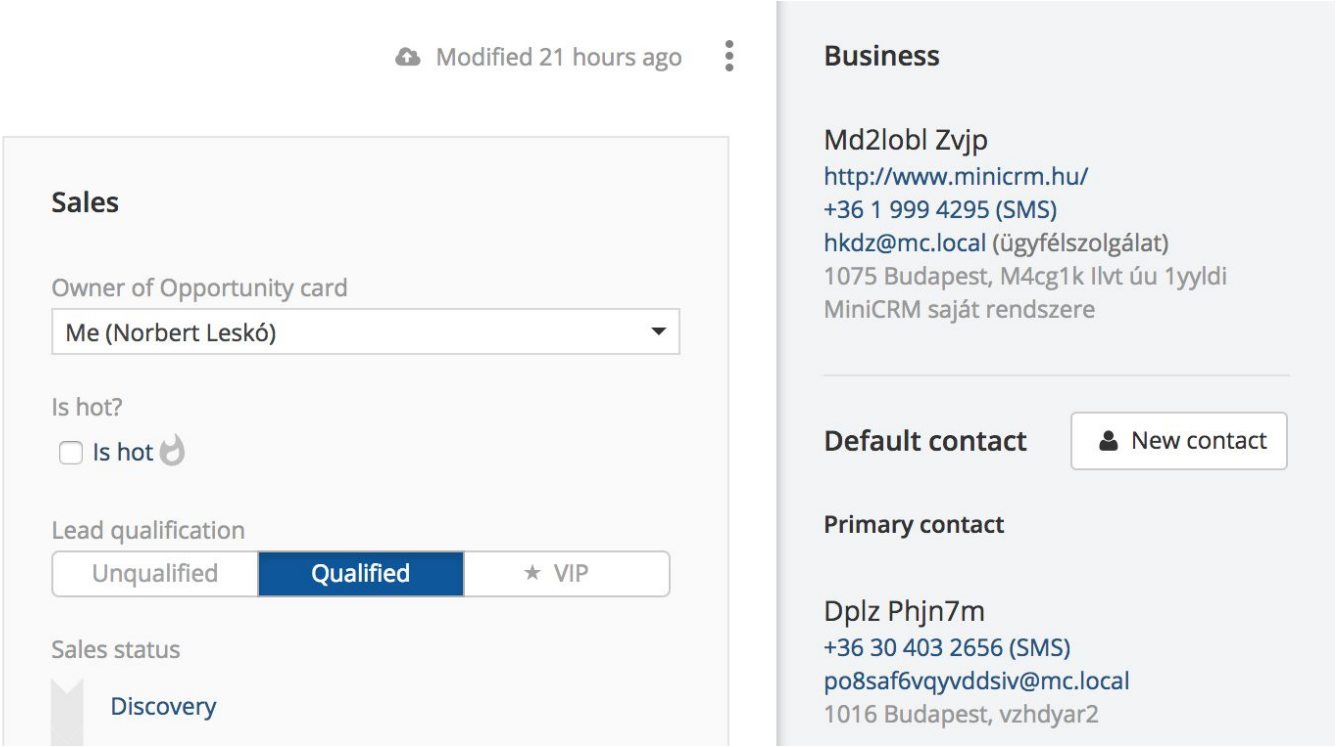
Belépés és API szempontból az egyetlen különbség a két környezet között a "-test" az URL-ben. A dokumentációban lévő összes példa alkalmazható TEST környezetben is, r3-test.minicrm.hu -ra cserélve r3.minicrm.hu -t.

Tömeges törlés, adathelyreállítás helyett

Sokkal könnyebb, nyugodtabb TEST környezetben fejleszteni. Így nem kell izgulni éles adatok miatt. Éles adatbázisban tömeges törlést, adathelyreállítás sokszor nagyon körülményes vagy akár lehetetlen megcsinálni.

Személyes adatok roncsolva

Adatvédelmi okokból PROD környezetünket személyes adatok nem hagyhatják el. Ezért a TEST környezetbe áttöltés előtt anonimizált mentést készít rendszerünk, ami véletlenszerű adatokkal tölti fel ügyfelek neveit, elérhetőségeit. A mezők testreszabása, rendszer beállításai nem változnak.

