tenHsServer Documentation

Version 2.6

tenHsServer is a single web page that processes requests specified in the URL, and displays a page (returns html that contains the result of request). Typically, an application running on a PC remote from HomeSeer, will request the *tenHsServer* page and parse the returned html to get the results of the request.

If the IP address of your server is **2.3.4.5** and you are running the HomeSeer web server on port 67, then your URL should be:

http://2.3.4.5:67/tenHsServer/tenHsServer.aspx?t=xx& followed by your parameters.

For example, to get the status of device A1, the URL would look like:

http://2.3.4.5:67/tenHsServer/tenHsServer.aspx?t=ab&f=DeviceStatus&d=A1

Note: the t= parameter can be anything, but it MUST be there. One suggestion is to change it each time, generally as a random number, to make sure that your request is not cached by the browser.

The f= parameter specifies the nature (function) of the request. The following functions have been implemented so far (not case sensitive):

DeviceStatus	Returns the status of one or more devices
GetDevice	Returns device information
ExecX10	Performs the Homeseer ExecX10 function
ToggleDevice	Toggles the devices ON/OFF
GetDevices	Returns, for all devices: DeviceCode,DeviceType,Location,Name
GetEvents	Returns EventNames for Events in the specified Group
RunEvent	Runs the Event specified by Name
Uptime	Returns Homeseer SystemUptime
GetIni	Returns an entry from an .ini file
SetIni	Sets an entry in an .ini file
Check	Check for tenHsServer Communications OK
GetValues	Returns the Value Strings for a Device
SetDeviceValue	Changes a Device’s Value

LoggedInUser	Returns the logged in HS web user
ValidateUser	Verifies if Userid and Password is valid
GetUsers	Returns list of HS web users and their access rights
DeviceOn	Turns one or more Devices ON
DeviceOff	Turns one or more Devices OFF
SetDeviceString	Sets Device String for a Device

Optionally, *tenHsServer* can require that a Userid and Password be provided on the URL via the **&u** and **&p** parameters. Security is controlled via parameters specified in the following file on the Homeseer server: **c:\Program Files\Homeseer HS2\Config\tenHsServer.ini**

The contents of this file look like this:

```
[Security]
Mode=FullAccess
```

The Mode parameter specifies that the userid in the URL must have a certain level of access for the request to succeed. The valid values for the Mode parameter are:

Mode=FullAccess	No userid/password required on the URL
Mode=Guest	The URL must contain a Userid/Password with at least Guest access
Mode=Normal	The URL must contain a Userid/Password with at least Normal access
Mode=Admin	The URL must contain a Userid/Password with Admin access

If the **tenHsServer.ini** file does not exist, it will be created with Mode=FullAccess

The Userid and Password specified in the URL must be defined to Homeseer via **Setup/Web Users**

These functions use the following URL parameters:

DeviceStatus	f=DeviceStatus	literal "DeviceStatus"
	d=A1.p3.k14	Device Codes separated by ".”
	Example: ?t=97&f=DeviceStatus&d=p2	
	Returns: Homeseer numerical device status for each device in the form: d1:s1;d2:s2;d3:s3; were d is the device code and s is the status	
DeviceValue	f=DeviceValue	literal "DeviceValue"
	d=A1.p3.k14	Device Codes separated by "."
	Example: ?t=33&f=DeviceValue&d=p2	
	Returns: Homeseer numerical device value for each device in the form: d1:v1;d2:v2;d3:v3; were d is the device code and v is the value	
GetValues	f=GetValues	
	d=a1.p3.]6	Device Codes
	Example: ?t=77&f=GetValues&d=b3.d2.^4	
	Returns: Homeseer Value Strings in the form: d1:vs1;d2:vs2 where d is the device code and v is the Value String	
RunEvent	f=RunEvent	
	d=EventName	
	Example: ?t=12&f=RunEvent&d=LeavingHouse	
Check	f=Check	
	Example: ?t=12&f=Check	
	Returns: tenHsServer OK	

DeviceString	f=DeviceString	literal "DeviceString"
	d=A1.p3.k14	Device Codes separated by "."

