



Duet Module Interface Specification

for a

**Yealink MS Teams Rooms  
Gateway  
Utility**



## TABLE OF CONTENTS

Introduction .....	3
Overview .....	3
Implementation .....	3
Port Mapping .....	4
Buttons .....	5
Channels .....	5
Levels .....	5
Command Control .....	6
Programming Notes.....	9
Protocol Notes.....	10
Module Web Configuration .....	12
NetLinx Platform Compatibility .....	12
Confidential Data Encryption.....	12
Setting Web Configuration Tool Name.....	12
Web Configuration Tool Tabs.....	12

## Revision History

Date	Version	Initials	Comments
5-2022	1.0.0	IRC	Initial release

## Introduction

This is a reference manual to describe the interface provided between an AMX NetLinX system and a **Yealink MS Teams Rooms device(s)**. The interface was tested using version **1.6.201** of the firmware.

This module has a **minimum AMX hardware** requirement of an ***NX-Series* controller with master firmware 1.6.201**

This module has a **minimum firmware** requirement for the *Yealink* device of **version 2.30.50.0**.

This module was written using CaféDuet firmware version v3.21.343, NetLinX Studio version 4.4 build 1914, CaféDuet application platform and runtime version 2.0.5, and CaféDuet application plug-in version 3.2.0.

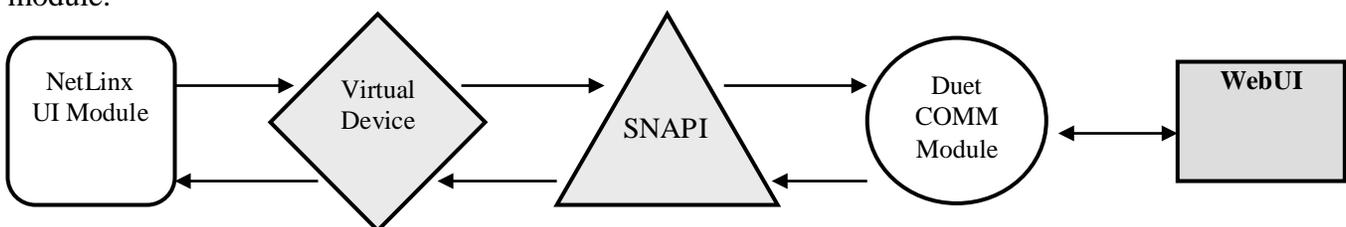
This module implements the **ITGWebServices Jett Server module** as the parent module and all instances must have the SERVER.PROPERTY-Duet-Device property set the same with a valid Duet Virtual Device DPS.

## Overview

The COMM module translates between the standard interface described below and the **Yealink MS Teams Rooms UI** protocol. It parses the buffer for responses from the **Yealink MS Teams Rooms UI**, sends strings to control the **Yealink MS Teams Rooms UI**, and receives commands from the UI module or telnet sessions.

This module uses the standard interface described below and parses the command responses for feedback.

The following diagram gives a graphical view of the interface between the interface code and the Duet module.



## Implementation

To interface to the AMX **Yealink MS Teams Rooms Gateway** module, the programmer must perform the following steps:

1. Define the device ID for each **TeamsUI** module instance.
2. Define the virtual device ID that each **TeamsUI** module instance will use to communicate with the main program and User Interface. Duet virtual devices use device numbers 41001 - 42000.

3. Define the virtual device ID that the **ITGWebService JettyServer** module will use to communicate with the main program and User Interface. Duet virtual devices use device numbers 41001 - 42000
4. This Duet module must be included in the program with a DEFINE\_MODULE command. This command starts execution of the module and passes in the following key information: the device ID of the parent to be controlled, and the virtual device ID for communicating to the main program.
5. Set the **SERVER.PROPERTY-Duet-Device,<DPS>** at the online event of all parent module instances.
6. Set the **webui-configuration-file** Property at the online event for each module instance

An example of how to do this is shown below.

