EKaBS

Electronic Cash Register Receipt Standard

Version 1.0.0 (As of 14/04/2021)

The aim of the Electronic Cash Register Receipt Standard, known in short as EKaBS, is to facilitate the general handling of receipts and thus serve the interests of consumers, users of cash registers, customers, the cash register system sector and also financial authorities.

The EKaBS thus explicitly stands above the specific interests of individual parties and was developed in the "Electronic Receipt" working group formed by the DFKA e.V. in cooperation with the ZDH (German Confederation of Skilled Crafts), the HDE (German Retail Association) and the DEHOGA (German Hotel and Restaurant Association).

1. General information

1.1. Legal basis

The obligation to issue receipts was enshrined in law in Section 146a (2) Sentence 1 of the Fiscal Code of Germany (AO) with effect from 01/01/2020. It is not sufficient to merely offer a receipt to the other party involved in the transaction, the receipt must actually be issued and made available in all cases and this process must be verifiable. The continuous provision of receipts is also important for facilitating the easy inspection of cash registers.

Receipts have rarely been issued in some sectors, e.g. in bakeries, up to now. However, there has been some public debate since the end of 2019 about the obligation to issue receipts because the waiver defined in law (Section 146a (2) Sentence 2 of the Fiscal Code of Germany) has now become unworkable in practice due to a stricter interpretation of the rules by the tax authorities. The preferred solution for politicians and administrative bodies is the issuing of electronic cash register receipts because they simultaneously comply with the obligation to issue receipts and also save paper.

In Section 6 of the German Cash Register Ordinance (KassenSichV), electronic receipts are expressly permitted in a standardised format. The Fiscal Code Application Decree (AEAO) on Section 146a defines various rules for the application of Section 146a.

A BMF Letter from 28/05/2020 provided some additional clarification on the Fiscal Code Application Decree with the aim of facilitating the use of electronic cash register receipts and making them more legally compliant (the clarifications in the new letter are underlined):

- 6.3: "The agreement of the customer for the issuing of an electronic receipt does not require a particular form and can also be implied."
- 6.4: "Simply displaying the receipt on a screen at the company (terminal/cash register display) without the possibility of making the electronic receipt available to the customer at the end of the process is not sufficient."
- 6.6: "No technical specifications exist for how the receipt should be made available or transmitted in electronic form. For example, it is permissible for the electronic receipt to be made available to the customer immediately via the screen display (e.g. in the form of a QR code). It can also be made available e.g. as a download link, via Near-Field-Communication (NFC), via e-mail or sent directly to a customer account."

The electronic receipt must be issued in a "standardised data format" (e.g. JPG, PNG or PDF). In this context, this merely means that "it must be possible for the electronic receipt to be received and viewed on the customer's end device using a free standard software" (Section 6.6). However, the receipts issued by the various different systems could differ to such an extent that it is impossible to uniformly manage and evaluate them using an automated process.

This standard aims to close this gap in the current rules.

1.2. Basic requirements

The project aims to take the following basic requirements into account:

- **Openness to different technologies:** This means (a) no specifications for the implementation method and (b) a large degree of freedom in the methods used to transmit the receipt to consumers.
- **Universal usability:** It should be possible to produce the standardised electronic cash register receipt in all sectors and with as many different systems as possible.
- **Implementation in stages:** This approach should enable the fastest possible application of the standard in practice.
- **Only issuing recommendations beyond the required standardisation:** The standardisation process is limited to those measures that are required to ensure the correct application and fulfilment of the legal requirements. To keep the standard as simple as possible, maximum recommendations are also given.

The standard is based on the following basic principles:

- The electronic display should contain the same information as the readable form of the receipt.
- Automated verification of the data issued by the technical security system (TSE) is possible.
- It is possible to verify this data by evaluating the security node without integrating other data – if necessary, an automated comparison with other content of the receipt can be subsequently carried out.
- Where possible, the standard will be broadly based on the taxonomy to simplify the definitions, documentation, implementation and application.
- The standard is currently restricted to the electronic display of cash register receipts that are similar to invoices. The standard does not currently apply to other types of receipt, such as delivery notes. However, enhancements are planned in the future.
- The standard is not intended as a "replacement for the taxonomy", i.e.:
    - All of the receipts together do not provide complete documentation of the cash management process.
    - The contents of the receipt do not provide the same depth of information as the taxonomy.

