



MB 29/04/2024
GEREGISTREERD KASSASYSTEEM –
TECHNISCHE VOORSCHRIFTEN –
DETAILBESCHRIJVING VAN DE
WERKING EN DE COMMUNICATIE
TUSSEN HET KASSASYSTEEM EN DE
FISCAL DATA MODULE

VERSIE 1.8 – 10/02/2025



CHANGELOG

Versie	Datum	Samenvatting
1.0	26/06/2024	Originele tekst
1.1	18/07/2024	<p>Foutcorrecties:</p> <ul style="list-style-type: none"> - Input SaleInput: fdmRefs (Hfdst. 2, 2.3.4) - lineTotal: opmerking composite product – formule doorsturen naar FDM (Hfdst. 2, 2.2.2) - type MessageItem: message (Hfdst. 2, 2.3.2) - input ReportUserXInput: reportBookingDate tweemaal (Hfdst. 2, 2.3.4) - input InvoiceInput: invoegen costCenter (Hfdst. 2, 2.3.4) - whitelist vervangen door allowlist (Hfdst. 2, 2.2.4 en 2.3.3.) <p>Verduidelijkingen</p> <ul style="list-style-type: none"> - Fysieke verbinding POS – FDM (Hfdst 2, 1.) - Object fdmRefs – (Hfdst. 2, 2.2.2) - lineTotal – (Hfst. 2, 2.2.2)
1.2	06/09/2024	<p>Foutcorrecties:</p> <ul style="list-style-type: none"> - typo's: ud pate (update), ON HOLD (ON_HOLD) <p>Aanvullingen:</p> <ul style="list-style-type: none"> - tb_messages - enum Code - copyOfEvent (definitie) <p>Aanpassingen:</p> <ul style="list-style-type: none"> - warnings en errors - message object (beschrijving en type system) - price change (groupingId en scope) <p>Verduidelijkingen:</p> <ul style="list-style-type: none"> - posID - fysieke verbinding POS-FDM
1.3	20/09/2024	<p>Verduidelijkingen:</p> <ul style="list-style-type: none"> - voorlopige rekening <p>Aanpassingen:</p> <ul style="list-style-type: none"> - voorbeeld en tekst object message in 2.2.5. - verwijderen item in tb_messages en in corresponderende enum
1.4	04/10/2024	<p>Aanpassingen:</p> <ul style="list-style-type: none"> - showPos: Boolean → enum - tb_messages: kolom display + unauthorized → unknown POS <p>Aanvullingen:</p> <ul style="list-style-type: none"> - enum Display - footerLigns in signResult (conform API PROT) <p>Verduidelijkingen:</p> <ul style="list-style-type: none"> - 2.4.2. quid null waarden - Product (geen boekingen op departementsniveau)
1.5	29/10/2024	<p>Aanpassingen</p> <ul style="list-style-type: none"> - unitPrice: 2 → 4 decimalen - transactionLines : verbeteringen aan formules - 1.8 : niet afdrucken 'dit is geen ...' bij kopie btw-kasticket + fdmRef

Versie	Datum	Samenvatting
1.6	06/12/2024	<p><u>Aanvullingen</u></p> <ul style="list-style-type: none"> - Punt 6.6: controlegegevens op btw-kasticket <p><u>Aanpassingen</u></p> <ul style="list-style-type: none"> - reportBookingDate: niet langer verplicht op X-rapport - communicatieregels (onder 2.2.2): vervangen request door mutatie <p><u>Verduidelijkingen</u></p> <ul style="list-style-type: none"> - message: wanneer en aantal keer tonen via UI POS - QR-code ingeval kopie van btw-kasticket - Proces flow – meldingen <p><u>Correcties</u></p> <ul style="list-style-type: none"> - transactions object: amount: verwijderen 'per departementId' - departements object: amount: formule gecorrigeerd - turnover object: overnemen benamingen uit type system <ul style="list-style-type: none"> o reason → negQuantityReason o count → negQuantityCount
1.7	09/12/2024	<p><u>Aanvulling</u></p> <ul style="list-style-type: none"> - terug toevoegen UNAUTHORIZED in tb_messages en enum <p><u>Correcties</u></p> <ul style="list-style-type: none"> - teveel request automatisch vervangen door mutatie (enkel NL-versie) <p><u>Aanpassingen</u></p> <ul style="list-style-type: none"> - footerLigns → footer - lineTotal: 2 decimalen - reportNo en reportBookingDate → enkel bij Z rapport (niet op X) – ook aangepast in type system <p><u>Verduidelijkingen</u></p> <ul style="list-style-type: none"> - message: bijvoegen 'minstens' bij optional
1.8	10/02/2025	<p><u>Verduidelijkingen:</u> Hoofdstuk1, 8.1 Afdrukken: wat moet op een kopie van een BTW kasticket afgedrukt worden (papier/digitaal)</p>

MB 29/04/2024	i
GEREGISTREERD KASSASYSTEEM – TECHNISCHE VOORSCHRIFTEN –	i
DETAILBESCHRIJVING VAN DE WERKING EN DE COMMUNICATIE TUSSEN het KASSASYSTEEM EN DE FISCAL DATA MODULE.....	i
INLEIDING.....	1
HOOFDSTUK 1 – TECHNISCHE EISEN KASSASYSTEEM	2
1. Registratie van events	2
1.1. Directe verkoop.....	2
1.2. Verkoopregistratie met volledige terugname eerder uitgereikt btw kasticket.....	2
1.3. Onderbroken verkoop	2
1.4. Het rekeningoverzicht (voorlopige rekening)	3
1.5. Registraties in trainingsmodus	3
1.6. Registraties van pure financiële verrichtingen.....	3
1.7. Aanmaak van facturen op basis van eerder aangemaakte events N.....	3
1.8. Kopieën van eerder aangemaakte events.....	3
1.9. Registraties van de aanwezigheid van het personeel.....	4
2. Nummering events	4
3. Voorwaarden registratie events	4
4. Bewaring gegevens.....	5
5. BTW-codes	6
6. Het btw-kasticket.....	7
6.1. Identificatie van het kassasysteem.....	7
6.2. Identificatie van de terminal	7
6.3. Identificatie van het input device	7
6.4. Identificatie van de gebruiker	7
6.5. QR-code	7
6.6. Controlegegevens op het btw-kasticket.....	8
6.7. Algemeen	8
7. Afrondingen van betalingen	8
7.1. De uitbater rondt enkel contante betalingen af.....	9
7.2. De uitbater rondt alle betaalmiddelen af.....	9
8. consolidatie- en afdrukregels.....	9
8.1. Afdrukken	10
8.2. Consolidatie	10
HOOFDSTUK 2 – WISSELWERKING POS - FDM	11
Definities en afkortingen.....	11
1. fysieke verbinding.....	14
2. communicatie tussen POS en FDM.....	15

2.1. Algemeen	15
2.2. Inhoud van de communicatie POS - FDM	17
2.2.1. POS → FDM – volledig overzicht	17
2.2.2. POS → FDM - detailbeschrijving	18
Taal	18
Afleverwijze	18
Pos Identificatiegegevens	18
Pos Datum en Tijd	19
Identificatiegegevens Onderneming & Gebruikers	19
Referenties	20
Registratie Transacties – Gebruik van Beheersmogelijkheden	20
Registratie Transacties - Transactielijnen	22
Registratie Transacties – Financiële Bewegingen	27
Rapporten	30
Communicatieregels	36
2.2.3. FDM → POS – volledige overzicht	37
2.2.4. FDM → POS – gedetailleerd overzicht	37
Message	40
2.2.5. FDM → POS errorhandling	43
2.3. GraphQL schema communicatie tussen POS en FDM	44
2.3.1. Mutations	44
2.3.2. Types	44
2.3.3. Enums	45
2.3.4. Input Objects	48
2.4. Digitale handtekening	57
2.4.1. Canonieke JSON voor de reproduceerbare hash	57
2.4.2. De 'enriched JSON' waarvan de inhoud digitaal wordt ondertekend	58
2.4.3. Te gebruiken certificaat voor de handtekening	58

INLEIDING

De detailbeschrijvingen geven op meer technische en gedetailleerde wijze de technische bepalingen van het ministerieel besluit van 29/04/2024 met betrekking tot de technische aspecten en de certificatie van het kassasysteem weer.

Dit is de technische detailbeschrijving met betrekking tot de werking van het kassasysteem en haar communicatie met de Fiscal Data Module.

Voor de werking van de Fiscal Data Module wordt verwezen naar de desbetreffende detailbeschrijving.

Dit document kan kleine wijzigingen ondergaan, wegens eventuele reglementaire beslissingen, voor het rechtzetten van eventuele fouten.

Bijkomende informatie kan ingewonnen worden bij de bevoegde dienst, NCO afdeling GKS via secr.gksce@minfin.fed.be.

HOOFDSTUK 1 – TECHNISCHE EISEN KASSASYSTEEM

Referentie MB:Titel II, Hoofdstuk 1

1. REGISTRATIE VAN EVENTS

Referentie MB: Afdeling 1, onderafdelingen 1, 7 en 8

1.1. DIRECTE VERKOOP

Hierbij wordt onmiddellijk het event N geregistreerd.

De boeking (en het GraphQL bericht) bevat steeds in chronologische volgorde de geregistreerde productlijnen. Prijswijzigingen worden aan die productlijnen opgehangen. Correcties in aantal producten zijn steeds afzonderlijke lijnen, die mee hun plaats innemen in de chronologie.

De uiteindelijke afdruk van het btw-kasticket mag geconsolideerd zijn en ook de chronologie mag gewijzigd worden (bijvoorbeeld voor groepering per departement). De afdruk mag ook meer informatie bevatten dan wat wettelijk wordt gevraagd (vb. Gtin, eenheidsprijs, ...).

De afdruk bevat steeds de vermelding BTW-KASTICKET.

Voorbeelden in de Use Cases (UC).

1.2. VERKOOPREGISTRATIE MET VOLLEDIGE TERUGNAME EERDER UITGEREIKT BTW KASTICKET

Indien een eerder aangemaakt event N VOLLEDIG wordt teruggenomen in een nieuw event N wordt de **referentie** van het oorspronkelijke btw-kasticket in het GraphQL bericht en op de afdruk van het btw-kasticket vermeld. Hiervoor wordt de 'fdmRefs' array gebruikt. Dit nieuwe event N mag bijkomende productlijnen bevatten. Per event N mag maximaal één eerder aangemaakt event N worden teruggenomen.

Indien het btw-kasticket enkel een volledige terugname van een eerder event N bevat (en bijgevolg enkel negatieve productlijnen bevat), zal de afdruk van dit ticket bijkomend de vermelding 'REFUND' bevatten (artikel 5, 3° van het MB).

Voorbeelden in de UC.

1.3. ONDERBROKEN VERKOOP

Hierbij wordt de verkoopregistratie onderbroken in de tijd en worden één of meerdere deelregistraties (boekingen) gedaan. Voorbeelden zijn tafelbeheer, klantenbeheer, overboeking naar hotelrekening, aannamen weborders, aannamen kioskorders, het on-hold plaatsen van een registratie.

Hiervoor wordt het event P gebruikt. De volledige verkoopregistratie dient na deze (één of meerdere) events P steeds te resulteren in een definitieve registratie event N voor het totaal van deze events P (behalve bij overboeking naar hotelrekening).

Om alle boekingen van een verkooptransactie bijeen te houden, wordt een 'reference' gebruikt.

Elke afzonderlijke boeking volgt voor het GraphQL bericht dezelfde regels als voor de directe verkoop (correcties, afdrukken, ...).

Elke afdruk van een dergelijke boeking bevat steeds de vermelding PRO FORMA en onderaan komt bijkomend de vermelding DIT IS GEEN GELDIG BTW-KASTICKET.

Voorbeelden in de UC.

1.4. HET REKENINGOVERZICHT (VOORLOPIGE REKENING)

Dit event P moet worden aangemaakt tijdens een registratie van 'onderbroken verkoop', waarbij via de functionaliteit 'voorlopige rekening/rekeningoverzicht' een overzicht van de geboekte orders en/of het te betalen bedrag wordt getoond en/of afgedrukt vooraleer de betaling wordt geregistreerd, het event 'normal' wordt aangemaakt en het btw-kasticket wordt afgeleverd.

Conform art. 34 van het MB mogen zowel op de afdruk als in het GraphQL bericht naar de FDM van dit rekeningoverzicht de productlijnen geconsolideerd worden. Nieuwe productlijnen toevoegen of wijzigingen aanbrengen aan bestaande productlijnen is niet toegelaten bij de aanmaak van het rekeningoverzicht.

Een afdruk vermeldt steeds de vermelding PRO FORMA én VOORLOPIGE REKENING. Onderaan komt bijkomend de vermelding DIT IS GEEN GELDIG BTW-KASTICKET.

Voorbeelden in de UC.

1.5. REGISTRATIES IN TRAININGSMODUS

Alle registraties die aangemaakt worden terwijl het volledige kassasysteem zich in trainingsmodus bevindt of door een gebruiker die zich in trainingsmodus bevindt, worden als training event doorgestuurd naar de FDM. Ze dragen dus altijd het label "T".

Een afdruk van een dergelijk event vermeldt altijd de benaming TRAINING. Onderaan komt bijkomend de vermelding DIT IS GEEN GELDIG BTW-KASTICKET.

Voorbeelden in de UC.

1.6. REGISTRATIES VAN PURE FINANCIËLE VERRICHTINGEN

Voorbeelden in de UC.

1.7. AANMAAK VAN FACTUREN OP BASIS VAN EERDER AANGEMAAKTE EVENTS N

Indien een kassasysteem de mogelijkheid heeft om een factuur, ongeacht het formaat, aan te maken, op basis van één of meerdere eerder uitgereikte btw-kastickets, wordt deze factuur opgenomen in het event 'INVOICE'. De productlijnen van deze btw-kastickets worden in de afdruk van dit event opgenomen (maar **niet** in het GraphQL bericht naar de FDM).

De betaallijnen van de oorspronkelijke btw-kastickets worden **niet** overgenomen.

De afdruk draagt altijd de vermelding FACTUUR en onderaan komt bijkomend de vermelding DIT IS GEEN GELDIG BTW-KASTICKET.

Voorbeelden in de UC.

1.8. KOPIEËN VAN EERDER AANGEMAAKTE EVENTS

Indien men een **kopie** van een eerder aangemaakt btw-kasticket (event "NORMAL") wenst af te drukken (bijvoorbeeld na een printerpanne of onleesbaar geworden ticket) dan dient hiervoor een copy event aangemaakt te worden. Hierbij wordt steeds een referentie naar het oorspronkelijke btw-kasticket opgegeven (op de afdruk wordt dit: fdmId-eventLabelCounter/totalCounter).

Dit event kan ook gebruikt worden voor kopieën van eerder aangemaakte 'P', 'I', 'S', 'F' of 'R' events.

Eventuele afdrukken bevatten steeds de vermelding KOPIE en onderaan komt bijkomend de vermelding DIT IS GEEN GELDIG BTW-KASTICKET, behalve bij een kopie van een btw-kasticket.

1.9. REGISTRATIES VAN DE AANWEZIGHEID VAN HET PERSONEEL

Elk kassasysteem moet verplicht in deze functionaliteit voorzien; het effectief gebruik is niet verplicht, behalve indien de werkgever voor bepaalde RSZ- of andere voordelen opteert.