DEFINE\_DEVICE

```
dvDevice      = 0:3:0      // The parent module device
vdvDevice     = 41001:1:0 // The parent module virtual device
vdvServer     = 41999:1:0 // The Server Module virtual device
```

```
DEFINE_MODULE 'Yealink_MSTeamsRooms_Gateway_dr1_0_0'UI1(vdvDevice, dvDevice)
```

DEFINE\_EVENT

```
data_event[vdvDevice]
{
    online:
    {
        send_command data.device,""SERVER.PROPERTY-Duet-Device,'
                                itoa(vdvServer.NUMBER),
                                ':',itoa(vdvServer.PORT),
                                ':',itoa(vdvServer.SYSTEM)";

        send_command data.device,'PROPERTY- webui-configuration-file,room1.json';

        send_command data.device,'REINIT';
    }
}
```

**Port Mapping**

This module uses multiple virtual devices in order distinguish events for one zone from another.

Virtual Device	Buttons	Channels	Levels	Control	Feedback
41001:1:0	<b>All</b>	<b>All</b>	<b>All</b>	All	<b>All</b>
41001:2:0 – 41001:20:0	<b>All</b>	<b>All</b>	<b>All</b>	n/a	<b>n/a</b>

**Table 1 - Port Mapping**

## **Buttons**

The **Microsoft TeamsUI** initiates system control via button events (NetLinx commands *press*, *release*, *holdf*) sent from the Gateway module. The buttons supported by the Gateway module are listed below.

<b>Button</b>	<b>Description</b>
<b>1.. 1024</b>	Button action initiated from the WebUI

## **Channels**

The UI module controls the **Microsoft TeamsUI** via channel events (NetLinx commands *pulse*, *on*, and *off*) sent to the Gateway module. The channels supported by the Gateway module are listed below.

<b>Channel</b>	<b>Description</b>
<b>1.. 1024</b>	Set button feedback at WebUI
<b>251</b>	ON: Successfully started the Server – feedback only (Port 1) OFF: Server is not started
<b>252</b>	ON: Server initialization complete – feedback only (Port 1) OFF: Server initialization is not complete

**Table 2 - Virtual Device Channel Events**

## **Levels**

The UI module controls the **Microsoft TeamsUI** via level events (NetLinx command *send\_level*) sent to the Gateway module. The levels supported by the Gateway module are listed below

<b>Level</b>	<b>Description</b>
<b>1.. 16</b>	Level control and feedback for the WebUI

**Table 3 - Virtual Device Level Events**

## Command Control

The UI module controls the **Microsoft TeamsUI** via command events (NetLinx command *send\_command*) sent to the module. The commands supported by the Gateway module are listed below.

Note: An ‘\*’ indicates an extension to the standard API.

Command	Description
*CLIENTS.CONNECTED-DISCONNECT,ALL	Disconnects all connected clients. CLIENTS.CONNECTED-DISCONNECT,ALL
*CLIENTS.CONNECTED-DISCONNECT,<host>	Disconnects all connected clients at specified host. CLIENTS.CONNECTED-DISCONNECT,192.168.2.100
*?CLIENTS.CONNECTED	Queries for all connected clients. Returns a list of all connected clients.
?DEBUG	Request the state of the debug feature
DEBUG-<value>	Set the state of debugging messages in the UI module and the Comm. module.  <value> : 1 = set only error messages on 2 = set error and warning messages on 3 = set error, warning & info messages on 4 = set all messages on  DEBUG-1
?PROPERTY-<key>	Get the value of a property <key>. If the value is not initialized, an empty string is returned.  <key> : webui-resources-zip webui-configuration-file  ?PROPERTY-webui-configuration-file

PROPERTY-<key>, <value>	<p>Set the value of property &lt;key&gt; to &lt;value&gt;. This must be followed by the REINIT command to take effect.</p> <p>&lt;key&gt; : webui-resources-zip          &lt;value&gt; : path and name of saved file          [default] : teams.zip</p> <p>&lt;key&gt; : webui-configuration-file          &lt;value&gt; : path and name of saved file</p> <p>PROPERTY-webui-resources-zip,/uifiles/teams_v1.zip</p>
REINIT	Reinitializes the module. Refreshes all file data as set at webui-resources-xxxx Properties.
*SERVER.PROPERTY-<key>, <value>	<p>Set the value of server property &lt;key&gt; to &lt;value&gt;.</p> <p>&lt;key&gt; : Duet-Device          &lt;value&gt; : DPS of server device</p> <p>SERVER.PROPERTY-Duet-Device,41999:1:0</p>
*?URL-HTTP	Query for WebUI URL.
*?URL-WS	Query for WebSocket direct access URL.