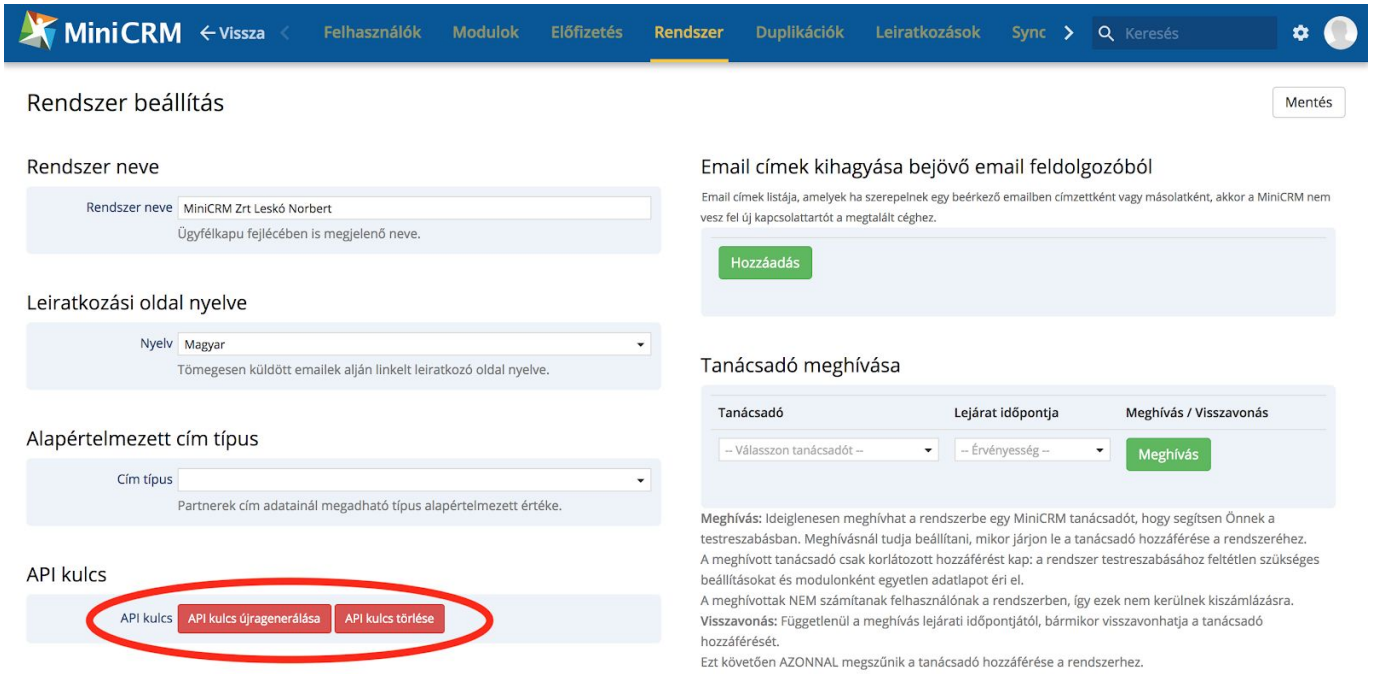

Modified 21 hours ago
Sales
Owner of Opportunity card
Me (Norbert Leskó)
Is hot?
Is hot
Lead qualification
Unqualified
Qualified
VIP
Sales status
Discovery
Business
Md2lobl Zvjp
http://www.minicrm.hu/
+36 1 999 4295 (SMS)
hkdz@mc.local (ügyfélszolgálat)
1075 Budapest, M4cg1k Ilvt úu 1yyldi
MiniCRM saját rendszere
Default contact
New contact
Primary contact
Dplz Phjn7m
+36 30 403 2656 (SMS)
po8saf6vqyvddsiv@mc.local
1016 Budapest, vzhdyar2

Api kulcs készítés

API kulcsot adminisztrátor jogosultságú felhasználóként tudsz igényelni a Beállítások > Rendszer képernyőn. Ehhez az Új API kulcs készítése gombra kell kattintani.

Ha már generáltál API kulcsot, lehetőséged van újat generálni, vagy törölni a meglévőt. Törölni akkor kell API kulcsot, ha szeretnéd lekapcsolni rendszered API-s elérését.

Egy rendszerben egyszerre csak egy API kulcs lehet érvényben, ha újat generálsz a régit töröljük.

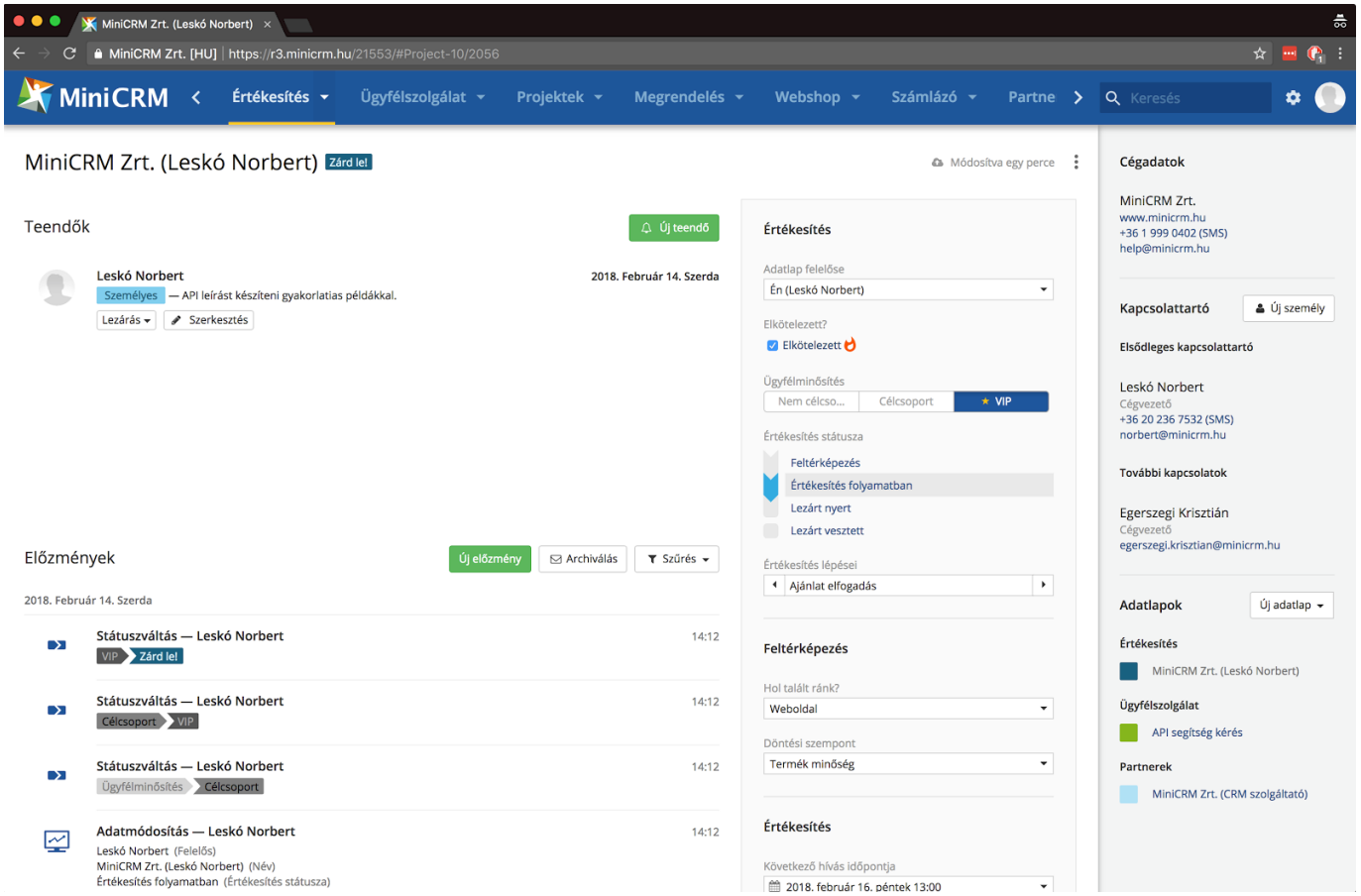


FONTOS: az API kulcs pont olyan jelszó mint amit Te használsz webes belépésnél. Soha ne küldd el senkinek emailben, sms-ben és más megbízhatatlan, lehallgatható csatornán keresztül!

Contact: Cég & kapcsolattartók végpont

Gyakran indul egy kereséssel a MiniCRM integráció folyamata. Például megtalálni egy személyt email cím alapján és rögzíteni ha még nincs meg a rendszerben vagy felvenni hozzá új információt, ha már megtalálható a rendszerben.

Példa egy teszt rendszerben a MiniCRM mint cég, két kapcsolattartó és 3 adatlap 3 különböző modulban.



