

# Customer Ledger

In this article, you can find guidelines for how to use the endpoints connected to the Customer Ledger.

For more information regarding the endpoints, query parameters and other endpoints for all areas,

Please read the documentation found here: [Swagger - Visma.Net Integrations Documentation](#)

<b>Customer</b>	<b>2</b>
Methods:	2
GET All Customers	2
Example of usage with parameters	2
POST Customer	3
JSON Request body	3
POST action/createDunningLetter	4
JSON Request body	4
PUT Customer	4
URL	4
JSON Request body	4
<b>Customer CreditNote</b>	<b>5</b>
Methods:	5
GET All Customer Credit Notes	5
Example of usage with parameters	5
POST CustomerCreditNote	6
JSON Request body	6
PUT CustomerCreditNote	6
URL	7
JSON Request body	7
<b>Customer DebitNote</b>	<b>7</b>
Methods:	7
GET All Customer Debit Notes	7
Example of usage with parameters	7
POST CustomerDebitNote	8

JSON Request body	8
PUT CustomerCreditNote	9
URL	9
JSON Request body	9
<b>Customer Invoice</b>	<b>9</b>
Methods:	10
GET All	10
Example of usage with parameters	10
POST CustomerInvoice	10
JSON Request body	10
PUT CustomerInvoice	11
URL	11
JSON Request body	11

# Customer

Common usage example for Customer(ScreenId=AR303000)

URL: <https://integration.visma.net/API/controller/api/v1/Customer>

## Methods:

GET All Customers

URL: <https://integration.visma.net/API/controller/api/v1/Customer>

Example of usage with parameters

### Query parameters:

- lastModifiedDate=YYYY-MM-DD

As of today Filtering Parameters does not accept certain characters such as W-Z

These are the formats for Filtering

**\*2001-01-01**

**\*2001-01-01 13:13:13**

**\*2001-01-01 13:13:13.133**

- lastModifiedDateCondition=Operator

### Supported comparative operators for LastModifiedDateTime Conditions on filtering

> - Greater than

< - Less than

<= - Less than or equal to

>= - Greater than or equal to

#### Usage example:

This query will return customers updated on 2020-01-14 or later.

```
GET https://integration.visma.net/API/controller/api/v1/Customer
?lastModifiedDateTime=2020-01-14
&lastModifiedDateTimeCondition=>=
( Lines are separated for readability)
```

#### POST Customer

```
URL: https://integration.visma.net/API/controller/api/v1/customer
```

This example shows the minimum fields you should send to be able to post a new Customer, if you are using automatic numbering you can omit the "number" field.

#### JSON Request body

```
{
  "name": {
    "value": "NewCustomer"
  },
  "number": {
    "value": "10000"
  },
  "overrideWithClassValues": true,
  "customerClassId": {
    "value": "1"
  },
  "mainAddress": {
    "value": {
      "addressLine1": {
        "value": "TestRoad 13"
      },
      "postalCode": {
        "value": "0101"
      },
      "countryId": {
        "value": "NO"
      }
    }
  }
}
```

If sent successfully, the API will return:

```
Status 201 Created
```

## POST action/createDunningLetter

URL: <https://integration.visma.net/API/controller/api/v1/customer/{customerNumber}/action/createDunningLetter>

This example shows what should be sent to create dunning letters for a specified customer.

## JSON Request body

```
{
  "dunningLetterDate": "2020-01-21T09:39:48.538Z",
  "levelFrom": 1,
  "levelTo": 2
}
```

If successful the API will return a message as below:

```
Status: 200 OK
Response body:
{
  "actionId": "2ddece9b-a89b-4f5a-a251-2cf14837bb2d",
  "actionResult": "Done"
}
```

## PUT Customer

URL: <https://integration.visma.net/API/controller/api/v1/customer/{customerNumber}>

When using PUT, you only send the fields you want to update. In the below example, we update the mainAddress of the customer:

## URL

PUT <https://integration.visma.net/API/controller/api/v1/customer/10000>

## JSON Request body

```
{
  "mainAddress": {
    "value": {
      "addressLine1": {
        "value": "TestRoad 50"
      },
      "postalCode": {
        "value": "0101"
      },
      "countryId": {
        "value": "NO"
      }
    }
  }
}
```

If successful, the API will return:

Status: 204 No Content

# Customer CreditNote

Common usage example for Endpoint(ScreenId=AR301000)

URL: <https://integration.visma.net/API/controller/api/v1/CustomerCreditNote>

## Methods:

GET All Customer Credit Notes

URL: <https://integration.visma.net/API/controller/api/v1/CustomerCreditNote>

## Example of usage with parameters

### Query parameters:

- financialPeriod=YYYYMM

Lets you define what financialPeriod to include in response

- status=string

Limits response to CreditNotes with a status

- expandAttachments=boolean

Lets you decide if you want attachments to be expanded or not.

- expandTaxDetails=boolean

Lets you decide if you want tax detail lines to be expanded or not.

### Usage example:

This will return Customer CreditNotes in the financial period 01-2020 that have the status "balanced", and will not expand attachments or tax detail lines.

```
GET https://integration.visma.net/API/controller/api/v1/customerCreditNote
?financialPeriod=202001
&status=balanced
&expandAttachments=false
```

**&expandTaxDetails=false**  
(Lines are separated for readability)

## POST CustomerCreditNote

URL: <https://integration.visma.net/API/controller/api/v1/CustomerCreditNote>

This example shows the minimum fields you should send to be able to post a new Customer Credit Note, if you are using automatic numbering you can omit the "referenceNumber" field.

## JSON Request body

```
{
  "currencyId": {
    "value": "NOK"
  },
  "externalReference": {
    "value": "ExRef"
  },
  "lines": [
    {
      "operation": "Insert",
      "inventoryNumber": {
        "value": "6"
      },
      "quantity": {
        "value": 1
      },
      "unitPriceInCurrency": {
        "value": 1
      }
    }
  ],
  "customerNumber": {
    "value": "10000"
  },
  "referenceNumber": {
    "value": "25698"
  },
  "documentDate": {
    "value": "2019-12-21"
  },
  "postPeriod": {
    "value": "122019"
  },
  "financialPeriod": {
    "value": "201912"
  }
}
```

If sent successfully, the API will return:

Status 201 Created

## PUT CustomerCreditNote

URL: <https://integration.visma.net/API/controller/api/v1/CustomerCreditNote/{creditNoteNumber}>

When using PUT, you only send the fields you want to update. In the below example, we update the first line of the Credit Note:

## URL

```
PUT https://integration.visma.net/API/controller/api/v1/CustomerCreditNote/25698
```

## JSON Request body

```
{
  "lines": [
    {
      "operation": "Update",
      "lineNumber": {
        "value": 1
      },
      "inventoryNumber": {
        "value": "6"
      },
      "quantity": {
        "value": 1
      },
      "unitPriceInCurrency": {
        "value": 1
      }
    }
  ]
}
```

If successful, the API will return:

```
Status: 204 No Content
```

# Customer DebitNote

Common usage example for Customer Debit Note(ScreenId=AR301000)

```
URL: https://integration.visma.net/API/controller/api/v1/CustomerDebitNote
```

## Methods:

GET All Customer Debit Notes

```
URL: https://integration.visma.net/API/controller/api/v1/CustomerDebitNote
```

Example of usage with parameters

**Query parameters:**

- Project=String

Will limit response to documents with the specified project

#### Usage example:

This will return Customer Debit Notes with projectID 20.

**GET** <https://integration.visma.net/API/controller/api/v1/customerDebitNote?project=20>

#### POST CustomerDebitNote

URL: <https://integration.visma.net/API/controller/api/v1/CustomerDebitNote>

This example shows the minimum fields you should send to be able to post a new Customer Debit Note, if you are using automatic numbering you can omit the "referenceNumber" field.