1.3. Licence

1.4. Standardisation in levels

To achieve the desired gradual standardisation of the process for issuing electronic receipts and thus quick practical application of the standard, the process has been divided into the levels described below that build upon one another.

This version of the standard only addresses levels 1 to 3.

1.4.1. Level 1 – Content of the electronic cash register receipt

What data should the electronic cash register receipt contain?

This has already been conclusively clarified by the legal regulations in, amongst other things, Section 6 KassenSichV.

1.4.2. Level 2 – Format of the structured data on the receipt

How is the data that needs to be automatically evaluated structured?

- Chosen solution:
  - Coding the content of the receipt in JSON format
  - This builds on the cash register data in the DFKA Taxonomy – there are both simplifications and also additions in comparison to this data
  - This method enables automatic signature verification
- Unfavoured alternatives:
  - XML structure used in the ZUGFeRD standard – significantly more complex and not designed for cash register receipts
  - CSV version of the taxonomy (DSFinV-K) – not well suited to the type of data to be depicted

1.4.3. Level 3 – Combining the readable receipt with structured data

How are human-readable and machine-readable data combined?

- Chosen solution:
  - Producing the receipt as a PDF file
  - Integration of structured data according to the PDF/A standard
- Optional alternatives for low-end systems are conceivable, e.g. the use of a simple text format instead of PDF
- Unfavoured alternatives:
  - ZUGFeRD – see above
  - HTML – not well suited for summarising multiple components of the receipt in a file or for permanently archiving the receipts

1.4.4. Level 4 – Methods for transmitting the receipt from the cash register to the recipient

How do customers receive the receipts?

- "Familiarity" will promote acceptance amongst users
- Long-term goal: Every customer (whether anonymous or with a customer account) can collect their receipt in a virtual basket
- Possibilities for …

    - Anonymous customers: QR code with a WWW link to the data, transmission via NFC
    - Customers with a customer account: System suppliers must provide standard web APIs for "receipt management solutions"
- The standardisation of an interface between the cash register and a receipt issuing and management system from a third party supplier (that handles the actual issuing and provision of the receipt) is also conceivable
- Definition for the current version of the standard:
    - No specifications
    - If there is sufficient demand, this level will be included in future versions

1.4.5. Level 5 – Electronic cash register receipts in accounting

How can the receipts be used as incoming invoices?

- This is an obvious enhancement to the standard
- However, this enhancement would be more relevant in the long term after the standard has become established
- Aspects:
    - Verification
    - Automatic booking

1.4.6. Level 6 – Overarching amalgamation of receipts

How can receipts from different companies be saved and evaluated, e.g. for data-driven business models?

- This level would only be interesting in the long term
- Demand and acceptance are uncertain
- Competitive and data protection aspects must be carefully examined

2. Structure of the receipt

2.1. PDF/A

The PDF/A ISO standard is an expanded version of the PDF file format. It was created for the long-term archiving of important documents.

The main differences to normal PDFs are that no references to external resources are permitted (all fonts and images must be embedded in the document), the files may not contain script languages such as Javascript, digital signatures are supported and it is possible and permitted to directly add annexes to the PDF.

Due to these features, PDF/A has now become the archive format used by most DMS systems as it can be guaranteed that the files can still be opened after they have been archived for multiple years because they are not dependent on external resources.

The structured data in the electronic cash register receipt that can be automatically evaluated is embedded in the PDF file as an annex in JSON format.

There are currently no specifications for the content and structure of the plain text for the readable invoice in the PDF file. The requirements for the content of invoices according to tax law must be observed.

It is recommended that the text "EKaBS" is added to the plain text for the readable invoice, preferably in combination with the TSE-specific data, so that it is clear that it contains structured data.

