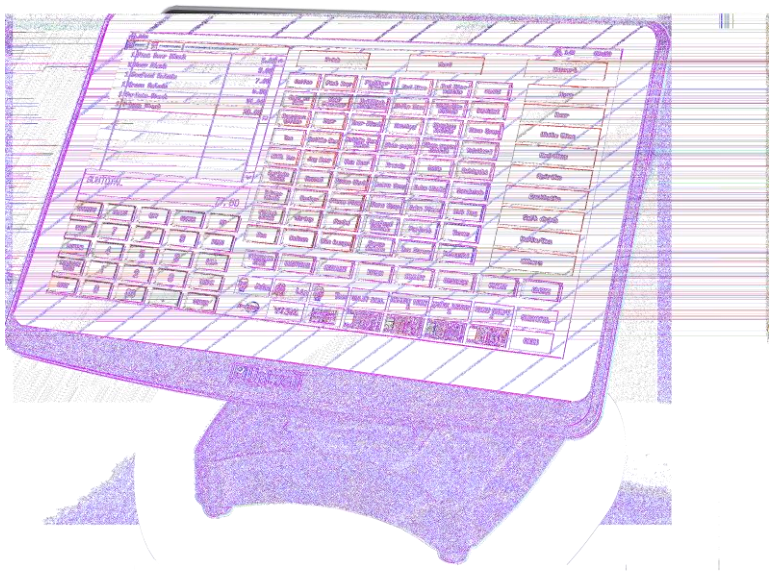
Uniwell
2019-06-21

WEB Ordering Protocol for Uniwell POS



Rev. 015

Document History

Date	Revision	ECR Version	Contents
2013-08-09	001		New Release
2013-09-20	002		4 points added, and 2 error messages added.
2013-10-16	003		"track" and <freetext> tag added.
2013-10-18	004		Increase the number of digits in the freetext to 40 Printer can be pecified by the attaribute of <request> tag.
2013-11-08	005		discount usage added
2015-07-15	006		"modifier" added to plu tag
2015-07-17	007		<discount>tag added and "price" added to plu tag Maximum quantity value changed 99 → 9999 Error code "908 Over limitation" added
2015-10-14	008		6 error messages added. (907, 909 to 913)
2016-08-04	009		Destination-related explanation added
2017-01-19	010		Destination-related explanation added
2018-07-17	011	V4.20	Separate payment, Tip, Item change, Cancel transaction added
2019-01-29	012	V4.42	Media closing of the charge-to-customer transactions via WOP
2019-02-27	013	V4.43	Destination and Order number (callNumber) added
2019-05-1	014	V4.60	Reservation protocol added
2019-06-17	015	V4.60	Error code 916 to 920 added Mobile_phone added to "Make reservation" Customer Order added

