Maksettujen verkkokauppatilausten muuttaminen kilpailun osallistujalistaksi ilman koodausta

1 · 2026-09-19

Kiipeilysali järjestää kilpailun ja myy osallistumismaksuja kuten mitä tahansa muuta tuotetta: rivi verkkokaupassa, hinta ja ehkä ikäryhmäkohtainen kategoria. Ilmoittautumiset saapuvat kolmen viikon aikana. Sitten, edellisenä iltana, joku vie tilaukset ulos järjestelmästä, suodattaa ne käsin ja liittää nimet taulukkolaskentaan, jotta ilmoittautumispisteellä on lista.

Juuri tuo viimeinen vaihe kannattaa poistaa. Ei siksi, että taulukkolaskenta olisi huono, vaan siksi, että kopiointi tapahtuu kerran, hieman liian myöhään, datasta, joka on muuttunut kolmen viikon ajan. Viimeisenä iltana ilmoittautuneet puuttuvat siitä.

Webhook korjaa ajoituksen, eikä työkalua. Sali pitää taulukkolaskentansa; taulukko täyttyy itsestään jokaisen maksetun ilmoittautumisen myötä.

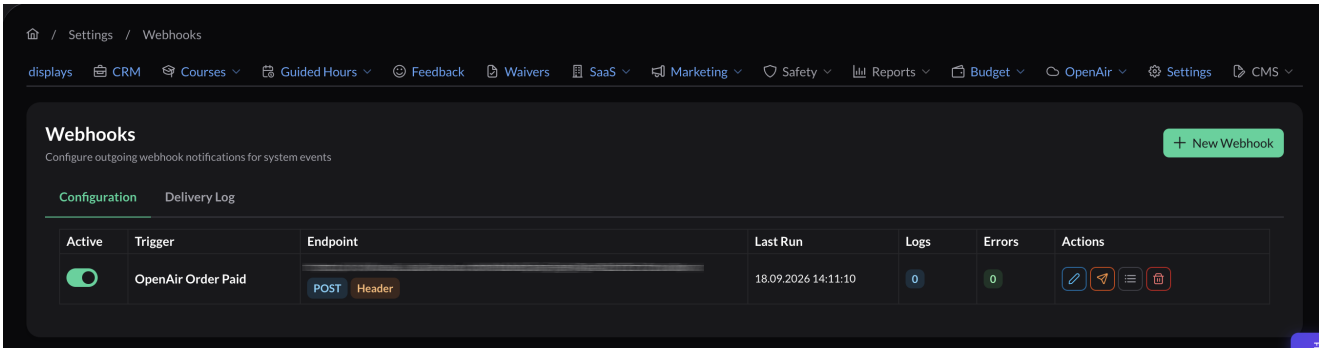
Mitä webhook tarkoittaa salinhallintajärjestelmässä?

Webhook on HTTP-pyyntö, jonka GymKeeper lähettää valitsemaasi URL-osoitteeseen heti, kun jokin tapahtuma toteutuu. Mikään ei pollaa, mitään ei ajasteta eikä sinulla tarvitse olla pääsyä tietokantaan. Annat GymKeeperille osoitteen, ja se lähettää sinne JSON-datan tapahtuman sattuessa.

GymKeeper laukaisee webhookeja tilauksille, jäsenyyksille, asiakkaille, laskuille, kursseille, lipuille ja ovikulunvalvonnalle. Tilauksille on viisi tapahtumaa — luotu, maksettu, maksu peruutettu, hylätty ja toimitettu — ja jokainen laukeaa kolmella nimellä, joten voit tilata tapahtumat yhdestä kaupasta tai kaikista kerralla:

Kaikki kaupat	OpenAir	Verkkokauppa
`order_paid`	`openair_order_paid`	`webshop_order_paid`

Kilpailun kannalta oleellinen on **maksettu**. Olemassa oleva tilaus ei ole ilmoittautuminen; maksettu tilaus on. Tilaamalla `order_created`-tapahtuman täyttäisit taulukon hylätyin ostoskorein.



Mitä tilauksen maksamisen webhook lähettää?

