FOODTEC

RESTAURANT SOLUTIONS

Order Acceptance API v2

FoodTec Solutions Inc. – Version 1.0, Thu Sep 11 02:56:21 EDT 2025

Table of Contents

Specifications subject to change.

The FoodTec POS supports a RESTful API that accepts orders from authenticated external sources, such as catering or internet ordering systems.

1. Overview

The Order Acceptance API is designed to have resource-oriented URLs (/orders) and to use HTTP error codes to indicate various error types. All the messages including errors, are formatted in JSON.

1.1. Authentication

The Order Acceptance API v2 uses the Basic HTTP Auth protocol to authorize calls.

1.2. Registering API client

In order for a third-party application to be able to use the Order Acceptance API v2, it should first get registered with FoodTec POS. Upon registration a client identifier (username) and client password will be issued to the third-party API client.

1.3. Error Handling

Invalid provided data can result into an error similar to the following:

```HTTP
HTTP/1.1 400 Bad Request
Content-Type: application/json;charset=UTF-8
{
    "meta": {
        "code": 1600,
        "error": "No coupon matching the 3dolloff",
        "info": "No coupon matching the 3dolloff"
    },
    "response": ""
}
```

Note that all these responses will have an HTTP 400 code with a customer error code in the *meta* object. The *error* and *info* fields will always have the same value and the response object is going to be empty.

1.3.1. Authentication Failure Request Example

```BASH
curl https://localhost/ws/store/v1/menu/ordertypes -i -H "Accept: application/json" -H "Content-Type: application/json"
```

1.3.2. Authentication Failure Response Example

```HTTP
HTTP/1.1 401 Unauthorized
Server: Apache-Coyote/1.1
WWW-Authenticate: Basic realm="store services"
Content-Length: 0
Date: Mon, 16 Mar 2015 13:20:28 GMT
```

1.4. Error Codes

The values of words prefixed with a dollar sign ($) depend on the context of the error.

1.4.1. General Errors

Code	Error/Info
1000	Unsupported order type. $message
1001	Not licensed for web
1002	ExternalRef cannot be empty or null
1003	Open order for externalRef $externalRef already exists
1004	Unsupported order source: $orderSource. $message
1005	Cannot find order with externalRef: $externalRef
1007	$field1 is required when $field2 is present
1010	Order could not be placed
1011	Internal address error
1012	Order price cannot be calculated
1020	Cannot calculate price for item $item ($category)
1021	Cannot calculate price for special $special
1022	Cannot calculate discount for special $special
1050	$order_could_not_be_voided_reason

1.4.2. Menu Errors

Code	Error/Info
1100	At least one item or special must be specified
1101	Category cannot be empty
1102	No such category $category
1110	Item cannot be empty
1111	No such item $item for category $category
1120	Item size cannot be empty for item $item ($category)
1121	No such size $size for item $item ($category)
1122	The size $size is not available for item $item ($category)
1123	The item $item ($category) is out of stock

Code	Error/Info
1124	Item $item contains an ingredient with empty name
1125	$invalid_ingredient_name_reason
1130	No such ingredient $ingredient for item $item ($category)
1131	No such qualifier $qualifier for item $item ($category)
1140	Quantity must be positive for item $item ($category)
1141	At least $cnt enforced ingredients is required for $category $item $choice choice.
1150	Special name cannot be empty
1151	No such special $special
1152	Special $special must contain items
1153	The selling price for the special $special cannot be empty for the special $special
1154	The selling price $price for the special $special is invalid
1155	The discount for the special $special cannot be empty
1156	The discount $discount for the special $special is invalid
1157	Unsupported special type. $message
1158	Unsatisfied special $message
1159	The $special is limited to $max per order.