2.2. JSON annex

The annex to the PDF with the data for the electronic receipt in JSON format has the file name "ekabs.json".

2.2.1. General information

JSON (JavaScript Object Notation) is a data exchange format that is easy to read and write for humans and easy to parse (analysis of the data structures) and generate for machines. It is based on a subset of JavaScript.

JSON is a text format that is completely independent of programming languages but which follows many of the conventions that are familiar from programming with the C-family of programming languages (including C, C++, C#, Java, JavaScript, Perl, Python and many others). These characteristics make JSON the ideal format for exchanging data.

A JSON Schema is a standardised description of JSON data that is used to check, or validate, whether the content and form of certain data correspond to a previously defined convention. The JSON Schema itself was also created in JSON format.

The JSON structure for electronic receipts contains the following data:

- **Invoice**: Forms the brackets around the entire invoice. (the term invoice has the same meaning as receipt in this context)

- **Cash register**: Describes the cash register that issued the receipt (currently just the serial number)
- **Header data**: General data required according to tax law that is issued precisely once per invoice, such as data on the issuer of the receipt, the recipient where relevant, date and time, etc.
- **Invoice data:**
  ° **Total amounts**, split according to tax rates and payment types
  ° **Items on the invoice**, i.e. the individual goods or services and all other information that flows into the sum totals, such as e.g. discounts
- **Security data**: Enables (cryptographic) validation that the data on the receipt has been given a signature by a TSE and is thus backed up securely
- **Miscellaneous**: Additional data that is not required according to tax law, such as data for cashless payment

All items on the invoice – including hierarchically linked items, such as discounts, tips, items in transit, sales for the accounts of third parties, etc. – are currently displayed in the same form to make things as easy as possible and to avoid problems with manufacturer-specific displays. Enhancements will be added as required, see Section 4.

2.2.2. Versioning

Every JSON file has a version number that indicates which version of the specifications the data complies with. The version number is divided into three parts:

MAJOR.MINOR.PATCH

e.g. "2.1.3".

These parts are defined as follows:

PATCH: The PATCH part of the version number is incremented when only backwards-compatible changes have been made, which do not affect the structure of the JSON Schema. This includes e.g. changes to descriptions, extensions to the data length or changing mandatory fields to optional fields. It is essential that any data that was validated with an older PATCH version of the JSON Schema within the same MAJOR.MINOR version can also be validated with the new PATCH version of the schema.

MINOR: The MINOR number is incremented when data within the same MAJOR version was validated with an older MINOR version of the JSON Schema and it is possible under certain circumstances that it can no longer be validated with the new version of the JSON Schema. This includes e.g. restricting the contents of fields and structural changes such as renaming or deleting fields or introducing new fields to better depict existing circumstances.

MAJOR: The MAJOR number is incremented when completely new circumstances have been depicted in the JSON Schema or certain circumstances can no longer be depicted in the new version. A large number of structural changes, which would each only justify the incrementation of the MINOR number on their own, should also result in the incrementation of the MAJOR number.

A full description of the JSON structure can be found in the following section.

2.2.3. Presentation of the gross, net and VAT amounts

To present the VAT amounts for the entire receipt in **$.data.vat_amounts[*]**, all three values (**incl_vat** (Gross), **excl_vat** (Net) and **vat** (VAT amount)) should be displayed alongside the VAT rate.

A differentiation is made for the individual data lines in **$.data.lines[*].vat_amounts[*]** between "gross method" and "net method".

If the gross method is used, only the gross amount including the VAT amount is shown. Only **incl_vat** is shown and not **excl_vat** and **vat**.

If the net method is used, the net amount and the associated VAT amount are shown. Only **excl_vat** and **vat** are shown and not **incl_vat**.