Contact keresés emailcím alapján

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact`**
- Paraméter: **`Email`**

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Contact?Email=norbert@minicrm.hu"
{
  "Count": 1,
  "Results": {
    "7506": {
      "Id": 7506,
      "Name": "Leskó Norbert",
      "Url": "https://r3.minicrm.hu/Api/R3/Contact/7506",
      "Type": "Person",
      "Email": "norbert@minicrm.hu",
      "Phone": "06 (20) 236 7532",
      "BusinessId": 7505
    }
  }
}
```

A válasz értelmezése:

- **Id**: a megtalált kontakt egyedi azonosítója.
- **Type**: Person személy esetén, Business cég esetén.
- **BusinessId**: a cég azonosítója, akihez a kontakt tartozik.

Contact keresés módosítás ideje alapján

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact`**
- Paraméter: **`UpdatedSince`**

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Contact?UpdatedSince=2018-02-14+10:00:00"
{
  "Count": 4,
  "Results": {
...
  }
}
```

Csak olyan elemek kerülnek listázásra, amik a megadott dátum után lettek módosítva.

Contact keresés kulcsszóval

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact`**
- Paraméter: **`Query`**

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Contact?Query=lesko"
{
  "Count": 3,
  "Results": {
    "7506": {
      "Id": 7506,
      "Name": "Leskó Norbert",
      "Url": "https://r3.minicrm.hu/Api/R3/Contact/7506",
      "Type": "Person",
      "Email": "norbert@minicrm.hu",
      "Phone": "06 (20) 236 7532",
      "BusinessId": 7505
    }
...
  }
}
```

Szabad szavas keresés az összes szöveges mezőben. Csak teljes szó egyezés esetén ad találatot. Kis/nagy betű és ékezet nem számít.

Contact keresés telefonszám alapján

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact`**
- Paraméter: **`Query`**

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Contact?Query=2367532"
{
  "Count": 1,
  "Results": {
    "7506": {
      "Id": 7506,
      "Name": "Leskó Norbert",
      "Url": "https://r3.minicrm.hu/Api/R3/Contact/7506",
      "Type": "Person",
      "Email": "norbert@minicrm.hu",
      "Phone": "06 (20) 236 7532",
      "BusinessId": 7505
    }
  }
}
```

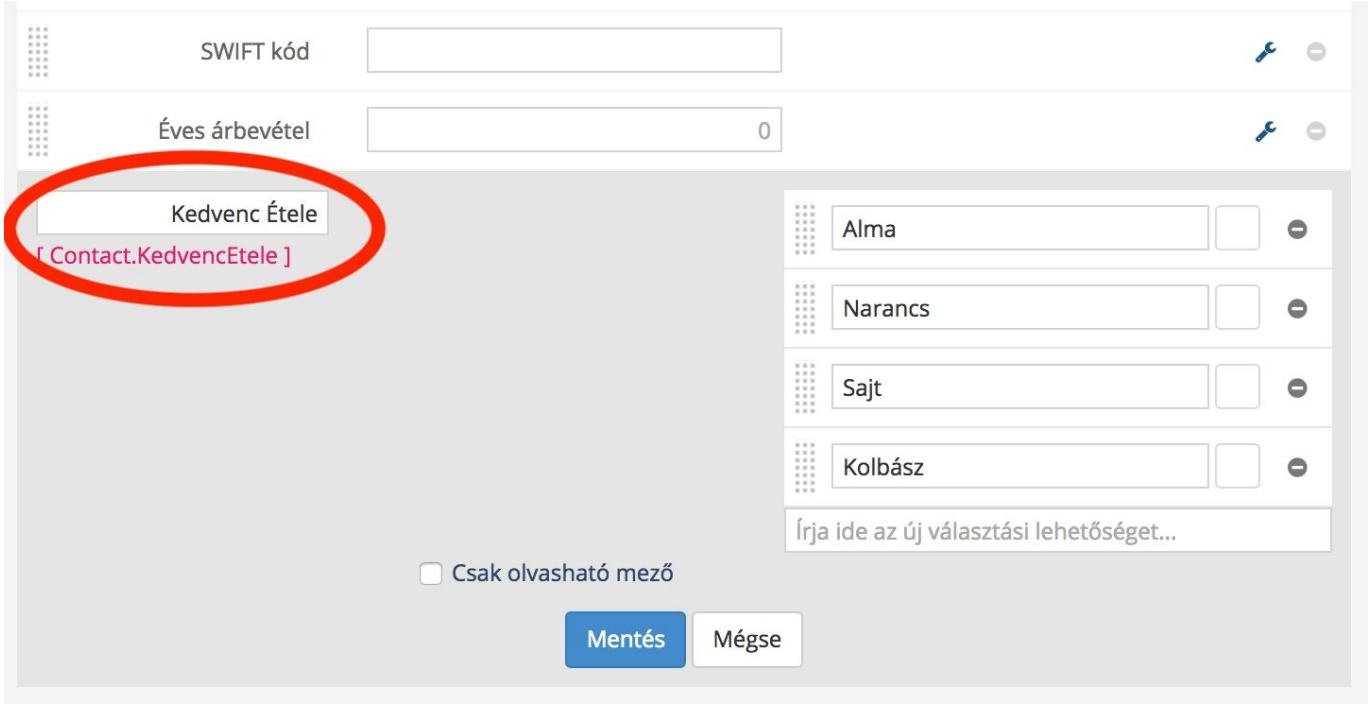
Telefonszámok keresésére ez a típusú keresés javasolt! Telefonszámok keresésekor a rendszer normalizálja a kapott telefonszámot és az alapján végzi a keresést. A keresés a telefonszámok végéről indul, így elég megadni a telefonszám utolsó számjegyeit, hogy releváns találatokat kapjunk.

