

Egyedi fejlesztésű webshop összekötése a MuFis rendszerrel - API készítés útmutató

Az útmutatóval kapcsolatos kérdéseit és javaslatait ide küldheti: it1@mufis.sk

Előszó

A MuFis információs rendszer számlázás, könyvelés, illetve fulfillment szolgáltatás céljából is összekapcsolható különböző webshop motorokat használó webshopokkal.

Pár konkrét ilyen webshop motor (a teljesség igénye nélkül):

- UNAS
- Shoprenter
- WooCommerce (WordPress)
- Shopify

Az összekötés lényege, hogy a webshopba beérkező megrendelések bekerüljenek a MuFis rendszerbe, ahol aztán számla készülhet a vásárlónak, fulfillment szolgáltatás esetén pedig megkezdődhessen a megrendelt áru raktári begyűjtése, csomagolása és útnak indítása. Szintén a fulfillment szolgáltatás kapcsán lényeges, hogy a csomagolás és szállítás egyes szakaszainak megfelelően a rendszer frissíteni tudja a webshopba a megrendelés állapotát, valamint hogy új áru, illetve visszáru érkezése esetén aktualizálni tudja a webshopba a raktárkészletet.

Ezeket a funkciókat a fent említett webshop motorok saját API-n keresztül biztosítják.

Az útmutató célja

Ezen útmutató célja, hogy olyan webshopok is igénybe vehessék a MuFis szolgáltatásait, melyek nem a fentebb említett webshop motorokra épülnek, hanem saját, egyedi fejlesztések. Ehhez nincs másra szükség, mint hogy a webshop képes legyen az útmutatóban foglalt specifikációk szerint értelmezni és kiszolgálni az alábbi MuFis-ből érkező request-eket:

- **megrendelés(ek) lekérése**
- **megrendelés(ek) státusz módosítása** (csak fulfillment szolgáltatás esetén szükséges)
- **termék(ek) lekérése** (csak fulfillment szolgáltatás esetén szükséges)
- **termék(ek) raktárkészletének módosítása** (csak fulfillment szolgáltatás esetén szükséges)

Előzetes egyeztetést követően beállítjuk, hogy a MuFis a webshop domain-en belül milyen útvonalra, és azon belül pontosan milyen endpoint-ra küldje az egyes request-eket a webshopnak. Az egyszerűség kedvéért az útmutatóban az alábbi endpoint megnevezéseket használjuk:

- *getOrder*
- *setOrder*
- *getProduct*
- *setProduct*

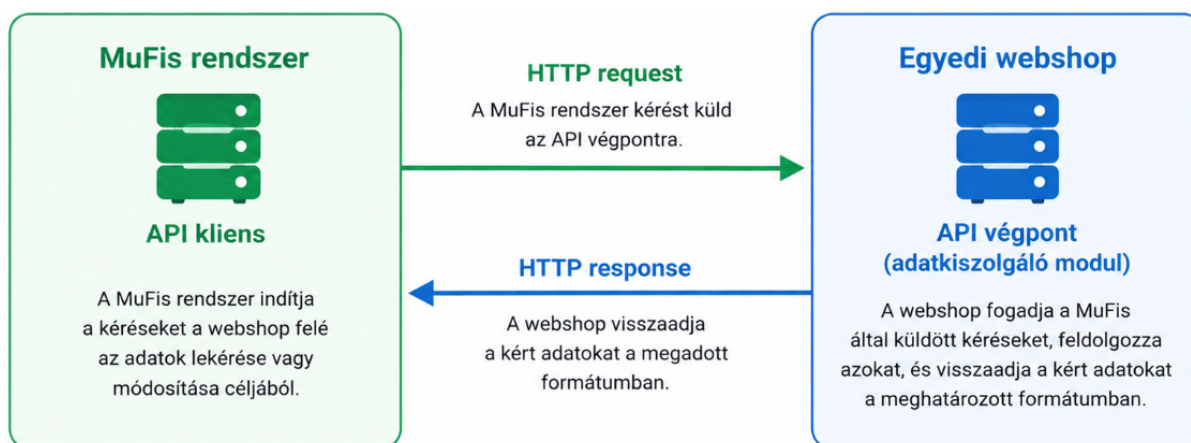
Ezek értelmében ha a domain neve *domain.hu*, az API elérési útvonal *api/mufis*, és a rendelések lekérésére a *getOrder* nevű endpoint lesz használva akkor a **https://domain.hu/api/mufis/getOrder** címre küldünk lekérést, ahol arra válaszként kell megkapnunk a kívánt adatokat.