**Table 4 – Send Command Definitions**

### Command Feedback

The COMM module provides feedback to the User Interface module for **Microsoft TeamsUI** changes via command events. The commands supported are listed below.

**PLEASE NOTE:** Feedback is only provided when there is a state change. If no state change resulted from the command sent in, then no feedback will be returned.

Command	Description
*CLIENTS.CONNECTED- <index>,<total>,<host/ip>,<username>	List of all connected clients at all parent modules' instances.  <index> : item # <total> : total items <host/ip> : client addressing <username> : Authorization  CLIENTS.CONNECTED-1,1,192.168.2.10,administrator
*CLIENT.CONNECT- <host/ip>,<username>	New client connected notification  CLIENT.CONNECT-192.168.2.74,administrator
*CLIENT.DISCONNECT- <host/ip>,<statusCode>	Connected client disconnect notification  StatusCodes: 1001: Server shutdown 1002: Protocol Error 1003: Bad Data received 1007: Bad Payload recieved 1008: Policy Violation - failed authentication or authorization 1010: Required Extension is missing 1011: Server Error 1012: Service Restarted 1013: Try Again Later 1014: Invalid Upstream Response 1015: Failed TLS Handshake  CLIENT.DISCONNECT-192.168.2.74,1008
PROPERTY-<key>,<value>	Feedback on the value of property <key>. <b>Note:</b> An empty string is returned if the property has no value  <key> : webui-resources-zip <value> : path and name of saved file  <key> : webui-configuration-file <value> : path and name of saved file  PROPERTY-webui-resources-zip,teams.zip

<pre>*URL- HTTP,&lt;protocol&gt;://&lt;host&gt;:&lt;port&gt;/web/ui/&lt;dps.NUMBER&gt;/home</pre>	<p>WebUI URL where &lt;host&gt; will be the static IP or FQDN Hostname if set.</p> <p>URL-HTTP, <a href="http://192.168.2.75:8080/web/ui/41101/home">http://192.168.2.75:8080/web/ui/41101/home</a></p>
<pre>*URL- WS,&lt;protocol&gt;://&lt;host&gt;:&lt;port&gt;/web/ui/&lt;dps.NUMBER&gt;/ws</pre>	<p>WebSocket direct access URL where &lt;host&gt; will be the static IP or FQDN Hostname if set.</p> <p>URL-WS, <a href="ws://192.168.2.75:8080/web/ui/41101/ws">ws://192.168.2.75:8080/web/ui/41101/ws</a></p>

**Table 5 - Command Feedback Definitions**

### **Device Notes**

This hardware requires its configuration to be updated with the URL of the module instance as published from the Command API or at the Web Configuration control tab.

- 1) Enter the Windows root machine as the device admin user.
- 2) Launch the Yealink plug-in software.
- 3) Press the “Settings” icon.
- 4) Select “Room Control” from the list.
- 5) Toggle the “Room Control” option on.
- 6) Enter the module instance URL in the “Web Address” input field.
- 7) Exit the Windows root machine and restart with the MTR Interface/App view.

Refer to the device manual for complete and detailed instructions.

### **Programming Notes**

This module has an external jar dependency (jose-4j-0.7.11.jar) which should be included with the Project package and loaded to the controller with the other jar files.