Contact keresés mezők alapján

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact`**
- Paraméter: **`Keresendő mező(k) nevei`**

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Contact?KedvencEtele=Alma"
{
  "Count": 16,
  "Results": {
...
  }
}
```

Az API lekérdezésnél használható mező neveket kétféleképpen lehet megtudni. MiniCRM webes felületén, az adatlap/kapcsolat mezők testreszabás képernyőn látszik a mező API-n elérhető neve:



Az elérhető mezők és lehetséges értékeik adott rendszerhez API-n keresztül is lekérdezhetőek a Schema végponton keresztül. Pl.:

- API végpont: **`https://r3.minicrm.hu/Api/R3/Schema/$Type`**
- Paraméter: **`$Type helyén Business v Person v Project/$CategoryId`**

```
$ curl -s --user $SystemId:$ApiKey "https://r3.minicrm.hu/Api/R3/Schema/Person"
{
  "Id": "Int",
  "BusinessId": "Int",
  "FirstName": "Text(255)",
  "LastName": "Text(255)",
  "Email": "Text(128)",
  "Phone": "Text(128)",
...
  "KedvencEtele": {
    "2001": "Alma",
    "2002": "Narancs",
    "2003": "Sajt",
    "2004": "Kolbász"
  }
}
```

Cég összes kapcsolattartója

Ha a fenti módszerek egyikével sikerül megtalálni egy kapcsolattartót, gyakran szükség van a hozzá tartozó cégre és további kapcsolattartókra.

A kapcsolat válasz BusinessId mező értéke alapján lehet megtalálni a további kapcsolatokat. Példa rendszeremben "BusinessId": 7505 választ kaptam magamra, ezért ezt használom.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact`**
- Paraméter: **`MainContactId`**

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Contact?MainContactId=7505"
{
  "Count": 3,
  "Results": {
    "7505": {
      "Id": 7505,
      "Name": "MiniCRM Zrt.",
      "Url": "https://r3.minicrm.hu/Api/R3/Contact/7505",
      "Type": "Business",
      "Email": "help@minicrm.hu",
      "Phone": "06 (1) 999-0402"
    },
    "7506": {
      "Id": 7506,
      "Name": "Leskó Norbert",
      "Url": "https://r3.minicrm.hu/Api/R3/Contact/7506",
      "Type": "Person",
      "Email": "norbert@minicrm.hu",
      "Phone": "06 (20) 236 7532",
      "BusinessId": 7505
    },
    "7507": {
      "Id": 7507,
      "Name": "Egerszegi Krisztián",
      "Url": "https://r3.minicrm.hu/Api/R3/Contact/7507",
      "Type": "Person",
      "Email": "egerszegi.krisztian@minicrm.hu",
      "Phone": "",
      "BusinessId": 7505
    }
  }
}
```

Kapcsolat vagy cég részletes adatai

Ha megvannak a kapcsolattartók és cégek akikre kíváncsi vagy, a válaszokban kapott "Url" mezőt lekérve API-n keresztül megkapod a kiválasztott kontakt adatait.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact/$Id`**
- Paraméter: **`$Id helyén a kapcsolat egyedi azonosítója`**

```
$ curl -s --user $SystemId:$ApiKey "https://r3.minicrm.hu/Api/R3/Contact/7506"
{
  "Id": 7506,
  "ParentId": 0,
  "Type": "Személy",
  "FirstName": "Norbert",
  "LastName": "Leskó",
  "Email": "norbert@minicrm.hu",
  "Phone": "+36 20 236 7532",
  "Description": "",
  "Deleted": 0,
  "CreatedAt": "2018-02-14 14:10:07",
  "Url": "",
  "BankAccount": "",
  "Swift": "",
  "VatNumber": "",
  "Position": "Cégvezető",
  "Industry": "",
  "Region": "",
  "YearlyRevenue": 0,
  "KedvencEtele": "Alma",
  "BusinessId": 7505,
  "Tags": []
}
```

Kapcsolattartó elérhetőségének módosítása

API-n keresztül lehetőség van elérhetőség, illetve kapcsolattartó adatainak módosítására a Contact végpont segítségével.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact/$Id`**
- Paraméter: **`$Id helyén az adatlap egyedi azonosítója`**
- HTTP metódus: PUT
- Adatok beküldése: JSON formátumban
- Érdemes csak azokat a mezőket küldeni, amiket módosítani szeretnél

Elérhetőség módosításánál (telefon, e-mail cím) lehetőség van többször adatbeküldésre is. Ilyenkor a legutóbb beküldött adat lesz az alapértelmezett. Email módosításhoz az “Email”, telefonszám módosításhoz a “Phone” azonosító használandó.

```
$ curl --user $SystemId:$ApiKey -XPUT "https://r3.minicrm.hu/Api/R3/Contact/7506"
-d '{"Email": "norbert+1@minicrm.io"}'

$ curl --user $SystemId:$ApiKey -XPUT "https://r3.minicrm.hu/Api/R3/Contact/7506"
-d '{"Email": "norbert+2@minicrm.io"}'
```

Sikeres mentés után az új adatlap azonosítóját Id property-ben tartalmazó json objektum a válasz.

Project: Adatlap végpont

Cég összes adatlapja

Gyakran szükség van egy céghez tartozó összes adatlapra. A kapcsolat válasz BusinessId mező értéke alapján lehet megtalálni az adatlapokat. Példa rendszeremben "BusinessId": 7505 választ kaptam magamra, ezért ezt használom.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Project`**
- Paraméter: **`MainContactId`**

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Project?MainContactId=7505"
{
  "Count": 3,
  "Results": {
    "2054": {
      "Id": 2054,
      "Name": "MiniCRM Zrt. (CRM szolgáltató)",
      "Url": "https://r3.minicrm.hu/Api/R3/Project/2054",
      "ContactId": 7506,
      "StatusId": 2538,
      "UserId": 22404,
      "BusinessId": 7505
    },
    "2055": {
      "Id": 2055,
      "Name": "API segítség kérés",
      "Url": "https://r3.minicrm.hu/Api/R3/Project/2055",
      "ContactId": 7506,
      "StatusId": 2521,
      "UserId": 22404,
      "BusinessId": 7505
    },
    "2056": {
      "Id": 2056,
      "Name": "MiniCRM Zrt. (Leskó Norbert)",
      "Url": "https://r3.minicrm.hu/Api/R3/Project/2056",
      "ContactId": 7506,
      "StatusId": 2622,
      "UserId": 22404,
      "BusinessId": 7505
    }
  }
}
```

Cég adatlapjai egy modulból

Gyakran szükség van egy céghez tartozó adatlapokra egy konkrét modulból. A kapcsolat válasz BusinessId mező értéke alapján lehet megtalálni az adatlapokat. Példa rendszeremben "BusinessId": 7505 választ kaptam magamra, ezért ezt használom.

Modul azonosító - CategoryId - értéke kiderülhet a dokumentáció elején mutatott módszerekkel.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Project`**
- Paraméter: **`MainContactId & CategoryId`**

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Project?MainContactId=7505&CategoryId=10"
{
  "Count": 1,
  "Results": {
    "2056": {
      "Id": 2056,
      "Name": "MiniCRM Zrt. (Leskó Norbert)",
      "Url": "https://r3.minicrm.hu/Api/R3/Project/2056",
      "ContactId": 7506,
      "StatusId": 2622,
      "UserId": 22404,
      "BusinessId": 7505
    }
  }
}
```

Ha curl-el tesztelve nem teszed " jelek közé az url-t a & utáni rész "elveszik".