Via het event "SOCIAL" kan de registratie van de aanwezigheid (start/eind) van de 'gebruikers' worden doorgestuurd naar de FDM. Hiervoor kan eender welke aanmeldroutine gebruikt worden, mits de kassa de correcte data voor het event aanmaakt en deze via het GraphQL bericht naar de FDM verzendt. Dit event wordt **niet** afgedrukt en is **geen** verkooptransactie.

2. NUMMERING EVENTS

Art. 13. verplicht een doorlopende nummering voor elk event. Hiermee wordt bedoeld één van onderstaande:

- een aparte doorlopende nummering per event type (label);
- een algemene doorlopende nummering over de events heen;
- een algemene doorlopende nummering over de events heen per terminal;
- een aparte doorlopende nummering per event type (label) per terminal.

3. VOORWAARDEN REGISTRATIE EVENTS

1) Een gebruiker die events wil registreren, moet zich eerst aanmelden op het kassasysteem. Op het kassasysteem mag geen enkele bewerking mogelijk zijn zonder dat een gebruiker is aangemeld. Voor het registreren van events van het type 'S' (Social) mag een afzonderlijke aanmeldprocedure worden voorzien.

Gebruikers van het kassasysteem, ongeacht hun functie binnen de onderneming, dienen duidelijk identificeerbaar te zijn aan de hand van hun INSZ-nummer, dat naargelang het geval bestaat uit het rijksregisternummer of BIS-nummer. Dit nummer wordt opgeslagen in de kelnerprogrammatuur of in de database "gebruikers" van het kassasysteem. Het INSZ-nummer is opgebouwd uit 11 numerieke karakters.

Een gebruiker van het kassasysteem, vreemd aan de onderneming (vb. een technicus), die handelingen registreert met behulp van het kassasysteem, wordt hiervoor in het kassasysteem steeds geïdentificeerd met het nummer "00000000097".

Voor online- en kioskbestellingen en alle toekomstige mogelijkheden tot bestellen die geen tussenkomst van een fysiek persoon vereisen, maar wel onder de volle verantwoordelijkheid van de uitbating vallen, dient als 'robotuser' het nummer "00000000029" gebruikt te worden.

Een overzicht van deze programma-instellingen of databasetabel moet op eenvoudige wijze - op verzoek - ter beschikking kunnen gesteld worden van de controlerende ambtenaar.

2) Het kassasysteem mag leveringen van goederen en/of diensten enkel kunnen registreren wanneer de FDM is aangesloten en volledig operationeel is. Een transactie, waarvan de registratie begonnen is, mag niet beëindigd kunnen worden zolang er geen handtekening van de FDM kan ontvangen worden.

4. BEWARING GEGEVENS

Referentie MB: Afdeling 1, onderafdeling 2

De belastingplichtige-gebruiker van een geregistreerd kassasysteem is verantwoordelijk voor de bewaring van de gegevens die door het kassasysteem zijn aangemaakt, conform de btw-wetgeving (en bij uitbreiding de boekhoudwetgeving). Zo is de belastingplichtige-gebruiker met name verantwoordelijk voor de bewaring van de gegevens op het kassasysteem en op de FDM. Gemaakte back-ups van kassadata blijven deel uitmaken van het kassasysteem en zijn bijgevolg aan dezelfde bewaringstermijnen onderhevig.

Op het kassasysteem worden de door het kassasysteem zelf aangemaakte gegevens (database, JSON, de data van de GraphQL berichten) en deze ontvangen van de FDM (JSON) in originele vorm bewaard, conform de vigerende fiscale bewaartermijn. Er wordt aan herinnerd dat, alhoewel geen enkel specifiek formaat wordt voorgeschreven, alle door het kassasysteem aangemaakte gegevens conform artikel 61, §1 van het Btw-Wetboek moeten kunnen voorgelegd worden in leesbare en verstaanbare vorm. Om kopieën van deze gegevens te vergemakkelijken, dient er tenminste één poort (van een courant type) van de kassa toegankelijk/geactiveerd te zijn voor een externe gegevensdrager.

De producent van het kassasysteem voorziet bij de aanvraag tot certificatie een gedetailleerde beschrijving van de database en meer in het bijzonder van alle tabellen die event-gegevens bevatten of een volledige beschrijving van elke andere manier van bewaring van de in het ministerieel besluit vermelde gegevens.

De producenten van kassasystemen worden er uitdrukkelijk op gewezen dat zij verantwoordelijk zijn voor de manier van beveiligen van de database gegevens met betrekking tot de voormelde events. De producent vermeldt bij zijn aanvraag tot certificatie de beveiligingsmechanismen die werden gebruikt (ongeacht of die in de software zelf zijn ingebouwd of via derden worden voorzien).

5. BTW-CODES

Referentie MB: Afdeling 1, onderafdeling 3

Het kassasysteem dient volgende btw-codes te voorzien

BTW-CODE	BTW-TARIEF OMSCHRIJVING	BTW-TARIEF
A	Hoog	21 %
B	Midden	12 %
C	Laag	6 %
D	Nultarief	0 %
X	Buiten btw-toepassingsgebied	Geen

De kassasoftware mag zelf de maatstaven van heffing en de verschuldigde btw berekenen. Dit resultaat wordt evenwel **niet** doorgestuurd naar de FDM.

De firmware van de FDM berekent zelf, op basis van de doorgestuurde productlijnen met bijhorende btw-code, de maatstaf van heffing en de verschuldigde btw. De tarieven waarmee de FDM rekening houdt, worden door de online verbinding met de servers van de FOD Financiën up-to-date gehouden. Bij een tariefwijziging is op de FDM dus geen enkele manuele handeling nodig.

Op het btw-kasticket worden de maatstaven van heffing en de btw-bedragen vermeld die van de FDM ontvangen werden. Dit geldt ook voor de aanmaak van de rapporten.

Gebruik van btw-code X: voor de registratie van de 'verkoop' van goederen en diensten die volledig buiten het toepassingsveld van de btw vallen.

Bedoeld worden: aanrekening en terugname van leeggoed, aanrekening en terugbetaling van waarborgen, de verkoop van een multipurpose voucher (MPV).

Belangrijke opmerking: artikelen met meerdere btw-codes.

In uitzonderlijke gevallen kunnen artikelen twee of meer btw-codes hebben. Typevoorbeelden zijn: een tijdschrift dat samen met een CD wordt verkocht (niet-horeca) of een all-in menu (horeca).

De gebruikte datastructuren maken het mogelijk om dergelijke artikelen correct naar de FDM te sturen als één product met verschillende btw-codes; elke btw-code wordt dan voorzien van het deel van de verkoopprijs (inclusief btw) waarop de btw-code betrekking heeft, aangevuld met de eventuele prijswijzigingen van toepassing op dit deel van de verkoopprijs.

Deze artikelen kunnen als één productlijn worden afgedrukt op het btw-kasticket, bij voorkeur met een gecombineerde btw-code (vb. AB). Indien dit technisch niet mogelijk is, mag de btw-code afgedrukt worden van het onderdeel met de grootste waarde, voor zover de werkelijke verdeling wordt doorgestuurd naar de FDM.

Er mag uiteraard geopteerd worden om een aparte productlijn per btw-code, met de corresponderende waarde, af te drukken.

Opgelet: indien een menu verkrijgbaar is met een drankenpakket waarvan de waarde afzonderlijk geafficheerd is, dan dient altijd geopteerd te worden voor twee afzonderlijke productlijnen (zowel bij de afdruk als in het doorsturen naar FDM).

6. HET BTW-KASTICKET

Referentie MB: Afdeling 1, onderafdeling 4

Zie ook eerder in dit document, registratie van event N.

6.1. IDENTIFICATIE VAN HET KASSASYSTEEM

Hiermee wordt het toegekende serienummer uit art. 46 bedoeld. Dit wordt in het GraphQL bericht naar de FDM weergegeven in het veld posld.

6.2. IDENTIFICATIE VAN DE TERMINAL

Indien een GKS-setting uit meerdere kassa's onder een zelfde posld bestaat, dan moet voor elke kassa een unieke terminalld gebruikt worden. Voorbeelden zijn een logische naam (bar, terras, ...) of enige andere identificatie (nummer, ...). Dit wordt in het GraphQL bericht naar de FDM weergegeven in het veld terminalld.

6.3. IDENTIFICATIE VAN HET INPUT DEVICE

Een input device is het fysieke toestel waarop het event wordt geregistreerd. Dit kan het stand alone kassatoestel zijn, de terminal, de kiosk, de handheld, de smartphone,...

Voor webshops die zich in de cloud bevinden of door een derde partij worden aangeboden, wordt de naam van die partij of de url van de webshop gebruikt.

In het GraphQL bericht tussen kassa en FDM wordt de identificatie opgenomen in het veld deviceld.

6.4. IDENTIFICATIE VAN DE GEBRUIKER

Op het btw-kasticket wordt **NOOIT** het INSZ van de gebruiker afgedrukt. De gebruiker dient wel, intern in het kassasysteem, via de op het ticket vermelde identificatie, gelinkt te kunnen worden aan dit INSZ, dat overigens als onderdeel van het GraphQL bericht naar de FDM wordt gestuurd.

6.5. QR-CODE

Deze wordt door de kassasoftware zelf gegenereerd op basis van de url die het van de FDM ontvangt.

De url bestaat uit 38 alfanumerieke karakters; deze passen in een 25x25 module type 2 QR met medium error correction feature (ECC).

Deze QR-code wordt afgedrukt op het papieren btw-kasticket. Bij een digitale afdruk wordt de url als link op het btw-kasticket vermeld.

Indien naderhand een COPY (C) van het btw-kasticket wordt aangemaakt en afgedrukt, ontvangt de kassa geen nieuwe url van de FDM, maar dient dezelfde QR code afgedrukt te worden, volgens de url ontvangen voor het oorspronkelijke NORMAL (N) event.

6.6. CONTROLEGEGEVENS OP HET BTW-KASTICKET

Het MB vermeldt welke gegevens op een btw-kasticket moeten worden vermeld. Met de controlegegevens, zoals vermeld in artikel 1, 7°, wordt dus uitdrukkelijk bedoeld:

- het fdmID
- de fdmDateTime
- het eventLabel
- de eventCounter
- de totalCounter
- de shortSignature.
-

6.7. ALGEMEEN

Het btw-kasticket mag meer informatie bevatten dan hetgeen wettelijk bepaald wordt.

7. AFRONDINGEN VAN BETALINGEN

Referentie MB: Afdeling 1, onderafdeling 9

De artikelen VI.7/1 en VI.7/2 van het Wetboek Economisch Recht verplichten de handelaar om het totaalbedrag van de betaling in speciën door de consument af te ronden naar het lagere of hogere veelvoud van 5 cent.

Concreet betekent dit dat betalingen eindigend op 1, 2, 6 of 7 cent worden afgerond naar het lagere veelvoud van 5 cent, betalingen eindigend op 3, 4, 8 of 9 cent worden afgerond naar het hogere veelvoud van 5 cent. Betalingen van minder dan 5 cent worden nooit afgerond.

Indien de handelaar alle betalingen, ongeacht het betaalmiddel, wil afronden dan dient het GKS in die mogelijkheid te voorzien. Gelet op het momenteel facultatief karakter van de afronding van andere betalingen dan in speciën, past het GKS de afronding voor andere betalingen dan contante betalingen niet toe wanneer de handelaar daar niet voor kiest.

Opgemerkt wordt dat deze keuze niet per handeling gebeurt, maar voor het geheel van alle andere betalingen dan in speciën. Voor betalingen in speciën is de afronding verplicht.

Opgelet: bovenstaande geldt **NOOIT** voor maaltijd-, ecocheques en waardebonnen, die immers altijd een vaste waarde hebben. Bij betaling met maaltijd-, ecocheques en waardebonnen moeten deze altijd als eerste behandeld worden.

Voor de te gebruiken soorten betaallijnen wordt verwezen naar Hoofdstuk 2 verder in dit document.

Belangrijke opmerking: Een btw-kasticket mag in totaal maximaal een afronding van 2 eurocent hebben (in plus of in min).

Indien het GKS binnen een horecaomgeving wordt gebruikt, is het niet ongebruikelijk dat bij grotere groepen de betalingen per persoon of per paar personen gebeuren. Naargelang de organisatie van de onderneming en/of het gebruikte kassasysteem zal hierbij gebruik gemaakt worden van directe verkoop per 'consument', 'splitsing tafel' of 'gesplitste betaling'.

Bij 'directe verkoop' wordt per consument exact in de kassa geregistreerd wat is geconsumeerd en wordt naargelang de betaalwijze (en keuze van de handelaar) afgerond of niet. Per consument wordt een btw-kasticket afgeleverd.

Bij 'splitsing tafel' wordt eveneens een btw-kasticket per consument afgeleverd (hetzij met het exact geconsumeerde, hetzij met zijn evenredig deel). Betalingsaf rondingen zijn identiek aan 'directe verkoop'.

Bij 'gesplitste betaling' echter, wordt uiteindelijk slechts één btw-kasticket uitgereikt voor alle consumenten samen. In een dergelijk geval zal de uitbater zelf methodisch te werk moeten gaan (dit geldt overigens niet enkel voor horeca en GKS).

Rekening houdend met hierboven vermelde belangrijke opmerking dient als volgt gehandeld te worden:

7.1. DE UITBATER ROND ENKEL CONTANTE BETALINGEN AF

Methodologie:

1. De betalingen van maaltijdcheques en waardebonnen (globaal of apart, naar keuze) worden geregistreerd. Deze komen dan als betaallijn op het ticket. De kassa maakt het saldo.
2. Daarna worden de betalingen in speciën verwerkt. Deze komen als één betaallijn op het ticket, AFGEROND. De kassa maakt opnieuw het saldo.
3. Als laatste komen de elektronische betalingen (NIET AFGEROND) voor het saldo. Indien gewenst kunnen meerdere betaallijnen per type betaling gebruikt worden.

Hierbij mag de totale afronding niet groter zijn dan 2 eurocent (in min of in plus).

7.2. DE UITBATER ROND ALLE BETAALMIDDELEN AF

Methodologie:

1. De betalingen van maaltijdcheques en waardebonnen (globaal of apart, naar keuze) worden geregistreerd. Deze komen dan als betaallijn op het ticket. De kassa maakt het saldo.
2. DE KASSA ROND DIT SALDO AF.
3. Daarna worden de betalingen in speciën verwerkt. Deze komen als één betaallijn op het ticket, AFGEROND. De kassa maakt opnieuw het saldo.
4. Als laatste komen de elektronische betalingen voor het saldo, wat onvermijdelijk een AFGEROND bedrag zal zijn. Indien gewenst kunnen meerdere betaallijnen per type betaling gebruikt worden.

Hierbij mag de totale afronding niet groter zijn dan 2 eurocent (in min of in plus).

De FOD Financiën stelt via de website www.geregistreerdkassasysteem.be voorbeelden van specifieke situaties ter beschikking van geïnteresseerde POS-ontwikkelaars.

8. CONSOLIDATIE- EN AFDRUKREGELS

Referentie MB: Afdeling 1, onderafdelingen 7 en 8

8.1. AFDRUKKEN

Er dient voor de toepassing van dit besluit steeds een duidelijk onderscheid behouden te worden tussen de volgende begrippen: EVENT <> AFDRUK (ticket) <> BOODSCHAP naar de FDM.

