

|  |
| --- |
| Fiserv uCommerce Services  Step by step guide to make a guest checkout  Document Version – 1.0 |

# How to make a guest checkout?

Making a guest checkout, or an anonymous checkout, as its often referred to, is a fairly simple process. A few bits of a data are needed to successfully process an anonymous checkout/order. For starters, an orderId, merchantId, or storeId (depending on the configuration) are needed to initiate a guest checkout.

## Step 1: Generate an order ID

### Description

In order to perform a guest checkout or sale using the ucom/v1/payments/sales API call, an order ID is needed to help identify the order or transaction. The order ID could any string of characters with a maximum 36 characters limit.

### Sample Order ID

| Sample Request –Create Recipient-Consumer |
| --- |
| "orderId": "34548sdfdsfer769" |

## Step 2: Pass through Merchant ID or Store ID

### Description

The sales API call requires either a merchant ID (MID) or store ID in the body to identify the merchant or store making the transaction. The MID or store ID need to be configured and provided by Fiserv.

### Sample Value

| Sample Request –Update Recipient-Consumer |
| --- |
| "merchantId": "1120542230" |

The minimal request can be any single entity that requires updating, any of the objects or entities in the parameters will work here.

## Step 3: Provide Requested Amount

### Description

An amount is needed to process a guest checkout transaction. This value must be the final price that the customer is required to pay for the goods or services received. Along with the requested amount field, a currency code is required based on the country and currency where the transaction is taking place. For example, to processor a payment using USD, simply use the code “840” as demonstrated below.

### Sample Value

| Sample Response – Get Recipient-Consumer |
| --- |
| "requestedAmount":1,  "currencyCode":{  "number":840 |

### Sample Response (400 – Bad Request)

| Sample Error Response |
| --- |
| {  "code": "404001",  "message": "No data found for the request.",  "category": "common",  "developerInfo": {  "developerMessage": "No data found for the request.",  "fieldError": [  {  "field": "recipientProfile",  "message": "Recipient does not exist."  }  ]  }  } |

## Step 4: Provide Funding Source

### Description

This portion of the API call/payload will require a funding source or payment method to process the transaction. Since all credit card details need to encrypted, a nonce token is needed to mask the credit card details. A nonce could be obtained through the POST - /v1/account-tokens API call. There are several ways one could obtain a nonce, please refer to the implementation guide for the full process.

### Sample Value

| Sample Nonce |
| --- |
| "token": {                  "tokenProvider": "UCOM",                  "tokenType": "CLAIM\_CHECK\_NONCE",                  "tokenId": "26d0487a-126d-4640-a128-e9ee2c3efc2e"              } |

## Full Payload Sample

### Description

This is sample sales API payload that combines all of the previous steps.

### Endpoint URL

HTTP Method: POST

Non-prod: [https://int.api.firstdata.com/ucom/](https://int.api.firstdata.com/ucom/v1/account-tokens)v1/payments/sales

Prod: [https://prod.api.firstdata.com/ucom/ v1/payments/sales](https://prod.api.firstdata.com/ucom/v1/account-tokens)

### Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Attributes | Data Type | Required | Max Length |
| fdCustomerId | String | Optional | Maximum 36 characters. |
| orderId | String | Required | Maximum 36 characters. |
| storeId/merchantId | String | Required | FD Issued - Alpha-Numeric |
| fundingSource | String | Optional | Credit, prepaid |
| requestedAmount | String | Required | Dollars. Cents |
| Number | Integer | Required | USD = 840 |
| nameOnCard | String | Required | Maximum 50 characters |
| cardType | String | Required | VISA, MASTERCARD,DISCOVER,AMEX |
| securityCode | String | Required | Payment card security code |
| Type | String | Optional | Canonical type values of "work", "home", and "other". |
| streetAddress | String | Optional | The full street address component, which may include house number, street name, P.O. box, and multi-line extended street address information. |
| Locality | String | Optional | The city or locality component. |
| Region | String | Optional | 4 |
| postalCode | String | Required | 5 |
| Country | String | Optional | The country name component. When specified, the value MUST be in ISO 3166-1 “alpha-2” code format; e.g., the United States and Sweden are “US” and "SE", respectively. |
| Formatted | String | Optional | The full mailing address, formatted for display or use with a mailing label. |
| Primary | Boolean | Optional | Indicates if this is the default address for the associated resource. It only has meaning when used in a set or array of addresses. |
| Month | String | Required | Month format ‘MM’ |
| Year | String | Required | Year format ‘YY’ |

### Headers

Content-Type:application/json

Client-Request-Id:{{$guid}}

Api-Key:{{clientKey}}

Authorization: Bearer {{tokenId}}

Timestamp:{{time}}

### Sample Request (with storeid)

| Sample Request – anonymous CC transaction using Nonce with storeId |
| --- |
| {  "sale":{  "orderId":"Order00122",  "storeId":"KNOWN\_STORE\_ID",  "requestedAmount":1,  "currencyCode":{  "number":840  },  "fundingSource":{  "saveToVault": false,  "token":{  "tokenId":"bab8a932-f9f1-483b-9cf9-bb9a899e1a55",  "tokenProvider": "UCOM",  "tokenType": "CLAIM\_CHECK\_NONCE"  }  }  } |

### Sample Request (with merchantid)

| Sample Request – anonymous CC transaction using Nonce with merchantId |
| --- |
| {  "sale":{  "orderId":"Order00122",  "merchantId":"M017311360001",  "requestedAmount":1,  "currencyCode":{  "number":840  },  "fundingSource":{  "saveToVault": false,  "token":{  "tokenId":"bab8a932-f9f1-483b-9cf9-bb9a899e1a55",  "tokenType":"CLAIM\_CHECK\_NONCE"  }  }  }  } |

### Sample Response (201 – Created)

| Sample Response – anonymous card sale transaction |
| --- |
| {  "fdSaleId": "2651f121f7e64f90871fa25ee5921027",  "status": "APPROVED",  "orderId": "Order0012",  "storeId": "KNOWN\_STORE\_ID",  "requestedAmount": 1,  "approvedAmount": 1,  "currencyCode": {  "code": "USD",  "number": 840  },  "transactionDateTime": "2020-05-07T18:15:52-0400",  "fundingSource": {  "type": "CREDIT",  "credit": {  "nameOnCard": "John Smith",  "alias": "1111",  "cardType": "VISA",  "billingAddress": {  "streetAddress": "100 Universal City Plaza",  "postalCode": "00000"  },  "expiryDate": {  "month": "04",  "year": "20"  },  "token": {  "tokenId": "8973365013201111",  "tokenProvider": "TRANS\_ARMOR"  }  }  },  "hostExtraInfo": [  {  "name": "APPROVAL\_NUMBER",  "value": "ET144721"  },  {  "name": "SEQUENCE\_NUMBER",  "value": "303417"  }  ]  } |