Payload kuvaa tilauksen selkokielisin nimin eikä tietokantasarakkeina:


```
{
  "event": "paid",
  "channel": "openair",
  "fired_at": "2026-09-18T14:11:10+00:00",
  "order": {
    "id": 25315,
    "total": 55,
    "paid": true,
    "paid_at": "2026-09-18 14:11:07",
    "payment_method": "paytrail",
    "status": 101,
    "status_key": "payment_accepted",
    "is_test": false
  },
  "customer": {
    "id": 982,
    "firstname": "Jarmo",
    "lastname": "Annunen",
    "email": "jarmo@annunen.fi",
    "phone": "0401234567"
  },
  "rows": [
    {
      "id": 37909,
      "title": "Boulder Battle 2026 – Adult",
      "product_id": 17,
      "feature_id": 17,
      "quantity": 1,
      "unit_price": 55,
      "vat_percent": 14,
      "total": 55
    }
  ]
}
```


Kaksi kenttää määrittää koko rakenteen. **`rows` on taulukko** — yksi tilaus voi sisältää kilpailuilmoittautumisen, magnesiumpussin ja päivälipun. Ja **`channel`** kertoo, mistä kaupasta tilaus tuli, joten yksi skenaario voi palvella molempia kauppoja ilman arvausta.

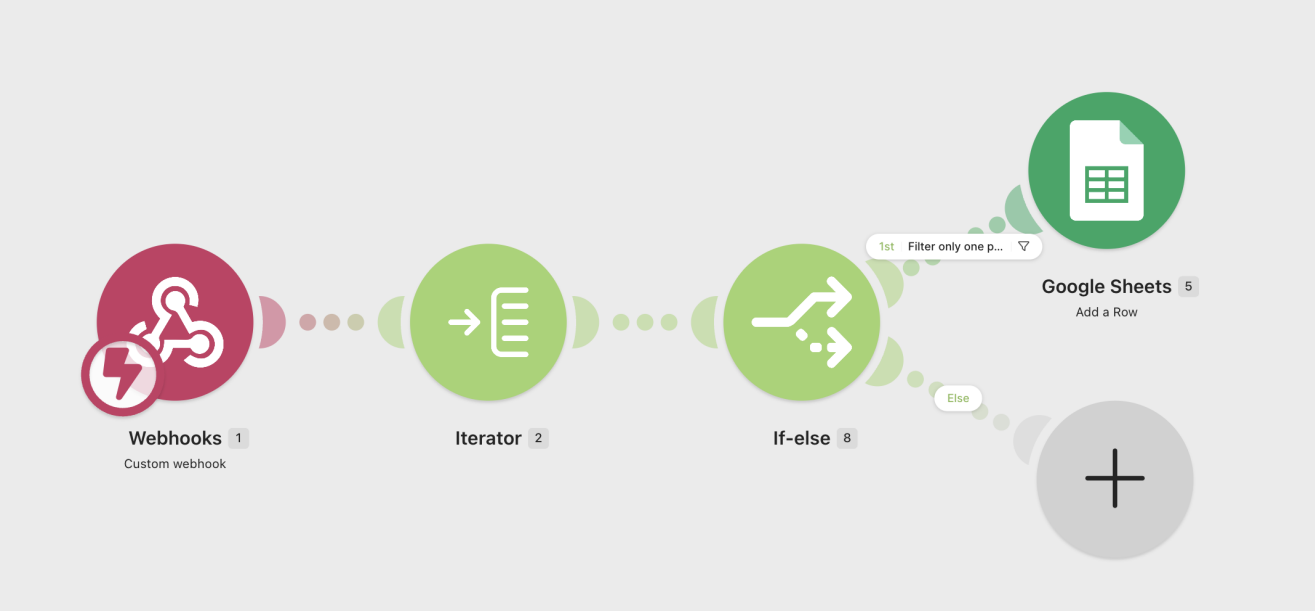
Miten saat yhden rivin kilpailijaa kohti Google Sheetsiin?