CONTENTS

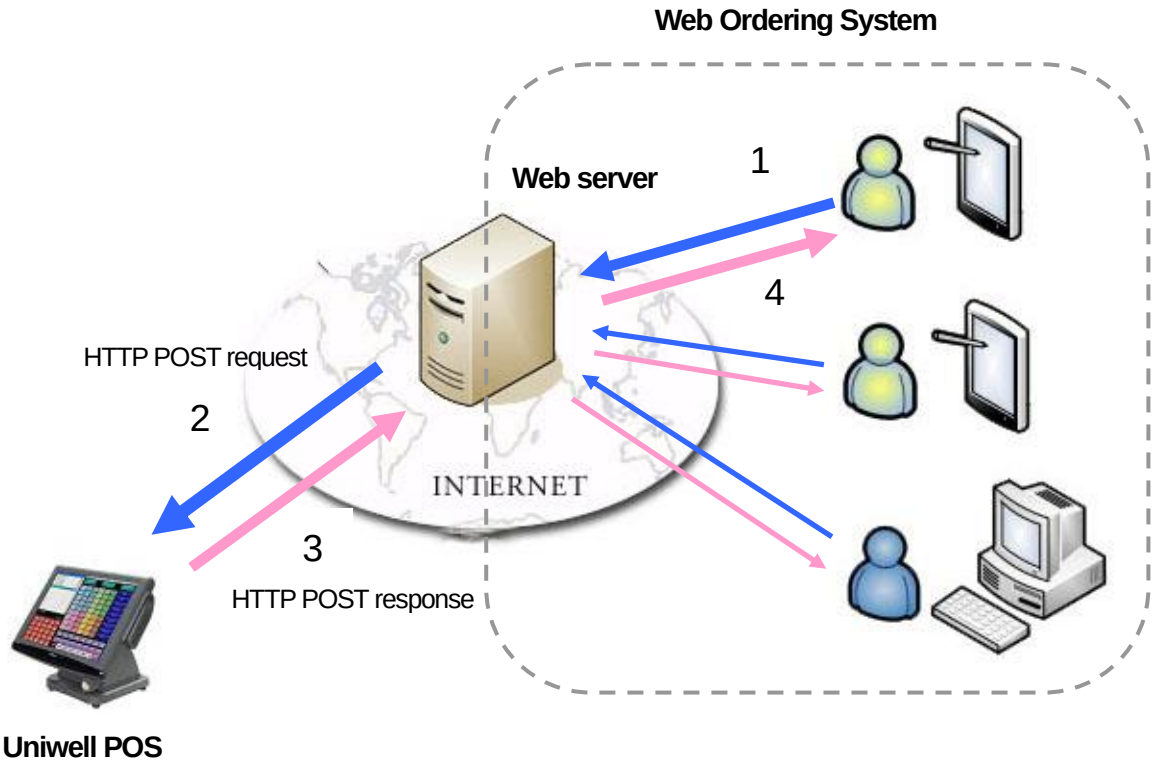
1. Overview

WEB Ordering Protocol (WOP) is a protocol to issue an order to Uniwell POS on the Internet.
Messages are sent/received between Uniwell POS (as server) and an Order terminal (as client).
WOP message is a HTTP-POST request, and a request body is in XML format.
The request is executed at the server, and the result is returned to the client in XML format as well.

Remote Ordering System

by HTTP POST Request /Response (Proposal)

System configuration and flow of order entry



Order entry flow

1	Web ordering system receives an order from a customer. The web site with such a facility is supposed to be prepared by a shop or a restaurant, or by a service provider as a portal site to cover multiple shops and restaurants. Items to order on the web site should match those existing on Uniwell POS, therefore, Program data including PLUs need to be obtained from Uniwell POS to analyze beforehand.
2	The contents of the order is converted into XML format (as stipulated in the next chapter), which then the web server requests to be processed in Uniwell POS by HTTP POST request.
3	Receiving request command from the web server, if UNIWELL POS in a status able to receive an order, Uniwell POS analyzes the order and process it if there's no problem, then sends back necessary information to the web server. In case Uniwell POS is not in a status to receive an order, or there's any problem in the order contents, then Uniwell POS sends back Error to the web server.
4	The web server checks if the order has been processed or not seeing the response from Uniwell POS, the result of which needs to be informed to the customer by some way or other.

Example of Request

```
POST /wop HTTP/1.1
Content-Type: text/xml
Content-Length: 246
Host: 192.168.251.90
User-Agent: Apache-HttpClient/UNAVAILABLE (java 1.4)

<?xml version="1.0" encoding="utf-8"?>
<request method="store" clerk=”1”>
   <track customerNumber=”1234”>
      <plu quantity="5">0000000000000001</plu>
      <plu quantity="2">0000000000000002</plu>
      <plu quantity="3">0000000000000003</plu>
   </track>
</request>
```

Example of Response

```
HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: 263

<?xml version="1.0" encoding="utf-8"?>
<response>
     <track orderNumber="732" subtotal="1225"/>
</response>
```

2. Header

URI of HTTP-POST request should be /wop.
Content-Type should be text/xml.
Content-Length is a must, and the correct value should be specified.
Host and User-Agent should be specified as well.

3. Payload

Payload is a data body in XML format, and it consists of a single <request>.
XML header <?xml version="1.0" encoding="utf-8"?> should be added all the time.
Encoding in XML should be UTF-8.

As for some characters which can not be used in XML as they are, use the description (by entity reference) as below.

Original character	Description by entity reference
&	&
<	<
>	>
'	'
"	"