3. Description of the JSON structure

3.1. Invoice (root)

```
{
 "version": " 1 . 0 . 0 " ,
 "type":     "INVOICE" ,
 "cash_register": {
   ...
 },
 "head": {
   ...
 },
 "data": {
   ...
 },
 "security": {
   ...
 },
 "misc": {
   ...
 }
}
```

$.version	
Type	String
Mandatory field	Yes
Range of values	^[0-9]+\.[0-9]+\.[0-9]+$
Description	Version number of the standard for electronic invoices

$.type	
Type	String
Mandatory field	Yes
Range of values	INVOICE
Description	Type of invoice, currently only **INVOICE** is permitted

$.cash_register	
Type	Object
Mandatory field	Yes
Description	Information on the cash register that created the invoice.

$.head	
Type	Object
Mandatory field	Yes
Description	Header data for the invoice

$.data	
Type	Object
Mandatory field	Yes
Description	Invoice data

$.security	
Type	Object
Mandatory field	No
Description	Data to securely back up the invoice via TSE

$.misc	
Type	Object
Mandatory field	No
Description	Optional additional information

3.2. Cash register (cash_register)

```
{
  "serial_number": "Q345/98Z"
}
```

$.cash_register.serial_number	
Type	String
Mandatory field	Yes
Description	Serial number of the cash register

3.3. Header data (head)

```
{
  "id": "45130a36-29ed-4e96-817b-e01d02087d14",
  "number": "2020/1001",
  "date": "2020-08-10T10:51:33.7892765+00:00",
  "delivery_period_start": "2020-08-07",
  "delivery_period_end": "2020-08-08",
  "seller": {
    ...
  },
  "buyer_text": "Joe Bloggs, Heidestraße 17, 51147 Cologne",
  "buyer": {
    ...
  }
}
```

$.head.id	
Type	String
Mandatory field	No
Description	Optional ID to identify the invoice which is used as a reference within the cash register system issuing the invoice

$.head.number	
Type	String
Mandatory field	Yes
Description	Consecutive invoice number
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.head.date	
Type	String
Mandatory field	Yes
Range of values	date-time (RFC 3339, Section 5.6.)
Description	Invoice date
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.data.head.delivery_period_start	
Type	String
Mandatory field	Only if $.data.head.delivery_period_end was stated
Range of values	ISO8601 / RFC 3339, Section 5.6 Date
Description	Start of the delivery period, if different from the invoice date

$.data.head.delivery_period_end	
Type	String
Mandatory field	No
Range of values	ISO8601 / RFC 3339, Section 5.6 Date
Description	End of the delivery period, if different from the invoice date

$.head.seller	
Type	Object
Mandatory field	No
Description	Information on the issuer of the invoice / company providing the service
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.head.buyer_text	
Type	String
Mandatory field	No
Description	Name and address of the recipient of the invoice in basic text form. Should only be used if the data is not available in structured form (see $.head.buyer)
Legal basis	UStDV Section 31 (2), UStG Section 14 (4), UStAE Section 14.5

$.head.buyer	
Type	Object
Mandatory field	No
Description	Name and address of the recipient of the invoice in structured form. If the data is not available in structured form, $.head.buyer_text should be used.
Legal basis	UStDV Section 31 (2), UStG Section 14 (4), UStAE Section 14.5

3.3.1. Issuer of the invoice (seller)

```
{
  "name": "Muster GmbH",
  "tax_number": "123/456/789",
  "tax_exemption": true,
  "tax_exemption_note": "Exempt from VAT in accordance with Section 19
  UStG", "address": {
    ...
  }
}
```

$.head.seller.name	
Type	String
Mandatory field	Yes
Description	Name of the company providing the service, correct company name
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.head.seller.tax_number	
Type	String
Mandatory field	Yes
Description	Tax number or VAT ID number of the company providing the service
Legal basis	Section 14 (4) No. 2 UStG

$.head.seller.tax_exemption	
Type	Boolean

Mandatory field	Yes, if there is a tax exemption
Default	False
Description	If there is a tax exemption, this field is set to true. In this case, head.seller.tax_exemption_note must be set.
Legal basis	§ 19 UStG

$.head.seller.tax_exemption_note	
Type	String
Mandatory field	Yes, if head.seller.tax_exemption is set to true
Description	Reason for the tax exemption if head.seller.tax_exemption is set to true.
Legal basis	§ 19 UStG