A továbbiakban ismertetett specifikáció megvalósításának megkezdése előtt kérjük, olvassa el az alábbi fejezetet, amely ismerteti az API működési modelljét és az integráció során betöltött szerepeket.

Fontos információ az API működéséről

Fontos kiemelni, hogy a webshopban **nem API kliens** fejlesztésről van szó. A MuFis rendszer már rendelkezik a szükséges kliens oldali funkcionalitással, amely HTTP request-eket küld a webshop felé.

A kommunikáció iránya:



Az integráció során az egyedi fejlesztésű webshop feladata olyan **API végpontok (adatkiszolgáló modulok) létrehozása**, amelyek:

- fogadják a MuFis rendszer által küldött request-eket;
- feldolgozzák a kapott paramétereket;
- a jelen dokumentumban meghatározott formátumú választ adják vissza.

A dokumentációban szereplő request-ek leírásai azt mutatják be, hogy a MuFis rendszer milyen HTTP kéréseket küld az egyes végpontokra. A response-ok leírásai pedig azt határozzák meg, hogy a webshopnak milyen formátumban kell visszaadnia a kért adatokat.

Autentikáció

Minden, a MuFis rendszer által küldött HTTP request tartalmaz egy **API-KEY header**-t, amelynek alapján a webshop ellenőrizheti, hogy a kérés valóban a MuFis rendszertől érkezett. Az API kulcs értékét a két fél az integráció bevezetése előtt egyezteti.

Biztonsági okokból javasolt, hogy az API kizárólag **HTTPS** kapcsolaton keresztül legyen elérhető, így az **API-KEY** értéke az átvitel során titkosított formában továbbítódik.

Amennyiben a tárhely- vagy szerverkörnyezet lehetővé teszi, javasolt IP-cím alapú szűrés alkalmazása is, amelynek segítségével a webshop kizárólag a MuFis szervereinek IP-címéről érkező request-eket fogadja el.

Érvénytelen vagy hiányzó autentikáció esetén a webshop **401 Unauthorized** HTTP státuszkóddal válaszoljon.

Pagination, limit, találatok sorrendje

Nem kötelező, de *getOrder* és *getProduct* requestek kiszolgálásához javasunk beépíteni valamilyen felső határt (a továbbiakban **limit**, **limitáció**) arra vonatkozóan, hogy a webshop egy response-ban maximum hány a megadott szűrési feltételeknek megfelelő resource-t (megrendelést vagy terméket) adjon vissza. Ezen érték pontos meghatározása a webshopra van bízva.

Amennyiben a webshop implementálja ezt a limitációt, akkor a *getOrder* és *getProduct* request-ekre adott response-oknak mindig tartalmazniuk kell az alábbi értékeket:

érték megnevezése	érték leírása
total_pages	int típus, a request-ben megfogalmazott feltételekre adott válasz összesen hány "oldal"
page	int típus, az éppen aktuálisan küldött adat hányadik "oldal" az összesből, 1-től számozva

Amennyiben ez a két érték nem szerepel a válaszban, a MuFis úgy veszi, hogy amit kapott adat, az a feltételekben megfogalmazottaknak megfelelő összes találat.

Javasoljuk továbbá, hogy a response-ban visszaadott resource-ok mindig a webshop adatbázisának belső azonosítója szerint legyenek rendezve, csökkenő sorrendben (a legújabb rekordok legyenek a lista elején).