ELK event resulteert ALTIJD in een boodschap naar de FDM.

Enkel de events NORMAL (N) en INVOICE (I) geven ALTIJD aanleiding tot een afdruk, op papier, digitaal of beide. Het btw-kasticket moet immers in uitvoering van artikel 21bis van het KB nr. 1 verplicht uitgereikt worden aan de eindklant. Hetzelfde geldt voor de aangemaakte factuur die krachtens artikel 53 §2 van het WBTW wordt uitgereikt.

Het event REPORT (R) resulteert ALTIJD in de aanmaak van een bestand, MAAR moet ook op eenvoudige wijze afgedrukt kunnen worden door de exploitant, hetzij op papier, hetzij op digitale wijze.

Alle andere events MOGEN (maar moeten niet) afgedrukt worden. Deze afdrukken bevatten, onderaan en duidelijk leesbaar, steeds de vermelding "DIT IS GEEN GELDIG BTW-KASTICKET" (met uitzondering van een COPY (C) van een NORMAL (N) btw-kasticket).

Een rekeningoverzicht draagt bijkomend duidelijk leesbaar de vermelding "VOORLOPIGE REKENING".

Daarenboven bevatten afdrukken van events TRAINING (T) en PRO FORMA (P), naast de vermelding 'DIT IS GEEN GELDIG BTW-KASTICKET', bovenaan en duidelijk leesbaar, respectievelijk de vermelding "TRAINING TICKET" en "PRO FORMA TICKET. Voor de afdrukken van event COPY (C) wordt bijkomend aan voorgaande de vermelding "KOPIE TICKET" vermeld.

Keuken- en bartickets vermelden NOOIT bedragen. Indien ze worden afgedrukt als direct gevolg van de aanmaak van een event PRO FORMA dienen ze eveneens de vermelding 'DIT IS GEEN GELDIG BTW-KASTICKET'te bevatten.

Bijzonderheden met betrekking tot de afdruk van een kopie van een BTW-KASTICKET.

Behalve de duidelijke vermelding KOPIE op het ticket wordt verduidelijkt dat:

- de afdruk (papier/digitaal) **identiek** is aan het oorspronkelijke BTW-kasticket (alle velden, inbegrepen de controledata);
- de afdruk (papier/digitaal) **bijkomend** de posDateTime en fdmRef van het copy event vermeldt.

8.2. CONSOLIDATIE

Om de leesbaarheid van de tickets te optimaliseren worden consolidaties in bepaalde gevallen toegelaten, met inbegrip van alle correcties (negatieve of positieve productlijnen) en prijswijzigingen (in min of in meer, gekoppeld aan de correcte productlijn). De kassasoftware mag hiervoor geen afzonderlijk 'P' event aanmaken.

Belangrijke opmerking: Deze consolidaties worden nooit toegepast op de berichten die het kassasysteem naar de FDM stuurt. Concreet betekent dit dat een ticket inhoudelijk overeenstemt met de boodschap, maar dat er formeel verschillen kunnen zijn.

Consolidatie is niet verplicht, elke producent bepaalt zelf of zijn kassasoftware al dan niet consolideert. Indien consolidatie wordt gebruikt, dienen onderstaande regels gevolgd te worden.

Directe verkoop (event 'N'):

De boodschap naar de FDM bevat alle ingebrachte gegevens, in chronologische volgorde, met inbegrip van correcties (negatieve of positieve productlijnen) en prijswijzigingen (in min of in meer, gekoppeld aan de correcte productlijn).

Voorbeeld:

1	Cola	2.50	A
1	Water	3.00	A
2	Spaghetti	20.00	B
-1	Cola	-2.50	A
1	Water	3.00	A

Bij de afdruk op het btw-kasticket mogen de correcties worden geconsolideerd en zelfs de volgorde mag naar inzicht van de kassasoftware aangepast worden (bijvoorbeeld groeperen op artikeldepartementen of per btw-tarief).

In het voorgaande voorbeeld wordt dit dan:

2	Water	6.00	A
2	Spaghetti	20.00	B

Onderbroken verkoop (tafelbeheer)

Dezelfde regels gelden als hierboven beschreven. Concreet zal dus het event P (dat de **oorspronkelijke** bestelling bevat) chronologisch en met inbegrip van de correcties worden doorgestuurd. De bijhorende afdruk (voorlopige rekening of keuken-/barticket) mag geconsolideerd en gehegroepeerd worden.

Bij het afsluitend event N mag zowel de boodschap als de afdruk van het finale btw-kasticket geconsolideerd worden (uiteraard voor zover er bij de aanmaak van dit event geen nieuwe gegevens werden geregistreerd).

HOOFDSTUK 2 – WISSELWERKING POS - FDM

Wanneer de FDM verbonden is met het kassasysteem, zal:

- de FDM specifieke event data ontvangen van het kassasysteem;
- de FDM zijn relevante tellers updaten;
- de FDM de controledata aanmaken;
- de FDM deze boodschappen en antwoorden opslaan;
- de FDM de controledata terugsturen naar het kassasysteem;
- het kassasysteem het ticket met de ontvangen controledata kunnen afdrukken, op voorwaarde dat ze een dergelijk antwoord krijgt van de FDM.

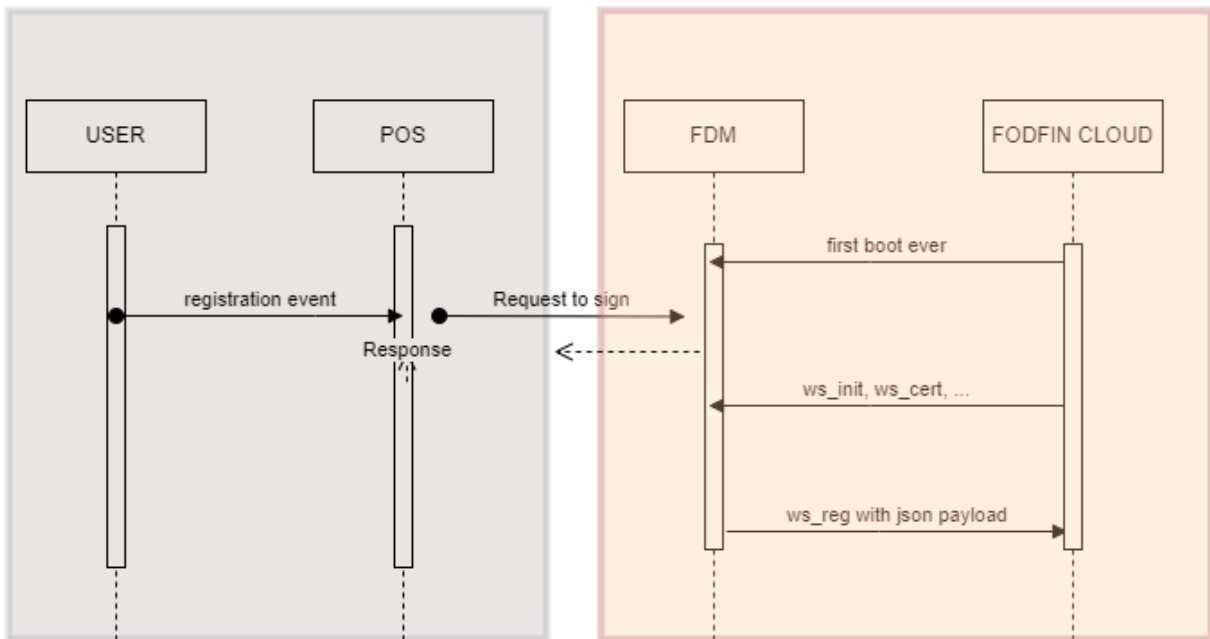
Alle geregistreerde kassasystemen, zoals bedoeld in het ministerieel besluit, moeten deze data communiceren aan de hand van het communicatieprotocol en volgens de dataformaten, zoals beschreven in dit document als onderdeel van hierboven vermeld ministerieel besluit.

DEFINITIES EN AFKORTINGEN

API	Application Programming Interface
EFT	Electronic Funds Transfer
ENUM	Enumeratie (opsomming)
FDM	Fiscal Data Module

GKS	Geregistreerd KassaSysteem
GRAPHQL	Graph Query Language
JSON	JavaScript Object Notation
LAN	Local Area Network
POS	Point Of Sale
RTC	Real Time Clock
TCP/IP	Transmission Control Protocol/Internet Protocol
TRANSACTIES	alle events waarbij gegevens worden vastgelegd
UTC	Coordinated Universal Time
UTF	Unicode Transformation Format
WiFi	Wireless Fidelity

De process flow van de gegevens kan beknopt als volgt samengevat worden:



Chronologisch:

1. De FDM wordt de allereerste keer opgestart en ontvangt de nodige instellingen van de FODFIN cloud.
2. De gebruiker registreert events op het kassasysteem.
3. De kassa stuurt deze events onmiddellijk door naar de FDM onder de vorm van JSON berichten, die naargelang het soort event een aantal berekeningen doet en over de verrijkte JSON zijn handtekening plaatst.
4. De FDM stuurt het antwoord met de controledata terug naar de kassa.
5. De FDM buffert de verrijkte JSON van de events.
6. De FDM contacteert op geregelde tijdstippen de webservice van FODFIN om zijn parameters of certificaten op te halen.
7. De FDM stuurt op regelmatige tijdstippen de gebufferde JSON berichten door naar de FODFIN cloud.
8. De ontvangen JSON berichten worden in de FODFIN cloud in een beveiligde omgeving bewaard, waar ze ter beschikking zijn voor analyses.
9. Het kassasysteem toont via zijn user interface meldingen van de FDM.

1. FYSIEKE VERBINDING

Kassa en FDM worden met elkaar verbonden met een kabel (LAN) of draadloos (WiFi). De configuratie hiervan gebeurt in de backoffice van zowel kassa als FDM.

BELANGRIJK: In theorie kan de draadloze verbinding ook via het internet verlopen. Bij wegvallen van het internet valt dan automatisch ook de verbinding kassa-FDM weg en stopt de kassa met registreren.

De FDM producent voorziet voldoende mogelijkheden om een goede beveiliging van de verbinding kassa – FDM mogelijk te maken. Typische voorbeelden zijn: HTTP met token, HTTPS met TLS met self signed certificaat, HTTPS met mTLS met self signed certificaat gebaseerd op het intermediate certificaat van de FDM producent.

De eindverantwoordelijkheid voor de correcte en beveiligde installatie en configuratie ligt bij de verdeler/installateur.

Kassa en FDM gebruiken het TCP/IP protocol voor het uitwisselen van hun boodschappen. Deze boodschappen worden hieronder verder beschreven. Andere boodschappen zijn enkel toegelaten voor zover zij de verplichte features niet beïnvloeden en bijdragen tot een goede werking van het geheel. De FDM stelt zijn GraphQL service aan de POS beschikbaar via HTTP(S).

De FDM is via het internet verbonden met de FODFIN cloud API en ontvangt via deze weg ook regelmatig instructies. In bepaalde gevallen, zie betreffende detailbeschrijving, kan dergelijke instructie leiden tot het tonen van een boodschap op het kassasysteem en/of een manuele interventie via het kassasysteem.

De FDM mag niet verbonden worden met toepassingen van derde partijen. Enkel de POS toepassingen, de FODFIN toepassingen en de toepassingen van de FDM-producent mogen verbonden worden met de FDM.

2. COMMUNICATIE TUSSEN POS EN FDM

De requests worden door de kassa naar de GraphQL service van de FDM gestuurd. Voor het versturen van de event gegevens wordt JSON data als payload gebruikt. Deze payload wordt naar het pad “/graphql” op de FDM verstuurd met HTTP verb “POST” en Content-Type: application/json.

Alle JSON objecten volgen de RFC 8259 standaard.

String velden bevatten **nooit** voorloop- of volgwitruimte.

De verstuurde gegevens omvatten onder andere:

- transactiegegevens (events P en N);
- financiële gegevens (event F);
- sociale gegevens (event S);
- kopieën (event C);
- training events (event T);
- facturen (event I);
- rapporten (event R).

Deze objecten en hun voorwaarden worden verder bij het volledige GraphQL schema beschreven.

2.1. ALGEMEEN

In het ministerieel besluit werden de volgende events bepaald:

- NORMAL (label N);
- PRO FORMA (label P);
- TRAINING (label T);
- COPY (label C);
- FINANCIAL (label F);
- SOCIAL (label S);
- INVOICE (label I);
- REPORT (label R).

Deze events worden allemaal naar de FDM gestuurd, om – als beveiliging tegen data manipulatie – digitaal ondertekend te worden. Hiervoor worden er diverse mutaties beschikbaar gesteld door de GraphQL service van de FDM. Deze mutaties kunnen gebruikt worden om de hieronder beschreven JSON structuur van de registraties op de FDM correct op te vullen. De JSON structuur van de FDM kan elk type event bevatten; via de verschillende mutaties aanvaardt de FDM voor elke soort event enkel die velden en objecten die toegelaten zijn.

Volgende mutaties worden gedefinieerd:

Voor het event S:

- signWorkIn
- signWorkOut

Voor het event I:

- signInvoice

Voor het event N:

- signSale

Voor het event P:

- signCostCenterChange
- signOrder
- signPreBill

Voor het event F:

- signMoneyInOut
- signDrawerOpen
- signPaymentCorrection

Voor het event R:

- signReportTurnoverX
- signReportTurnoverZ
- signReportUserX
- signReportUserZ

Voor het event C:

- signCopy

2.2. INHOUD VAN DE COMMUNICATIE POS - FDM

De payload van de FDM naar de FODFIN (zie detailbeschrijving FDM) is volledig gebaseerd op de hierna beschreven structuur en bijhorend validatieschema.

Bijgevolg moeten de event data van de mutaties ook die structuur en validatieregels respecteren. In sectie 2.3. wordt de omzetting naar het GraphQL schema beschreven.

2.2.1. POS → FDM – volledig overzicht

```
{  
  language (scalar value)  
  ticketMedium (scalar value)  
  posId (scalar value)  
  posFiscalTicketNo (scalar value)  
  posSwVersion (scalar value)  
  terminalId (scalar value)  
  deviceId (scalar value)  
  posDateTime (scalar value)  
  bookingPeriodId (scalar value)  
  bookingDate (scalar value)  
  vatNo (scalar value)  
  estNo (scalar value)  
  employeeld (scalar value)  
  customerVatNo (scalar value)  
  invoiceNo (scalar value)  
  fdmRefs (array van fdmRef objecten)  
  costCenter (object)  
  transfer (object)  
  transaction (object)  
  drawer (object)  
  financials (array van paymentLine of moneyInOutLine objecten)  
  reportNo (scalar value)  
  reportBookingDate (scalar value)  
  posDevices (array van posDevice objecten)  
  fdmDevices (array van fdmDevice objecten)  
  turnover (object)  
  users (array van user objecten)  
}
```

2.2.2. POS → FDM - detailbeschrijving

Taal

language (enum, verplicht)

Taal waarin het kassasysteem de berichten van de FDM wenst te ontvangen, te kiezen uit onderstaande waarden:

language	Omschrijving
EN	Engels
NL	Nederlands
FR	Frans
DE	Duits

Afleverwijze

ticketMedium (enum, verplicht)

Soort ticket, te kiezen uit onderstaande tabel:

ticketMedium	Omschrijving
NONE	Geen afdruk
PAPER	Afdruk op papier
DIGITAL	Digitale aflevering
PAPER_DIGITAL	Afdruk op papier en elektronische aflevering

Pos Identificatiegegevens

posId (string, verplicht, 14 tekens in hoofdletters)

Serienummer van de POS, zoals geregistreerd via de e-service GKS. Voorbeeld: CFOD0061234567.