Az egyes szűkítési feltételek kombinálhatóak szabadon, pl adott ügyfél adatlapjai egy konkrét státuszból:

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/R3/Project?MainContactId=7505&StatusId=2622"
{
  "Count": 1,
  "Results": {
    "2056": {
      "Id": 2056,
      "Name": "MiniCRM Zrt. (Leskó Norbert)",
      "Url": "https://r3.minicrm.hu/Api/R3/Project/2056",
      "ContactId": 7506,
      "StatusId": 2622,
      "UserId": 22404,
      "BusinessId": 7505
    }
  }
}
```

Adatlap részletes adatai

Ha megvan az adatlap azonosítója - vagy API Url-je - akkor itt tudod lekérni a részletes adatokat. Fenti példáimban az "Id": 2056 választ kaptam adatlap keresésnél, ezt használom itt.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Project/$Id`**
- Paraméter: **`$Id helyén az adatlap egyedi azonosítója`**

```
$ curl -s --user $SystemId:$ApiKey "https://r3.minicrm.hu/Api/R3/Project/2056"
{
  "Id": 2056,
  "CategoryId": 10,
  "ContactId": 7506,
  "UserId": "Leskó Norbert",
  "Name": "MiniCRM Zrt. (Leskó Norbert)",
  "StatusUpdatedAt": "2018-02-14 14:12:41",
  "Deleted": 0,
  "CreatedAt": "2018-02-14 14:12:31",
  "AutoSalesV3_Qualification": "VIP",
  "AutoSalesV3_IsHot": "Igen",
  "AutoSalesV3_SalesStatus": "Értékesítés folyamatban",
  "AutoSalesV3_LeadStep": "Ajánlat elfogadás",
  "AutoSalesV3_CustomerStep": "",
  "DontesiSzempont": "Termék minőség",
  "HolTalaltRank": "Weboldal",
  "VarhatoArbevetel": 1000000,
  "KovetkezoHivasIdpontja": "2018-02-16 13:00:59",
  "VarhatoLezarasDatuma": "2018-02-23 23:59:59",
  "BusinessId": 7505
}
```

Adatlap módosítása

Ha megvan az adatlap azonosítója - vagy API Url-je - akkor ezen keresztül módosítani is tudod az adatokat. Fenti példáimban az "Id": 2056 választ kaptam adatlap keresésnél, ezt használom itt is.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Project/$Id`**
- Paraméter: **`$Id helyén az adatlap egyedi azonosítója`**
- HTTP metódus: PUT
- Adatok beküldése: JSON formátumban
- Érdemes csak azokat a mezőket küldeni, amiket módosítani szeretnél.

```
$ curl -s --user $SystemId:$ApiKey -XPUT
"https://r3.minicrm.hu/Api/R3/Project/2056" -d '{"AutoSalesV3_Qualification":
"Célcsoport"}'
{"Id":2056}
```

A legördülő listás mezőknél mint például a StatusId és UserId mentéskor küldheted az értéket a felületen látható módon és a rendszerben tárolt egyedi azonosítójával is (ami épp kényelmesebb).

Sikeres mentés után az adatlap azonosítóját Id property-ben tartalmazó json objektum a válasz.

Fájl feltöltés adatlapra

Az adatlap azonosító vagy API Url segítségével csatolmány fájl beküldése is lehetséges, amely megjelenik az adatlapon letölthető fájlként az előzmények között.

Ahhoz, hogy fájlt tudj feltölteni, egy publikus szerverre kell kihelyezned, hogy a MiniCRM az Url segítségével letölthesse azt.

Az adatlapon létre kell hoznod egy fájl típusú mezőt.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Project/$Id`**
- Paraméter: **`$Id helyén az adatlap egyedi azonosítója`**
- HTTP metódus: PUT
- Adatok beküldése: JSON formátumban

```
$ curl -s --user $SystemId: $ApiKey -XPUT
"https://r3.minicrm.hu/Api/R3/Project/2056" -d '{"$FileColumnName”:”$FileUrl”}'
{"Id":2056}
```

- **`$FileColumnName`** helyén az adatlapon létrehozott fájl mező neve.
- **`$FileUrl`** helyén a beküldendő fájl elérhető publikus útvonala.

Sikeres mentés után az új adatlap azonosítóját Id property-ben tartalmazó json objektum a válasz.

Új adatlap rögzítése

Az adatlap módosításhoz hasonló, azzal a különbséggel, hogy elhagyjuk az adatlap azonosítóját, és az alábbi kötelező adatokat küldjük be JSON formátumban:

- “Name” - Adatlap neve
- “UserId” - Adatlap felelőse
- “CategoryId” - Modul
- “ContactId” - Kapcsolattartó
- “StatusId” - Modulban szereplő státusz

```
$ curl -s --user $SystemId:$ApiKey -XPUT "https://r3.minicrm.hu/Api/R3/Project/" -d
'{"Name":"Új adatlap","UserId":53602,"CategoryId":30,"ContactId":351,"StatusId":2867}'
{"Id": 539}
```

Sikeres mentés után az új adatlap azonosítóját Id property-ben tartalmazó json objektum a válasz.

Cég hozzáadás

Ha cég keresés nem hoz eredményt, és új céget szeretnél rögzíteni a MiniCRM-be, akkor így tudod megtenni.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact/`**
- HTTP metódus: PUT
- Adatok beküldése: JSON formátumban
- Type: Business adja meg, hogy cég jöjjön létre és a név (Name) mező kitöltése kötelező.

```
$ curl -s --user $SystemId:$ApiKey -XPUT "https://r3.minicrm.hu/Api/R3/Contact/"
-d '{"Type": "Business", "Name": "Példa Kft."}'
{"Id":7520}
```

Kontakt személy hozzáadás

Ha az új cég bekerült a rendszerbe, ugyanezen a végponton tudsz hozzá új kapcsolat tartót rögzíteni.

- API végpont: **`https://r3.minicrm.hu/Api/R3/Contact/`**
- HTTP metódus: PUT
- Adatok beküldése: JSON formátumban
- Type: Person adja meg, hogy kapcsolattartó személy jöjjön létre.
- BusinessId paraméter köti az előbb rögzített céghez a személyt (az ott kapott Id-t küldöm be).

- FirstName mező kitöltése kötelező.