$.head.seller.address	
Type	Object
Mandatory field	Yes
Description	Address of the company providing the service in structured form
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

```
{
  "street": "Heidestraße 17",
  "postal_code": "51147",
  "city": "Cologne",
  "country_code": "DEU"
}
```

$.head.seller.address.street	
Type	String
Mandatory field	Yes
Range of values	Length 0 - 60
Description	Street name and house number of the company providing the service
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.head.seller.address.postal_code	
Type	String
Mandatory field	Yes
Range of values	Length 0 - 10
Description	Postal code of the company providing the service
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.head.seller.address.city	
Type	String

Mandatory field	Yes
Range of values	Length 0 - 62
Description	City/town of the company providing the service
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.head.seller.address.country_code	
Type	None
Mandatory field	Yes
Default	DEU
Range of values	ALA, AFG, ALB, DZA, ASM, AND, AGO, AIA, ATA, ATG, ...
Description	Country code of the company providing the service according to ISO 3166 alpha-3 country code
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

3.3.2. Recipient of the invoice (buyer)

```
{
  "customer_number": "K1298",
  "name": "Joe Bloggs",
  "tax_number": "DE987654321",
  "address": {
    ...
  }
}
```

$.head.buyer.customer_number	
Type	String
Mandatory field	No
Description	Customer number of the recipient of the service

$.head.buyer.name	
Type	String
Mandatory field	Yes
Description	Name/company name of the recipient of the service

$.head.buyer.tax_number	
Type	String
Mandatory field	No
Description	Tax number or VAT number of the recipient of the service

$.head.buyer.address	
Type	Object
Mandatory field	Yes
Description	Address of the recipient of the service in structured form

```
{
  "street": "Heidestraße 17",
  "postal_code": "51147",
  "city": "Cologne",
  "country_code": "DEU"
}
```


$.head.buyer.address.street	
Type	String
Mandatory field	No
Range of values	Length 0 - 60
Description	Street and house number of the recipient of the service

$.head.buyer.address.postal_code	
Type	String
Mandatory field	No
Range of values	Length 0 - 10
Description	Postal code of the recipient of the service

$.head.buyer.address.city	
Type	String
Mandatory field	No
Range of values	Length 0 - 62

Description	City/town of the recipient of the service

$.head.buyer.address.country_code	
Type	None
Mandatory field	No
Range of values	ALA, AFG, ALB, DZA, ASM, AND, AGO, AIA, ATA, ATG, ...
Description	Country code of the recipient of the service according to ISO 3166 alpha-3 country code

3.4. Invoice data (data)

```
{
  "currency": "EUR",
  "full_amount_incl_vat": 10.49,
  "payment_types": [
    {
      ...
    }
  ],
  "vat_amounts": [
    {
      ...
    }
  ],
  "lines": [
    {
      ...
    }
  ]
}
```

$.data.currency	
Type	String
Mandatory field	Yes
Default	EUR
Range of values	ISO 4217 currency code (alphabetic code)
Description	This field defines the currency to be used for all of the data on the invoice if no other currency is explicitly stated.

$.data.full_amount_incl_vat	
Type	Numeric
Mandatory field	Yes
Range of values	Max. 2 decimal places
Description	The gross total amount on the invoice
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.data.payment_types	
Type	Array
Mandatory field	Yes
Description	List of all payment types that were used to settle this invoice.

$.data.vat_amounts	
Type	Array
Mandatory field	Yes
Description	Total amounts per tax rate – corresponds to the same amounts that are also shown on the printed invoice.
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)

$.data.lines	
Type	Array
Mandatory field	Yes
Description	List of individual items for the services provided
Legal basis	Section 14 (4) in combination with Section 14a (5) German VAT Law (UStG)


```
{
 "name": "CASH",
 "amount": 10.49,
 "foreign_amount": 12.33
 "foreign_currency": "USD",
}
```


$.data.payment_types[*].name	
Type	String
Mandatory field	Yes
Description	Description of the payment type, there must be a distinction between at least cash and non-cash.

