

Yealink DECT Phones XML Browser Developer's Guide

Contents

About This Guide.....	3
Who should use this guide?.....	3
XML and Yealink DECT Phones.....	3
What is XML?.....	3
XML Format.....	4
Functionality.....	5
How does it work?.....	5
Phone-initiated Application.....	5
Yealink DECT Phone XML Objects.....	6
XML Object Definitions.....	6
TextMenu Object.....	6
TextScreen Object.....	10
InputScreen Object.....	13
Customizable Soft Keys.....	20
Some Development Guidelines.....	22
Configuring the HTTP Server.....	23
Configuring an XML Browser Key.....	23
Troubleshooting.....	25
Troubleshooting Tools.....	26
Parsing Error Debug Example.....	26

About This Guide

XML browser is a simple browser based on XML language and http / https service. You can dynamically generate XML files that meet your requirements for the phone functions on the server side according to the established syntax, and then download them to the DECT phones.

This guide shows you how to use XML API to control the LCD screen display of DECT phones as well as its configuration.

This guide applies to the following Yealink devices:

Product	Device	Firmware Version
W80 DECT IP Multi-Cell System	W80DM/W80B	103.83.0.70.rom or later
	W56H	61.83.0.90.rom or later
	W53H	88.83.0.90.rom or later

Who should use this guide?

This guide is designed specifically to provide development engineers, system administrators, or network engineers with information for developing and deploying customized client services on DECT phones using the XML browser feature.

This guide is not intended for end users and does not provide user-level information on how to use any specific XML applications.

Before reading this guide, you should be familiar with the following:

- Basic text editors, or full IDE-like Eclipse or Microsoft Visual Studio for creating or writing code.
- General application and software development.
- Adequate planning, creating, and testing resources needed to produce a fully deployable web-based application.
- DECT IP multi-cell system and provisioning methods.
- How to use an XML editor.
- The XML-based schema and syntax.

XML and Yealink DECT Phones

What is XML?

XML stands for Extensible Markup Language. It is a markup language much like HTML. HTML is designed to display data and to focus on how data looks, while XML is designed to describe data and focus on what data is.

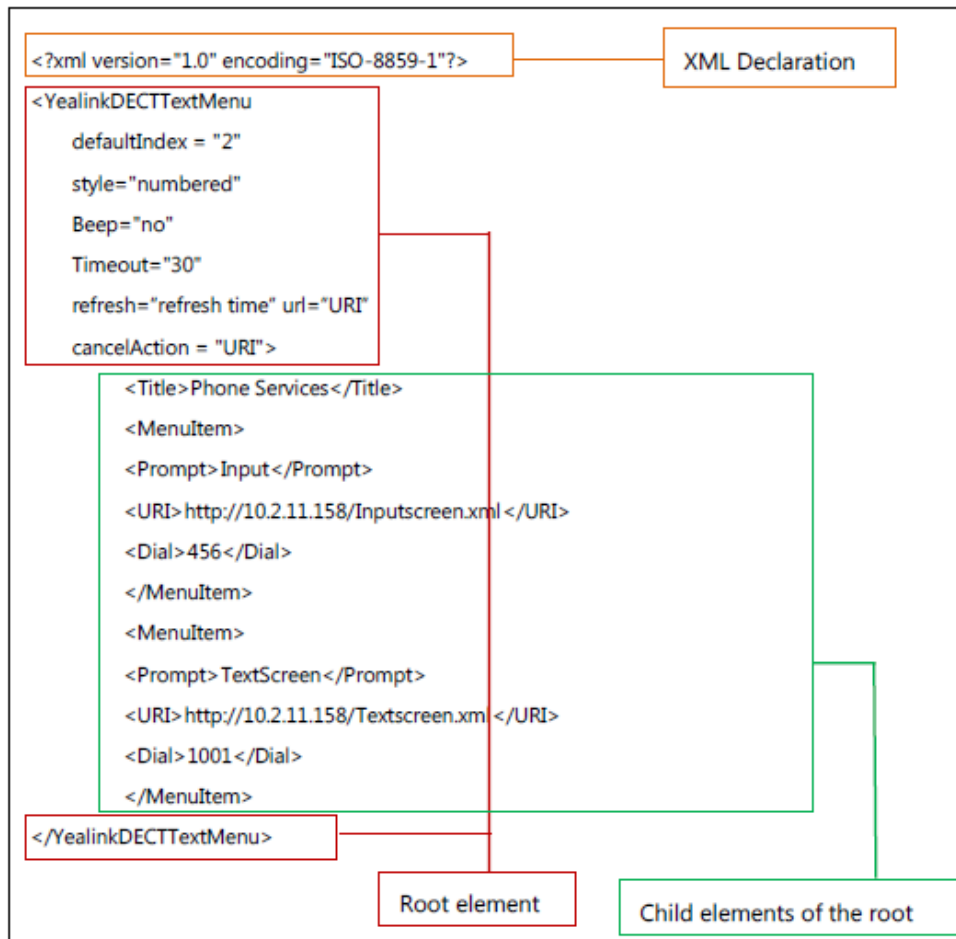
XML enables DECT phones to serve as output devices for many exciting applications. The XML infrastructure allows the phones to interact with external applications in a flexible and programmable manner.