Ongelma on muodoltaan se, että webhook saapuu **tilausta** kohti, mutta taulukko haluaa rivin **ilmoittautumista** kohti. Nämä eivät ole sama määrä. Make.comissa ketju koostuu kolmesta moduulista:

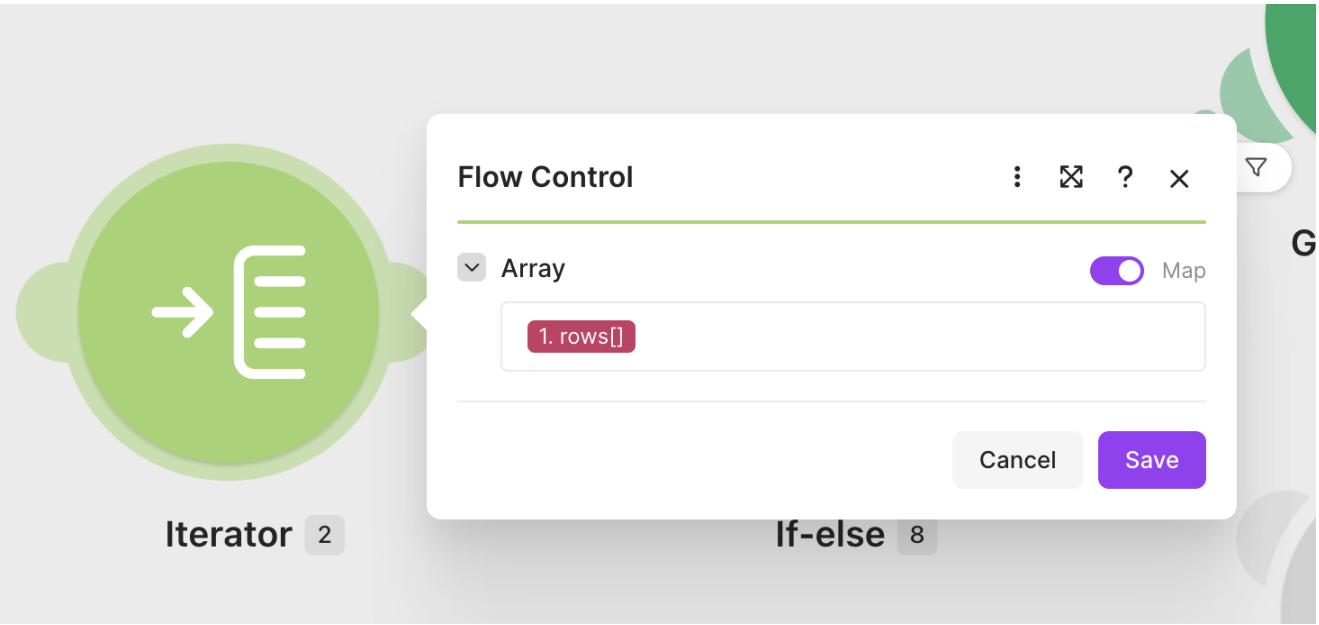
1. **Custom webhook** — Make antaa sinulle URL-osoitteen. Liitä se GymKeeperiin

`openair_order_paid`-tapahtuman endpointiksi. Lähetä yksi tilaus läpi, jotta Make oppii tietorakenteen.

2. **Iterator** — osoita se `rows`-taulukkoon. Kaikki tämän jälkeen suoritetaan kerran riviä kohti eikä kerran tilausta kohti, mikä muuttaa kolme ilmoittautumista ostavan perheen kolmeksi riviksi.
3. **Suodatin**, sitten **Google Sheets → Add a row**. Suodatin päättää, onko käsiteltävä rivi kilpailuilmoittautuminen; Sheets-moduuli kirjoittaa sen.



Mappaa kilpailijan nimi ja sähköposti `customer`-kentästä — joka pysyy käytettävissä iteratorin jälkeenkin, koska se kuuluu tilaukseen — ja kategoria rivin `title`-kentästä. Lisää `order.paid_at`, jos pisteen täytyy selvittää, kuka ilmoittautui ensimmäisenä kiintiölliseen kategoriaan.