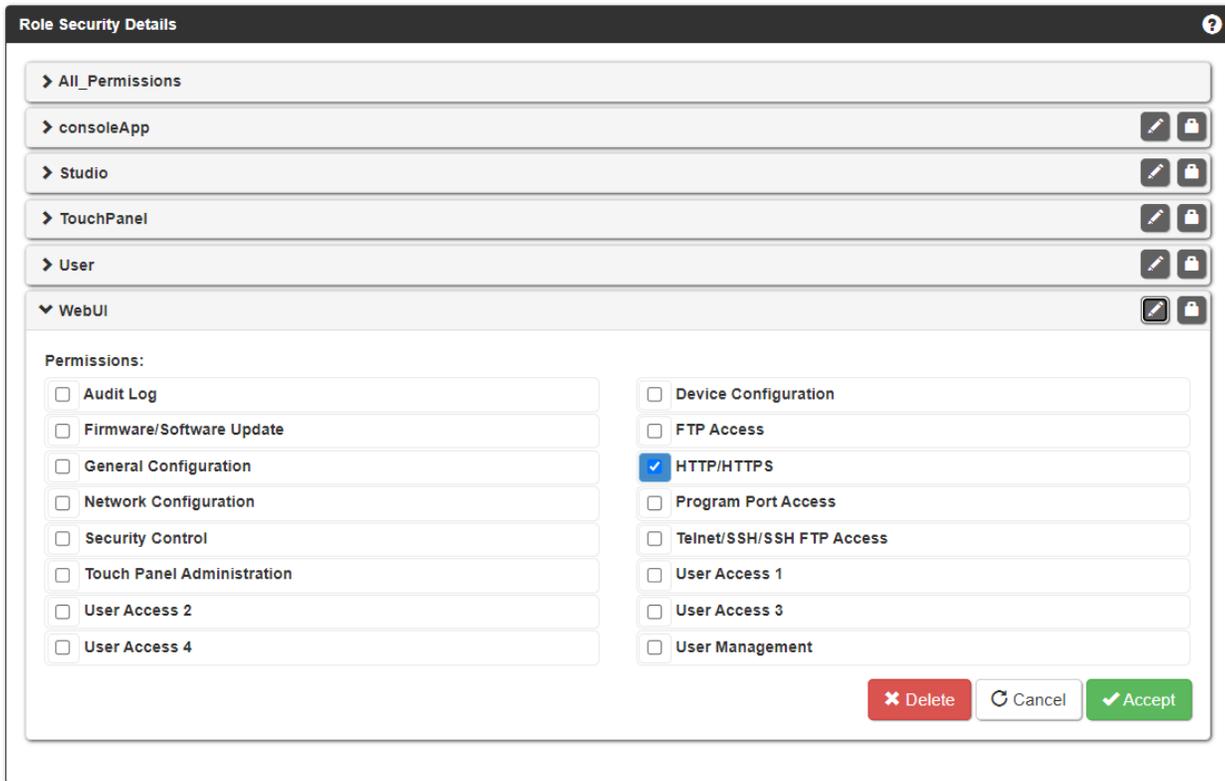
The **SERVER.PROPERTY-Duet-Device,<DPS>** must be set the same at all module instances that implement the **ITGWebServices Jetty Server** module.

The **ITGWebService Jetty Server** module is embedded in this module and does not require a separate download or instantiation.

Calling REINIT at the module instance or rebooting the controller is required to read in updated webui files (**webui-configuration-file** or **webui-resources-zip** Properties).

## Protocol Notes

Authentication is required for all connections to the server regardless of Protocol. The provided credentials are validated against the users in the Controller's Security settings. It is recommended to create a WebUI role access limited to only HTTP/HTTPS then create user(s) with the WebUI role for each connection with the Type set to Device. Once Authenticated, persistent Authorization is enabled by storing a signed and encrypted JSON Web Token (JWT) as an httpOnly Cookie on the client machine.



### Add New User ✕

**User Name:**  **Type:**

**Roles:**

Your new password must be between 4 to 20 characters.

**New Password:**

**Confirm Password:**

**Permissions:**

## Module Web Configuration

This module implements a Web Configuration tool that allows viewing with an option to edit select Properties required to establish and maintain communications to the device. Data applied at the Web Configuration tool will be persistently stored on the controller and will have precedence over data set through the virtual device Property command API when the module is reinitialized. Selecting the *Restore Defaults* option will delete the persistent data storage and precedence will revert to the virtual device Property command API.

## NetLinx Platform Compatibility

Accessing the Web Configuration tool requires a minimum NetLinx Master firmware version of 1.6.x (NX-Series).

## Confidential Data Encryption

All confidential data is encrypted in transit from the controller and masked at the client. The encryption mode is AES 256 GCM; the key and initialization vector (IV) are unique to each controller. Confidential data applied to the module configuration at the client is sent back to the controller as clear text and encrypted at rest in the persistent storage. **It is strongly recommended that HTTP is explicitly disabled with HTTPS enabled exclusively in the General System Security settings of the master.**

## Setting Web Configuration Tool Name

The name of the Web Configuration tool as it appears in the icon on Web Controls menu on the controller System Configuration page can be set from NetLinx code through the noted virtual device Property command APIs. When not validly set through a Property the default tool name is “MS Teams UI <Virtual-Device.NUMBER>”.