Het posId CFOD010000001 is voorbehouden (en wordt steeds gebruikt) voor de certificatieprocedure. Het definitieve prefix (CXXXNN) wordt gegenereerd na het succesvol beëindigen van deze procedure.

posFiscalTicketNo (numeriek, verplicht, waarde tussen 1 en 999999999)

Intern ticketnummer van de POS. Dit unieke nummer mag gegenereerd worden over alle events heen, over alle terminals heen, per event type of per terminal, voor zover de combinatie van dit nummer en respectievelijk het event type of het terminal id uniek is. Wanneer de maximale waarde wordt bereikt, start de POS opnieuw met het nummer "1".

posSwVersion (string, verplicht, lengte min. 1 en max. 36 tekens) Huidige softwareversie van de kassasoftware. Voorbeeld: 1.8.3

terminalId (string, verplicht, lengte min. 1 en max. 600 tekens)

Indien een GKS-setting uit meerdere kassa's onder een zelfde posId bestaat, dan moet voor elke kassa een unieke terminalId gebruikt worden. Voorbeelden zijn een logische naam (bar, terras, ...) of enige andere identificatie (nummer, ...). Het begrip terminal staat volledig los van enige hardwarecomponent. Indien de GKS-setting hiervan geen gebruik maakt, dient in dit veld steeds dezelfde waarde ingevuld te worden. Voorbeeld: "1", "n/a", vaste guid.

deviceld (string, verplicht, lengte min. 1 en max. 600 tekens)

De hardware-identificatie van het apparaat waarop de transactie werd geregistreerd.

Pos Datum en Tijd

posDateTime (string, verplicht)

Datum en tijd van het kassasysteem, in lokale tijd (Brussel), volgens onderstaande ISO 8601 formaat.

Voorbeeld: 2022-10-20T15:01:25+02:00 (zomertijd), 2022-11-03T15:01:25+01:00 (wintertijd)

bookingPeriodId (string, verplicht, formaat = GUID met streepjes en kleine letters)

Werkingsperiode van de kassa: meestal is dit een openingsdag, waardoor deze periode gelijk valt met de periode van een fiscaal dagrapport. Kan ook een shift zijn als onderdeel van een openingsdag. In dit laatste geval zal een fiscaal dagrapport gegevens van meerdere booking periods bevatten.

Voorbeeld: dffcd829-a0e5-41ca-a0ae-9eb887f95637

bookingDate (string, verplicht)

Boekhoudkundige datum waarop de transactie geboekt wordt, in het onderstaande datum ISO 8601 formaat. Deze waarde verwijst naar de datum waarop de bookingPeriodId betrekking heeft. Een booking period van 00.30 tot 04.30 's morgens kan bijvoorbeeld wel degelijk opgenomen worden in de dagomzet van de dag (datum) ervoor.

Voorbeeld: 2023-10-20

Identificatiegegevens Onderneming & Gebruikers

vatNo (string, verplicht, formaat: BE gevolgd door 10 cijfers, totale lengte = 12)

Btw-nummer van de onderneming. Het numeriek deel begint steeds met een '0' of een '1'. De FDM zal een controle uitvoeren op de geldigheid van dit nummer (door toepassing van de modulo 97).

Voorbeeld: BE0499999960

estNo (string, verplicht, lengte = 10)

Vestigingseenheidnummer van de uitbating. Een vestigingseenheidnummer begint steeds met een '2' tot en met '8'. De FDM zal een controle uitvoeren op de geldigheid van dit nummer (door toepassing van de modulo 97).

Voorbeeld: 8789456149

employeeld (string, verplicht, lengte = 11)

Inschrijvingsnummer Sociale Zekerheid (INSZ), bestaande uit ofwel het rijksregister- ofwel het bisnummer van de operator van het tewerkgesteld personeelslid. De FDM zal een controle uitvoeren op de geldigheid van dit nummer (door toepassing van de modulo 97).

Voorbeeld: 75061189731

Referenties

customerVatNo (string, conditioneel, lengte min. 1 en max. 600 tekens)

Bij het event 'I' (Invoice) is dit veld verplicht te gebruiken en wordt hier het btw-nummer van de medecontractant vermeld. Hiervoor wordt het formaat zoals beschreven in VatNo gebruikt of het buitenlandse equivalent indien de klant geen Belgische onderneming is.

invoiceNo (string, conditioneel, lengte min. 1 en max. 600 tekens)

Factuurnummer, enkel te gebruiken bij event type 'I'.

fdmRefs (optioneel, array van fdmRef objecten, zie fdmRef)

Deze array bevat de referenties toegekend door de FDM aan eerdere events waarnaar verwezen wordt. Dit kan een btw-kasticket (event N) zijn dat **volledig** geannuleerd wordt, één of meerdere btw-kastickets die deel uitmaken van een factuur (event I), het btw-kasticket waarvan de betaalwijze wordt gecorrigeerd (event F) of het maken van een kopie van een eerder event (event C).

Het fdmRef object wordt beschreven in hoofdstuk 4 – RESPONSE.

Registratie Transacties – Gebruik van Beheersmogelijkheden

De tussentijdse registraties van verkooptransacties kunnen worden toegewezen aan tafels, stoelen, klanten, kamers ... Dergelijke boekingen kunnen zich veelvuldig opvolgen of van toewijzing veranderen vooraleer dit resulteert in een definitief btw-kasticket. De objecten 'costCenter' en 'transfer' worden gebruikt om dit correct te registreren.

costCenter

costCenter (object, conditioneel)

Identificeert de 'plaats' van de transactie. Verplicht te gebruiken voor transacties waarbij de registratie wordt onderbroken (zoals tafelbeheer, klantbeheer, ...). Het object wordt als volgt opgebouwd.

{

id (string, verplicht, lengte min. 1 en max. 600 tekens)

Omschrijving van het costCenter. Voorbeeld: "T17", "K08"

type (enum, verplicht)

Een waarde te kiezen uit de tb_costCenterType.

tb_costCenterType

type
TABLE
CHAIR
ROOM
CUSTOMER
ON_HOLD
OTHER

Opmerking: de waarde **CHAIR** mag enkel gebruikt worden indien op hetzelfde of een hoger niveau de waarde **TABLE** werd gebruikt.

reference (string, verplicht, lengte min. 1 en max. 600 tekens)

Sessie van een costCenter, houdt alle geconnecteerde transacties samen.

Voorbeeld: "e4eb2509-dce9-44db-8322-e445aaeb19c4"

costCenter (costCenter object, optioneel, max. 2 niveau's diep)

Deze geneste structuur laat toe om bijvoorbeeld tegelijk op tafel- én stoelniveau registraties te doen.

}

transfer

transfer (object, conditioneel)

Het transfer object wordt gebruikt om reeds geregistreerde leveringen van goederen/diensten te verplaatsen vanaf het costCenter waarop ze geboekt werden naar een ander costCenter. Voorbeelden hiervan zijn transfer tafel, split tafel, join tafels, overboeking naar kamer, overboeking naar klantenrekening, ...

{

from (array van één of meerdere transferItem objecten)

Oorspronkelijk(e) costCenter(s), *van waar de transfer vertrekt*

to (array van één of meerdere transferItem objecten)

Nieuw(e) costCenter(s), *waar de transfer toekomt*

}

De transferItem objecten in from en to kunnen een 'één-op-één', een 'meerdere-op-één' of een 'één-op-meerdere' relatie hebben. Een relatie van 'meerdere-op-meerdere' waar goederen of diensten van meerdere costCenters naar meerdere andere costCenters verplaatst worden, is verboden.

transferItem

{

costCenter (object, verplicht, zie costCenter)

Identificatie van het costCenter van waar de transfer vertrekt of waarop een transfer toekomt.

transaction (object, verplicht, zie transaction)

De geleverde goederen/diensten die van costCenter verplaatst worden.

}

Belangrijke opmerking.

Bij het gebruik van tafelbeheer wordt een tafel in normale omstandigheden afgesloten wanneer alle registraties geresulteerd hebben in event 'N', met aanmaak van het btw-kasticket.

Hierop is één uitzondering mogelijk, namelijk bij het doorboeken van de inhoud van de tafel naar de hotelkamer, waarbij die inhoud wordt overgenomen op de globale hotelrekening. Er worden twee mogelijke manieren aanvaard om de tafel af te sluiten:

- hetzij afsluiten met een event 'N' en betaalwijze 'ROOM_CREDIT' (de boekingen op de tafel figureren in dit geval WEL in de dagomzet van het omzetrapport),
- hetzij afsluiten met een event 'P', waarbij een transfer wordt geregistreerd van een costCenter TABLE naar een costCenter ROOM (hier komen de boekingen NIET in de dagomzet van het kassasysteem).

Registratie Transacties - Transactielijnen

transaction (object, conditioneel)

Dit object bevat de registraties van de leveringen van goederen en/of diensten in het kassasysteem.

Dit object is **verplicht** bij de volgende eventOperations:

- SALE;
- PRE_BILL;
- ORDER;
- COST_CENTER_CHANGE.

Dit object is **verboden** te gebruiken bij alle andere eventOperations.

Het transaction object bevat steeds een transactionLines array. Deze kan leeg zijn in het geval van een nulbon.

{

transactionLines (array van transactionLine objecten, beschrijving zie transactionLine)

Transactielijnen inclusief alle producten en hun (prijs-)wijzigingen.

transactionTotal (numeriek, twee decimalen)

Totaalprijs van het volledige event (som van alle lineTotal). Dit bedrag kan zowel positief, negatief als nul (0) zijn.

}

transactionLine

Dit object bevat één transactielijn waarin maximaal één product wordt in opgenomen.

{

lineType (enum, verplicht)

Waarde te kiezen uit de tabel hieronder.

tb_lineType

lineType
SINGLE_PRODUCT
COMPOSITE_PRODUCT

mainProduct (product object, verplicht)

Bevat alle informatie omtrent het product vervat in de transactionLine. Indien het een samengesteld product betreft (bijvoorbeeld een menu) dient steeds ook de array van subProduct objecten gebruikt te worden. Zie de beschrijving verderop.

subProducts (array van product objecten, conditioneel)

Bevat alle informatie omtrent de producten die vervat zitten in het samengesteld product vermeld in het mainProduct object. Zie de beschrijving verderop.

costCenter (object, optioneel)

Wordt gebruikt indien de kassa toelaat om producten tot op het laagste niveau toe te wijzen aan een costCenter (bijvoorbeeld: stoel). Het costCenter object werd eerder beschreven.

Opmerking: de waarde CHAIR mag enkel gebruikt worden indien op hetzelfde of een hoger niveau de waarde TABLE werd gebruikt.

lineTotal (numeriek, verplicht, 2 decimalen)

Totaalprijs van de transactielijn, inclusief btw, na toepassing van prijswijzigingen. Kan positief, negatief of nul zijn.

In het geval van een SINGLE_PRODUCT is er enkel een mainProduct object en is de formule: $Som(vat.price + som(priceChange.amount))$.

}

Belangrijke opmerking: composite_product

Het werken met een composite_product biedt de mogelijkheid om producten, die standaard in het assortiment aangeboden worden, te combineren tot één nieuw product, waarbij al dan niet kortingen worden toegepast om tot één prijs te komen.

Een composite_product kan onderworpen zijn aan meerdere btw-tarieven. Volgende richtlijnen dienen bij het gebruik van een composite_product in acht te worden genomen:

→ **berekening** van de maatstaf van heffing en het btw-bedrag door de FDM:

De FDM zal zich hierbij baseren op de informatie die door de kassa wordt meegegeven over de subproducten.

De basis wordt dan, per vat per subProduct: $som(vat.price + som(priceChange.amount))$.

→ **doorsturen** van lineTotal naar de FDM:

Voor het mainProduct wordt hiervoor steeds de som(vat.price + som(priceChange.amount)) van alle betrokken subProducten doorgestuurd.

→ **afdrukken** van de artikellijn van het composite_product:

Op de afdruk van het btw-kasticket komt op productlijnniveau enkel het composite_product, met alle betrokken btw-tarieven, met als totaalprijs de som(vat.price + som(priceChange.amount)) van alle betrokken subProducten.

product

Maakt het onderwerp uit van een lijn in een transactie waar het de details bevat van de levering van een goed of dienst of onderdeel is van een samengesteld product. Leveringen van goederen of diensten moeten in de POS tot op productniveau worden gedetailleerd (een rechtstreekse registratie op een departement is bijgevolg niet toegelaten).

{

gtin (string, optioneel, lengte min. 8 en max. 20 tekens)

Global Trade Item Number.

Voorbeeld: "0811571013579"

productId (string, verplicht, lengte min. 1 en max. 600 tekens)

Intern productnummer in het kassasysteem.

productName (string, verplicht, lengte min. 1 en max. 600 tekens)

Productnaam zoals opgeslagen in het kassasysteem.

departmentId (string, verplicht, lengte min. 1 en max. 600 tekens)

Intern departement-id van het kassasysteem, waaraan het betrokken product is gekoppeld.

departmentName (string, verplicht, lengte min. 1 en max. 600 tekens)

Naam van het departement waaraan het betrokken product in het kassasysteem is gekoppeld.

quantity (numeriek, verplicht, formaat = veelvoud van 0.0001)

Aantal eenheden van het product opgenomen in deze artikellijn. Aantal kan positief, negatief of nul (0) zijn.

Voorbeelden: 1245.4789, 2.5, -17.23.

negQuantityReason (enum, conditioneel).

Reden waarom een negatieve hoeveelheid werd geregistreerd, te kiezen uit onderstaande tabel.

negQuantityReason
REFUND
CORRECTION
PRICE_CHANGE
COST_CENTER_CHANGE
PRODUCT_SUBSTITUTION
VOUCHER
OTHER

- Onder *refund* wordt verstaan: een 'goed' of 'dienst', onderdeel van een eerder volledig afgesloten verkooptransactie, dat wordt teruggenomen.
- Onder *correction* wordt verstaan: elke aanpassing van een hoeveelheid in een nog niet definitief afgesloten verkooptransactie (voorbeelden: lijnannulatie, correctie, ticketafbreking)
- *price_change*: wanneer een 'kunstmatige' productlijn wordt toegevoegd om een prijswijziging op een eerder geboekte productlijn in een nog niet definitief afgesloten transactie correct te kunnen registreren.

Voorbeeld: de oorspronkelijke bestelling bevatte 2 pils; na doorzenden van de bestelling wordt later een korting op 1 van die 2 pils gegeven; hiervoor zal eerst 1 pils gecorrigeerd worden om daarna opnieuw 1 pils met korting te registreren.

quantityType (enum, verplicht)

De eenheid waarin de quantity is uitgedrukt, waarde te kiezen uit de tb_quantityType

quantityType
PIECE
KILOGRAM
METER
LITRE
HOUR

unitPrice (numeriek, vier decimalen, verplicht)

Normale prijs van 1 eenheid van het product, opgenomen in de productlijn, vóór toepassing van prijswijzigingen. Kan positief, negatief of nul (0) zijn.

vats (array van vat objecten, verplicht).