1.4.3. Customer Errors

Code	Error/Info
1200	Customer cannot be null
1210	The name of the customer must be specified
1220	The phone of the customer must be specified
1221	Invalid customer phone $phone. $message

1.4.4. Delivery address Errors

Code	Error/Info
1300	Address and zip fields must be specified for delivery orders
1301	Address or zip is missing
1303	Location not found
1304	Coordinates not found
1305	Did not find $field1 field corresponding to $field2

1.4.5. Deferring & Date Errors

Code	Error/Info
1400	$message
1401	Cannot set defer time for a past date: $deferDate
1402	Store is closed
1410	Cannot set defer time. Store will be closed on : $deferDate
1411	Cannot set defer time. Timeslot is full for the date: $deferDate
1420	$message

1.4.6. Pricing Errors

Code	Error/Info
1506	Invalid order price $price
1507	Invalid order tax $tax
1508	Invalid order tip $tip
1509	Invalid order discout $discount
1510	Invalid order delivery charge $charge
1511	Invalid price $price for item $item
1512	Order Price cannot be empty
1513	The selling price for item $item cannot be empty
1514	The price for ad-hoc $ingredient cannot be empty

Code	Error/Info
1515	Invalid Service Charge price $price

1.4.7. Pricing Mismatches

Code	Error/Info
1560	Provided order price $provided does not match calculated price $calculated
1561	Provided order tax $provided does not match calculated tax $calculated
1562	Provided order discount $provided does not match calculated discount $calculated
1563	Provided order delivery charge $provided does not match calculated delivery charge $calculated

1.4.8. Coupon Application Errors

Code	Error/Info
1600	No coupon matching the $code
1601	The coupon does not qualify for the order. $terms
1602	The coupon $code has been expired.
1603	The coupon $code has been already used.
1604	The coupon $code cannot be applied.
1605	The coupon $code cannot be matched with this special.
1606	$coupon_redemption_failed_reason.
1607	Cannot apply $code, one adjustment per order allowed.

1.4.9. Adjustment Application Errors

Code	Error/Info
1190	Adjustment name cannot be empty
1191	Adjustment discount cannot be empty
1192	Invalid adjustment discount $discount

1.5. Order Object

This object defines an **order** object

Property	Type	Description
orderNum	*string*	The order number of the accepted order. *Returned after acceptance*
type	*enum*	The type of the order. Allowed values: Pickup, Delivery. *Required*
source	*enum*	The source of the order. Allowed values (*case insensitive*): Door Dash, GrubHub, Postmates, Uber Eats, CyborgOps, Third Party. *Required*
paid	*boolean*	Whether the order was paid. *Required*
isThirdPartyPickup	*boolean*	For Pickup orders, whether a third party will pick the order from the store, instead of the customer. *Optional*
price	*float*	The total price of the order (including tax, tip etc). *Required*
tax	*float*	The total order tax. *Optional*
taxableSubtotal	*float*	The order's taxable subtotal. *Optional*
tip	*float*	The order tip. *Optional*
discount	*float*	The total order discount. *Optional*
deliveryCharge	*float*	The Delivery charge if applicable. *Optional*
serviceCharge	*float*	The Service charge if applicable. *Required when serviceChargeName is present*
serviceChargeName	*string*	The Service charge name if applicable. *Required when serviceCharge is present*
timeOrdered	*date_time* [1]	The time that the order has been accepted. *Returned after acceptance*

Property	Type	Description
deferTime	*date_time* [1]	If the order is to be deferred, this is the time at which the order is due to the customer. *Optional*
promiseTime	*date_time* [1]	The time the order will be ready. *Returned after acceptance*
customer	*Customer object*	The order customer details.
items	*Array of Item objects*	The order items. *Required if specials are empty*
specials	*Array of Special objects*	The order specials. *Required if items are empty*
adjustment	*Adjustment object*	The order adjustment. *Optional*
instructions	*string*	The order instructions provided by the consumer. Max 150 characters allowed. *Optional*
externalRef	*string*	Is a unique identifier, provided by the client. It is used to prevent duplicate orders and to void the order (if needed) via the Void Order endpoint. *Required on acceptance API only*
externalNumber	*string*	Is an identifier which is provided by the client for orders from 3rd party services such as GrubHub that can be entered via the Order Acceptance API. It can be used when a customer calls the store, the employee will be able to locate the customer's order.