```
$ curl -s --user $SystemId:$ApiKey -XPUT "https://r3.minicrm.hu/Api/R3/Contact/"
-d '{"Type": "Person", "FirstName": "Péter", "LastName": "Kovács", "Email":
"kovacs.peter@example.com", "BusinessId": 7520}'
{"Id":7522}
```

Kontakt személy végleges törlése

Egy kapcsolattartó törölhető, amennyiben egyetlen adatlapon sem elsődleges.

- API végpont: **`https://r3.minicrm.hu/Api/R3/PurgePerson/$ContactId`**
- Paraméter: **`$ContactId helyén a kapcsolattartó egyedi azonosítója`**

```
$ curl -s --user $SystemId:$ApiKey "https://r3.minicrm.hu/Api/R3/PurgePerson/7522"
```

A törlés előtt szükséges lehet, hogy váltsuk adatlap(ok) elsődleges kapcsolattartóját adatlap végponton új ContactId beküldésével.

Hibakezelés

Ha a beküldeni kívánt adatok nem megfelelőek, a szerver HTTP 400 Bad Request választ fog adni.

Parancssori kísérletezésnél alap esetben nem látszik, hogy milyen kódot (200 v más) válaszolt a MiniCRM. A -w paraméter használatával lehet hozzáfűzni bármelyik parancshoz a válasz kódot.

Hiba esetén a válasz nem JSON kódolású, hanem egy "sima" hibaüzenet.

Az alábbihoz hasonló esetben a lehetséges értékeket ki lehet deríteni a fent említett Schema végponton keresztül vagy a MiniCRM webes felületén.

```
$ curl -s --user $SystemId:$ApiKey -w "\nHTTP CODE: %{http_code}\n" -XPUT
"https://r3.minicrm.hu/Api/R3/Project/2056" -d
'{"AutoSalesV3_Qualification": "Ilyen érték nincs is CRM-ben"}'

Project (PUT): Érvénytelen választható érték 'Ilyen érték nincs is CRM-ben'
a Ügyfélminősítés mezőhöz.
HTTP CODE: 400
```

Adatlap státuszelőzmény lekérdezése

Van lehetőség lekérdezni a az adatlap státuszváltozás történetét API-n keresztül:

API végpont:

```
https://$SystemId:$ApiKey@r3.minicrm.hu/$SystemId/Api/R3/ProjectHistory/$ProjectId/Type=StatusHistory
```

$ProjectId: Adatlap azonosítója

Az adatlap jelenlegi státusza a Status_New_Id változó értéke, a régi pedig a Status_Old_Id.
A Status_Old_Name és Status_New_Name, a fenti azonosítókhoz tartozó státusz nevek a rendszerben.

Példakód:

```
curl https://$SystemId:$ApiKey@r3.minicrm.hu/$SystemId/Api/R3/ProjectHistory/29592?Type=StatusHistory
| jq
```

Válasz:


```
{
 "Count": "1",
 "Results": {

       "30027": {
       "Id": "30027",
       "ProjectId": "29592",
       "Type": "StatusHistory",
       "UserId": "53986",
       "ClosedAt": "2018-07-18 01:44:20",
       "Status_Old_Id": "",
       "Status_New_Id": "2876",
       "Status_Old_Name": "",
       "Status_New_Name": "Ügyfélminősítés"
}}}
```

Megrendelések végpont

A MiniCRM-ben található megrendelések kezelésére szolgáló API. Megrendelések keresése/listázása jelenleg nem támogatott az API által.

Új megrendelés létrehozása

Ezzel a funkcióval létrehozható egy megrendelés a rendszerbe. Hasonló adatstruktúrát vár, mint a lekérdezés (lásd: “Megrendelés lekérdezése”). A funkciót **POST** kéréssel lehet elérni. Az adatokat **JSON encode**-olt formában várja a rendszer. **CustomerId** vagy **ReferenceId** megadása kötelező!

- API végpont: **https://r3.minicrm.hu/Api/Order**
- **CustomerId**: Rendszerben található kontakt azonosító száma.
- **ReferenceId**: SyncFeed által létrehozott adatlap külső azonosító száma.

A rendeléshez a számlázási cím kerül felhasználásra. Ha ilyen cím nem létezik, akkor a kontakt címei közül az alapértelmezett. A **Customer** tömb használata opcionális, itt lehet megadni a vevő adatait. Ha a cím már létezik a kontaktnál, akkor azt használjuk, ha nem, akkor új (alapértelmezett) számlázási címként rögzül. **FONTOS!** Az adószám és bankszámlaszám mezőket ilyen esetben felülírja az API!

Azonosítás szükséges, példa url:

```
$ curl -s --user $SystemId:$ApiKey -XPOST
"http://r3.minicrm.hu/Api/Order/" -d '
{
    "CustomerId": "225",
    "CurrencyCode": "HUF",
    "Customer": {
            "Name": "Gigabit Távközlési Kft.",
           "Country": "36",
            "PostalCode": "9245",
                "City": "Mosonszolnok",
            "Address": "Alkotás u. 49.",
           "AccountNumber": "92826253-45043722-59650728",
            "VatNumber": "17823443-1-26",
            "EUVatNumber": ""
                    },
        "Subject": "",
        "Items": [
            {
```

```
            "SKU": "",
            "Name": "Optikai kábel",
            "Description": "",
            "PriceNet": "500.00",
            "Quantity": "15.00",
             "Unit": "m",
             "VAT": "27"
             }
          ]
}'
```

Válaszban visszakapjuk a rendelés összes adatát, a sikeresen létrejött megrendelés azonosítóját (Order.Id), illetve létrehozzuk a rendeléshez tartozó számított mezőket (Pl: AmountVat, AmountNet). A válasz struktúrája hasonló a “Megrendelés lekérdezése” válasz struktúrájához.

Megrendelés lekérdezése

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId**
- Paraméter: **$OrderId** egy korábban létrehozott rendelés azonosítója. **FONTOS**! A rendelés azonosító nem egyezik meg a rendeléshez tartozó projekt azonosítójával!