The following are characteristics of XML:

- XML tags are not predefined. You must define your own tags.
- XML uses an XML schema to describe the data.

- XML with an XML schema is designed to be self-descriptive.
- XML is a W3C Standard Recommendation.

Sample of Basic XML document:



XML Format

XML is written in the form of XML elements consisting of tags enclosed in angle brackets (for example, `<YealinkDECTTextMenu>`). XML contains 3 kinds of tags: the start tag, the end tag and the empty-element tag. The empty-element tag has two kinds of form: in pairs (for example, `<MenuItem> </MenuItem>`) and not in pairs (for example, `<MenuItem/>`). With the exception of the empty-element tag not in pairs, XML tags most commonly come in pairs like `<YealinkDECTTextMenu>` and `</YealinkDECTTextMenu>`. The first tag in a pair is the start tag (for example, `<YealinkDECTTextMenu>`), and the second tag is the end tag (for example, `</YealinkDECTTextMenu>`).

XML mainly follows rules as below:

- XML must have root element.
- The end tag must have the character `"/`.
- XML tags are case-sensitive.
- Each attribute value should be within double quotations.
- The texts within `<!-- -->` are considered as comments.

XML provides escape facilities for including characters which are problematic to include directly. For example, the characters `"<` and `"&` are key syntax markers and may never appear in the content. XML has five predefined entities.

The XML conversion table is shown as below:

Character	Name	Escape Sequence
&	Ampersand	&
"	Quote	"
'	Apostrophe	'
<	Left angle bracket	<
>	Right angle bracket	>

To respect XML recommendations, the following header can be set at the beginning of the XML document,

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

Or

```
<?xml version="1.0" encoding="UTF-8"?>
```

Yealink provides XML object files beginning with the XML declaration "<?xml version="1.0" encoding="ISO-8859-1"?>".

For more information on XML, refer to <http://www.xml.com/>.

Functionality

The XML browser allows users to develop and deploy custom services which to meet the user requirements of phone functions on the server.

Phone service developers should take it into consideration that the phone is not a web browser so it cannot parse HTML. Although the content is delivered to the phone through HTTP messages using a web server, keep in mind that the content is not HTML. All content comes to the phone either as plain text or text packaged in XML objects.

DECT phones support 3 proprietary XML objects, which allow the creation of powerful XML applications.

The supported objects are:

- TextMenu object
- TextScreen object
- InputScreen object

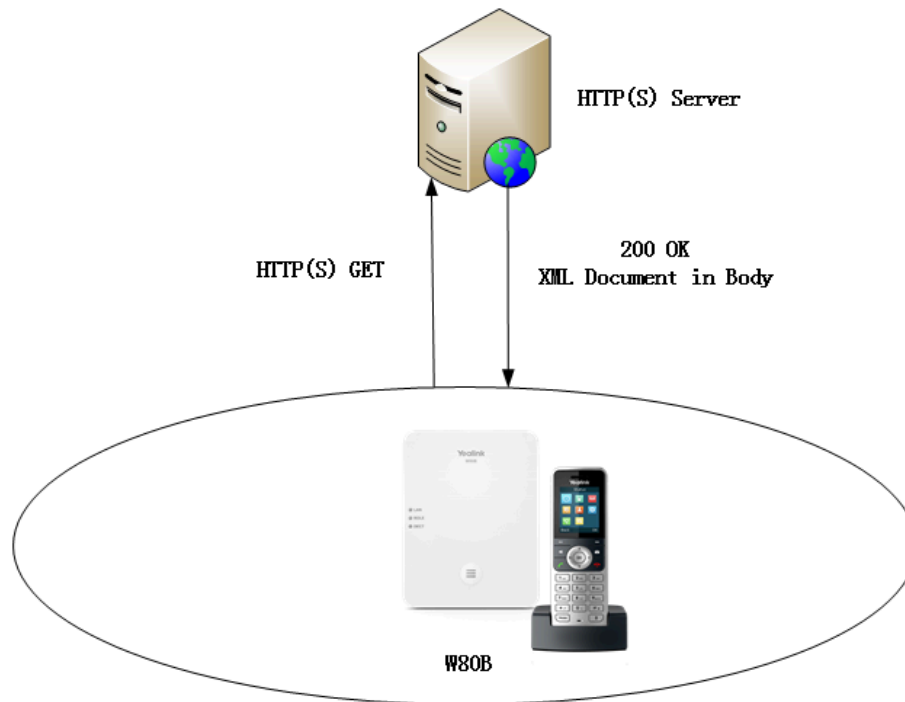
How does it work?

Depending on the IP infrastructure, Yealink is capable of developing the XML browser of the phones using HTTP. DECT phones support phone-initiated mode for XML browser applications.