Example: ?t=33&f=DeviceString&d=p2

Returns: The Device String from the specified devices
in the form: d1:s1;d2:s2;d3:s3;
were d is the device code and s is the string

Note: Because the DeviceString result can be html, the result is encoded. All special characters are replaced with an escape sequence consisting of “% followed by the 2-digit hex character code. For example, “<abc>” will be encoded as:

%3Cabc%3E%0D%0A

You can easily decode this using your languages built-in functions. Javascript, for instance, provides the string.unescape built-in function.

SetDeviceValue	f=SetDeviceV alue	
	d=a1	Device Code
	a=75	New Value

Example: ?t=90&f=SetDeviceValue&d=p7&a=100

GetDevice

f=GetDevice	literal "GetDevice"
d=A1.p3.k14	Device Codes separated by "."

Example: ?t=72&f=GetDevice&d=p2

Returns: Current Device fields in the form:
d:loc:name:type:misc:loc2:dim:status:value:string:time:LastChange
where,
d=device code
loc=location
name=device name
type=device type
misc=misc binary field
loc2=location 2
dim=can dim
status=device status
value=device value
string= encoded DeviceString (see DeviceString)
time=device time (in minutes)
LastChange=device LastChange date/time

ExecX10

f=ExecX10	literal "ExecX10"
d=a1	Device Code
a=Off	Device Action
s=15	Parm1
n=10	Parm2

Example: ?t=34&f=ExecX10&d=c13&a=Dim&s=50

Returns: DeviceStatus=nn
where, nn=Device Status after Device Action

GetDevices

f=GetDevices	literal "GetDevices"

Example: ?t=99&f=GetDevices

Returns: multiple device strings, one for each device, separated by ";"
Each device string is formatted as:
DeviceCode:DeviceType:Location:Name:Location2

An example if only two devices were defined in Homeseer:
a1:Lamp Module:Kitchen:Counter Pucks:loc2;a2:Applicance Module:Den:Ceiling Light:Loc2;

GetEvents

f=GetEvents	
d=group	group name of Events to retrieve

Example: ?t=4&f=GetEvents&d=default

Returns: Group1;Group2;Groupn;

Uptime

f=Uptime

Example: ?t=41&f=Uptime

Returns: SystemUptime, for example:
0 Days 16 Hours 22 Minutes 50 Seconds

GetIni

f=GetIni	literal "GetIni"
s=MySection	section name
n=MyFile	file name
d=MyKey	key name

Example: ?t=23&f=GetIni&s=Settings&d=WindowSize&n=MyApp.ini

Returns: Value of section/key in the specified file. File must be in Homeseer\config\

SetIni

f=SetIni	literal "SetIni"
s=MySection	section name
n=MyFile	file name
d=MyKey	key name
a=NewValue	value to be stored

Example:
?t=63&f=SetIni&s=Settings&d=WindowSize&a=large&n=MyApp.ini

ToggleDevice

f=ToggleDevice	literal “ToggleDevice”
d=a5.p3.m1	Device Codes separated by “.”

Example: ?t=22&f=ToggleDevice&d=a1.a2

Returns: New DeviceStatus as a1:2;a2:3
Note: If a DeviceStatus was not ON or OFF, then “?” is returned

LoggedInUser

f=LoggedInUser

Example: ?t=1213&f=LoggedInUser
Returns: The logged in Web User

GetUsers

f=GetUsers

Example: ?t=9375&f=GetUsers
Returns: guest:1;user2:6;user3:4;
1=Guest 2=Admin 8=Normal 4=Local   (can be ORed together)

ValidateUser

f=ValidateUser

d=user01	a Userid
a=mypassword	a Password

Example: ?t=4375&f=ValidateUser&d=user03&a=mypass
Returns: True if good userid/password   False if not

DeviceOn

f=DeviceOn	literal “**DeviceOn**”
d=]4.[56.A1	Device codes separated with “**.**”

Example: ?t=123&f=DeviceOn&d=A1.c3.[45
Returns: Resultant Device Status for each Device
A1:s;c3:s;[45:s;
where s is the resultant device status

DeviceOff

f=DeviceOff	literal “**DeviceOff**”
d=]4.[56.A1	Device codes separated with “**.**”

Example: ?t=123&f=DeviceOff&d=A1.c3.[45
Returns: Resultant Device Status for each Device
A1:s;c3:s;[45:s;
where s is the resultant device status

SetDeviceString

f=**SetDeviceString**	literal “**SetDeviceString**”
d=A1	Device
a=new device string	String to store into the Device