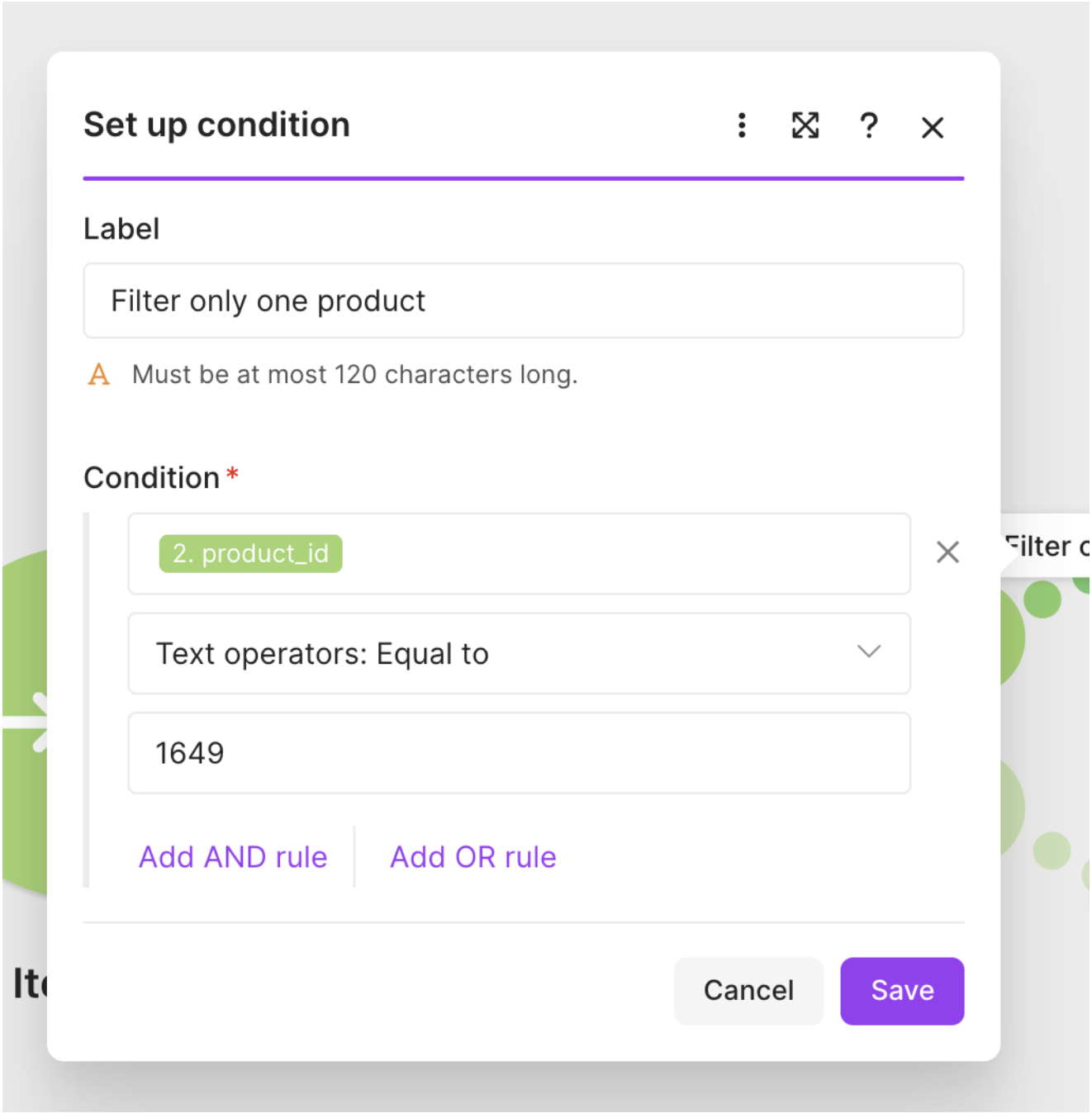


Miten suodatin tunnistaa kilpailuilmoittautumisen?

Käytä **`product_id`-kenttää**, älä otsikkoa.

Otsikko on tuotteen nimi sellaisena kuin se oli myyntihetkellä, ja tuotenimiä muokataan: "Boulder Battle 2026" muuttuu muotoon "Boulder Battle 2026 — Adult", kun nuorisokategoria lisätään, ja suodatin, joka tarkistaa `title contains "Boulder Battle 2026"`, toimii oikein aina siihen asti, kunnes joku nimeää sen uudelleen muotoon "Boulder Battle '26". Virhe on äänetön. Ilmoittautumiset saapuvat edelleen, skenaario palauttaa onnistumisen, ja taulukko yksinkertaisesti lakkaa kasvamasta.