Phone-initiated Application

You can press the predefined XML Browser key to trigger the phone-initiated application of XML browser. After you press the key, the DECT phone issues an HTTP(s) GET request message to the server, waits for the answer, decodes and displays this response message like any web browser, such as Microsoft Internet Explorer or Firefox, and would do as a web client.

Figure1 DECT phone acting as a client



Yealink DECT Phone XML Objects

Creating interactive service applications is relatively easy when you understand the XML objects that are defined for DECT phones and the behavior that each XML object generates.

Regardless of what causes the phone to load an XML page, the phone always behaves appropriately after it loads a page. Appropriate behavior depends only on the type of data delivered on the page.

XML Object Definitions

This section details each proprietary XML object supported by DECT phones.

You can ask the distributor or Yealink FAE for XML object files or obtain XML object files online: <http://support.yealink.com/documentFront/forwardToDocumentFrontDisplayPage>.



Note: XML objects do not support Chinese characters.

TextMenu Object

The TextMenu object allows users to create a list of menu items on the DECT phones. You can browse the menu items by linking HTTP requests.

XML description of the TextMenu object:

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<***TextMenu
defaultIndex = "integer"
style = "numbered/none"
Beep = "yes/no"
Timeout = "integer"
refresh="refresh time" url="URI"
cancelAction = "URI"
>
<Title >Menu Title</Title>
<MenuItem>
<Prompt>Prompt</Prompt>
<URI>URI</URI>
<Dial>Number to dial</Dial >
<Selection>Selection</Selection>
</MenuItem>
<!--Additional menu items may be added (up to 30) -->
<!--Additional Softkey items may be added (up to 6) -->
</***TextMenu>

```

The parameters of the TextMenu object are listed in the following table:

Parameter	Position	Type	Value	Description
TextMenu	Root tag	mandatory	none	The root element of the TextMenu object. Note: "" in the "***TextMenu" can be any string or an empty string.
defaultIndex	Root tag	optional	Integer	Position of the cursor. If the value is not specified or exceeds the number of menu items, the cursor will be positioned on the first menu item. The default value is 1.
style	Root tag	optional	"numbered" "none"	numbered (default): Add a digit before each menu item for index. none: No sign before each menu item.
Beep	Root tag	optional	"yes" "no"	Whether to play a tone when the XML object is opened. The default value is "yes".

Parameter	Position	Type	Value	Description
Timeout	Root tag	optional	integer	<p>If there is no operation at a fixed interval (in seconds) on the phone, the phone will automatically exit from the TextMenu screen.</p> <p>If it is set to 0, the phone will not automatically exit from the TextMenu screen until pressing the "Back" soft key/ On-hook key, or long pressing the On-hook key.</p> <p>The default value is 45.</p>
refresh	Root tag	optional	integer	<p>Define the time interval (in seconds) to automatically refresh the text menu by calling the URI defined by "url".</p> <p>If it is set to 0, the phone will not automatically refresh the text menu.</p>
url	Root tag	optional	URI	Define the URI to be called to refresh the text menu.
cancelAction	Root tag	optional	URI	Define the URI to be called when the user cancels the XML object by pressing the On-hook key.
Title	Body	mandatory	string	The title of the text menu.
MenuItem	Body	mandatory	none	The element of the menu item. (Up to 30 instances, the minimum is 1)
Prompt	MenuItem body	mandatory	string	The label of the menu item. Note: Only one line can be displayed.
URI	MenuItem body	mandatory	URI	URI is used if the user presses the "Select" soft key, "OK" key, or Right navigation key (if it is not customized) with the cursor on this menu item.
Dial	MenuItem body	optional	Phone number	Define what number will be dialed when the user presses the Speakerphone key or the Off-hook key.
Selection	MenuItem body	optional	string	<p>If "URI" is set to an HTTP URL, the "?selection= xxx (defined by the Selection parameter)" will be appended to the URI when the user presses the "Select" soft key or the "OK" key.</p> <p>Example: http://10.1.0.105/menu1.xml?selection=0&menu_pos=1</p>
SoftKey	Body	optional	string	Refer to Customizable Soft Keys for more information.

If there is no soft key defined in the TextMenu object, the LCD screen displays the following default soft keys:

SoftKey Index	Label	URI
1	Back	SoftKey:Back
2	Select	SoftKey:Select

The function keys and soft keys are listed in the following table:

Key Name	Operation	Function
Up/Down Key	Pressing the up/down key	Browse the menu item up and down.
Select	Pressing the "Select" soft key	Execute the content of the URI field assigned to the selected menu item.
Back/Exit	Pressing the "Back"/"Exit" soft key	Return to the idle screen.
Off-hook Key/ Speakerphone Key	Pressing the Speakerphone Key/ Off-hook key	If there is a number contained in the "Dial" tag, the phone will dial out the number. If there is no number contained in the "Dial" tag, the phone will not response to any operation.
Cancel Key	Pressing the On-hook key	If "cancelAction" is not left blank, the function of On-hook key is calling the URI defined by "cancelAction". If "cancelAction" is left blank, the function of On-hook key is returning to the idle screen. Note: Long pressing the On-hook key can also return to the idle screen.
OK Key	Pressing the "OK" key	The function of the "OK" key is the same as that of "Select".

An example of the TextMenu object:

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<YealinkDECTTextMenu
defaultIndex="2"
style="numbered"
Beep="no"
Timeout="30"
refresh="refresh time" url="http://10.2.11.158/XML/TextMenu.xml"
cancelAction = "http://10.2.11.158/test.xml"
>
<Title>Phone Services</Title>
<MenuItem>
<Prompt>Input</Prompt>
<URI>http://10.2.11.158/XML/InputScreen.xml</URI>
<Dial>456</Dial>
<Selection>12345</Selection>
</MenuItem>
<MenuItem>
<Prompt>TextScreen</Prompt>
<URI>http://10.2.11.158/XML/TextScreen.xml</URI>
<Dial>1001</Dial>
<Selection>4567</Selection>
</MenuItem>
</YealinkDECTTextMenu>