The **tenHsServer.aspx** page processes the URL parameters, performs the requested actions, and returns a response formatted as an HTML page. The returned values are embedded in the returned HTML, within a span element as:

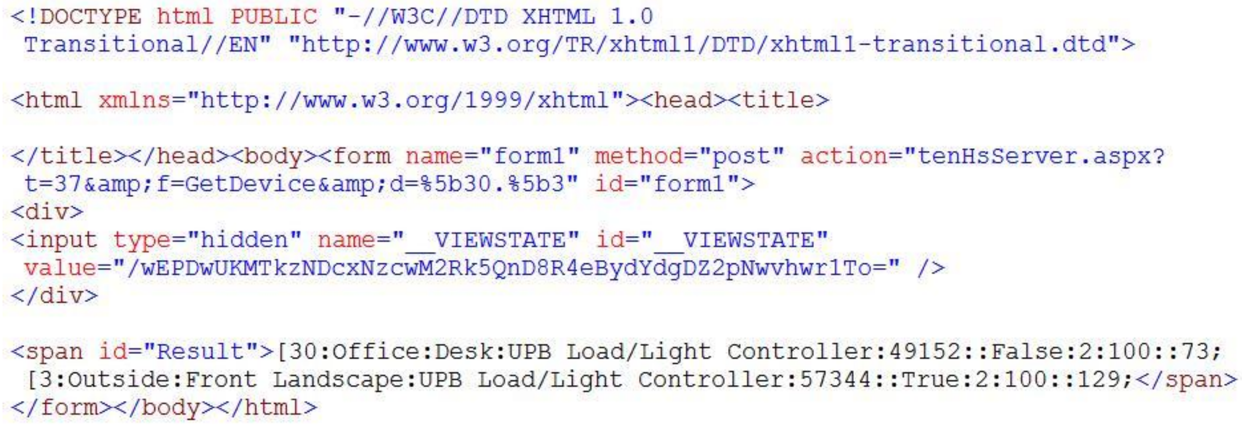
<span id="Result">**2**</span>

Where **2** is the value returned from *tenHsServer*.

The following URL:

http://homeseer:81/tenHsServer/tenHsServer.aspx?t=37&f=GetDevice&d=[30.[3

will return HTML containing something like the following:



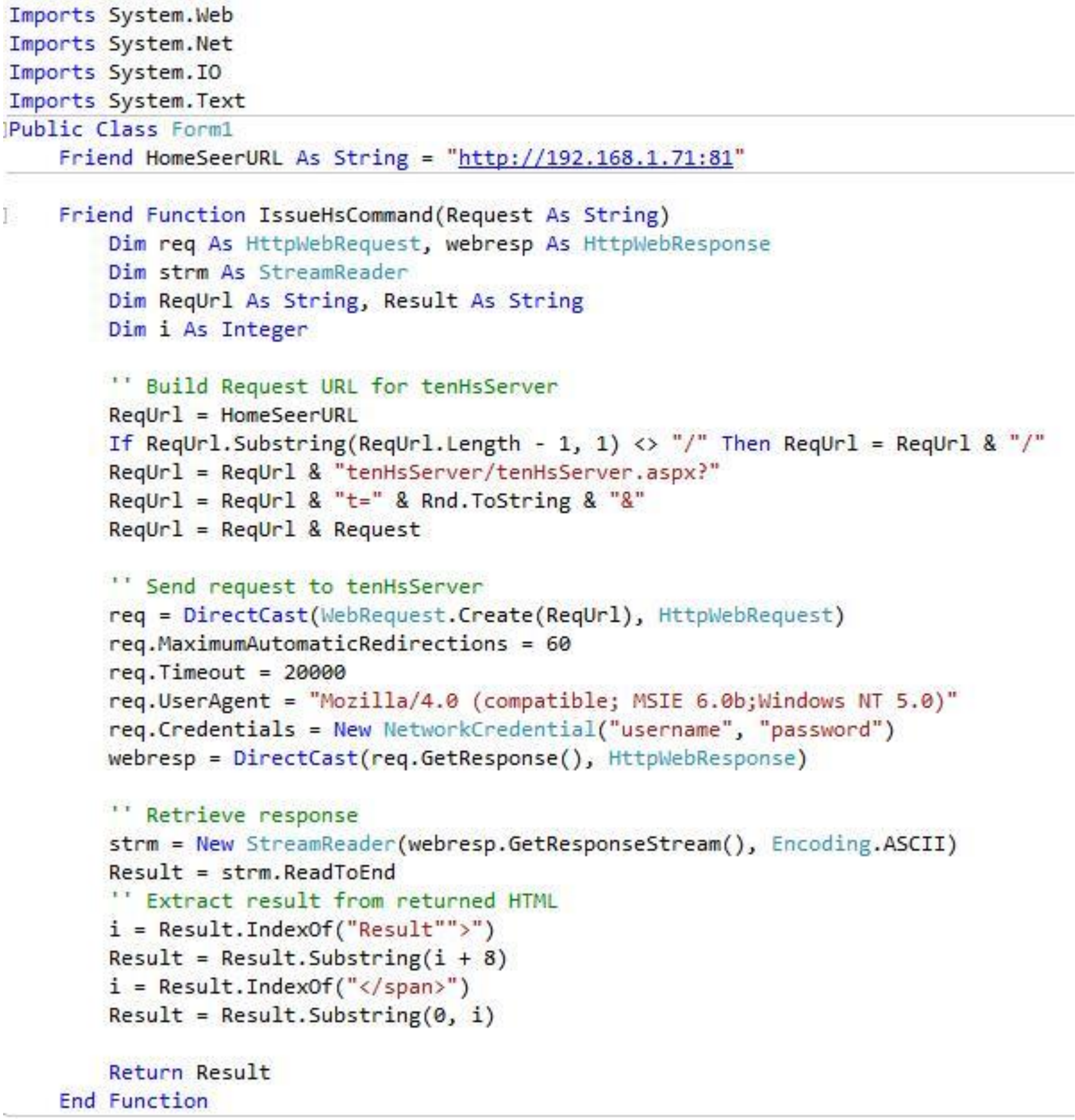
```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"><head><title>

</title></head><body><form name="form1" method="post" action="tenHsServer.aspx?
 t=37&amp;f=GetDevice&amp;d=%5b30.%5b3" id="form1">
<div>
<input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE"
 value="/wEPDwUKMTkzNDcxNzcwM2Rk5QnD8R4eBydYdgDZ2pNwvhwr1To=" />
</div>

<span id="Result">[30:Office:Desk:UPB Load/Light Controller:49152::False:2:100::73;
 [3:Outside:Front Landscape:UPB Load/Light Controller:57344::True:2:100::129;</span>
</form></body></html>
```

Here is a VB.NET function that will send a request to *tenHsServer* and parse the returned response:



```
Imports System.Web
Imports System.Net
Imports System.IO
Imports System.Text
Public Class Form1
    Friend HomeSeerURL As String = "http://192.168.1.71:81"

    Friend Function IssueHsCommand(Request As String)
        Dim req As HttpWebRequest, webresp As HttpWebResponse
        Dim strm As StreamReader
        Dim ReqUrl As String, Result As String
        Dim i As Integer

        '' Build Request URL for tenHsServer
        ReqUrl = HomeSeerURL
        If ReqUrl.Substring(ReqUrl.Length - 1, 1) <> "/" Then ReqUrl = ReqUrl & "/"
        ReqUrl = ReqUrl & "tenHsServer/tenHsServer.aspx?"
        ReqUrl = ReqUrl & "t=" & Rnd.ToString & "&"
        ReqUrl = ReqUrl & Request

        '' Send request to tenHsServer
        req = DirectCast(WebRequest.Create(ReqUrl), HttpWebRequest)
        req.MaximumAutomaticRedirections = 60
        req.Timeout = 20000
        req.UserAgent = "Mozilla/4.0 (compatible; MSIE 6.0b;Windows NT 5.0)"
        req.Credentials = New NetworkCredential("username", "password")
        webresp = DirectCast(req.GetResponse(), HttpWebResponse)

        '' Retrieve response
        strm = New StreamReader(webresp.GetResponseStream(), Encoding.ASCII)
        Result = strm.ReadToEnd
        '' Extract result from returned HTML
        i = Result.IndexOf("Result"">")
        Result = Result.Substring(i + 8)
        i = Result.IndexOf("</span>")
        Result = Result.Substring(0, i)

        Return Result
    End Function
```

Ed Tenholder
ed@tenholder.net