Per btw-label dat gelinkt is aan het betreffende product wordt een vat object toegevoegd. Deze array kan maximaal 5 vat objecten bevatten.

Opmerking: Bij een composite_product zal de vats array bij het mainProduct **leeg** zijn. Alle waarden worden immers opgenomen onder de subProducts.

Het vat object wordt beschreven in Vat.

}

vat

Elke lijn op een transactie is minstens gevat door één btw-tarief (zie nr. 18 van het ministerieel besluit). Hieraan kunnen ook één of meerdere prijswijzigingen verbonden zijn.

{

label (enum, verplicht)

De btw-code, zoals voorzien in nr. 18 van het ministerieel besluit, dat van toepassing is op het betrokken artikel.

label	BTW-TARIEF OMSCHRIJVING	BTW-TARIEF
A	Hoog	21 %
B	Midden	12 %
C	Laag	6 %
D	Nultarief	0 %
X	Buiten btw-toepassingsgebied	Geen

price (numeriek, twee decimalen, verplicht)

Gedeelte van de verkoopprijs (incl. btw) van het betrokken artikel dat onderhevig is aan voormeld btw-tarief, vóór toepassing van prijswijzigingen. Dit bedrag kan zowel positief, negatief als nul zijn.

priceChanges (array van priceChange objecten)

Deze array wordt enkel gebruikt indien aan dit gedeelte van de verkoopprijs van het betrokken artikel prijswijzigingen werden toegepast tijdens dit event. Deze array kan meerdere objecten bevatten indien meerdere prijswijzigingen van toepassing zijn. Er worden max. 99 objecten toegelaten.

Het priceChange object wordt beschreven bij priceChange.

}

priceChange

Per toegepaste prijswijziging wordt een object gebruikt. Deze wijziging kan in + of in - zijn.

{

groupingId (numeriek, verplicht, waarde > 0)

Dit nummer groepeert een prijswijziging van toepassing op meerdere producten. Dit kunnen meerdere producten zijn binnen de scope van één bepaalde transactielijn of binnen de scope van de volledige transactie.

id (string, verplicht, lengte min. 1 en max. 600 tekens)

Id van de prijswijziging, zoals voorzien in het kassasysteem, taalafhankelijk.

name (string, verplicht, lengte min. 1 en max. 600 tekens)

Naam van de prijswijziging, zoals voorzien in het kassasysteem. Voorbeelden: "menukorting", "happy hour", "on the house".

scope (enum, verplicht)

Duidt aan waarop de prijswijziging van toepassing is. Waarde te kiezen uit de tabel tb_priceChangeScope.

scope	omschrijving
LINE	transactionLine price change
EVENT	Event price change

type (enum, verplicht)

Identificatie van het soort prijswijziging, te kiezen uit de tabel tb_priceChangeType.

type	omschrijving
PUBLIC	Zichtbaar voor klant
INTERNAL	Interne herberekening

amount (numeriek, twee decimalen, verplicht)

Bedrag van de prijswijziging. Dit kan zowel positief, negatief als nul zijn.

}

Registratie Transacties – Financiële Bewegingen

Drawer

drawer (object, optioneel)

Mag in de root van de JSON structuur enkel gebruikt worden met een eventOperation DRAWER_OPEN, wanneer de geldlade opengaat zonder transactie en als zodanig geregistreerd wordt in het kassasysteem. drawer kan gebruikt worden indien de onderneming exact wil meegeven welke geldlade/geldbeugel wordt/werd gebruikt en voor zover de kassa dit ook ondersteunt.

Voor het gebruik van drawer in de overige transactietypes, zie financialLine.

{

id (string, verplicht, lengte min. 1 en max. 600 tekens)

Identificatie van de geldlade/-beugel.

name (string, verplicht, lengte min. 1 en max. 600 tekens)

Naam van de geldlade/-beugel, zoals voorzien in het kassasysteem.

}

Geldebewegingen

financials (array van paymentLine of moneyInOutLine objecten)

Deze array bevat alle financiële lijnen van een event, in EUR of valuta. Deze array kan gebruikt worden bij events 'N', wanneer ze betrekking hebben op een verkooptransactie, of 'F' wanneer het gaat om een loutere geldebeweging, of 'P', wanneer het een voorlopige rekening betreft (indien de klant al op voorhand vouchers, cadeaubon, ... heeft voorgelegd dient dit uiteraard al in mindering van het te betalen bedrag getoond te worden).

De objecten paymentLine en moneyInOutLine omvatten een geldebeweging van een betaalmiddel en de betekenis van het bedrag.

Betaallijnen

Bij een event 'N' betekent dit de registratie van een betaling of uitgestelde betaling. Indien het te betalen bedrag nul is wordt een lege financials array toegelaten.

In het geval van een uitgestelde betaling wordt het betaalmiddel 'CUSTOMER_CREDIT' gebruikt, bij een overboeking naar een hotelkamer binnen een event N wordt ROOM_CREDIT gebruikt.

Loutere geldebewegingen

Bij een event 'F' betekent dit de registratie van financiële bewegingen zonder verband met verkopen of een wijziging van een betaalmiddel van een reeds afgesloten verkooptransactie.

paymentLine (object)

Het object paymentLine laat toe om een **betaling** correct te registreren om een perfect beheer van de ontvangsten mogelijk te maken.

{

id (string, verplicht, lengte min. 1 en max. 600 tekens)

Id van het betaalmiddel, zoals voorzien in het kassasysteem, taalonafhankelijk.

name (string, verplicht, lengte min. 1 en max. 600 tekens)

Naam van het gebruikte betaalmiddel, zoals gecodeerd in het kassasysteem. Voorbeelden: "Cash", "Debetkaart".

type (enum, verplicht)

Type van betaalmiddel, te kiezen uit de tabel tb_paymentTypes

tb_paymentTypes

type	Opmerking
UNKNOWN	Type niet geregistreerd
CASH	Contant
CARD_DEBIT	Kaartbetaling met debetkaart
CARD_UNKNOWN	Kaartbetaling met niet gespecificeerde kaart
CARD_CREDIT	Kaartbetaling met kredietkaart
CARD_OTHER	Kaartbetaling niet eerder vermeld
CHEQUE_MEAL	Betaling met maaltijdcheque
CHEQUE_OTHER	Betaling cheque (eco, cultuur, ...)
APP	Betaling met betalingsapp
ONLINE	Online betaling
CUSTOMER_CREDIT	Betaling op krediet/uitgesteld
ROOM_CREDIT	Betaling via globale hotelfactuur
LOYALTY_REWARDS	Betaling met loyalty rewards
VOUCHER_STORE	Betaling met voucher van de zaak
VOUCHER_SUPPLIER	Betaling met voucher van leverancier
VOUCHER_OTHER	Betaling met niet eerdere vermelde voucher
OTHER	Elke andere niet eerdere vermeld

provider (string, optioneel, max. lengte = 600 tekens)

Uitreiker van het betaalmiddel (voorbeeld: Visa, MasterCard, Edenred, ...).

inputMethod (enum, verplicht)

Eén van de waarden uit de tabel tb_inputMethod:

Type	omschrijving
MANUAL	Manueel geregistreerd
AUTOMATIC	EFT, andere koppelingen

Indien het kassasysteem rechtstreeks verbonden is met de betaalterminal (en dus de betaling via EFT registreert), dan stuurt de betaalterminal zelf terug welke soort kaart werd gebruikt.

amount (numeriek, twee decimalen, verplicht)

Bedrag van de betaallijn of geldbeweging, uitgedrukt in euro. Dit bedrag kan zowel positief, negatief als nul (0) zijn.

amountType (enum, verplicht)

Beschrijft waarop het bedrag slaat, soort te kiezen uit de tb_ amountType:

Type	omschrijving
PAYMENT	Betaling
TIP	Fooi
ROUNDING	Afronding bij betaling

- Tip: wordt gebruikt om ontvangen fooien te registreren. Een fooi kan ontvangen worden binnen het kader van een transactie of kan geregistreerd worden met een geldbeweging.
- Rounding: verplicht te gebruiken om de afrondingen op betaalwijzen te registreren¹.

Voorbeeld: te betalen bedrag, 9,97 en de klant betaalt cash. Kassa moet afronden naar 9,95. Dit betekent dat twee betaallijnen moeten vermeld worden.

```
{ "name" : "CONTANT", "type" : "CASH", "inputMethod": "MANUAL", "amount" : 9.97, "amountType": "PAYMENT"},
```

```
{ "name" : "CONTANT", "type" : "CASH", "inputMethod": "MANUAL", "amount": -0.02, "amountType": "ROUNDING" }
```

foreignCurrency (object, optioneel, zie foreignCurrency)

Dit object wordt enkel gebruikt indien het kassasysteem betalingen in vreemde valuta ondersteunt.

Opgelet: dit object mag niet gebruikt worden in combinatie met het amountType ROUNDING;

{

amount (numeriek, verplicht)

Het bedrag in vreemde valuta. Dit bedrag kan zowel positief, negatief als nul (0) zijn.

iso (string, verplicht, lengte = 3)

ISO 4217 notatie (alfabetische code in hoofdletters) van de betrokken munt.
Voorbeelden: "USD", "GBP", "SEK".

}

reference (string, optioneel, lengte min. 1 en max. 600 tekens)

Indien gebruikt door het kassasysteem, kan hier de betaalreferentie meegegeven worden.

drawer (object, optioneel, beschrijving zie eerder)

}

moneyInOutLine (object)

¹ De artikelen VI.7/1 en VI.7/2 van het Wetboek Economisch Recht verplichten de handelaar het totaalbedrag van de betaling in specien door de consument af te ronden naar het lagere of hogere veelvoud van 5 cent. Mits aan bepaalde voorwaarden wordt voldaan (zie website FOD Economie), mag de handelaar ook alle andere betaalwijzen afronden. Zie punten 46 en volgende van dit ministerieel besluit voor alle informatie hieromtrent.

Het object moneyInOutLine laat toe om een **geldbeweging** correct te registreren om een perfect beheer van de ontvangsten mogelijk te maken.

```
{  
  id (string, verplicht, lengte min. 1 en max. 600 tekens)  
    Id van paymentLine, zoals voorzien in het kassasysteem, taalafhankelijk.  
  name (zie paymentLine.name)  
  type (zie paymentLine.type)  
  provider (zie paymentLine.provider)  
  inputMethod (zie paymentLine.inputMethod)  
  amount (zie paymentLine.amount)  
  amountType (enum, verplicht)
```

Beschrijft waarop het bedrag slaat, soort te kiezen uit de tb_ amountType:

amountType	omschrijving
MONEY_IN_OUT	Wijziging in + of – van de inhoud van de geldlade
TIP	Fooi
DRAWER_DECLARATION	Telling van inhoud geldlade

Tip: wordt gebruikt om ontvangen foaien te registreren. Een fooi kan ontvangen worden binnen het kader van een transactie of kan geregistreerd worden met een geldbeweging.

```
  foreignCurrency (zie paymentLine.foreignCurrency)  
  reference (zie paymentLine.reference)  
  drawer (zie paymentLine.drawer)  
}
```

Rapporten

Deze worden opgebouwd met de inhoud van onderstaande values, objecten of arrays.

reportNo (numeriek, enkel bij Z rapport, waarde tussen 1 en 999999999)

Doorlopende nummering van het rapport (enkel Z), zie nr. 27 van het ministerieel besluit. Maakt deel uit van zowel het omzet- als gebruikersrapport.

reportBookingDate (string, enkel bij Z rapport)

De bookingDate waarop dit rapport betrekking heeft. in het eerder vermelde ISO 8601 formaat.

posDevices (array van posDevice objecten, verplicht)

Bevat de informatie van alle kassa's waarvan cijfers zijn opgenomen in dit rapport. Maakt deel uit van zowel het omzet- als gebruikersrapport.

fdmDevices (array van fdmDevice objecten, verplicht)

Bevat de informatie van alle FDM's, gekoppeld aan de kassa en waarvan cijfers zijn opgenomen in dit rapport. Maakt deel uit van zowel het omzet- als gebruikersrapport.

turnover (object, verplicht onderdeel van het omzetrapport)

Bevat alle omzetgerelateerde informatie voor de betrokken reportBookingDate.

users (array van objecten, verplicht onderdeel van het gebruikersrapport)

Bevat alle relevante informatie omtrent de kassagebruikers voor de betrokken reportBookingDate.

posDevice

Dit object bevat de identiteit en het timeframe van alle POS waarvan de gegevens zijn opgenomen in dit rapport.

{

posId (string, verplicht)

De posId zoals gebruikt tijdens de registratie van de events.

terminalId (string, verplicht)

De terminalId zoals gebruikt tijdens de registratie van de events.

firstPosDateTime (string, verplicht)

De timestamp van de eerste registratie met deze posId en terminalId combinatie voor de gegeven bookingDate

lastPosDateTime (string, verplicht)

De timestamp van de laatste registratie met dit posId en terminalId combinatie voor de gegeven bookingDate.

}

fdmDevice

Dit object bevat de detailgegevens van alle FDM waarvan cijfers zijn opgenomen in het rapport en de referentie naar de eerste en laatste transactie die er in opgenomen zijn.

{

fdmId (string, verplicht)

firstFdmDateTime (string, verplicht)

firstTotalCounter (numeriek, verplicht, min. 1 en max. 999999999)

lastFdmDateTime (string, verplicht)

lastTotalCounter (numeriek, verplicht, min. 1 en max. 999999999)

}

turnover

Dit conditioneel object bevat de totaal gerealiseerde omzet, de omzet per departement en per btw-tarief. Het wordt enkel gebruikt in het omzetrapport.

{

transactions (array van alle gebruikte eventLabels met hun totalen)

[

{

eventLabel (string, verplicht, zie eerdere beschrijving)

ticketCount (integer, verplicht)

Totaal van het aantal events van het betrokken eventLabel tijdens de betrokken bookingDate.

amount (numeriek, verplicht, twee decimalen)

Som(VatInput.Price)+Som(VatInput.PricesChanges.Amount) van elke ProductInput.

}

]

departments (array van objecten die de omzetgegevens voor alle gebruikte departementen bevatten, verplicht)

[

{

departmentId (string, verplicht, zie eerdere beschrijving)

departmentName (string, verplicht, zie eerdere beschrijving)

amount (numeriek, twee decimalen, verplicht)

som(VatinInput.Price)+som(VatInput.PriceChanges.Amount), beperkt tot het eventLabel "N" voor het betrokken Departement tijdens de betrokken bookingDate.