`product_id` on tuotteen tunniste GymKeeperissä eikä muutu nimen vaihtuessa. Katso se kerran tuoteeditorista ja suodata numeron perusteella. Jos kilpailulla on useita kategorioita erillisina tuotteina, suodata id-joukon perusteella ja anna otsikon viedä kategoria taulukkoon — näin taulukko ryhmittyy oikein ja suodatin selviää silti uudelleennimeämisestä.



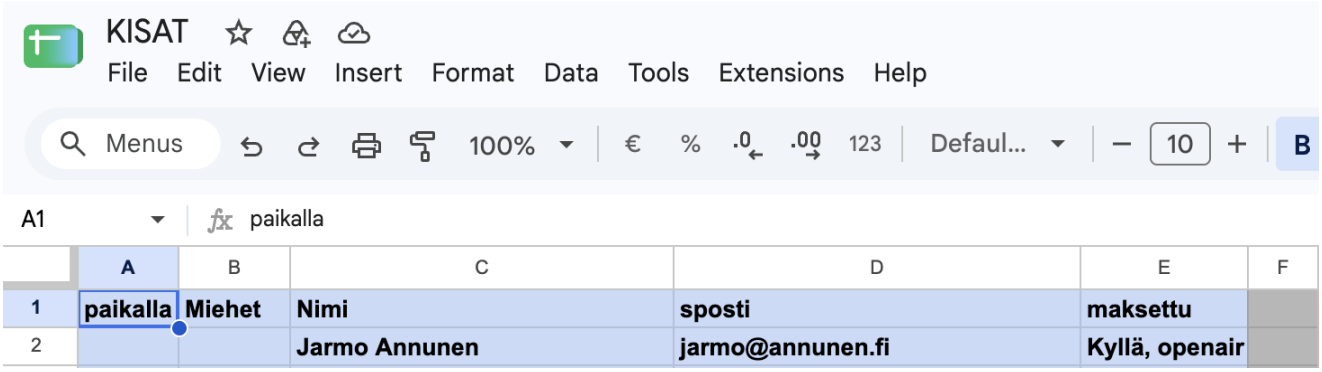
Neljä asiaa, jotka tulevat yllättämään sinut

Testitilaukset näyttävät täsmälleen oikeilta. Jokainen payload sisältää `is_test`-kentän, joka asetetaan tilauksen oman testitunnisteen mukaan. Jos kokeilet virtausta testimaksulla — ja sinun pitäisi — kyseinen ilmoittautuminen päätyy taulukkoon oikeiden joukkoon, eikä kukaan huomaa sitä ennen kuin ilmoittautumispisteellä törmätään nimeen, joka ei kuulu kellekään. Lisää `is_test = false` suodattimeen ennen kilpailun avaamista, ei sen jälkeen.

Toimitus on yksittäinen yritys. GymKeeper lähettää pyynnön kerran ja kirjaa tuloksen; automaattista uudelleenlähetystä ei tapahdu, jos endpointisi on alhaalla tai hidas. Tallennettua uudelleenyrityskertojen määrää ei tällä hetkellä noudateta. Make.comilla on oma virheenkäsittely ja jonottaminen, joten anna sen hoitaa uudelleenyritys: skenaariosta, joka epäonnistuu, voi toipua Make'n historianäkymästä, mutta webhookia, jota ei koskaan hyväksytty, ei voi palauttaa. Tarkista `webhooks_log` hallintapaneelista, jos olet epävarma siitä, toimitettiinko jokin pyyntö — se kirjaa jokaisen yrityksen vastauksen ja tilauskoodin.

Nollan minuutin viive asettaa endpointisi maksun uudelleenohjauksen sisälle. Ilman viivettä GymKeeper lähettää pyynnön samalla, kun asiakasta ohjataan takaisin maksuntarjoajalta, joten hidas endpoint saa heidät odottamaan "maksu hyväksytty" -sivulla. Yhden minuutin viiveen asettaminen siirtää toimituksen ajastettuun tehtävään, eikä asiakkaan uudelleenohjaus enää riipu automaatiostasi. Taulukon kannalta, jota ihminen lukee seuraavana aamuna, viive ei maksa mitään.