```

In the example, with the first menu item selected:

- Pressing the Speakerphone key/ Off-hook key, the phone will dial "456" using the available account.
- Pressing the **Select** soft key or **OK** key, the phone will call <http://10.2.11.158/XML/InputScreen.xml>.

TextScreen Object

The TextScreen object allows users to display some texts on DECT phones.

XML description of the TextScreen object:

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<**TextScreen
Beep = "yes/no"
doneAction = "URI"
Timeout = "integer"
refresh="refresh time" url="URI"
cancelAction = "URI"
>
<Title>Text Title</Title >
<Text>The screen text goes here</Text>
<!--Additional Softkey items may be added -->
</**TextScreen>

```

The parameters of the TextScreen object are listed in the following table:

Parameter	Position	Type	Value	Description
TextScreen	Root tag	mandatory	none	The root element of the TextScreen object. Note: "*" in the "TextScreen" can be any string or an empty string.
Beep	Root tag	optional	"yes" "no"	Whether to play a tone when the XML object is opened. The default value is "yes".
doneAction	Root tag	optional	URI	Define the URI to be called when the user presses the "OK" key.
Timeout	Root tag	optional	integer	If there is no operation at a fixed interval (in seconds) on the phone, the phone will automatically exit from the TextScreen screen. If it is set to 0, the phone will not exit from the TextScreen screen until pressing the "Exit" soft key. The default value is 45.
refresh	Root tag	optional	integer	Define the time interval (in seconds) to automatically refresh the screen text by calling the URI defined by "url". If it is set to 0, the phone will not automatically refresh the screen text.
url	Root tag	optional	URI	Define the URI to be called to refresh the screen text.

Parameter	Position	Type	Value	Description
cancelAction	Root tag	optional	URI	Define the URI to be called when the user cancels the XML object by pressing the On-hook key.
Title	Body	mandatory	string	The title of the text. Note: Only one line can be displayed.
Text	Body	mandatory	string	The content of the text. Note: Only one page can be displayed.
SoftKey	Body	optional	string	Refer to Customizable Soft Keys for more information.

If there is no soft key defined in the TextScreen object, the LCD screen displays the following default soft key:

SoftKey Index	Label	URI
1	Back	SoftKey:Back
2	OK	SoftKey:OK

The function keys and soft keys are listed in the following table:

Key Name	Operation	Function
Back/Exit	Pressing the "Back"/"Exit" soft key	Return to the idle screen.
Cancel Key	Pressing the On-hook key	If "cancelAction" is not left blank, the function of On-hook key is calling the URI defined by "cancelAction". If "cancelAction" is left blank, the function of On-hook key is returning to the idle screen. Note: Long pressing the On-hook key can also return to the idle screen.
OK Key	Pressing the "OK" key	The function of the "OK" key is calling the URI defined by "doneAction".

An example of the TextScreen object:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<YealinkDECTTextScreen
Beep = "yes"
doneAction = "http://10.2.11.158/test.xml"
Timeout = "15"
refresh="refresh time" url="http://10.2.11.158/TextScreen.xml"
cancelAction = "http://10.2.11.158/XML/InputScreen.xml"
>
<Title>Screen Title </Title>
<Text>The screen text goes here. When you see this, congratulations to you complete the
configuration.</Text>
</YealinkDECTTextScreen>
```

InputScreen Object

The InputScreen object allows users to create a screen capable of gathering user input. It constructs and displays an input form, which prompts the users to input content, then sends the input content to the target URL. You can use InputScreen object for user login or saving some information to the server. You can define the content and format of the input content.