#### JSON Request body

```
{
  "lines": [
    {
      "operation": "Insert",
      "description": {
        "value": "Test"
      },
      "quantity": {
        "value": 2
      },
      "unitPriceInCurrency": {
        "value": 2
      },
      "uom": {
        "value": "STK"
      }
    }
  ],
  "customerNumber": {
    "value": "10000"
  },
  "referenceNumber": {
    "value": "000050"
  },
  "documentDate": {
    "value": "2020-01-16"
  },
  "postPeriod": {
    "value": "012020"
  },
  "financialPeriod": {
    "value": "202001"
  },
  "invoiceText": {
    "value": "DebitNote"
  }
}
```



If sent successfully, the API will return:

```
Status 201 Created
```

PUT CustomerCreditNote

```
URL: https://integration.visma.net/API/controller/api/v1/CustomerDebitNote/{debitNoteNumber}
```

When using PUT, you only send the fields you want to update. In the below example, we update the first line of the Credit Note:

URL

```
PUT https://integration.visma.net/API/controller/api/v1/CustomerDebitNote/000050
```

JSON Request body

```
{
  "lines": [
    {
      "operation": "Insert",
      "lineNumber": {
        "value": 1
      },
      "inventoryNumber": {
        "value": "6"
      },
      "quantity": {
        "value": 1
      },
      "unitPriceInCurrency": {
        "value": 1
      }
    }
  ]
}
```

If successful, the API will return:

```
Status: 204 No Content
```

# Customer Invoice

Common usage example for CustomerInvoice(ScreenId=AR301000)

```
URL: https://integration.visma.net/API/controller/api/v1/CustomerInvoice
```

## Methods:

### GET All

URL: <https://integration.visma.net/API/controller/api/v1/CustomerInvoice>

### Example of usage with parameters

#### Query parameters:

- lastModifiedDate=YYYY-MM-DD

As of today Filtering Parameters does not accept certain characters such as W-Z

These are the formats for Filtering

\*2001-01-01

\*2001-01-01 13:13:13

\*2001-01-01 13:13:13.133

- lastModifiedDateCondition=Operator

Supported comparative operators for LastModifiedDate Conditions on filtering

> - Greater than

< - Less than

<= - Less than or equal to

>= - Greater than or equal to

#### Usage example:

This query will return customers updated on 2020-01-14 or later.

```
GET https://integration.visma.net/API/controller/api/v1/CustomerInvoice
?lastModifiedDate=2020-01-14
&lastModifiedDateCondition=>=
( Lines are separated for readability)
```

### POST CustomerInvoice

URL: <https://integration.visma.net/API/controller/api/v1/CustomerInvoice>

This example shows the minimum fields you should send to be able to post a new Customer Invoice, if you are using automatic numbering you can omit the "referenceNumber" field.

### JSON Request body

```
{
```

```

"invoiceLines": [
  {
    "operation": "Insert",
    "inventoryNumber": {
      "value": "222"
    },
    "quantity": {
      "value": 500
    },
    "uom": {
      "value": "STK"
    }
  }
],
"sendToAutoInvoice": {
  "value": true
},
"referenceNumber": {
  "value": "156589"
},
"customerNumber": {
  "value": "10000"
},
"documentDate": {
  "value": "2020-01-27"
},
"invoiceText": {
  "value": "Payment for potatoes"
}
}

```

If sent successfully, the API will return:

```
Status 201 Created
```

PUT CustomerInvoice

```
URL: https://integration.visma.net/API/controller/api/v1/CustomerInvoice/{InvoiceNumber}
```

When using PUT, you only send the fields you want to update. In the below example, we update the first line of the Invoice:

URL

```
PUT https://integration.visma.net/API/controller/api/v1/CustomerInvoice/003547
```

JSON Request body

```

{
  "lines": [
    {
      "operation": "Insert",
      "lineNumber": {
        "value": 1
      },
      "inventoryNumber": {
        "value": "6"
      },
      "quantity": {

```

```
    "value": 1
  },
  "unitPriceInCurrency": {
    "value": 1
  }
]
}
```

If successful, the API will return:

Status: 204 No Content