```
$ curl -s --user $SystemId:$ApiKey "https://r3.minicrm.hu/Api/Order/44"
{
   "Id": "44",
   "ProjectId": "206",
   "CustomerId": "225",
   "Number": "",
   "Customer": {
      "Name": "Gigabit Távközlési Kft.",
      "CountryId": "36",
      "PostalCode": "9245",
      "City": "Mosonszolnok",
      "Address": "Alkotás u. 49.",
      "AccountNumber": "92826253-45043722-59650728",
      "VatNumber": "17823443-1-26",
      "EUVatNumber": "",
      "RegistrationNumber": "32-88-574193",
      "GroupIdentificationNumber": ""
         },
   "Subject": "",
   "Performance": "2018-11-20",
```

```
    "CurrencyCode": "HUF",
    "AmountNet": "7500",
    "AmountVat": "2025",
    "Amount": "9525",
    "Items": [
          {
          "Id": "54",
          "SKU": "",
          "EAN": "",
          "Name": "Optikai kábel",
          "Description": "",
          "PriceNet": "500.00",
          "Quantity": "15.00",
          "Unit": "m",
          "PriceNetTotal": "7500.00",
          "VAT": "27%",
          "PriceTotal": "9525.00"
          }
          ],
    "Project": {
       "SzallitasExtraInfo": ""
          }
}
```


Megrendelés adatainak módosítása

Megrendelést csak “Előkészítés” státuszban lehet módosítani. **CustomerId** vagy **ReferenceId** megadása minden módosító művelethez kötelező! **CurrencyCode** értéket szerkesztéskor már nem lehet módosítani, azt a rendelés létrehozásakor kötelező megadni.

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId**
- Paraméter: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey -XPOST
"http://r3.minicrm.hu/Api/Order/44" -d '{"CustomerId": 225, "Subject": "Közvetlen szállítást kér"}'
```

Válaszban megkapjuk a rendelés összes adatát, az aktuálisan beküldött adatot (Pl: Subject) már a frissített értékkel kapjuk vissza.

Megrendelés tétel hozzáadása:

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId**
- Paraméter: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey -XPOST
"http://r3.minicrm.hu/Api/Order/44" -d
    '{
    "CustomerId": 225,
    "Items": [
        {
            "SKU": "",
            "Name": "Linksys IP telefon",
            "Description": "",
            "PriceNet": "12000.00",
            "Quantity": "3.00",
            "Unit": "darab",
            "VAT": "27"
        }
        ]
    }'
```

Válaszban megkapjuk a rendelés összes adatát, illetve frissülnek a rendeléshez tartozó számított mezők (Pl: AmountVat, AmountNet).

Példa válasz egy részlete:

```
{
. . .
"AmountNet": "43500",
"AmountVat": "11745",
"Amount": "55245",
. . .
"Items": [
{
 "Id": "54",
    "SKU": "",
 "EAN": "",
    "Name": "Optikai kábel",
 "Description": "",
    "PriceNet": "500.00",
 "Quantity": "15.00",
    "Unit": "m",
```

```
 "PriceNetTotal": "7500.00",
     "VAT": "27%",
          "PriceTotal": "9525.00"
 },
 {
 "Id": "55",
     "SKU": "",
 "EAN": "",
     "Name": "Linksys IP telefon",
 "Description": "",
     "PriceNet": "12000.00",
          "Quantity": "3.00",
          "Unit": "darab",
     "PriceNetTotal": "36000.00",
          "VAT": "27%",
          "PriceTotal": "45720.00"
 }
 ],
. . .
}
```


Megrendelés tétel szerkesztése:

Azonosító (Item.Id) megadással felülírhatjuk a hivatkozott tételsort. Például kicserélhetünk egy terméket, vagy megváltoztathatjuk a mennyiséget.

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId**
- Paraméter: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey -XPOST
"http://r3.minicrm.hu/Api/Order/44" -d
'{
"CustomerId": 225,
"Items": [
          {
"Id": 55,
          "SKU": "",
          "Name": "Cisco IP telefon",
          "Description": "",
          "PriceNet": "15000.00",
          "Quantity": "2.00",
          "Unit": "darab",
          "VAT": "27"
```

```
        }
        ]
}
}'
```

Válaszban megkapjuk a rendelés összes adatát, illetve frissülnek a rendeléshez tartozó számított mezők (Pl: AmountVat, AmountNet).
Példa válasz egy részlete:


```
{
. . .
"AmountNet": "37500",
"AmountVat": "10125",
"Amount": "47625",
. . .
"Items": [
{
 "Id": "54",
     "SKU": "",
 "EAN": "",
     "Name": "Optikai kábel",
 "Description": "",
     "PriceNet": "500.00",
 "Quantity": "15.00",
     "Unit": "m",
 "PriceNetTotal": "7500.00",
     "VAT": "27%",
         "PriceTotal": "9525.00"
 },
 {
 "Id": "55",
     "SKU": "",
 "EAN": "",
     "Name": "Cisco IP telefon",
 "Description": "",
     "PriceNet": "15000.00",
         "Quantity": "2.00",
         "Unit": "darab",
     "PriceNetTotal": "30000.00",
         "VAT": "27%",
         "PriceTotal": "38100.00"
 }
 ],
. . .
}
```

Megrendelés tétel törlése:

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId/DeleteItem/$ItemId**
- **$OrderId** egy korábban létrehozott rendelés azonosítója.
- **$ItemId** tételsor azonosítója

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/Order/44/DeleteItem/55"
```

Válaszban megkapjuk a rendelés összes adatát, illetve frissülnek a rendeléshez tartozó számított mezők (Pl: AmountVat, AmountNet).

Megrendelés egyedi mezők szerkesztése

Amennyiben a rendeléshez tartozó adatlaphoz létrehoztunk egyedi mezőket van lehetőség ezeket szerkeszteni (új rendelés, és létező rendelés esetén is, a rendelés jelenlegi státuszától függetlenül)

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId/Project**
- Paraméterek: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey -XPOST
"http://r3.minicrm.hu/Api/Order/44/Project" -d '{"SzallitasExtraInfo": "A csomagot 9 és dél között tudja átvenni"}'
```

Válaszban megkapjuk a rendelés összes adatát. Példa válasz egy részlete:

```
{
. . .
"Project": {
"SzallitasExtraInfo": "A csomagot 9 és dél között tudja
átvenni"
}
. . .
}
```

Megrendelés véglegesítése:

A megrendelést “**Előkészítés**” státuszból “**Folyamatban**” státuszúra állítja. A rendelés csak “Előkészítés” státuszban módosítható!

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId/Finalize**
- Paraméterek: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/Order/44/Finalize"
```

Megrendelés szerkesztése:

A megrendelést “**Folyamatban**” státuszból “**Előkészítés**” státuszúra állítja. A rendelés csak “Előkészítés” státuszban módosítható!

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId/Edit**
- Paraméterek: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/Order/44/Edit"
```

Megrendelés teljesítése:

A megrendelést “**Folyamatban**” státuszból “**Teljesítve**” státuszúra állítja.