XML description of the InputScreen object:

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<**InputScreen
type = "IP/string/number/timeUS/timeInt/dateUS/dateInt"
Beep = "yes/no"
Password = "yes/no"
Timeout = "integer"
refresh="refresh time" url="URI"
cancelAction= "URI"
defaultIndex = "integer from 1 to 7">
<Title>Title string</Title>
<URL>Target receiving the input</URL>
<InputField
type = "IP/string/number/timeUS/timeInt/dateUS/dateInt/empty"
password = "yes/no"
editable = "yes/no"
>
<Prompt>Guidance for the input</Prompt>
<Parameter> parameter name add to URL</Parameter>
<Selection>Selection</Selection>
<Default>Default Value</Default>
</InputField>
<!--Additional input field Items may be added (up to 7) -->
<!--Additional Softkey items may be added (up to 6)-->
</**InputScreen >

```

The parameters of the InputScreen object are listed in the following table:

Parameter	Position	Type	Value	Description
InputScreen	Root tag	mandatory	none	The root element of the InputScreen object. Note: "*" in the "InputScreen" can be any string or an empty string.

Parameter	Position	Type	Value	Description
type	Root tag	mandatory	"IP" "string" "number" "timeUS" "timeInt" "dateUS" "dateInt" "empty"	Specifies the type of input. Data input options: <ol style="list-style-type: none"> 1. IP 2. string(default) 3. number 4. timeUS, 12-hour format Format: HH:MM:SS AM/PM HH:1-12, MM:0-59, SS:0-59 AM/PM stand for the forenoon/afternoon. Example: 02:00:23 AM 12:59:00 PM 5. timeInt, 24-hour format Format: HH:MM:SS HH:0-23, MM:0-59, SS:0-59 Example: 23:25:00 6. dateUS Format: MM/DD/YYYY MM:1-12,DD:1-31,YYYY:0000-9999 Example: 12/31/2009 7. dateInt Format: DD/MM/YYYY DD:1-31,MM:1-12,YYYY:0000-9999 Example: 31/01/2010 8. empty
Beep	Root tag	optional	"yes" "no"	Whether to play a tone when the XML object is opened. The default value is "yes".

Parameter	Position	Type	Value	Description
Password	Root tag	optional	"yes" "no"	Whether to mask the input by the "*" character . The default value is "no".
Timeout	Root tag	optional	integer	If there is no operation at a fixed interval (in seconds) on the phone, the phone will automatically exit from the InputScreen screen. If it is set to 0, the phone will not exit from the InputScreen screen until pressing the "Exit" soft key. The default value is 45.
refresh	Root tag	optional	integer	Define the time interval (in seconds) to automatically refresh the input screen by calling the URI defined by "url". If it is set to 0, the phone will not automatically refresh the input screen.
url	Root tag	optional	URI	Define the URI to be called to refresh the input screen.
cancelAction	Root tag	optional	URI	Define the URI to be called when the user cancels the XML object by pressing the On-hook key.
defaultIndex	Root tag	optional	integer	Position of the cursor. If the value is not specified or exceeds the number of input boxes, the cursor is positioned on the first input box. The default value is 1.
Title	Body	mandatory	string	The title of the text.
URL	Body	mandatory	URL	Specify the target URL to receive the user input.
InputField	Body	optional	none	Set several input boxes.

Parameter	Position	Type	Value	Description
type	InputField tag	optional	"IP" "string" "number" "timeUS" "timeInt" "dateUS" "dateInt" "empty"	Specifies the type of input. Data input options: <ol style="list-style-type: none"> 1. IP 2. string(default) 3. number 4. timeUS, 12-hour format Format: HH:MM:SS AM/PM HH:1-12, MM:0-59, SS:0-59 AM/PM stand for the forenoon/afternoon. Example: 02:00:23 AM 12:59:00 PM 5. timeInt, 24-hour format Format: HH:MM:SS HH:0-23, MM:0-59, SS:0-59 Example: 23:25:00 6. dateUS Format: MM/DD/YYYY MM:1-12,DD:1-31,YYYY:0000-9999 Example: 12/31/2009 7. dateInt Format: DD/MM/YYYY DD:1-31,MM:1-12,YYYY:0000-9999 Example: 31/01/2010 8. empty
password	InputField tag	optional	"yes" "no"	Whether to mask the input by the "*". The default value is "no".

Parameter	Position	Type	Value	Description
editable	InputField tag	optional	"yes" "no"	Whether to allow users to input something. The default value is "yes". Users can not input anything and modify the default input if it is set to "no". Applicable scenario: only allow some users to login.
Prompt	InputField body	optional	string	The prompt of user input.
Parameter	InputField body	mandatory	string	Name of the parameter to be appended to the URL. The "?parameter name= the user input" will be appended to the URL when the user presses the "Select" soft key or the "OK" key.
Selection	InputField body	optional	string	The "selection= the value of the Selection parameter" will also be appended to the URL when the user presses the "Select" soft key or the "OK" key.
Default	InputField body	optional	string	Default value to be displayed in the input field.
SoftKey	Body	optional	string	The soft keys displayed will be changed according to the attribute value of type. Refer to Customizable Soft Keys for more information.



Note: The InputField parameter in the XML file is optional. You can use this parameter to customize more input fields on the DECT phone.


If there is no soft key defined in the InputScreen object, the LCD screen displays the following default soft keys:

SoftKey Index	Label	URI
1	Back	SoftKey:Back
2	Submit	SoftKey:Submit

The function keys and soft keys are listed in the following table:

Key Name	Operation	Function
Up/Down Key	Pressing the up/down key	Browse the input box up and down.
Left/Right Key	Pressing the left/right key	Move the cursor left and right.

Key Name	Operation	Function
Keypad	Pressing the digit keys 1~9, *, and # keys	If the value of the "editable" is "yes", then input character; otherwise no response.
BackSpace	Pressing the "BackSpace" soft key	Delete the character before the cursor in the input box.
Dot (.)	Pressing the "Dot" soft key	Input a "." in the input box at the cursor position.
Submit	Pressing the "Submit" soft key	Execute the command comprised of the URI and input content.
Back/Exit	Pressing the "Back"/"Exit" soft key	Return to the idle screen.
NextSpace	Pressing the "NextSpace" soft key	Input a space in the input box at the cursor position.
Cancel Key	Pressing the On-hook key	If "cancelAction" is not left blank, the function of On-hook key is calling the URI defined by "cancelAction". If "cancelAction" is left blank, the function of On-hook key is returning to the idle screen. Note: Long pressing the On-hook key can also return to the idle screen.
OK Key	Pressing the "OK" key	The function of the "OK" key is the same as that of "Select".

 **Note:** You can press the # key to switch the input modes among "Abc", "123", "ABC" or "abc".

An example of the InputScreen object:

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<YealinkDECTInputScreen
type="string"
Beep="yes"
Timeout="15"
LockIn="no">
<Title wrap="yes">Proxy Server</Title>
<URL>http://10.3.5.5/XML/input.xml</URL>
<InputField>
<Prompt>User Name:</Prompt>
<Parameter>user</Parameter>
<Default></Default>
<Selection>1</Selection>
</InputField>
<InputField>
<Prompt>Password:</Prompt>
<Parameter>password</Parameter>
<Default></Default>
<Selection>2</Selection>
</InputField>
</YealinkDECTInputScreen>