}

]

vats (array van objecten die de omzetgegevens voor alle gebruikte btw-tarieven bevatten, verplicht)

[

{

label (string, verplicht ,zie eerdere beschrijving)

rate (numeriek, verplicht, min. waarde = 0 en max. waarde = 100, 2 decimalen, zie beschrijving onder 2.2.4)

taxableAmount (numeriek, twee decimalen, verplicht)

Het totaalbedrag van de waarden van de taxableAmount (zie beschrijving onder 2.2.4) voor het betrokken label.

vatAmount (numeriek, twee decimalen, verplicht)

Het totaalbedrag van de waarden van de vatAmount (zie beschrijving onder 2.2.4) voor het betrokken label

totalAmount (numeriek, twee decimalen, verplicht)

Het totaalbedrag van de waarden van alle totalAmount (zie beschrijving onder 2.2.4) voor het betrokken label

```
}  
]
```

payments (array van objecten die de ontvangen bedragen met betrekking tot de eerder vermelde omzetten bevatten, per type betaalwijze, verplicht)

```
[  
{
```

id (string, verplicht, zie eerdere beschrijving)

name (string, verplicht, zie eerdere beschrijving)

type (enum, verplicht, zie eerdere beschrijving)

amount (array van PaymentTotalAmount objecten, verplicht)

```
[  
{
```

type (enum PaymentLineType, verplicht)

normalAmount (numeriek, twee decimalen, verplicht)

Bevat het totaalbedrag van alle betalingen met betrekking tot de gerealiseerde omzet, te vermelden in euro (Sale).

negativeCorrections (numeriek, twee decimalen, verplicht)

Bevat het totaalbedrag van de negatieve betalingscorrecties.

positiveCorrections (numeriek, twee decimalen, verplicht)

Bevat het totaalbedrag van de positieve betalingscorrecties.

correctionsCount (integer, verplicht)

Het totaal aantal betalingscorrecties.

totalAmount (numeriek, twee decimalen, verplicht)

Het totaal van de normalAmount, negativeCorrections en positiveCorrections

normalForeign (array van foreignCurrency objecten, conditioneel)

Bevat het totaalbedrag per valuta van alle betalingen in vreemde valuta met betrekking tot de gerealiseerde omzet (Sale).

correctionsForeign (array van foreignCurrency objecten, conditioneel)

Bevat het totaalbedrag van de betaalcorrecties in vreemde valuta. Deze bedragen zitten al verrekend in euro in de andere velden (negativeCorrections, positiveCorrections en totalAmount).

```
}
```

```
]
}
```

drawersOpenCount (numeriek, min. 0 en max. 999999999, verplicht)

Bevat het aantal keren dat één of meerdere geldlade(s) werd(en) geopend (via het kassasysteem) zonder verkoopregistratie.

negQuantities (array van objecten, verplicht)

Bevat het aantal en totaalbedrag voor alle gebruikte negatieve aantallen.

```
[
{
```

negQuantityReason (enum, verplicht, zie eerdere beschrijving)

negQuantityCount (numeriek, verplicht, min. 1 en max. 999999999)

Het aantal keer dat betrokken negatieve registratie werd gebruikt gedurende de bookingDate.

ticketCount (numeriek, min. 1 en max. 999999999, verplicht)

Het aantal tickets voor die reason.

amount (numeriek, twee decimalen, verplicht)

Het totaalbedrag van deze negatieve registraties voor dit type.

```
}
]
```

priceChanges (array van objecten, verplicht)

Bevat het volledige overzicht van de geboekte prijswijzigingen

```
[
{
```

id (string, verplicht, lengte min. 1 en max. 600 tekens)

Id van de prijswijziging, zoals voorzien in het kassasysteem, taalafhankelijk.

name (string, verplicht, zie eerdere beschrijving van priceChange.name)

type (enum, verplicht, zie eerdere beschrijving van priceChange.type)

Te beperken tot de priceChange.type PUBLIC.

amount (array van objecten, verplicht)

Bevat het totaalbedrag alle prijswijzigingen tijdens betrokken bookingDate, uitgesplitst per gebruikt btw-tarief en naargelang het positieve of negatieve prijswijzigingen betreft)

```
[
{
```

label: (string, verplicht, zie eerdere beschrijving)

negative: (numeriek, twee decimalen, verplicht)

Is de som van alle **negatieve** prijswijzigingen voor het gegeven btw-tarief.

positive (numeriek, twee decimalen, verplicht)

Is de som van alle **positieve** prijswijzigingen voor het gegeven btw-tarief.

}

]

}

]

invoices (array van objecten, conditioneel)

Overzicht van de met de POS opgestelde facturen tijdens de betrokken bookingDate.

[

{

invoiceNo (string, verplicht, zie eerdere beschrijving)

amount (numeriek, 2 decimalen, verplicht, zie eerdere beschrijving van transactionTotal)

}

]

}

users

Deze array met objecten bevat de informatie van het gebruikersrapport, uitgesplitst per gebruiker.

[

{

employeed (string, verplicht, zie eerdere beschrijving)

totalAmount (numeriek, twee decimalen, verplicht).

Het totaalbedrag van alle transactionTotal die werden geregistreerd via de events 'N' door de betrokken gebruiker.

firstPosDateTime (string, verplicht, formaat zoals beschreven voor posDateTime)

Timestamp van de **eerste** registratie van een event door de betrokken gebruiker in deze periode.

lastPosDateTime (string, verplicht, formaat zoals beschreven voor posDateTime)

Timestamp van de **laatste** registratie van een event door de betrokken gebruiker in deze periode.

socialEvents (array van objecten, verplicht)

Bevat de timestamps van het in-/uitloggen van de kassa voor de betrokken gebruiker via het gebruik van het event 'S'.

[


```

{
  posDateTime (string, verplicht, zie eerdere beschrijving)
    Timestamp van de registratie van het betrokken event 'S'.

  inOut (enum, verplicht)
    Waarde te kiezen uit de TB_inOut



| inOut      | omschrijving |
|------------|--------------|
| <b>IN</b>  | Work In      |
| <b>OUT</b> | Work Out     |



}
]

```

payments (array van objecten, verplicht)

Deze array bevat de samenvatting van alle registraties van alle betalingen voor de betrokken gebruiker voor deze bookingDate, uitgesplit per gebruikte soort). Deze heeft dezelfde structuur als deze bij het turnover object.

Communicatieregels

Om een correcte afhandeling van de mutaties te kunnen garanderen en deze uniek te kunnen onderscheiden worden volgende sleutelvelden bepaald:

- **posId**
- **posDateTime**
- **terminalId**
- **eventLabel**
- **posFiscalTicketNo**

Wanneer een POS binnen een tijdspanne van 10 minuten een mutatie doorstuurt naar de FDM, waarbij deze 5 sleutelvelden identiek zijn aan deze verstuurd in een vorig bericht, dan moet de FDM verifiëren of dit al of niet een mutatie betreft waarop al eerder werd geantwoord (inclusief het verhogen van de betrokken tellers op de FDM).

Volgende situaties kunnen zich voordoen:

1. De combinatie van sleutelvelden is verschillend: de FDM behandelt deze mutatie op de normale manier.
2. De inhoud van de mutaties is volledig identiek: de FDM behandelt deze mutatie als een "dubbele" zending en stuurt in het response dezelfde gegevens terug als in de oorspronkelijke mutatie; de FDM verhoogt zijn tellers ook niet; deze dubbele transactie wordt niet bijgehouden op de FDM.
3. De combinatie van sleutelvelden is identiek, maar de andere waarden in de mutatie zijn verschillend dan die in de eerder verwerkte mutatie: de FDM weigert deze mutatie te behandelen en stuurt een errorcode mee in zijn antwoord.

2.2.3. FDM → POS – volledige overzicht

Na ontvangst van de hierboven beschreven query, voert de FDM de benodigde berekeningen uit, past de nodige tellers aan, maakt het canonical JSON object aan, ondertekent de data.

De GraphQL service van de FDM stuurt de gevraagde velden terug naar de POS.

Hieronder het volledige response overzicht:

posId (scalar value)
posFiscalTicketNo (scalar value)
posDateTime (scalar value)
terminalId (scalar value)
deviceId (scalar value)
eventOperation (scalar value)
fdmRef (object)
fdmSwVersion (scalar value)
digitalSignature (scalar value)
shortSignature (scalar value)
verificationUrl (scalar value)
vatCalc (array van objecten)
bufferCapacityUsed (scalar value)
warnings (array van objecten)
informations (array van objecten)

2.2.4. FDM → POS – gedetailleerd overzicht

Een aantal arrays, objecten of values die nog niet eerder beschreven werden, worden in dit punt beschreven.

eventOperation (enum, verplicht)

Waarde uit de tabel `tb_eventOperation`.

`tb_eventOperation`

eventOperation	eventLabel	omschrijving
WORK_IN	S	Inklokken op de kassa
WORK_OUT	S	Uitklokken op de kassa
SALE	N	Afsluiting verkooptransactie
INVOICE	I	Factuur van eerder aangemaakt btw-kasticket
COST_CENTER_CHANGE	P	Wijziging van de toewijzing aan cost center
ORDER	P	Registratie bestelling op costcenter
PRE_BILL	P	Aanmaak voorlopige rekening
MONEY_IN_OUT	F	Registratie geldbeweging buiten verkoop
DRAWER_OPEN	F	Opening geldlade buiten verkoop
PAYMENT_CORRECTION	F	Wijziging eerder geboekte betaalwijze
REPORT_TURNOVER_X	R	Omzetrapport X
REPORT_TURNOVER_Z	R	Omzetrapport Z
REPORT_USER_X	R	Gebruikersrapport X
REPORT_USER_Z	R	Gebruikersrapport Z
COPY	C	Kopie van een hierboven vermelde event operation
(eventOperation)	T	Alle vorige eventOperation in trainingsmodus

fdmRef (object, verplicht)

Bevat de identificatie, timestamp en tellerstanden van de FDM. Hierdoor wordt de getekende transactie op een unieke manier geïdentificeerd.

{

fdmId (string, verplicht)

Het serienummer van de FDM die het request beantwoordt. Lengte = 11.
Voorbeeld: FOD01987654

fdmDateTime (string, verplicht)

De datum en tijd van het antwoord komende van de RTC van de FDM. Hiervoor wordt steeds **UTC+00** gebruikt (dus: geen gebruik van zomer- of wintertijd). Het volgende ISO 8601 formaat wordt hierbij verplicht gebruikt.

Voorbeeld: 2022-10-20T15:01:26Z

eventLabel (enum, verplicht, lengte = 1)

Het eventLabel dat correspondeert met de mutatie die door de POS werd verstuurd naar de GraphQL service van de FDM, op basis waarvan de FDM de juiste teller heeft verhoogd. In dit antwoord wordt dus het eventtype zelf teruggestuurd (zie hoofdstuk 3).

Voorbeeld: 'N'.

eventCounter (numeriek, verplicht, min. 1 en max. 999999999)

De event teller. De FDM heeft een aantal ingebouwde tellers, zie nr. 64 van het ministerieel besluit. De FDM geeft hier de laatste update (na aanpassing op basis van de inhoud van betreffend request) van de teller van het betrokken event mee.

Voorbeeld: 46895

totalCounter (numeriek, verplicht, min. 1 en max. 999999999)

De totaalevent teller. De FDM heeft een aantal ingebouwde tellers, zie nr. 64 van dit besluit. De FDM geeft hier de laatste waarde (na aanpassing op basis van de inhoud van betreffend request) van de totaal teller mee.

Voorbeeld: 53896

}

digitalSignature (string, verplicht, min 1 karakter)

Dit veld bevat de digitale handtekening die werd geplaatst door het handtekencertificaat van de FDM. Deze handtekening wordt geplaatst volgens de regels beschreven in punt 2.4. van dit document.

shortSignature (string, conditioneel, min 1 karakter)

Dit veld bevat de verkorte digitale handtekening. Het is deze handtekening die op het btw-kasticket zal worden **afgedrukt**. Bijgevolg wordt ze **enkel** meegestuurd in het geval van een event 'NORMAL'. Ze wordt als volgt berekend:

shortSignature = Hex(SHA1(Base64Decode(digitalSignature))).

verificationUrl (string, conditioneel, min. 1, max. 60)

Deze url wordt door de FDM gegenereerd om als QR code te worden afgedrukt op het btw-kasticket (zie sectie 6.5). De prefix wordt door de FODFIN meegegeven aan de FDM die de url verder aanmaakt op basis van bepaalde event data.

Deze string wordt enkel aan de response JSON toegevoegd voor events van het type 'N'.

fdmSwVersion (string, verplicht, min. 1 en max. 10 tekens)

De versie van de FDM firmware op het ogenblik van antwoord. Voorbeeld: "1.2.0".

bufferCapacityUsed (numeriek, verplicht, 2 decimalen)

De opgebruikte buffercapaciteit uitgedrukt in percentage.

vatCalc (array van vatCalc objecten, verplicht indien vat-objecten in het request vervat zitten)

De FDM staat in voor de correcte berekening van de maatstaf van heffing en de btw voor elk toepasselijk tarief. Deze berekening gebeurt enkel voor de transactietype/eventlabel 'N'.

Per btw-code wordt de maatstaf van heffing en het btw-bedrag berekend.

De inhoud van de velden vatId en rate wordt bepaald volgens de gegevens die door de FOD Financiën werden verstrekt (art. 66 van het ministerieel besluit). De inhoud van de velden taxableAmount en vatAmount wordt door de FDM berekend.

Enkel voor de vatLabels waarvoor in het request productlijnen werden doorgestuurd wordt een object ingevuld.

[

{

label (enum, verplicht)

Het btw-label zoals ontvangen van de FOD Financiën. Voorbeeld: "B"

rate (numeriek, verplicht, min. waarde = 0 en max. waarde = 100, 2 decimalen)

Het percentage voor het betrokken btw-label, zoals door de FDM ontvangen van de FOD Financiën.

Voorbeeld: 12

taxableAmount (numeriek, twee decimalen, verplicht)

Maatstaf van heffing, voor het betrokken btw-tarief. Voor de berekening wordt verwezen naar nr. 66 van het ministerieel besluit. Deze waarde kan zowel positief, negatief als nul (0) zijn.

Voorbeeld: 8.93

vatAmount (numeriek, twee decimalen, verplicht)

Btw-bedrag, voor het betrokken btw-tarief en berekend op basis van de hierboven doorgegeven maatstaf van heffing. Voor de berekening wordt verwezen naar nr. 66 van het ministerieel besluit. Deze waarde kan zowel positief, negatief als nul (0) zijn.

Voorbeeld: 1.07

totalAmount: (numeriek, twee decimalen, verplicht)

outOfScope: (true of false, verplicht)

}

]

warnings (array van message objecten, optioneel, max. 50 objecten, zie beschrijving)

Lijst met waarschuwingen en/of errors waarop de FDM de kassa attent wil/moet maken.

Message

{

message (string, verplicht)

Omschrijving van de boodschap, zoals gedefinieerd in dit besluit of door de producent van de FDM. De gebruikte taal komt overeen met de taal zoals aangegeven in het language veld van de mutatie. Voor de boodschappen gelieerd aan de SPF_FOD categorie dienen deze gebaseerd te zijn op de messages uit tb_messages hieronder. De kolom nature geeft weer of het een warning of error betreft. De kolom display verduidelijkt:

- **Mandatory:** de melding wordt verplicht getoond op de UI van POS
- **Optional:** de POS bepaalt zelf of deze melding getoond wordt op de UI of niet. (Dit kan bijvoorbeeld een instelling op de kassa zijn: 'alle meldingen tonen' of 'enkel de verplichte meldingen tonen')
- **Never:** deze meldingen mogen nooit getoond worden op de UI van de POS.