4. Store

Store the order at the server.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<request method="store" clerk=”1” printer="VIRTUAL PRINTER 01">
    <track customerNumber="1234" destination=”1”>
      <freetext>                Free Text1                </freetext>
      <freetext>                Free Text2                </freetext>
      <freetext>                Free Text3                </freetext>
      <plu quantity="3">0000000000000000001</plu>
        <plu quantity="3" condiment="true">0000000000000000120</plu>
        <plu quantity="2" condiment="true">0000000000000000122</plu>
      <plu quantity="2" price="220">0000000000000000002</plu>
        <instruction>Medium Rare</instruction>
      <plu quantity="1" modifier="1" >0000000000000000003</plu>
       <discount>100</discount>
       <percent number="2">500</percent>
    </track>
</request>
```

Attribute “method” of <request> tag should be “store”.

As an attribute of <request> tag, a clerk number can be specified.

Attribute	Vaule	Explanation	Type
clerk	1 ~ 255	clerk sequential number	mandatory

Recommended setting and operation:
Program one specific clerk only for Web Ordering (he/she will never sign on the Server/POS), and use the clerk number (clerk sequential number) for Web Ordering operation.

Destination setting for the clerk (in Program > Clerk > Clerk General) is applied to the transaction so that Destination-related function can be used. Destination information is useful to notify that it is an order by Web ordering when it is printed on KP, etc. Also, settings related to the destination (in Program > Otherers > Destination) are also effective. However, “Destination input compulsory” settings in Destination System Option are not supported.

Printer can be specifed by the attaribute of <request> tag.

Attribute	Vaule	Explanation	Type
printer	upto 20 characters.	Printer name	option

track

As an attribute of <track> tag, a customer number (which is registered in the server) can be specified.

Attribute	Vaule	Explanation	Type
customerNumber	1 ~ 99999999999999999	Customer number	optional
destination (V4.43)	1 ~ 10	Destination	optional

Describe the ordered items in a <track> tag.

<plu> tag

Specify the registered PLUs.

As an attribute of <plu> tag, PLU code should be specified. PLU code is an 18-digit number, and preceding 0 (zero) can not be skipped.

Attributes of <plu> tag

Attribute	Vaule	Explanation	Type
quantity	1 ~ 9999		mandatory
price	1 ~ 999999999		optional
condiment	true / false	true: registered as a condiment of the preceding PLU	optional
modifier	1, 2		optional

<instruction> tag

Cooking instruction of the preceding PLU is specifed by the attaribute of <instruction> tag.

Specify it with upto 20 characters.

<freetext> tag

Free text is specifed by the attaribute of <freetext> tag.

Specify it with upto 40 characters.

Notes:

Please be sure not to put freetext between PLU and subsidiary item of that PLU (Condiment or Instruction etc.).

```
<plu quantity="1">00000000000000001</plu>
<freetext>                Free Text                </freetext>        <-- N.G
  <plu quantity="1" condiment="true">00000000000000120</plu>

<plu quantity="1">00000000000000001</plu>
  <plu quantity="1" condiment="true">00000000000000120</plu>
<freetext>                Free Text                </freetext>        <-- N.G
  <instruction>Medium Rare</instruction>

<plu quantity="1">00000000000000001</plu>
  <plu quantity="1" condiment="true">00000000000000120</plu>
  <instruction>Medium Rare</instruction>
<freetext>                Free Text                </freetext>        <-- O.K
<plu quantity="1">00000000000000002</plu>
```

<persent> tag

With an attribute “number” of <percent> tag, kind of percent is specified. And, if any number is specified in element, it will override the programmed percent rate.
Other settings (Item% or Subtotal %, etc.) will depend on the settings in POS.

Attribute	Vaule	Explanation
number	1 ~ 10	Percent number (depending on the server POS setting)

ex.) %2, 5.00%

```
<percent number="2">500</percent>
```

<discount> tag

Discount amount is specifed by the attaribute of <discount> tag. (Max. 999999999)
Other settings (Taxable affects, etc.) will depend on the settings in POS.

ex.) -1.00

```
<discount>100</discount>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response>
   <track orderNumber="732" subtotal="1225" callNumber=”001”/>
</response>
```

With an attribute of <track> tag, response to “store” is returned.

Attribute	Vaule	Explanation	Note
orderNumber	1 ~ 99999999	Stored order number	
subtotal	0~999999999	Subtotal of stored order	decimal point is NOT added
callNumber (V4.43)	001 ~999	Calling number	optionnal

Notes:
- Position of decimal point is defined by the setting of Cash media at the server POS.
- When an error occurs, error condition is notified by the faulse response (explained later).
- callNumber is added only when the following setting is enabled on the POS.
  Program > System > System Option > Track > Page3 > Use order number

5. Media Close

Media close notifies the payment of an order to the server which is settled at the client.
Such an order should be stored and get an order number in advance.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<request method="close" clerk=”1” printer="VIRTUAL PRINTER 01">
   <track orderNumber="732"/>
   <tip>200</tip>
   <media number="1">725</media>
   <media number="2">1225</media>
</request>
```

Attribute “method” of <request> tag should be “close”.
With an attribute “orderNumber” of <track> tag, an order to be paid is specified.
With an attribute “number” of <media> tag, kind of media is specified, and with element, paid amount is specified.