$.data.payment_types[*].amount	
Type	Number
Mandatory field	Yes
Description	Payment amount in the local currency ($.data.currency)

$.data.payment_types[*].foreign_amount	
Type	Number
Mandatory field	Yes, if foreign_currency is used
Description	The amount in the foreign currency if paying in a foreign currency. In this case, the foreign currency must be set in $data.payment_types[*].foreign_currency.

$.data.payment_types[*].foreign_currency	
Type	String
Mandatory field	Yes, if foreign_amount is stated
Range of values	ISO 4217 currency code (alphabetic code)
Description	The code for the foreign currency if paying in a foreign currency


```
{
 "percentage": 19,
 "incl_vat": 10.49,
 "excl_vat": 8.82,
 "vat": 1.67
}
```


$.data.vat_amounts[*].percentage	
Type	Number
Mandatory field	Yes

Range of values	Min 0 / Max 100 / 2 decimal places
Description	VAT rate * 100, 19% is displayed as 19.00

$.data.vat_amounts[*].incl_vat	
Type	Number
Mandatory field	Yes
Range of values	2 decimal places
Description	Gross amount for this tax rate

$.data.vat_amounts[*].excl_vat	
Type	Number
Mandatory field	Yes
Range of values	2 decimal places
Description	Net amount for this tax rate

$.data.vat_amounts[*].vat	
Type	Number
Mandatory field	Yes
Range of values	2 decimal places
Description	Absolute tax amount, incl_val - excl_vat = vat

```
{
 "text": "Raincoat",
 "additional_text": "Factory
 second", "vat_amounts": [],
 "item": {
   ...
 },
 "delivery_period_start": "2020-08-07",
 "delivery_period_end": "2020-08-08"
}
```


$.data.lines[*].text	
Type	String
Mandatory field	Yes
Description	Describes the type of item / service or other business transaction. In general, this field is used to give a sufficiently clear description of the item or service but can also be used to describe a discount or similar.
Legal basis	Section 6 (5) KassenSichV, AEAO to Section 146a, Section 5.4 No. 3

$.data.lines[*].additional_text	
Type	String
Mandatory field	No
Description	Additional information on invoice items which go beyond the legal requirements for invoices but which could be of interest for the recipient of the invoice, e.g. serial number of the sold item, percentage discount or similar.

$.data.lines[*].vat_amounts	
Type	Array
Mandatory field	No
Description	Amounts for the individual lines (items) on the invoice split according to the tax rate. One item on the invoice can be split between multiple tax rates if e.g. a group of products (menu) consisting of food and drink is offered in the catering sector and is sold to "take away".

$.data.lines[*].item	
Type	Object
Mandatory field	No
Description	Detailed description of an item on the invoice. Items on the invoice can be described in more detail in the field "item" by providing information on the quantity, item number, etc. The "item name" is stated as a text field at a line level ($.data.lines[*].text) and is thus not repeated in the item object.

$.data.lines[*].delivery_period_start	
Type	String
Mandatory field	Only if $.data.lines[*].delivery_period_end was stated
Range of values	ISO8601 / RFC 3339, Section 5.6 Date
Description	Start of the delivery period if different from the invoice date

$.data.lines[*].delivery_period_end	
Type	String
Mandatory field	No
Range of values	ISO8601 / RFC 3339, Section 5.6 Date
Description	End of the delivery period if different from the invoice date

```
{
 "number": "0815",
 “gtin”: “4013595313208”,
 "quantity": 1,
 "quantity_measure":
 "Pieces", "price_per_unit":
 10.49
```

$.data.lines[*].item.number	
Type	String
Mandatory field	Yes
Description	Item number for the sold item/product or the supplied service

$.data.lines[*].item.gtin	
Type	String
Mandatory field	No
Range of values	At least 1 and a max. of 50 characters

Description	GTIN/EAN of the item

$.data.lines[*].item.quantity	
Type	Number
Mandatory field	Yes
Range of values	3 decimal places
Description	Quantity of the sold item/product or the supplied service