The module Properties will be evaluated with the following priority to set the name of the Web Configuration tool:

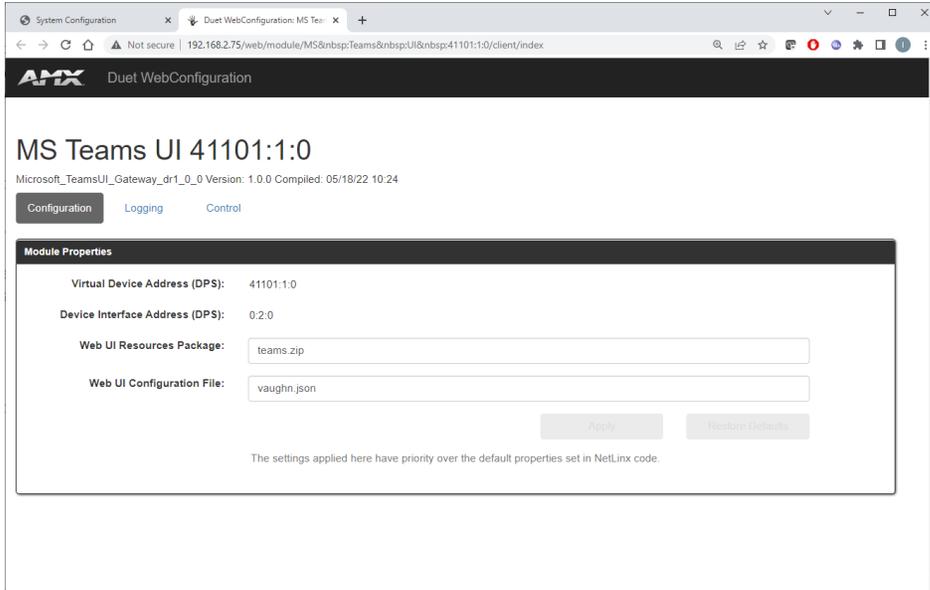
1. `send_command vdvDevice,"PROPERTY-WEB.CONFIG.NAME,'<name>";`
2. `send_command vdvDevice,"PROPERTY-RMS-ASSET-NAME,'<name>";`

## Web Configuration Tool Tabs

The following tabs are enabled in the web configuration tool.

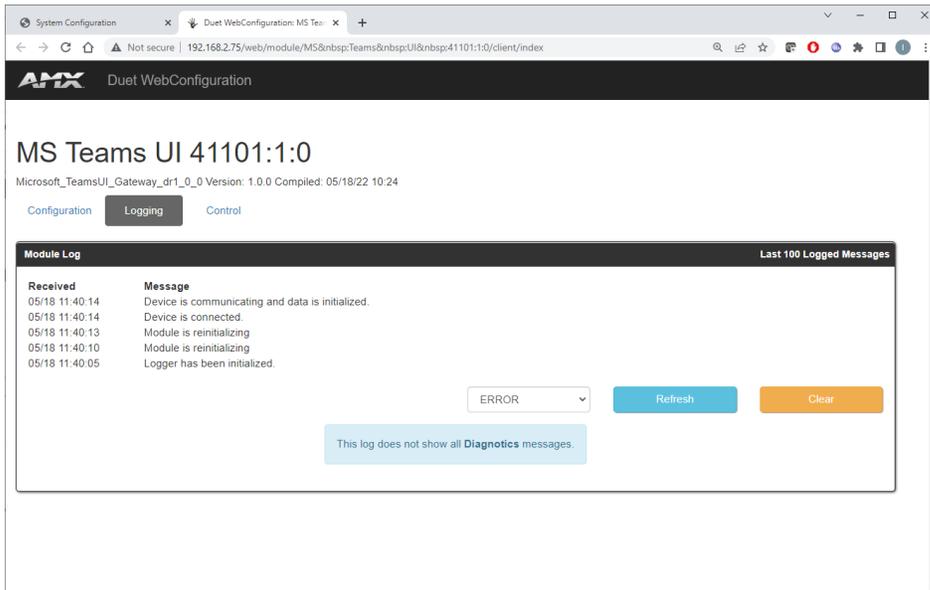
## Configuration

The Configuration Tab enables access to select Module Properties required for establishing and maintaining communications to the controlled device.



## Logging

The Logging Tab enables monitoring of select events and setting the Debug level in the module.



## Control

The Control Tab enables basic device control and status feedback for the purpose of module commissioning.