Attribute	Vaule	Explanation
Number	1 ~ 20	Media number (depending on the server POS setting)

Notes:
- Payment amount should agree with the subtotal amount of the order.
- Payment with multiple media supported. (V4.20)

<tip> tag (V4.20)

Tip amount is specifed by the attaribute of <tip> tag. (Max. 999999999)

ex.) 2.00

```
<tip>200</tipt>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response>
</response>
```

When media close is done correctly, <response> tag is returned without any contents.
When an error occurs, error condition is notified by the faulse response (explained later).

6. Reservation (V4.60)

Make reservation

```
<?xml version="1.0" encoding="utf-8"?>
<request method="make_reservation" clerk="1">
  <booking>
    <guest lastname= "Taro" firstname= "Osaka" phone="06-1234-5678" mobile_phone="080-2345-6789"
    email="taro@osaka.co.jp"/>
    <datetime in="2019-05-14T19:00:00" out ="2019-05-14T21:00:00" />
    <table>1234</table>
    <person>2</person>
    <deposit media="1"> 12345 </deposit>
    <freetext> Free text (40 digits) </freetext>
  </booking>
</request>
```

Change reservation

ex.) out time 21:00 →22:00, person 2→3

```
<?xml version="1.0" encoding="utf-8"?>
<request method="change_reservation" clerk="1">
  <booking id="123456">
    <datetime out="2019-05-14T22:00:00"/>
    <person>3</person>
  </booking>
</request>
```

Cancel reservation

```
<?xml version="1.0" encoding="utf-8"?>
<request method="cancel_reservation" clerk="1">
  <booking id="123456"/>
</request>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response>
  <booking id="123456"/>
</response>
```

Inquiry (reservation status)

ex.) Reservation status on 14/May/2019

```
<?xml version="1.0" encoding="utf-8"?>
<request method="inquiry_reservation" clerk="1">
  <booking>
     <datetime  in="2019-05-14T00:00:00"/>
  </booking>
</request>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response>
   <booking id="123456">
      <guest  lastname= "Taro" firstname= "Osaka" phone="06-1234-5678" email="taro@osaka.co.jp"/>
      <datetime  in="2019-05-14T17:00:00"  out ="2019-05-14T19:00:00" />
      <table>1001</table>
      <person>2</person>
      <deposit> 12345 </deposit>
      <freetext>          Free text (40 digits)          </freetext>
   </booking>
   <booking id="123457">
      <guest  lastname= "Jiro" firstname= "Tokyo" phone="03-1234-5678" email="Jiro@tokyo.co.jp"/>
      <datetime  in="2019-05-14T19:00:00"  out ="2019-05-14T21:00:00" />
      <table>1002</table>
      <person>2</person>
      <deposit> 23456 </deposit>
      <freetext>          Free text (40 digits)          </freetext>
   </booking>
</response>
```

ex.) Reservation status on 14/May/2019 after 19:00

```
<?xml version="1.0" encoding="utf-8"?>
<request method="inquiry_reservation" clerk="1">
  <booking>
     <datetime  in="2019-05-14T19:00:00"/>
  </booking>
</request>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<response>
   <booking id="123457">
      <guest  lastname= "Jiro" firstname= "Tokyo" phone="03-1234-5678" email="Jiro@tokyo.co.jp"/>
      <datetime  in="2019-05-14T19:00:00"  out ="2019-05-14T21:00:00" />
      <table>1002</table>
      <person>2</person>
   </booking>
</response>
```

7. Customer Order (V4.60)

Existing Customer

```
<?xml version="1.0" encoding="utf-8"?>
<request method="customerorder" clerk="1">
    <track customerNumber="0777700002">
        <plu quantity="1">0000000000000000001</plu>
        <plu quantity="1">0000000000000000002</plu>
        <plu quantity="1">0000000000000000003</plu>
        </track>
</request>
```

New Customer

```
<?xml version="1.0" encoding="utf-8"?>
<request method="customerorder" clerk="1">
    <track>
        <guest lastname= "Taro" firstname= "Osaka" phone="06-1234-5678" mobile_phone="080-1234-5678" email="taro@osaka.co.jp"/>
        <plu quantity="1">0000000000000000001</plu>
        <plu quantity="1">0000000000000000002</plu>
        <plu quantity="1">0000000000000000003</plu>
        </track>
</request>
```