$.data.lines[*].item.quantity_measure	
Type	String
Mandatory field	No
Description	Unit for the amount stated under quantity such as e.g. “Pieces”, “kg”, “Litre”, “Bundle”, …

$.data.lines[*].item.price_per_unit	
Type	Number
Mandatory field	Yes
Range of values	5 decimal places
Description	Gross price of the item/product or service per unit. price_per_unit * quantity = SUM(vat_amounts[*].incl_vat)

Gross:

```
{
 "percentage": 19.0,
 "incl_vat": 10.49
}
```

Net:

```
{
 "percentage": 19.0,
 "excl_vat": 8.82,
 "vat": 1.67
}
```

$.data.lines[*].vat_amounts[*].percentage	
Type	Number
Mandatory field	Yes
Range of values	Min 0 / Max 100 / 2 decimal places
Description	Tax rate

$.data.lines[*].vat_amounts[*].incl_vat	
Type	Number
Mandatory field	Yes, if excl_vat and vat are not set
Range of values	5 decimal places
Description	Gross amount – not used if excl_vat and vat are used instead

$.data.lines[*].vat_amounts[*].excl_vat	
Type	Number
Mandatory field	Yes, if incl_vat is not set
Range of values	5 decimal places
Description	Net amount - not used if incl_vat is used

$.data.lines[*].vat_amounts[*].vat	
Type	Number
Mandatory field	Yes, if incl_vat is not set
Range of values	5 decimal places
Description	VAT amount - not used if vat is used

3.5. Security data (security)

```
{
 "tse": {
   ...
 }
}
```

$.security.tse	
Type	Object
Mandatory field	No
Description	Security features that are relevant for verifying the invoice in accordance with KassenSichV.
Legal basis	KassenSichV; 146a AO

```
{
 "serial_number": "623323B6C170DF2200...8F3A78E5BA7C4BA60B",
 "signature_algorithm": "ecdsa-plain-SHA384",
 "log_time_format": "unixTime",
 "certificate": "LS0tLS...S0tCg==",
 "timestamp_start": "2020-08-10T10:51:33.8049447+00:00",
 "timestamp_end": "2020-08-10T10:51:33.8049447+00:00",
 "first_order": "2020-08-10T10:51:33.8049447+00:00",
 "transaction_number": 50,
 "signature_number": 121,
 "process_type": "Invoice-V1",
 "process_data":
 "Invoice^42.31_16.26_0.00_0.00_0.00^58.57:Cash", "signature":
 "AjEGS...dhRitb"
}
```

$.security.tse.serial_number	
Type	String
Mandatory field	Yes
Range of values	Length 64 - / ^([0-9a-fA-F]{2})+$
Description	Serial number of the TSE in hexadecimal format
Legal basis	KassenSichV; 146a AO

$.security.tse.signature_algorithm	
Type	String
Mandatory field	Yes
Default	ecdsa-plain-SHA256
Range of values	ecdsa-plain-SHA224, ecdsa-plain-SHA256, ecdsa-plain-SHA384, ecdsa-plain-SHA512, ecdsa-plain-SHA3-224, ecdsa-plain-SHA3-256, ecdsa-plain-SHA3-384, ecdsa-plain-SHA3-512, ecsdsa-plain-SHA224, ecsdsa-plain-SHA256, ...
Description	Signature algorithm used by the TSE
Legal basis	KassenSichV; 146a AO

$.security.tse.log_time_format	
Type	String
Mandatory field	Yes

Default	unixTime
Range of values	unixTime, utcTime, utcTimeWithSeconds, generalizedTime, generalizedTimeWithMilliseconds
Description	Time format used by the TSE
Legal basis	KassenSichV; 146a AO

$.security.tse.certificate	
Type	String
Mandatory field	Yes
Description	Total certificate chain for the TSE in PEM format
Legal basis	KassenSichV; 146a AO

$.security.tse.timestamp_start	
Type	String
Mandatory field	Yes
Description	Time stamp for the start of the transaction. The time stamp corresponds to the time stamp returned by the TSE when the start() function is executed.