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId/Complete**
- Paraméterek: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/Order/44/Complete"
```

Sikeres megrendelés:

A megrendelést “**Teljesítve**”, *vagy* “**Sikertelen**” státuszból “**Sikeres**” státuszúra állítja.

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId/Paid**
- Paraméterek: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/Order/44/Paid"
```

Sikertelen megrendelés:

A megrendelést “**Teljesítve**”, *vagy* “**Sikeres**” státuszból “**Sikertelen**” státuszúra állítja.

- API végpont: **https://r3.minicrm.hu/Api/Order/$OrderId/Storno**
- Paraméterek: **$OrderId** egy korábban létrehozott rendelés azonosítója.

```
$ curl -s --user $SystemId:$ApiKey
"https://r3.minicrm.hu/Api/Order/44/Storno"
```

A státuszváltások csak a fent meghatározott módón végezhetőek, azaz például “**Sikeres**” megrendelést már nem lehet “**Folyamatban**” státuszúra változtatni. Ilyen esetben az API

405 Method Not Allowed tipusú hibával válaszol és nem végzi el a kért műveletet.

Webhook

Nem mindig egy külső rendszer kezdeményezi a változtatást. Van, hogy a MiniCRM-ben történt változásra szeretne reagálni egy másik rendszer, pl.:

- Komplex CRM automatizálás. Webhook-on keresztül értesül egy script az új érdeklődőről/hibajegyről/projektről/rendelésről/számláról. Ez alapján számításokat végez, elemez majd egy vagy több mezőt módosít, teendőt vesz fel, stb…
- Külső rendszer szinkronizációja. Ha módosul egy adatlap MiniCRM-ben, arról értesüljön a külső rendszer is. Legyenek szinkronban az ügyféladatok, minimális késleltetéssel.

Adatlap,cég és személy módosításokat tudjuk ügyfél által megadott Api végpontra POST-olni.

Végpont HTTP 200-as válasz küldésével tudja jelezni, hogy sikeresen feldolgozta az adatokat.

Amennyiben NEM HTTP 200-at válaszol a végpont úgy vesszük, hogy hiba lépett fel a feldolgozás során és később újra próbálkozunk, összesen maximum 6 alkalommal, a következő idő-eltolással:

1. 10 másodperc
2. 1 perc
3. 5 perc
4. 15 perc
5. 30 perc
6. 1 óra.

Az idő-eltolás az előző próbálkozástól értendő, tehát adatlap módosítás után 1 óra 51 perc és 10 másodperc áll rendelkezésére a fogadó félnek, hogy feldolgozza az adatot. Ha ennyi idő alatt nem dolgozta fel, többet nem próbáljuk, a módosítást elavultnak tekintjük.

A kommunikáció mindkét irányban application/json alapú.

A végpont válaszát vissza tudjuk menteni ugyanabba az entitásba, melynek módosulásáról ment az értesítés. Fontos tudni, hogy a webhook válasza mentéséből eredő módosításokról nem lesz újabb értesítés a végpontra, elkerülve ezzel az esetleges végtelen párbeszédet. A mentés az R3 API mentéséhez hasonló, ugyanazok a limitációi.

Webhook szűrési feltételek

Túlterhelés elkerülése érdekében több paraméterrel is szűrhető, hogy pontosan mikor hívjuk meg az adott végpontot.

FONTOS: ha a meghívott program a kapott adatok alapján módosítást végez MiniCRM REST API-n keresztül, arról ugyanúgy kaphat ismételt értesítést. Így kialakulhat végtelen ciklus. Ilyen esetekben különösen körültekintően kell megválasztani a hívás feltételeit. Csak olyan mezők változásaira figyeljen a végpont, amiket az később nem módosít.

Típus

- Contact (személy és cég is)
- Project (adatlap)

Modul

Csak adatlap - Project - esetén értelmezett feltétel. Egy kiválasztott modulban történt módosításokról menjen értesítés.

Ha több modulra szeretnél értesítést kapni, de nem az összesre, akkor több Webhook-ot lehet beállítani, mindegyik modulra külön-külön egyet.

Mezők

Szűrni bizonyos mezők változásaira. Akkor hívjuk a végpontot, ha a felsorolt mezők közül legalább az egyik módosult. Ha a végponthoz nincs megadva mező, akkor a végpontot bármely mező módosulásra behívjuk.

Modul és mező filterek kombinálhatóak.

Webhook példák

Az értékesítés modul a 3-as azonosítón van és szeretném ha előfizetéskor/lemondáskor a kizárólag előfizetőimnek szóló tartalmakat kiszolgáló weboldalam értesüljön.

```
     Type: Project
CategoryId: 3
    Fields: ["StatusId"]
    ApiUrl: https://$User:$Pass@example.com/StatusChanged/
```

Az ügyfélszolgálat modul a 4-es azonosítón van és szeretném ha az új bejelentéseket egy script automatikusan elő-feldolgozná, hogy komplex automatizmusokat tudjunk megvalósítani.

```
     Type: Project
CategoryId: 4
    Fields: ["CreatedAt"]
    ApiUrl: https://$User:$Pass@example.com/TicketAdded/
```

Példa üzenet

```
{
  "Id": "10",
  "Type": "Project",
  "Data": {
    "CategoryId": "8",
    "ContactId": "16",
    "MainContactId": "15",
    "StatusId": "11684",
    "UserId": "2",
    "Name": "További kérdéseim vannak",
    "CreatedBy": "2",
    "CreatedAt": "2018-03-02 23:35:41",
    "UpdatedBy": "2",
    "UpdatedAt": "2018-03-02 23:55:42",
    "InternalUrl": "https://r3.minicrm.hu/123456/#Project-8/10"
  },
  "Changed": [
    "StatusId",
    "UserId",
    "Name",
    "UpdatedAt"
  ],
  "Config": ["Bármi beállítható, amit a megadott végpont használhat"]
  "Schema": [
    "CategoryId" => [
      16 => "Értékesítés",
      20 => "HelpDesk",
    ],
    "ContactId" => "Int",
    "MainContactId" => "Int",
    "UserId" => [
      6553 => "Olivér Gál",
      2 => "Norbert Leskó",
    ],

    "Name" => "Text(512)",
  ],
}
```

Példa válasz

```
{
    "UserId": "6553",
}
```

A fenti válasszal lehet például az adatlap felelősét Olivérre állítani.

Webhook beállítás

Jelenleg webhook-ok éles-próba üzemben mennek. Saját belső rendszereinket ennek segítségével kötöttük össze éles környezetben is.

A webes felületre azonban még nincs kivezetve a webhook beállítás. Ügyfélszolgálatunkon keresztül lehet kérni további felvilágosítást és webhook beállítást.