Request-ek struktúrája

Minden a MuFis által küldött request az alábbiakat követi:

HTTP metódus	POST
Content-Type header	application/x-www-form-urlencoded
API-KEY header	a használatra kiválasztott érték

Megjegyzés: A MuFis rendszer minden API végpontot HTTP **POST** metódussal hív meg, beleértve a kizárólag adatlekérdezésre szolgáló végpontokat is (pl. *getOrder*, *getProduct*). Ennek oka a kommunikációs interfész egységes kialakítása, ezért a webshop API-jának minden dokumentált végponton POST kéréseket kell fogadnia.

Az egyes request-ek lehetséges paramétereinek részletes leírása lejjebb, az adott request specifikációjánál találhatóak.

Response-ok várt struktúrája

Minden request **JSON** formátumban várja a webshop választ, illetve **200-as HTTP státuszú** response header-t, ami igazolja, hogy maga a kérés sikeres volt és a válasz releváns. Az egyes request-ekre várt **JSON response** részletes struktúrája lejjebb, az adott request specifikációjánál található.

getOrder request és lehetséges paraméterei

A *getOrder* a webshopba beérkező megrendelések adatainak lekérésére szolgál.

A MuFis az alábbi paramétereket küldheti **POST**-ban:

paraméter megnevezése	paraméter leírása
page	- amennyiben a webshop implementálta a limitációt, ez a paraméter jelenti azt, hogy a MuFis éppen hányadik oldalt kéri - ha nem szerepel a kérésben a page paraméter, a webshop vegye úgy, mintha az első oldalt kérné a MuFis
order_number	konkrét megrendelés lekérése a megrendelés száma alapján
order_id	konkrét megrendelés lekérése a megrendelés belső azonosítója alapján
updated_at_min	- időpont, CET (közép-európai) időzónában értve - formátum: <i>Y-m-d H:i:s</i> (pl. <i>2023-01-20 12:23:48</i>) - a paramétert tartalmazó lekérés azokat a megrendeléseket várja válaszul, melyek a megadott időpont után keletkeztek vagy módosultak a webshopban (beleértve a megadott időpontot is)
status	- csak a megadott státuszban lévő megrendeléseket kérjük - egyszerre több státusz is kérhető legyen, vesszővel elválasztva - a webshop által használt státuszok pontos megnevezéseit és azok jelentését a két fél előzetesen egyezteteti
date_from	- dátum, CET (közép-európai) időzónában értve - formátum: <i>Y-m-d</i> (pl. <i>2025-06-15</i>) - a paramétert tartalmazó lekérés azokat a megrendeléseket várja válaszul, melyeket a megadott napon, vagy attól később adtak le
date_to	- dátum, CET (közép-európai) időzónában értve - formátum: <i>Y-m-d</i> (pl. <i>2025-06-15</i>)

	a paramétert tartalmazó lekérés azokat a megrendeléseket várja válaszul, melyeket a megadott napon, vagy attól korábban adtak le
--	--

Az **order_number**, **order_id**, **updated_at_min**, **status**, **date_from** és **date_to** paraméterek mindegyikére érvényes, hogy csak akkor kell a szűrést alkalmazni, ha az adott paraméter szerepel a request-ben. Tehát ha pl. a **status** paraméter nem szerepel a lekérésben, akkor a webshop egyáltalán ne szűrjön a státuszra.

Továbbá érvényes, hogy amennyiben a kérés több lehetséges paramétert is tartalmaz, pl. jelen van a **status**, az **updated_at_min** és a **date_from** paraméter is, akkor válaszul azokat a rendeléseket várjuk, melyek MINDEN szűrési feltételnek megfelelnek.

getOrder-re adott response várt struktúrája

Limitációt implementáló webshop esetében elvárt mezők a response-ban:

total_pages

page

Ezekről részletek a **Pagination**, **limit**, **találatok sorrendje** fejezetben.

orders = a request paraméterekben megadott szűrési feltételeknek megfelelő megrendelések listája (array) 0-tól indexelve. Ha nincs találat, akkor is legyen feltüntetve, üres array-ként. Pontosan 1 találat (pl. **order_number** szerinti lekérés) esetén egy 1-elemű array.