8. Other functions

Item change to the already-stored order (V4.20)

"orderNumber" is added to the request method "store", then it will be considered as an addition/change to the already-stored order.

ex.) The command to add 1 pc of PLU1 and delete 2 pcs of PLU2 in the already-ordered transaction with Order number "1" will be as below.

```
<request method="store" clerk="1">
  <track orderNumber="1">
    <plu quantity="1">0000000000000001</plu>
    <plu quantity="2" delete="true">0000000000000002</plu>
  </track>
</request>
```

In case there is no orderNumber in "store", it is considered as a new order as before.

```
<request method="store" clerk="1">
  <track>
    <plu quantity="1">0000000000000001</plu>
    <plu quantity="2">0000000000000002</plu>
  </track>
</request>
```

Cancel transaction (V4.20)

Order cancel will be possible by adding "cancel" to the request method.
At that timing, an orderNumber will be added to the track tag.

```
<request method="cancel" clerk="1">
<track orderNumber="1"/>
</request>
```

Media closing of the charge-to-customer transactions via WOP (V4.42)

It is now possible to close the charge-to-customer transactions by media via WOP (Web Ordering protocol).
There are following three ways to specify which customer-charged transactions should be closed by media.

1. By Receipt number

```
<request method="chargeclose" clerk="1">
<track customerNumber="123456" receiptNumber="123"/>
<media number="1">100</media>
</request>
```

2. By Bill number

```
<request method="chargeclose" clerk="1">
<track customerNumber="123456" orderNumber="123"/>
<media number="1">100</media>
</request>
```

[In case both Receipt number and Bill number exist]

POS will search the transaction which has the same Receipt number and Bill number, and close it (if either of them is different, then, POS will return "Not found" error).

3. By Receipt number and Bill number

```
<request method="chargeclose" clerk="1">
<track customerNumber="123456" receiptNumber="123"/> orderNumber="456"/>
<media number="1">100</media>
</request>
```

In addition, "Receipt OFF" is supported. It was already possible to specify the destination printer. From this version, it is now possible not to print. In order not to print, add the red-character part below, when sending the command. In order to specify the destination printer, enter the virtual printer name, instead of "off".

```
<request method="chargeclose" clerk="1" printer="off">
<track customerNumber="123456" receiptNumber="123"/>
<media number="1">100</media>
</request>
```

9. Fault Response

When the server detects a problem during request processing, it returns <fault>.

```
<?xml version="1.0" encoding="utf-8"?>
<response>
      <fault code="901">Track Not Found</fault>
</response>
```

With an attribute “code” of <fault> tag, an error code is specified, and with element, an error message is specified.

List of Error code and message

Code	Message	Meaning
900	Server Busy	Server is in a maintenance condition, and can not process the request.
901	Track Not Found	The designated order can not be found.
902	Amount Not Agree	Payment amount does not agree with the subtotal amount.
903	Customer Not Found	The designated customer can not be found.
904	PLU Not Found	The designated PLU can not be found.
905	Clerk busy	The clerk already signs on in the server, or a track is in “Clerk store” condition to the clerk.
906	Track busy	A track with a number designated by the orderNumber is already picked up in the server.
907	Printer not found	The designated printer not found
908	Over limitation	Quanity or Amount value exceeds the maximum limitation.
909	Stockout	Ordered PLU stock is zero or negative.
910	Zero price entry	If PLU with “Zero price entry prohibited” is registered with zero price.
911	Modifier prohibited	Modifier entry is prohibited with ordered PLU.
912	Req.Auto Bill number	“Automatic bill number system” is not programmed in the POS.
913	Invalid percent entry	Invalid percent entry (ex. item percent prohibited etc.)
914	Invalid payment	Not supported media number (ex. Charge cuatomer etc.)
915	Short tendering	Payment amount not enough for Subtotal amount at Media close
916	Deactivated PLU	Deactivated PLU cannot be registered
917	Reservation server busy	Reservation (Deposit book) server is busy (in program or Report mode)
918	Reservation full	Reservation(Deposit book) memory is full Please issue a Z-Deposit book report and delete the "used" record.
919	Reservation not found	The Reservation (Deposit book) record with the specified ID cannot be found.
920	Reservation duplicated	The reservation for the requested date and time has already been registered.
999	Internal Server Error	The requests can not be handled, because of the problem other than above listed issues