```

In this example, when the user presses the **Submit** soft key or **OK** key on the phone after entering "admin" as the User Name and "222" as the password, the phone will call the following URLs:

- <http://10.3.5.5/XML/input.xml?password=222&user=admin&selection=1>, if the position of the cursor is located in the **User Name** input box.
- <http://10.3.5.5/XML/input.xml?password=222&user=admin&selection=2>, if the position of the cursor is located in the **Password** input box.

Customizable Soft Keys


DECT phones allow users to create soft keys with customizable labels, positions, and actions to be taken when the soft keys are pressed. The customizable soft keys can override the default soft keys in each XML object.

XML descriptions of customizable soft keys:

```

<SoftKey index = "1-6">
<Label>Text</Label>
<URI>http://someserver/somepage OR SoftKey:someaction</URI>
</SoftKey>

```

 **Note:** If you use the customizable soft keys, the default soft keys of the XML object will not be displayed anymore. This means they have to be recreated as customizable soft keys.


The URI of the custom soft key is case sensitive.

The parameters of the soft key are listed in the following table:

Parameter	Type	Value	Description
SoftKey	mandatory	none	The soft key.
Index	mandatory	Integer	Indicate the soft key number. (Value ranges from 1~6.)
Label	mandatory	String	The label of the soft key.
URI	mandatory	String	The action of the soft key.

The supported actions for each UI XML object are described in the following table:

Label	URI	Function
TextMenu Object		
Exit/Back	SoftKey:Exit/ SoftKey:Back	Return to the idle screen.
Select	SoftKey:Select	Execute the URI defined by "Selection".
Send	SoftKey:Dial	Dial out the number of the highlighted menu item.
TextScreen Object		
Exit/Back	SoftKey:Exit/ SoftKey:Back	Return to the idle screen.
InputScreen Object		
Exit/Back	SoftKey:Exit/ SoftKey:Back	Return to the idle screen.
Submit	SoftKey:Submit	Execute the command comprised of the URI and input content.
Dot (.)	SoftKey:Dot	Input a "." in the input box at the cursor position.
BackSpace	SoftKey:BackSpace	Delete the character before the cursor in the input box.
NextSpace	SoftKey:NextSpace	Insert a space in the input box at the cursor position.

 **Note:** If you configure more than two soft keys, **Options** is displayed. You may need to select **Options** first and then select the corresponding soft key.