Het tonen van de boodschap gebeurt:

- bij warnings: minstens éénmaal per bookingDate, na de ontvangen respons (met de melding) op de gestuurde mutatie;
- bij errors: elke keer na ontvangst van de response (met de melding) op de gestuurde mutatie.

tb_messages

Code	Message	category	nature	display
CLIENT_CERT_NEAR_EXPIRATION	Client certificate expires in less than 4 months	SPF_FOD	Warning	O
CLIENT_CERT_EXPIRED	Client certificate has expired	SPF_FOD	Warning	M
RTC_SYNC_FAILED	FDM could not synchronize real-time clock.	FDM/SPF_FOD	Warning	O
UPDATE_URLS_FAILED	FDM could not update URLs.	SPF_FOD	Warning	O
UPDATE_TRUST_CERT_FAILED	FDM fails to update trust certificates	SPF_FOD	Warning	O
UPDATE_TASK_LIST_FAILED	FDM could not download the task list.	SPF_FOD	Warning	O
TASK_FEEDBACK_FAILED	FDM could not report the task result	SPF_FOD	Warning	O
NOP_FAILED	FDM could not post a NOP message	SPF_FOD	Warning	O
BUFFER_NEAR_FULL	FDM buffer usage exceeds 70 %	SPF_FOD	Warning	M
SERVER_CERT_RESOLVE_FAILED	FDM could not resolve server certificate	SPF_FOD	Warning	O
TRANSACTION_UPLOAD_FAILED	FDM has more than 10 failed attempts to post business transaction	SPF_FOD	Warning	O
INITIALIZATION_FAILED	FDM initialization has failed	SPF_FOD	Warning	O
RTC_NOT_INITIALIZED	FDM Real Time Clock not synchronized	SPF_FOD	Warning	N
CORRUPT_RECORD_ENCOUNTERED	FDM memory contains corrupt transaction	SPF_FOD	Warning	N
UPDATE_PARAMS_FAILED	FDM could not update parameters	SPF_FOD	Warning	N

Code	Message	category	nature	display
UPDATE_CLIENT_CERT_FAILED	FDM could not update the client certificate	SPF_FOD	Warning	N
UPDATE_VAT_RATES_FAILED	FDM could not update the vat rates	SPF_FOD	Warning	N
UPDATE_POS_ALLOWLIST_FAILED	FDM could not update the POS allowlist	SPF_FOD	Warning	N
DUPLICATE_REQUEST	FDM received a duplicate of a earlier request	FDM	Warning	N
BUFFER_FULL	FDM buffer is full	SPF_FOD	Error	M
FDM_LOCKED	FDM is locked	SPF_FOD	Error	M
TOO_MANY_MEMORY_ERRORS	FDM's memory contains too many corrupt transactions	SPF_FOD	Error	N
UNAUTHORIZED	FDM received unauthorized request	FDM	Error	O
INVALID_REQUEST	FDM could not treat the request	FDM	Error	O
INTERNAL_ERROR	FDM encountered a technical issue	FDM	Error	O
UNDEFINED_ERROR	FDM encountered a undefined error	FDM	Error	O
UNDEFINED_OTHER	Any other error encountered	SPF_FOD/FDM	Error/ Warning/ Information	O
FDM_NOT_OPERATIONAL	FDM is not functioning	FDM	Error	M
UNKNOWN_POS	FDM received message from POS not in posAllowList	FDM	Error	M

locations (array van locationItem objecten, optioneel)

Bevat de locaties in het request waarop de message betrekking heeft.

```
[
{
  line (integer, verplicht, waarde > 0)
  column (integer, verplicht, waarde > 0)
}
]
```

extensions (object, verplicht)

Bevat de volledige inhoud van de message.

```
{
  category (enum, verplicht)
```

Waarde te kiezen uit de tabel tb_messageCategory.

tb_messageCategory

category	Omschrijving
SPF_FOD	Codes gedefinieerd door de administratie. Hierbij wordt verplicht de codes uit tb_messages gebruikt.
FDM	Boodschappen afkomstig van de FDM – vrij door de producent in te vullen
OTHER	Andere boodschappen.

code (enum, verplicht)

De inhoud van de waarschuwing, error of informatie die verstrekt wordt aan de kassa door de FDM. Voor de SPF_FOD categorie worden enkel de codes uit de tabel tb_messages gebruikt. Voor de andere categorieën zijn die vrij te kiezen door de producenten.

data (array van data objecten, verplicht)

Kan bijkomende nuttige informatie bevatten, bedoeld voor de ontwikkelaars of support medewerkers. Is niet bedoeld om te tonen via de user interface van de kassa.

```
[  
{
```

name (string, verplicht)

Bevat de naam van de waarde. Bijvoorbeeld: Offset (toont waar de fout zit), Expected (wat er verwacht werd), Encountered (wat er gestuurd werd).

value (string, verplicht)

De extra waarde die meegegeven wordt aan een Message.

```
}
```

```
]
```

showPos (enum, verplicht)

Deze waarde bepaalt of de boodschap wordt getoond aan de gebruiker via de user interface van de POS. Waarden te kiezen uit de tb_display.

tb_display

Waarde	Omschrijving
MANDATORY	Boodschap moet getoond worden via de UI van de POS
OPTIONAL	Boodschap mag getoond worden via de UI van de POS (POS beslist hierover)
NEVER	Boodschap wordt nooit getoond via de UI van de POS

```
}
```

Voorbeeld:

```
{  
  "message": "FDM buffer usage exceeds 70%",  
  "locations": [],  
  "extensions": {  
    "category": "FPS_FIN",  
    "code": "BUFFER_NEAR_FULL",  
    "data": [],  
    "showPos": "MANDATORY"  
  }  
}
```

informations (array van Message objecten, optioneel, zie eerdere beschrijving)

Lijst met extra berichten waarover de FDM de kassa wil informeren. Bijvoorbeeld dat er een nieuw certificaat beschikbaar is.

footer (array van strings)

Hiermee kan de FODFIN informatie doorgeven die verplicht in de footer van het kasticket moeten worden afgedrukt. Deze informatie wordt via de FDM aan de POS doorgegeven. Default is deze array leeg.

2.2.5. FDM → POS errorhandling

Fouten worden geretourneerd in de errors array volgens het formaat van GraphQL. Elk error object in de array bevat verplicht een 'message' string, die een beschrijving van de fout geeft, optioneel een 'locations' array die aangeeft waar in de query de fout optrad en optioneel een 'path' array die het pad weergeeft naar het betrokken veld in de response.

Alle fouten met betrekking tot de syntax, validatie en runtime worden door de FDM bepaald.

Voorbeeld:

```
{
  "errors": [
    {
      "message": "Variable '$data' got invalid value 'N' at 'data.language'; Value 'N' does not exist in 'Language' enum. Did you mean the enum value 'EN' or 'NL'?",
      "locations": [
        {
          "line": 1,
          "column": 20
        }
      ]
    }
  ]
}
```

Voor het message object geldt dezelfde structuur als het message object in de warnings en informations array.

Voorbeeld error response:

```
{
  "errors": [
    {
      "message": "FDM buffer is full.",
      "extensions": {
        "category": "SPF_FOD",
        "code": "BUFFER_FULL",
        "showPos": "MANDATORY"
      }
    }
  ]
}
```

Algemene opmerking omtrent de behandeling van errors.

Bepaalde fouten en waarschuwingen worden getoond aan de gebruiker via de user interface van de POS. De FDM producent voorziet, bijkomend aan de bepalingen hierboven, de nodige bijkomende informatie voor de gebruiker zodat die gepast kan handelen om de fout op te lossen.

Dit wordt uitvoerig beschreven in de bij de certificatieaanvraag meegeleverde documentatie.

2.3. GRAPHQL SCHEMA COMMUNICATIE TUSSEN POS EN FDM

Hieronder wordt het volledige schema beschreven.

2.3.1. Mutations

Mutaties beschikbaar op de FDM:

- signWorkIn(data: WorkInOutInput!, isTraining: Boolean! = false): SignResult
- signWorkOut(data: WorkInOutInput!, isTraining: Boolean! = false): SignResult
- signInvoice(data: InvoiceInput!, isTraining: Boolean! = false): SignResult
- signSale(data: SaleInput!, isTraining: Boolean! = false): SignResult
- signCostCenterChange(data: CostCenterChangeInput!, isTraining: Boolean! = false): SignResult
- signOrder(data: OrderInput!, isTraining: Boolean! = false): SignResult
- signPreBill(data: PreBillInput!, isTraining: Boolean! = false): SignResult
- signMoneyInOut(data: MoneyInOutInput!, isTraining: Boolean! = false): SignResult
- signDrawerOpen(data: DrawerOpenInput!, isTraining: Boolean! = false): SignResult
- signPaymentCorrection(data: PaymentCorrectionInput!, isTraining: Boolean! = false): SignResult
- signReportTurnoverX(data: ReportTurnoverXInput!, isTraining: Boolean! = false): SignResult
- signReportTurnoverZ(data: ReportTurnoverZInput!, isTraining: Boolean! = false): SignResult
- signReportUserX(data: ReportUserXInput!, isTraining: Boolean! = false): SignResult
- signReportUserZ(data: ReportUserZInput!, isTraining: Boolean! = false): SignResult
- signCopy(data: CopyInput!, isTraining: Boolean! = false): SignResult

2.3.2. Types

```
type SignResult {  
  posId: String!  
  posFiscalTicketNo: Int!  
  posDateTime: String!  
  terminalId: String!  
  deviceId: String!  
  eventOperation: EventOperation!  
  fdmRef: FdmReference!  
  fdmSwVersion: String!  
  digitalSignature: String!  
  shortSignature: String  
  verificationUrl: String  
  vatCalc: [VatCalcItem!]  
  bufferCapacityUsed: Float!  
  warnings: [MessageItem!]
```

```
informations: [MessageItem!]
footer: [String!]
}
```

```
type FdmReference {
  fdmId: String!
  fdmDateTime: String!
  eventLabel: EventLabel!
  eventCounter: Int!
  totalCounter: Int!
}
```

```
type VatCalcItem {
  label: VatLabel!
  rate: Float!
  taxableAmount: Float!
  vatAmount: Float!
  totalAmount: Float!
  outOfScope: Boolean!
}
```

```
type MessageItem {
  message: String!
  locations: [LocationItem!]
  extensions: ExtensionItem!
}
```

```
type LocationItem {
  line: Int!
  column: Int!
}
```

```
type ExtensionItem {
  category: Category!
  code: Code!
  data: [DataItem!]
  showPos: Display!
}
```

```
type DataItem {
  name: String!
  value: String!
}
```

2.3.3. Enums

In dit onderdeel worden de verschillende enums beschreven. Raadpleeg steeds de meest recente versie via de website www.geregistreerdkassasysteem.be .

```
enum Language {
  EN
  NL
  FR
  DE
}
```

```
enum TicketMedium {  
  NONE  
  PAPER  
  DIGITAL  
  PAPER_DIGITAL  
}
```

```
enum InOut {  
  IN  
  OUT  
}
```

```
enum PriceChangeType {  
  PUBLIC  
  INTERNAL  
}
```

```
enum PriceChangeScope {  
  LINE  
  EVENT  
}
```

```
enum VatLabel {  
  A  
  B  
  C  
  D  
  X  
}
```

```
enum EventLabel {  
  N  
  P  
  F  
  S  
  I  
  R  
  C  
  T  
}
```

```
enum QuantityType {  
  PIECE  
  KILOGRAM  
  METER  
  LITRE  
  HOUR  
}
```

```
enum TransactionLineType {  
  SINGLE_PRODUCT  
  COMPOSITE_PRODUCT  
}
```

```
enum NegQuantityReason {  
  REFUND  
  CORRECTION  
}
```

```
PRICE_CHANGE
COST_CENTER_CHANGE
PRODUCT_SUBSTITUTION
VOUCHER
OTHER
}
```

```
enum PaymentType {
  UNKNOWN
  CASH
  CARD_UNKNOWN
  CARD_DEBIT
  CARD_CREDIT
  CARD_OTHER
  CHEQUE_MEAL
  CHEQUE_OTHER
  APP
  ONLINE
  CUSTOMER_CREDIT
  ROOM_CREDIT
  LOYALTY_REWARDS
  VOUCHER_STORE
  VOUCHER_SUPPLIER
  VOUCHER_OTHER
  OTHER
}
```

```
enum InputMethod {
  MANUAL
  AUTOMATIC
}
```

```
enum PaymentLineType {
  PAYMENT
  TIP
  ROUNDING
}
```

```
enum MoneyInOutLineType {
  MONEY_IN_OUT
  TIP
  DRAWER_DECLARATION
}
```

```
enum CostCenterType {
  TABLE
  CHAIR
  ROOM
  CUSTOMER
  ON_HOLD
  OTHER
}
```

```
enum EventOperation {
  WORK_IN
  WORK_OUT
  SALE
  INVOICE
  COST_CENTER_CHANGE
  ORDER
  PRE_BILL
  MONEY_IN_OUT
  DRAWER_OPEN
}
```

```

PAYMENT_CORRECTION
COPY
REPORT_TURNOVER_X
REPORT_TURNOVER_Z
REPORT_USER_X
REPORT_USER_Z
}
enum Category {
  SPF_FOD
  FDM
  OTHER
}
enum Display {
  MANDATORY
  OPTIONAL
  NEVER
}
enum Code {
  CLIENT_CERT_NEAR_EXPIRATION
  CLIENT_CERT_EXPIRED
  RTC_SYNC_FAILED
  UPDATE_URLS_FAILED
  UPDATE_TRUST_CERT_FAILED
  UPDATE_TASK_LIST_FAILED
  TASK_FEEDBACK_FAILED
  NOP_FAILED
  BUFFER_NEAR_FULL
  SERVER_CERT_RESOLVE_FAILED
  TRANSACTION_UPLOAD_FAILED
  INITIALIZATION_FAILED
  RTC_NOT_INITIALIZED
  CORRUPT_RECORD_ENCOUNTERED
  UPDATE_PARAMS_FAILED
  UPDATE_CLIENT_CERT_FAILED
  UPDATE_VAT_RATES_FAILED
  UPDATE_POS_ALLOWLIST_FAILED
  DUPLICATE_REQUEST
  BUFFER_FULL
  FDM_LOCKED
  TOO_MANY_MEMORY_ERRORS
  UNAUTHORIZED
  INVALID_REQUEST
  INTERNAL_ERROR
  UNDEFINED_ERROR
  UNDEFINED_OTHER
  FDM_NOT_OPERATIONAL
  UNKNOWN_POS
}

```

2.3.4. Input Objects

```

input FdmReferenceInput {
  fdmId: String!
  fdmDateTime: String!
  eventLabel: EventLabel!
  eventCounter: Int!
  totalCounter: Int!
}