Az **orders** lista egyes elemeinek várt struktúrája:

érték megnevezése	érték leírása
order_id	a megrendelés belső azonosítója a webshopban erre az értékre vonatkozik a <i>getOrder</i> kérésben az order_id paraméter
order_number	a megrendelés száma erre az értékre vonatkozik a <i>getOrder</i> kérésben az order_number paraméter
order_date	a megrendelés leadásának dátuma <i>Y-m-d</i> formátumban pl. 2023-01-23 erre az értékre vonatkozik a <i>getOrder</i> kérésben a date_from és a date_to paraméter
status	a megrendelés állapota erre az értékre vonatkozik a <i>getOrder</i> kérésben a status paraméter
date_mod	a megrendelés utolsó módosításának időpontja <i>Y-m-d H:i:s</i> formátumban pl. 2023-01-08 15:02:18 erre az értékre vonatkozik a <i>getOrder</i> kérésben az updated_at_min paraméter

billing_name	számlázási cím - név
billing_name2	számlázási cím - név második sora
billing_city	számlázási cím - település
billing_streetnum	számlázási cím - utca és házszám
billing_zip	számlázási cím - irányítószám
billing_country	számlázási cím - kétbetűs országcód pl. <i>HU</i> vagy <i>SK</i>
shipping_name	szállítási cím - név <i>Megj.: amennyiben a válasz nem tartalmazza a szállítási címre vonatkozó adatokat vagy azok mindegyike üres, a MuFis úgy veszi, hogy a szállítási cím megegyezik a számlázási címmel.</i>
shipping_name2	szállítási cím - név második sora
shipping_city	szállítási cím - település
shipping_streetnum	szállítási cím - utca és házszám
shipping_zip	szállítási cím - irányítószám
shipping_country	szállítási cím - kétbetűs országcód pl. <i>HU</i> vagy <i>SK</i>
email	vevő e-mail címe
phone	vevő telefonszáma, nemzetközi formátumban, szóközök nélkül pl. +36201234567 vagy +421907123456 a telefonszám elején szereplő + karakter elhagyható, vagy helyettesíthető kettő darab 0-val
eu_vat_number	vásárló EU-s adószáma pl. HU1234567890 vagy SK1234568790
lang	vásárló által használt nyelv kétbetűs kódja - számla küldése esetén a számla illetve a számlát tartalmazó e-mail szövegének nyelvét határozza meg pl. <i>hu</i> vagy <i>sk</i> vagy <i>en</i>
currency	megrendelés pénznemének 3-betűs kódja pl. <i>EUR</i> vagy <i>HUF</i>
total_price	megrendelés végösszege (beleértve mindent, tehát a szállítási díjat, ÁFA-t, kedvezményeket stb.) szóköz nélküli float típusban, tizedespontot használva, a currency -ben meghatározott pénznemben pl. 159.99
payment_type	megrendelés fizetési módja, lehetséges értékek: COD = utánvét BANK = banki átutalás

	CARD = kártyás fizetés PAYPAL = PayPal-os fizetés OTHER = egyéb nem utánvétes fizetési mód
shipping_type	a vásárló által választott szállítási mód megnevezése pl. <i>Házhozszállítás</i>
delivery_point_group	csomagpontra történő szállítás esetén a csomagpont típusa (szolgáltatója) pl. <i>packeta</i> vagy <i>gls</i> vagy <i>foxpost</i>
delivery_point_id	csomagpontra történő szállítás esetén az adott szolgáltatóhoz tartozó csomagpont azonosítója
packeta_carrier_id	Packeta csomagpontra történő szállítás esetén (tehát ha a delivery_point_group értéke packeta) itt kell megadni annak a külső Packeta partnernek az azonosítóját, amelyhez a delivery_point_id -ben megadott átvételi pont tartozik. Amennyiben a csomagpont a Packeta saját pontja vagy automatája, a mezőt üresen kell hagyni , vagy a mező ne szerepeljen a válaszban. <i>Megj.: A külső szállító partner azonosítója a Packeta widgetben a választott csomagponthoz tartozó carrierId mezőben található, azt az értéket kell itt feltüntetni.</i>
order_items	megrendelésben szereplő tételek listája (array), részletes struktúra lejjebb
meta_data	opcionális, itt bármilyen extra adat küldhető, ami a rendeléshez tartozik maga a meta_data lista típus (array), az egyes elemei pedig key és value értékpárban JSON-ban tartalmazzák az adott adat megnevezését és értékét pl. [{key: 'adat1', value: 'ertek1'} , {key: 'adat2', value: 'ertek2'}] a meta_data -ban küldött értékek jelentését, és azok Mufis-ban történő használatát a két fél előzetesen egyeztetni

order_items lista egyes elemeinek várt struktúrája:

érték megnevezése	érték leírása
sku	termék cikkszám
name	termék neve
quantity	a megrendelt mennyiség
unit_price	egységár
price_type	a feltüntetett egységár típusa, lehetséges értékek: with_vat = ha a feltüntetett egységár tartalmazza az ÁFA-t without_vat = ha a feltüntetett egységár nem tartalmazza az

	ÁFA-t megj.: ha az ÁFA százalék 0, akkor mindegy, hogy a kettő közül melyik érték
vat_rate	az ÁFA százalékban kifejezett mértéke

Amennyiben a rendelés a termékeken kívül egyéb tételeket is tartalmaz (pl. szállítási díj, utánvét kezelési díj, kedvezmény stb.), ezeket a tételeket szintén az **order_items** listában kell szerepeltetni, a megadott struktúrában.

setOrder request és lehetséges paraméterei

(csak fulfillment szolgáltatás esetén szükséges implementálni)

A *setOrder* request egy megrendelés státuszának módosítására szolgál, továbbá ha már csomagszámmal rendelkező megrendelésről van szó, akkor ezt a csomagszámot is elküldi a webshopnak, hogy az elmenthesse azt. Más adatot a MuFis nem módosít a megrendelésekben. A request lehetséges **POST** paraméterei:

paraméter megnevezése	paraméter leírása
order_number	a frissíteni kívánt megrendelés száma
status	az adott megrendelés új státusza
package_number	a megrendeléshez tartozó csomagszám több csomag esetén JSON lista az összes a rendeléshez tartozó csomagszámról csak akkor szerepel paraméterként, ha már regisztrálva van a csomag a szállítást végző futárszolgálatnál
tracking_link	a megrendeléshez tartozó csomag nyomkövetési linkje a szállítást végző futárszolgálatnál több csomag esetén JSON lista az összes a rendeléshez tartozó csomag linkjével csak akkor szerepel paraméterként, ha már regisztrálva van a csomag a szállítást végző futárszolgálatnál
multiple_packages	1 vagy 0, attól függően, hogy 1 vagy több csomag készült a megrendeléshez (nagy megrendelés esetén előfordulhat, hogy nem fér be 1 csomagba) ha ennek az értéke 0, akkor a package_number és tracking_link értékek string-ként tartalmazzák az adatokat ha ennek az értéke 1, akkor a package_number és tracking_link értékek JSON listában tartalmazzák az adatokat

Az adatforgalom csökkentése érdekében a webshop implementálhatja a *setOrder* funkció olyan verzióját is, amely egyszerre több megrendeléshez tartozó adatot is képes feldolgozni. Az ilyen köteget *setOrder* request struktúrája csak annyiban tér el az előzőekben leírtaktól, hogy a **POST**-ban csak egy **data** paraméter szerepel, és ennek az értéke pedig egy **JSON** formátumú lista, melynek egyes elemei a fenti struktúrát követik. A két fél egyezteteti, hogy a

webshop melyik verzióját implementálja a funkciónak, a MuFis pedig ez alapján fogja küldeni a request-eket.

setOrder-re adott response várt struktúrája

érték megnevezése	érték leírása
ok	a küldött adatok sikeres beírása esetén 1-es érték, ellenkező esetben 0
error	hiba esetén (ok = 0) ez a mező tartalmazhat egy szöveges leírást a hibáról
order_number	csak kötegetelt request esetén, ez a mező jelzi, hogy az adott ok és error melyik megrendelésre vonatkozik

Kötegetelt request esetén a response egy **orders** mező legyen, mely listaként tartalmazza a fent említett **ok**, **error** és **order_number** mezőket - minden a requestben szereplő megrendeléshez 1 elemet.

getProduct request és lehetséges paraméterei

(csak fulfillment szolgáltatás esetén szükséges implementálni)

A *getProduct* a webshopban árusított termékek adatainak lekérésére szolgál.