$.security.tse.timestamp_end	
Type	String
Mandatory field	Yes

Description	Time stamp for the end of the transaction. The time stamp corresponds to the time stamp returned by the TSE when the end() function is executed.

$.security.tse.first_order	
Type	String
Mandatory field	Yes, if orders for this invoice have been backed up securely.
Description	Time stamp for the start of the first order transaction. The time stamp corresponds to the time stamp returned by the TSE when the start() function for the first order is executed.

$.security.tse.transaction_number	
Type	Integer
Mandatory field	Yes
Range of values	Min 0
Description	The transaction number issued by the TSE

$.security.tse.signature_number	
Type	Integer
Mandatory field	Yes
Range of values	Min 0
Description	The signature number issued by the TSE. The signature number is the answer received when executing the end() function for the invoice transaction.

$.security.tse.process_data	
Type	String
Mandatory field	No
Description	Content of the process_data when executing the end() function for the invoice

$.security.tse.process_type	
Type	String
Mandatory field	No
Default	Invoice-V1
Range of values	Length - 30
Description	Content of the process_type when executing the end() function for the invoice
Legal basis	DSFinV-K

$.security.tse.signature	
Type	String
Mandatory field	Yes
Description	Check value/signature issued by the TSE when executing the end() function for the invoice transaction

3.6. Miscellaneous

```
{
 "logo": {
  ...
 },
 "footer_text": "Please also take a look at our special summer offers!",
 "additional_receipts": [
  {
   ...
  }
 ]
}
```

$.misc.logo	
Type	Object
Mandatory field	No
Description	Logo to be shown on the printed / displayed invoice

$.misc.footer_text	
Type	String
Mandatory field	No
Description	Optional footer text to appear at the end of the invoice

$.misc.additional_receipts	
Type	Array

Mandatory field	No
Description	Additional receipts in text form, as an image or PDF (e.g. card terminal receipt)

```
{
  "content_type": "image/png",
  "content": „..."
}
```

$.misc.logo.content_type	
Type	String
Mandatory field	Yes
Description	MIME type for the logo, e.g. image/png or text/x-uri

$.misc.logo.content	
Type	String
Mandatory field	Yes
Description	Base64 encoded content of the logo. If the logo is a URL (text/x-uri), the Base64 encoding is not required.

```
{
  "content_type": "text/plain",
  "content": "* * EC customer receipt * *\nMuster GmbH\n..."
}
```

$.misc.additional_receipts[*].content_type	
Type	String
Mandatory field	Yes
Description	MIME type for the additional receipt, e.g. image/png, application/pdf or text/plain

$.misc.additional_receipts[*].content	
Type	String
Mandatory field	Yes
Description	Base64 encoded content of the additional receipt

4. Future enhancements

This section contains a list of the current suggestions for enhancements. No decisions have currently been made about their implementation or the order in which they will be implemented. These decisions will be made based on practical experience of the standard and feedback from all stakeholders.

The following proposals are currently being considered:

- Standardisation of levels 4 to 6 described in Section 1.4
- Detailed display options for the items ($.data.lines) on the invoice that are compatible with older versions of the standard for:
    - Discounts
    - Coupons
    - Deposits
    - Differentiation between the purchased items (items, services, gift cards/vouchers)
    - Tips (employees and employers)
    - Agency business, items in transit
- Hierarchical linking of items on the invoice – a typical application case would be showing the available side dishes in the catering sector
- Other fields for the master data of the issuer of the invoice (under $.head.seller) such as the telephone number, e-mail, website
- The ability to pseudonymise the recipient of the invoice, only recording the customer number without the name and address data
- Clear rules for manufacturer-specific enhancements to ensure that the keys are always unique
- Enhancements for the transmission of the invoice data in JSON structure to a system that then creates the entire invoice including the integration of graphics, text formatting, etc. (also with manufacturer-specific enhancements) – a typical application case would be sending the invoice data from the cash register system to a printer that either prints a paper copy of the invoice or produces an electronic invoice via a cloud solution