```

```
input DrawerInput {
  id: String!
  name: String!
}
```

```
input CostCenterInput {
  id: String!
  type: CostCenterType!
  reference: String!
  costCenter: CostCenterInput
}
```

```
input PriceChangeInput {
  groupingId: Int!
  id: String!
  name: String!
  scope: PriceChangeScope!
  type: PriceChangeType!
  amount: Float!
}
```

```
input VatInput {
  label: VatLabel!
  price: Float!
  priceChanges: [PriceChangeInput!]
}
```

```
input ProductInput {
  gtin: String
  productId: String!
  productName: String!
  departmentId: String!
  departmentName: String!
  quantity: Float!
  quantityType: QuantityType!
  negQuantityReason: NegQuantityReason
  unitPrice: Float!
  vats: [VatInput!]
}
```

```
input ForeignCurrencyInput {
  amount: Float!
  iso: String!
}
```

```
input TransactionLineInput {
  lineType: TransactionLineType!
  mainProduct: ProductInput!
  subProducts: [ProductInput!]
  costCenter: CostCenterInput
  lineTotal: Float!
}
```

```
input TransactionInput {
  transactionLines: [TransactionLineInput!]!
  transactionTotal: Float!
}
```

```
input PaymentLineInput {
  id: String!
  name: String!
  type: PaymentType!
  provider: String
  inputMethod: InputMethod!
  amount: Float!
  amountType: PaymentLineType!
  foreignCurrency: ForeignCurrencyInput
  reference: String
  drawer: DrawerInput
}
```

```
input TransferInput {
  from: [TransferItemInput!]!
  to: [TransferItemInput!]!
}
```

```
input TransferItemInput {
  costCenter: CostCenterInput!
  transaction: TransactionInput!
}
```

```
input MoneyInOutInput {
  language: Language!
  vatNo: String!
  estNo: String!
  posId: String!
  posFiscalTicketNo: Int!
  posDateTime: String!
  posSwVersion: String!
  terminalId: String!
  deviceId: String!
  bookingPeriodId: String!
  bookingDate: String!
  ticketMedium: TicketMedium!
  employeeId: String!
  financials: [MoneyInOutLineInput!]!
}
```

```
input MoneyInOutLineInput {
  id: String!
  name: String!
  type: PaymentType!
  provider: String
  inputMethod: InputMethod!
  amount: Float!
  amountType: MoneyInOutLineType!
  foreignCurrency: ForeignCurrencyInput
  reference: String
  drawer: DrawerInput
}
```

```
input WorkInOutInput {
  language: Language!
  vatNo: String!
  estNo: String!
```

```
posId: String!  
posFiscalTicketNo: Int!  
posDateTime: String!  
posSwVersion: String!  
terminalId: String!  
deviceId: String!  
bookingPeriodId: String!  
bookingDate: String!  
ticketMedium: TicketMedium!  
employeeId: String!  
}
```

```
input InvoiceInput {  
  language: Language!  
  vatNo: String!  
  estNo: String!  
  posId: String!  
  posFiscalTicketNo: Int!  
  posDateTime: String!  
  posSwVersion: String!  
  terminalId: String!  
  deviceId: String!  
  bookingPeriodId: String!  
  bookingDate: String!  
  ticketMedium: TicketMedium!  
  employeeId: String!  
  invoiceNo: String!  
  customerVatNo: String!  
  costCenter: costCenterInput  
  fdmRefs: [FdmReferenceInput!]!  
}
```

```
input SaleInput {  
  language: Language!  
  vatNo: String!  
  estNo: String!  
  posId: String!  
  posFiscalTicketNo: Int!  
  posDateTime: String!  
  posSwVersion: String!  
  terminalId: String!  
  deviceId: String!  
  bookingPeriodId: String!  
  bookingDate: String!  
  ticketMedium: TicketMedium!  
  employeeId: String!  
  fdmRef: FdmReferenceInput  
  costCenter: CostCenterInput  
  transaction: TransactionInput!  
  financials: [PaymentLineInput!]!  
}
```

```
input CostCenterChangeInput {  
  language: Language!  
  vatNo: String!  
  estNo: String!  
  posId: String!  
  posFiscalTicketNo: Int!  
  posDateTime: String!  
  posSwVersion: String!
```



```
terminalId: String!  
deviceId: String!  
bookingPeriodId: String!  
bookingDate: String!  
ticketMedium: TicketMedium!  
employeeId: String!  
transfer: TransferInput!  
}
```

```
input OrderInput {  
  language: Language!  
  vatNo: String!  
  estNo: String!  
  posId: String!  
  posFiscalTicketNo: Int!  
  posDateTime: String!  
  posSwVersion: String!  
  terminalId: String!  
  deviceId: String!  
  bookingPeriodId: String!  
  bookingDate: String!  
  ticketMedium: TicketMedium!  
  employeeId: String!  
  costCenter: CostCenterInput  
  transaction: TransactionInput!  
}
```

```
input PreBillInput {  
  language: Language!  
  vatNo: String!  
  estNo: String!  
  posId: String!  
  posFiscalTicketNo: Int!  
  posDateTime: String!  
  posSwVersion: String!  
  terminalId: String!  
  deviceId: String!  
  bookingPeriodId: String!  
  bookingDate: String!  
  ticketMedium: TicketMedium!  
  employeeId: String!  
  costCenter: CostCenterInput  
  transaction: TransactionInput!  
  financials: [PaymentLineInput!]  
}
```

```
input DrawerOpenInput {  
  language: Language!  
  vatNo: String!  
  estNo: String!  
  posId: String!  
  posFiscalTicketNo: Int!  
  posDateTime: String!  
  posSwVersion: String!  
  terminalId: String!  
  deviceId: String!  
  bookingPeriodId: String!  
  bookingDate: String!  
  ticketMedium: TicketMedium!  
  employeeId: String!
```

```
drawer: DrawerInput
}
```

```
input PaymentCorrectionInput {
  language: Language!
  vatNo: String!
  estNo: String!
  posId: String!
  posFiscalTicketNo: Int!
  posDateTime: String!
  posSwVersion: String!
  terminalId: String!
  deviceId: String!
  bookingPeriodId: String!
  bookingDate: String!
  ticketMedium: TicketMedium!
  employeeId: String!
  financials: [PaymentLineInput!]!
  fdmRef: FdmReferenceInput!
}
```

```
input CopyInput {
  language: Language!
  vatNo: String!
  estNo: String!
  posId: String!
  posFiscalTicketNo: Int!
  posDateTime: String!
  posSwVersion: String!
  terminalId: String!
  deviceId: String!
  bookingPeriodId: String!
  bookingDate: String!
  ticketMedium: TicketMedium!
  employeeId: String!
  fdmRef: FdmReferenceInput!
}
```

```
input ReportTurnoverXInput {
  language: Language!
  vatNo: String!
  estNo: String!
  posId: String!
  posFiscalTicketNo: Int!
  posDateTime: String!
  posSwVersion: String!
  terminalId: String!
  deviceId: String!
  bookingPeriodId: String!
  bookingDate: String!
  ticketMedium: TicketMedium!
  employeeId: String!
  posDevices: [PosDeviceInput!]!
  fdmDevices: [FdmDeviceInput!]!
  turnover: TurnoverInput!
}
```

```
input PosDeviceInput {
  posId: String!
```

```
terminalId: String!  
firstPosDateTime: String!  
lastPosDateTime: String!  
}
```

```
input FdmDeviceInput {  
  fdmId: String!  
  firstFdmDateTime: String!  
  firstTotalCounter: Int!  
  lastFdmDateTime: String!  
  lastTotalCounter: Int!  
}
```

```
input TurnoverInput {  
  transactions: [EventTotalInput!]!  
  departments: [DepartmentTotalInput!]!  
  vats: [VatTotalInput!]!  
  payments: [PaymentTotalInput!]!  
  drawersOpenCount: Int! = 0  
  negQuantities: [NegQuantityTotalInput!]!  
  priceChanges: [PriceChangeTotalInput!]!  
  invoices: [InvoiceTotalInput!]!  
}
```

```
input EventTotalInput {  
  eventLabel: EventLabel!  
  ticketCount: Int!  
  amount: Float!  
}
```

```
input DepartmentTotalInput {  
  departmentId: String!  
  departmentName: String!  
  amount: Float!  
}
```

```
input VatTotalInput {  
  label: VatLabel!  
  rate: Float!  
  taxableAmount: Float!  
  vatAmount: Float!  
  totalAmount: Float!  
}
```

```
input PaymentTotalInput {  
  id: String!  
  name: String!  
  type: PaymentType!  
  amount: [PaymentTotalAmountInput!]!  
}
```

```
input PaymentTotalAmountInput {  
  type: PaymentLineType!  
  normalAmount: Float!  
  negativeCorrections: Float!  
  positiveCorrections: Float!  
  correctionsCount: Int!  
  totalAmount: Float!  
  normalForeign: [ForeignCurrencyInput!]!
```

```
correctionsForeign: [ForeignCurrencyInput!]
}
```

```
input NegQuantityTotalInput {
  negQuantityReason: NegQuantityReason!
  negQuantityCount: Int!
  ticketCount: Int!
  amount: Float!
}
```

```
input PriceChangeTotalInput {
  id: String!
  name: String!
  type: PriceChangeType!
  amount: [PriceChangeVatTotalInput!]
}
```

```
input PriceChangeVatTotalInput {
  label: VatLabel!
  negative: Float!
  positive: Float!
}
```

```
input InvoiceTotalInput {
  invoiceNo: String!
  amount: Float!
}
```

```
input ReportTurnoverZInput {
  language: Language!
  vatNo: String!
  estNo: String!
  posId: String!
  posFiscalTicketNo: Int!
  posDateTime: String!
  posSwVersion: String!
  terminalId: String!
  deviceId: String!
  bookingPeriodId: String!
  bookingDate: String!
  ticketMedium: TicketMedium!
  employeeId: String!
  reportNo: Int!
  reportBookingDate: String!
  posDevices: [PosDeviceInput!]!
  fdmDevices: [FdmDeviceInput!]!
  turnover: TurnoverInput!
}
```

```
input ReportUserXInput {
  language: Language!
  vatNo: String!
  estNo: String!
  posId: String!
  posFiscalTicketNo: Int!
  posDateTime: String!
  posSwVersion: String!
  terminalId: String!
  deviceId: String!
}
```

```
bookingPeriodId: String!  
bookingDate: String!  
ticketMedium: TicketMedium!  
employeeId: String!  
posDevices: [PosDeviceInput!]!  
fdmDevices: [FdmDeviceInput!]!  
users: [UserItemInput!]!  
}
```

```
input UserItemInput {  
  employeeId: String!  
  totalAmount: Float!  
  firstPosDateTime: String!  
  lastPosDateTime: String!  
  socialEvents: [InOutItemInput!]!  
  payments: [PaymentTotalInput!]!  
}
```

```
input InOutItemInput {  
  posDateTime: String!  
  inOut: InOut!  
}
```

```
input ReportUserZInput {  
  language: Language!  
  vatNo: String!  
  estNo: String!  
  posId: String!  
  posFiscalTicketNo: Int!  
  posDateTime: String!  
  posSwVersion: String!  
  terminalId: String!  
  deviceId: String!  
  bookingPeriodId: String!  
  bookingDate: String!  
  ticketMedium: TicketMedium!  
  employeeId: String!  
  reportNo: Int!  
  reportBookingDate: String!  
  posDevices: [PosDeviceInput!]!  
  fdmDevices: [FdmDeviceInput!]!  
  users: [UserItemInput!]!  
}
```

2.4. DIGITALE HANDTEKENING

De in sectie 4.2. vermelde handtekening wordt geplaatst op de 'enriched JSON' die verder wordt beschreven. De handtekening wordt geplaatst met behulp van het private certificaat, na toepassing van de eveneens verder beschreven JSON hercodering. Dit laat toe om de reproduceerbaarheid van de handtekening te garanderen.

2.4.1. Canonieke JSON voor de reproduceerbare hash

De flexibiliteit dat JSON bied laat toe om dezelfde informatie op verschillende manieren te encoderen. Om steeds een zelfde handtekening te bekomen voor dezelfde brondata worden een aantal regels toegepast. Deze regels leiden tot een canonieke JSON.

Volgende regels dienen door de FDM strikt toegepast te worden op de verder in punt 2.5.2. vermelde velden uit de JSON.

- **Getallen** mogen enkel de karakters 0 tot en met 9 bevatten, het minteken (-) en een decimaal punt. Dit betekent o.a. dat waarden in wetenschappelijke notities gehercodeerd moeten worden naar hun meest basale vorm. Achterliggende nullen na een decimaal punt zijn niet toegestaan.
- Alle **witruimtes**, zoals beschreven in de JSON specificaties, worden verwijderd, er wordt geen witruimte toegevoegd.
- Tekens met een **codepunt > U+001F en < U+007F** worden **nooit** geëscapet, met uitzondering van de codepunten **U+0022** (dubbel aanhalingsteken) en **U+005C** (backslash) die volgens de JSON specificaties altijd worden geëscapet. Deze twee moeten worden geëscapet in kortst mogelijke notitie (respectievelijk \" en \\).
- Tekens met een **codepunt <U+0020 of > U+007E** worden **altijd** geëscapet en dit steeds in hun kortst mogelijke notatie. Backspace, formfeed, linefeed, carriage return en horizontale tab hebben dergelijke korte notaties (respectievelijk \b, \f, \n, \r en \t).
- Bij het escapen van een codepunt als één of meer sets van 4 hexadecimale tekens, worden de hexadecimale tekens altijd in **hoofdletters** uitgedrukt (bijvoorbeeld: \uAAFF).
- De JSON-decoder moet zich houden aan de **strikte** interpretatie van de JSON specificaties. Bijvoorbeeld: de literalen true, false, null zijn altijd in kleine letters en zonder aanhalingstekens, de namen van de naam/waarde paren in de objecten moeten steeds tussen aanhalingstekens, ...
- De **naam/waarde paren** van elke object in JSON worden **gesorteerd**, in oplopende volgorde, op de numerieke waarde van de codepunten in de naam. Er wordt hierbij aangenomen dat er, bij wijze van best practice, geen dubbele namen worden gebruikt, gezien er geen JSON standaard hieromtrent bestaat.

2.4.2. De 'enriched JSON' waarvan de inhoud digitaal wordt ondertekend

Het 'enriched' JSON object (**enrichedEventData**) bestaat uit onderstaande values en objecten, naargelang ze al of niet gebruikt werden in het bericht.

Optionele velden die een null waarde bevatten of optionele array's die leeg zijn, worden **niet** opgenomen in het object en dus ook niet in de handtekening.

```
{
  language
  posId
  vatNo
  estNo
  terminalId
  deviceId
  posDateTime
  posFiscalTicketNo
  ticketMedium
  eventOperation
  copyOfEvent
  employeed
  customerVatNo
  invoiceNo
  fdmRefs
  bookingPeriodId
  bookingDate
  costCenter
  transfer
  transaction
  vatCalc
  financials
  drawer
  reportNo
  reportBookingDate
  posDevices
  fdmDevices
  turnover
  users
  fdmSwVersion
  posSwVersion
  bufferCapacityUsed
  verificationUrl
  fdmId
  fdmDateTime
  eventLabel
  eventCounter
  totalCounter
}
```

De waarde copyOfEvent wordt door de FDM zelf bepaald en toegevoegd.

copyOfEvent (enum, verplicht)

Het eventLabel zoals gedetecteerd door de FDM bij ontvangst van de boodschappen van de POS. Waarden te gebruiken uit enum EventLabel (zie eerdere beschrijving).

2.4.3. Te gebruiken certificaat voor de handtekening

De handtekening dient geplaatst te worden met behulp van het hardwarecertificaat, waarvan de details beschreven staan in detailbeschrijving van de werking van en de communicatie tussen de fiscale data module en de cloud service fod.