A MuFis az alábbi paramétereket küldheti **POST**-ban:

paraméter megnevezése	paraméter leírása
page	- amennyiben a webshop implementálta a limitációt, ez a paraméter jelenti azt, hogy a MuFis éppen hányadik oldalt kéri - ha nem szerepel a kérésben a page paraméter, a webshop vegye úgy, mintha az első oldalt kérné a MuFis
product_id	konkrét termék lekérése a termék belső azonosítója alapján
sku	- csak a megadott cikkszámú terméket kérjük - egyszerre több termék is kérhető legyen, a több cikkszámot vesszővel elválasztva küldjük
active	- értéke 1 vagy 0, azt fejezi ki, hogy csak az aktív vagy csak az inaktív termékeket kéri a MuFis - aktív termék = jelenleg árulják a webshopban (raktárkészlettől függetlenül) - inaktív termék = létezik a terméklistában, de jelenleg nincs árusítva a webshopban

	• ha nem szerepel a requestben ez a paraméter, akkor nem kérünk szűrést eszerint a tulajdonság szerint • ha a webshop nem különbözteti meg ilyen módon a termékeit, hagyja figyelmen kívül ezt a paramétert
--	--

Ha a request-ben nem szerepel sem a **product_id** sem a **sku** paraméter (tehát nem konkrét termék(ek) adatait kérjük le), akkor az összes termék adatait várjuk a válaszban..

getProduct-ra adott válasz várt struktúrája

Limitációt implementáló webshop esetében elvárt mezők a response-ban:

total_pages

page

Ezekről részletek a **Pagination, limit, találatok sorrendje** fejezetben.

products = a request paraméterekben megadott szűrési feltételeknek megfelelő megrendelések listája (array) 0-tól indexelve. Ha nincs találat, akkor is legyen feltüntetve, üres array-ként. Pontosan 1 találat (pl. **sku** szerinti lekérés) esetén egy 1-elemű array.

A **products** lista egyes elemeinek várt struktúrája:

érték megnevezése	érték leírása
product_id	a termék belső azonosítója a webshopban
sku	a termék egyedi cikkszáma
name	a termék neve
active	1 vagy 0, attól függően, hogy az adott terméket jelenleg árulja-e a webshop (raktárkészlettől függetlenül) ha a webshop nem különbözteti meg ilyen módon a termékeket, ebben a mezőben szerepeljen mindig 1-es
stock_quantity	a webshop adatbázisa szerint hány darab van jelenleg ebből a termékből raktáron
barcode	a terméken fellelhető vonalkód

setProduct request és lehetséges paraméterei

(csak fulfillment szolgáltatás esetén szükséges implementálni)

A **setProduct** request egy termék raktárkészletének aktualizálására szolgál. A request lehetséges **POST** paraméterei:

paraméter megnevezése	paraméter leírása
sku	az aktualizálni kívánt termék cikkszáma

stock_quantity	az aktuális raktáron lévő mennyiség
-----------------------	-------------------------------------

Az adatforgalom csökkentése érdekében a webshop implementálhatja a *setProduct* funkció olyan verzióját is, amely egyszerre több termékhez tartozó adatot is képes feldolgozni. Az ilyen kötegetelt *setProduct* request struktúrája csak annyiban tér el az előzőekben leírtaktól, hogy a **POST**-ban csak egy **data** paraméter szerepel, és ennek az értéke pedig egy **JSON** formátumú lista, melynek egyes elemei a fenti struktúrát követik. A két fél egyezteteti, hogy a webshop melyik verzióját implementálja a funkciónak, a MuFis pedig ez alapján fogja küldeni a request-eket.

setProduct-ra adott response várt struktúrája

érték megnevezése	érték leírása
ok	a küldött adatok sikeres beírása esetén 1-es érték, ellenkező esetben 0
error	hiba esetén (ok = 0) ez a mező tartalmazhat egy szöveges leírást a hibáról
sku	csak kötegetelt request esetén, ez a mező jelzi, hogy az adott ok és error melyik termékre vonatkozik

Kötegetelt request esetén a response egy **products** mező legyen, mely listaként tartalmazza a fent említett **ok**, **error** és **sku** mezőket.