Yksi webhook, kaksi kauppaa. Jos sali myy sekä OpenAirin että verkkokaupan kautta, pelkästään `openair_order_paid`-tapahtumaan tilaaminen jättää hiljaa puolet ilmoittautumisista huomiotta. Tilaa joko yhdistetty `order_paid` ja lue `channel`-kenttä, jos sinun täytyy erottaa ne toisistaan, tai tilaa molemmat kauppakohtaiset tapahtumat. Ennen kauppakanavan tallentamista tehdyt tilaukset lasketaan verkkokaupan tilauksiksi.



	A	B	C	D	E	F
1	paikalla	Miehet	Nimi	sposti	maksettu	
2			Jarmo Annunen	jarmo@annunen.fi	Kyllä, openair	

Endpointin suojaaminen

Webhook-URL on julkinen osoite, ja kuka tahansa sen tietävä voi lähettää sinne pyyntöjä. GymKeeper tarjoaa neljä tapaa todistaa, että pyyntö on aidosti peräisin saliltasi: mukautettu otsikko, kyselyparametri, bearer-token tai basic auth — asetetaan tilauksittain endpointin yhteydessä. Make.com voi vaatia vastaavan otsikon omalla puolellaan. Kilpailulistan kohdalla riskit ovat pienet; jos kirjoitat taloushallintajärjestelmään, aseta jokin näistä.

Mihin tämä malli sopii

Samat kolme moduulia — iteroi rivit, suodata tuotteen mukaan, kirjoita jonnekin — kattavat kokonaisen joukon tehtäviä, joista muuten tulee käsityötä juuri väärällä hetkellä:

- kurssin tai klinikan osallistujalista, joka täyttyy paikkojen maksamisen myötä
- ryhmävarauksen vuokrausvastuulista
- kenkäkokolista aloittelijoiden tunnille, jossa koko on tuotevariantti
- kaupan tavarankeräyslista, suodatettuna fyysisiin tuotteisiin

Mikään näistä ei vaadi koodia. Ne tarvitsevat tapahtuman, joka saapuu samalla hetkellä kuin raha.

Usein kysytyt kysymykset

Toimiiko tämä myös verkkokaupassa OpenAirin lisäksi? Kyllä. Tilaa `webshop_order_paid`, tai `order_paid` molemmille kaupoille kerralla. Payload on identtinen lukuun ottamatta `channel`-kenttää.

Mitä tapahtuu, jos joku peruuttaa tai pyytää hyvitystä maksamisen jälkeen? Maksettu webhook on jo lauennut, joten rivi on jo taulukossa. Tilaa `order_payment_cancelled`, jos tarvitset myös toisen puolen tarinasta; henkilökunnan käsittelemä hyvitys muuttaa tilauksen statusta, jota voit seurata `order_delivered`-tapahtuman ja status-kentän kautta sen sijaan, että olettaisit taulukon pysyvän itsestään oikeana.

Voinko lähettää tämän jonnekin muualle kuin Make.comiin? Kyllä. Webhook on tavallista JSON-dataa HTTP:n yli, joten Zapier, n8n, Google Apps Script -verkkosovellus tai oma endpointisi toimivat samalla tavalla. Make.comia käytetään tässä, koska sen iterator tekee yksi-rivi-per-ilmoittautuminen-vaiheen näkyväksi.

Tarvitsenko kehittäjän? Ei. Kaikki yllä oleva on konfigurointia: URL liitetään GymKeeperiin, kolme moduulia rakennetaan Make.comiin ja product id kopioidaan tuoteeditorista.

GymKeeper on kiipeilysalien hallintaohjelmisto, jonka on kehittänyt GymKeeper Oy Suomessa. Se kattaa jäsenyydet, laskutuksen, kulunvalvonnan, kassan, verkkokaupan ja integraatiot. Jos olet uusi laajemman järjestelmän käyttäjä, aloita lukemalla mitä kiipeilysalin hallintaohjelmisto kattaa, tai katso, miten jäsenyydet ja sarjakortit myydään saman verkkokaupan kautta, josta nämä tilaukset tulevat.