1.5.1. Customer

This object defines a **customer** object

Property	Type	Description
name	string	The name of the customer. *Required*
phone	string	The customer's phone, formatted as AAA-XXX-NNNN. *Optional*

Property	Type	Description
address	string	The street number and address ("123 W Main St"). *Optional*
zip	string	The 5-digit zip code. Required when an address is required
city	string	City. *Optional*
state	string	The 2-letter state. *Optional*
value1	string	The apartment #, suite #, dorm name, etc. *Required when 'diff1' field is included in the request*
diff1	string	This is what Value1 is. *Required when 'value1' field is included in the request*
value2	string	Secondary location differentiator, e.g. the room # within the dorm. *Required when 'diff2' field is included in the request*
diff2	string	This is what Value2 is. *Required when 'value2' field is included in the request*
latitude	double	Latitude. *Required* if store is configured to accept customer coordinates
longitude	double	Longitude. *Required* if store is configured to accept customer coordinates
deliveryInstructions	string	Delivery instruction, if a Delivery order. *Optional*

1.5.2. Item

This object defines an order **item** object

Property	Type	Description
category	*string*	The category the item belongs to (such as "Pizza"). *Required*
item	*string*	The name of the item within the category (such as "Chesse"). *Required_*

Property	Type	Description
size	*string*	The size of the item (such as "Med"). *Required*
quantity	*integer*	The quantity of the item. *Optional (default: 1)*
sellingPrice	*float*	The selling price of the item. *Required*
ingredients	*Array of Ingredient objects*	Items' ingredients. *Optional*
externalRef	*string*	Is a unique identifier, provided by the client for identifying items. *Optional*
isInBucket	*boolean*	Indicates if the item is part of a bucket. *Optional*

1.5.3. Ingredient

This object defines an order item **ingredient** object

Property	Type	Description
ingredient	*string*	The name of the ingredient. *Required*
isLeftHalf	*boolean*	For a left-half ingredient. *Optional*
isRightHalf	*boolean*	For a right-half ingredient . *Optional*
ordered	*boolean*	A default ingredient that was de-selected. *Optional*
isAdHoc	*boolean*	For an Ad-Hoc ingredient. *Optional*
price	*boolean*	The price for an Ad-Hoc ingredient. *Optional*

1.5.4. Special

This object defines an order item **special** object

Property	Type	Description
special	*string*	The name of the special or coupon. *Required*
type	*enum*	Allowed values Special, Coupon. *Required*

Property	Type	Description
sellingPrice	*float*	The discounted price of the special or coupon. *Required*
discount	*float*	The special or coupon discount. *Required*
items	*Array of Item objects*	The items that this special or coupon applies to. *Required*
isCombo	*boolean*	Indicates if the special is a combo. *Optional*
couponCode	*string*	The coupon code of the special when applied. *Optional*

1.5.5. Adjustment

This object defines an order item **adjustment** object

Property	Type	Description
name	*string*	The name of the adjustment. *Required*
discount	*float*	The adjustment discount. *Required*

2. Resources

2.1. Orders

2.1.1. Create new order

Creates a third party order. `orderNum` will be ignored and assigned automatically by the POS.