Megrendelések automatikus importja a MuFis rendszerbe

Amennyiben a webshop kizárólag a MuFis számlázási funkcióját veszi igénybe, elegendő a MuFis felületén beállítani, hogy a rendszer naponta egyszer automatikusan beolvassa a webshopból az új megrendeléseket, majd azok alapján elkészítse a számlákat.

Fulfillment szolgáltatás esetén azonban fontos, hogy az új vagy módosított megrendelések a lehető legrövidebb időn belül megjelenjenek a MuFis rendszerben. Ennek érdekében javasolt a webshopban egy **webhook** létrehozása, amely új megrendelés létrehozásakor, illetve meglévő megrendelés módosításakor HTTP kérést küld a MuFis szerverének.

A webhook kizárólag arra szolgál, hogy értesítse a MuFis rendszert arról, hogy a webshopban változás történt. A kérésnek **nem szükséges sem paramétereket, sem request body-t** tartalmaznia, mivel a webhook URL-je már eleve tartalmazza azokat az azonosítókat, amelyek alapján a MuFis egyértelműen azonosítani tudja az érintett céget és a hozzá tartozó webshopot.

A MuFis rendszer a webhook fogadását követően automatikusan elindít egy importfolyamatot, amelynek során a *getOrder* végpontot meghívva, az **updated_at_min** paraméter segítségével lekéri a webshopból az utolsó import óta létrejött vagy módosított megrendeléseket, majd elvégzi a szükséges szinkronizációt.

A webhook pontos URL-jét az integráció megkezdése előtt biztosítjuk.

Amennyiben a webshop nem kíván webhookot implementálni, a MuFis munkanapokon munkaidőben óránként automatikusan lekéri a webshopból az új és módosított megrendeléseket. Ebben az esetben a változások legfeljebb egy órán belül jelennek meg a MuFis rendszerben.

Megrendelés törlése

Előfordulhat, hogy egy korábban leadott megrendelést mégsem kell teljesíteni, ezért azt törölni szükséges.

Fontos, hogy a webshop a megrendelést ne törölje fizikailag az adatbázisból, hanem állítsa egy előre egyeztetett „törölt” státuszba. Ennek oka, hogy a MuFis a *getOrder* végpont segítségével kizárólag a webshop által visszaadott megrendelésekről értesül.

Amennyiben a megrendelés törölt státuszban szerepel, a MuFis a következő szinkronizálás során átveszi ezt a módosítást, és saját adatbázisában is töröltként jelöli a megrendelést.

Ha a webshop a megrendelést véglegesen eltávolítja az adatbázisból, a MuFis a következő szinkronizálás során már nem kap róla semmilyen információt. Ennek következtében a MuFis adatbázisában a megrendelés továbbra is aktív állapotban maradhat, ami hibás számlázást vagy fulfillment szolgáltatás esetén a megrendelés téves teljesítését is eredményezheti.

A „törölt” státusz pontos megnevezését és értékét a két fél az integráció bevezetése előtt egyezteti.

change log

2026-08-05 - v1.2.2

- Új fejezet: *"Fontos információ az API működéséről"*
- Pontosításra kerültek a következő fejezetek_
 - *„Megrendelések automatikus importja a MuFis rendszerbe”*
 - *„Megrendelés törlése”*

2026-04-01 - v1.2.1

- `packeta_carrier_id` a *getOrder* válaszban

2026-03-06 - v1.2

- `order_items` pontosítása a *getOrder* válaszban

2025-07-17 - v1.1

- `date_from` és `date_to` paraméterek a *getOrder* kérésben
- `meta_data` a *getOrder* válaszban

2023-01-24 - v1.0