An example of the customizable soft keys used with the TextMenu object:

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<YealinkDECTTextMenu
defaultIndex="2"
style="numbered"
Beep="no"
Timeout="30"
refresh="refresh time" url="http://10.2.11.158/XML/TextMenu.xml"
cancelAction = "http://10.2.11.158/test.xml"
>
<Title>Phone Services</Title>
<MenuItem>
<Prompt>Input</Prompt>
<URI>http://10.2.11.158/XML/InputScreen.xml</URI>
<Dial>456</Dial>
<Selection>12345</Selection>
</MenuItem>
<MenuItem>
<Prompt>TextScreen</Prompt>
<URI>http://10.2.11.158/XML/TextScreen.xml</URI>
<Dial>1001</Dial>
<Selection>4567</Selection>
</MenuItem>
<SoftKey index="1">
<Label>Select</Label>
<URI>SoftKey:Select</URI>
</SoftKey>
<SoftKey index="2">
<Label>Custom</Label>
<URI>http://10.1.0.105/8.8.8.54.rom</URI>
</SoftKey>
</YealinkDECTTextMenu>

```

Some Development Guidelines

There are some simple rules that you had better follow when you develop XML applications for DECT phones.


- Place custom soft keys as they are for the standard objects, also it is better to use the same labels.

- If you want to access data from the Internet, it is preferable to use an RSS feed or a SOAP interface than Web scraping as Web sites frequently change their layout interface.

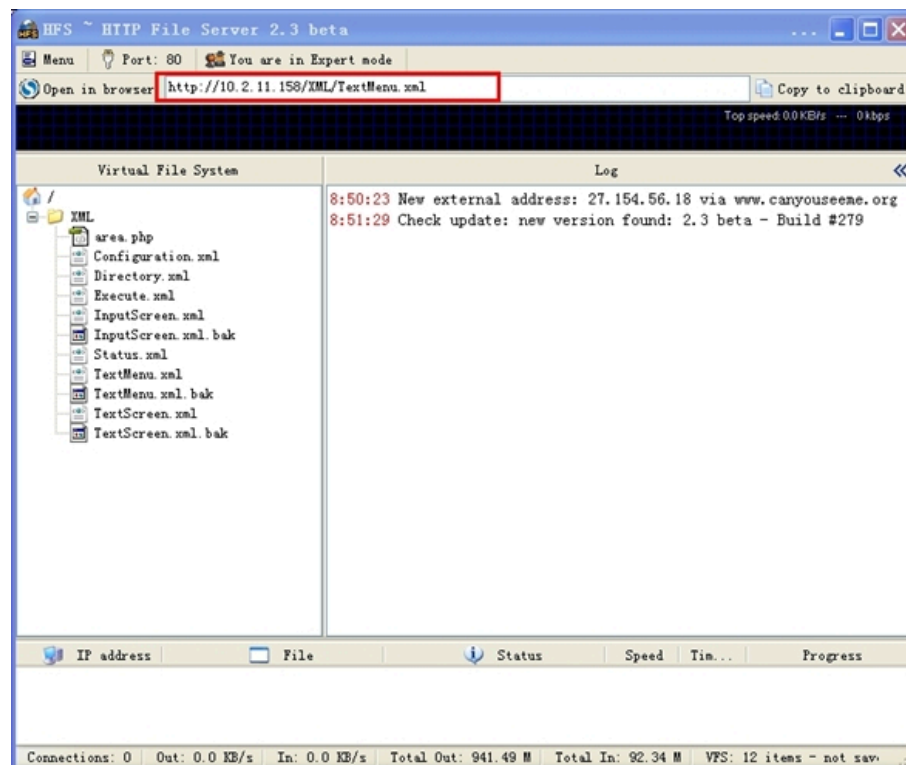
Configuring the HTTP Server

DECT phones use the HTTP (HTTPS) protocol for downloading. You can set up the HTTP(s) server, and place some XML files on the server for downloading. This section provides you with some instructions on how to configure the HTTP server and obtain the access URL of the XML files downloaded by the DECT phones.

Procedure

1. Double click the HFS.exe.
2. Click **Menu** on the main page and select the IP address of the PC from **IP address**.
The default HTTP port is 80. You can also reset the HTTP port (make sure the port isn't in use before reset).
3. Right click the  icon on the left of the main page, select **Add folder from disk** to add the HTTP Server root directory.
4. Locate the root directory from your local computer. Select your desired folder.
5. Select one of the XML files, then the access URL of the selected XML file is displayed in the address bar.

The screenshot for reference is shown as below:



Configuring an XML Browser Key

To use the XML browser feature, you must configure an XML key via configuration files in advance.

Procedure

1. Add/Edit XML Browser key parameters in the configuration file (for example, y000000000103.cfg).

The following table shows the information of parameters:

Parameters	Permitted Values	Default
custom.handset.defined_left_key.type	25	0
<p>Description: It configures the role of the Left Softkey on the idle screen. The number 25 stands for the key type as XML Browser.</p>		
custom.handset.defined_left_key.xml_url	String	Blank
<p>It configures the available access URL for the Left Softkey to browse the XML object. Note: It works only if "custom.handset.defined_left_key.type" is set to 25 (XML Browser).</p>		
custom.handset.defined_right_key.type	25	0
<p>Description: It configures the role of the Right Softkey on the idle screen. The number 25 stands for the key type as XML Browser.</p>		
custom.handset.defined_right_key.xml_url	String	Blank
<p>It configures the available access URL for the Right Softkey to browse the XML object. Note: It works only if "custom.handset.defined_right_key.type" is set to 25 (XML Browser).</p>		
custom.handset.defined_direction_left_key.type	25	0
<p>Description: It configures the role of the left navigation key on the idle screen. The number 25 stands for the key type as XML Browser.</p>		
custom.handset.defined_direction_left_key.xml_url	String	Blank
<p>It configures the available access URL for the left navigation key to browse the XML object. Note: It works only if "custom.handset.defined_direction_left_key.type" is set to 25 (XML Browser).</p>		
custom.handset.defined_direction_right_key.type	25	0
<p>Description: It configures the role of the right navigation key on the idle screen. The number 25 stands for the key type as XML Browser.</p>		
custom.handset.defined_right_key.xml_url	String	Blank
<p>It configures the available access URL for the right navigation key to browse the XML object. Note: It works only if "custom.handset.defined_direction_right_key.type" is set to 25 (XML Browser).</p>		

Parameters	Permitted Values	Default
custom.handset.defined_direction_up_key.type	25	0
<p>Description: It configures the role of the up navigation key on the idle screen. The number 25 stands for the key type as XML Browser.</p>		
custom.handset.defined_direction_up_key.xml_url	String	Blank
<p>It configures the available access URL for the up navigation key to browse the XML object. Note: It works only if "custom.handset.defined_direction_up_key.type" is set to 25 (XML Browser).</p>		
custom.handset.defined_direction_down_key.type	25	0
<p>Description: It configures the role of the down navigation key on the idle screen. The number 25 stands for the key type as XML Browser.</p>		
custom.handset.defined_direction_down_key.xml_url	String	Blank
<p>It configures the available access URL for the down navigation key to browse the XML object. Note: It works only if "custom.handset.defined_direction_down_key.type" is set to 25 (XML Browser).</p>		

The following shows an example of configuring an XML Browser key in the configuration file:

```
custom.handset.defined_right_key.type = 25
```

```
custom.handset.defined_right_key.xml_url = http://10.2.1.158/TextMenu.xml
```

2. Reference the configuration file in the boot file (for example, y000000000000.boot).

Example:

```
include:config "http://10.2.1.158/HTTP Directory/y000000000103.cfg"
```

3. Upload the boot file and configuration file to the root directory on the provisioning server.
4. Trigger DECT phone to perform an auto provisioning for a configuration update.

For more information on auto provisioning, refer to the latest Auto Provisioning Guide on [Yealink Technical Support](#).

Troubleshooting

This chapter provides general troubleshooting information to help to solve the problems you might encounter when developing XML applications for DECT phones. Besides, it also provides related troubleshooting tools and parsing error debug example for you to trace errors. If you require additional information or assistance, contact your system administrator.

Why does the phone display "Loading error!" when executing the XML object?

- Ensure that the network is reachable.
- Ensure that the server is running properly.
- Ensure that the HTTP server application is running properly.
- Ensure that the URI is resolved.

- Ensure that the XML object file exists on the server.
- Ensure that the URI is parsed.

Why does the phone display "File layout error!" when executing the XML object?

- Check whether the XML object files' syntax is right. For example, XML tags are in pairs.

Troubleshooting Tools

The following tools will help you troubleshoot problems with the XML services.

- Standard web browser (Microsoft Internet Explorer 6.0 or a later)
 - Verify the connectivity.
 - Verify the validity of the URI called by the phone.
- Network packet analyzer such as Wireshark
 - Verify what is exchanged between the phone and the server.
- HTTP Server log
 - Verify if the HTTP GET reaches the server.
 - Verify the parameters of the HTTP GET.
- Phone log (syslog)
 - Verify how the phone processes an XML request.

Parsing Error Debug Example

Some of the XML editors (such as UltraEdit and Dreamweaver) can also verify the XML syntax and detail the related error in the output window. The following takes the Dreamweaver tool as an example for reference.

After you execute the XML object, the LCD screen will prompt "File layout error !".

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<YealinkDECTTextMenu
defaultIndex="2"
style="numbered"
Beep="no"
Timeout="30"
refresh="refresh time" url="http://10.2.11.158/XML/TextMenu.xml"
cancelAction = "http://10.2.11.158/test.xml"
>
<Title>Phone Services</Title>
<MenuItem>
<Prompt>Input</Prompt>
<URI>http://10.2.11.158/XML/InputScreen.xml</URI>
<Dial>456</Dial>
<Selection>12345</Selection>
</MenuItem>
<MenuItem>
<Prompt>TextScreen</Prompt>
<URI>http://10.2.11.158/XML/TextScreen.xml</URI>
<Dial>1001</Dial>
<Selection>4567</Selection>
</MenuItem>
<YealinkDECTTextMenu>

```

The Dreamweaver tool tells us where the problem is. The error is on the last line, it should be </YealinkDECTTextMenu>.

```

! H:\firmware\D... 23 The input ended before all started tags were ended. Last tag started was 'YealinkDECTTextMenu' [xml]

```