URL	/ws/store/v1/orders
HTTP Method	POST
Available Since	9.1

Supported request params

suspend	*boolean*	Create the new order as suspended. This param can be used when there are missing information about the order *Optional*

Example request

```
$ curl 'https://localhost/ws/store/v1/orders?suspend=false' -i -X POST -H 'Accept: application/json' -H 'Content-
Type: application/json' -d '{
  "type" : "Pickup",
  "source" : "Third Party",
  "paid" : false,
  "isThirdPartyPickup" : false,
  "price" : 14.61,
  "tax" : 1.76,
  "taxableSubtotal" : 10.85,
  "tip" : 2.0,
  "discount" : 0.0,
  "deliveryCharge" : null,
  "deferTime" : null,
  "instructions" : "Ford Raptor",
  "customer" : {
    "name" : "Foodtec Joe",
    "phone" : "781-453-8620",
    "address" : "175 Highland Avenue",
    "zip" : "02494",
    "city" : null,
    "state" : "MA",
    "diff1" : "Apt",
    "value1" : "12G",
    "diff2" : "Left",
    "value2" : "Entrance",
    "diff3" : "5",
    "value3" : "B",
    "latitude" : 0.0,
    "longitude" : 0.0,
    "deliveryInstructions" : "Take the stairs or elevator to the fourth floor"
  },
  "items" : [ {
    "category" : "Pizza",
    "item" : "Cheese",
    "size" : "10\"",
    "quantity" : 1,
    "sellingPrice" : 11.85,
    "ingredients" : [ {
      "ingredient" : "Pepperoni",
      "isLeftHalf" : false,
      "isRightHalf" : false,
      "ordered" : true,
      "qualifiers" : [ "Extra" ]
    } ],
    "externalRef" : "df4bcc81-78da-46c1-af96-7dccc910e25e",
    "isInBucket" : false
  } ],
  "specials" : [ ],
  "adjustment":{
      "name":"Special Discount",
       "discount":1.0
  },
  "externalRef" : "b9a8fdc0-7f7c-441a-9b00-8bc580ce9146",
  "externalNumber" : "0123456789"
}'
```

Example response

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Content-Length: 1535

{
  "orderNum" : "5",
  "type" : "Pickup",
  "source" : "Third Party",
  "status": "made",
  "paid" : false,
  "isThirdPartyPickup" : "false",
  "price" : 14.61,
  "tax" : 1.76,
  "taxableSubtotal" : 10.85,
  "tip" : 2.0,
  "discount" : 0.0,
  "deliveryCharge" : null,
  "deferTime" : null,
  "promiseTime" : 1649064913000,
  "instructions" : "Ford Raptor",
  "customer" : {
    "name" : "Foodtec Joe",
    "phone" : "781-453-8620",
    "address" : "175 Highland Avenue",
    "zip" : "02494",
    "city" : null,
    "state" : "MA",
    "diff1" : "Apt",
    "value1" : "12G",
    "diff2" : "Left",
    "value2" : "Entrance",
    "diff3" : "5",
    "value3" : "B",
    "latitude" : 0.0,
    "longitude" : 0.0,
    "deliveryInstructions" : "Take the stairs or elevator to the fourth floor"
  },
  "items" : [ {
    "category" : "Pizza",
    "item" : "Cheese",
    "size" : "10\"",
    "quantity" : 1,
    "sellingPrice" : 11.85,
    "ingredients" : [ {
      "ingredient" : "Pepperoni",
      "isLeftHalf" : false,
      "isRightHalf" : false,
      "ordered" : true,
      "qualifiers" : [ "Extra" ]
    } ],
    "externalRef" : "10393283-fc19-4b43-9089-bd57e4150eac",
    "isInBucket" : false
  } ],
  "specials" : [ ],
  "orderTaker" : null,
  "externalRef" : "91461f24-d5a4-4d92-861f-98ea5d471b75",
  "externalNumber" : "0123456789"
}
```

2.1.2. Void order

Voids the third party order identified by the externalRef path variable.

URL	/ws/store/v1/orders/<externalRef>/void
HTTP Method	PUT
Available Since	9.4

Example request

```
$ curl 'https://localhost/ws/store/v1/orders/bb839647/void' -i -X PUT -H 'Accept: application/json' -H 'Content-
Type: application/json' -d
'{"reason": "The customer placed the order by mistake"}'
```

Example response

```
HTTP/1.1 204 No Content
```

1. *date_time* values are formatted in milliseconds since January 1, 1970, 00:00:00 GMT