

Micasaverde /GetVera is producing a product family of zWave controllers called Vera 3, Vera Lite and Vera Edge now. These product come with a user interface layer called UI and which exists in 2 versions as we speak : UI5 and UI7.

Unfortunately the long awaited UI7 has been kind of a disappointment , especially with its unresponsiveness , not really well optimized screen real estate, and difficult to deal with for 3rd party plug in writers. It was also promising a mobile user interface and the application does not resize well on phone or on desktops and the mobile version of the application is not user friendly.

I started to work on a UI replacement.

This document covers:

1. The overall project objectives & “big” rules
2. The screen shots
3. The installation instructions
4. Some internal explanation of the source code & architecture

Table of Contents

The project initial objectives.....	2
Objectives	2
DONE and functional so far.....	3
Screen shots:.....	5
UI5 Installation Instructions (similar for UI7).....	32
Architecture and Source Code organization	48
Extensibility	49
Mechanisms to extend.....	49
Javascript modules for customizable plugins	49
Full Source code	50
Source Files:	51
Basic rules for developers:.....	53

The project initial objectives

I am not fully satisfied with UI5 or UI7 and I think we can provide very quick improvement. French users of Orange HomeLive system on internet seems to be Highly largely unsatisfied by the UI and we could improve this relatively easily adopting a refreshed approach & architecture. (remains to be seen if orange is going to be open to this but we should try)

Objectives

1. **Fast & immediately responsive** (except LUA Jobs of course, cant control that). Asynchronous / threaded programming as much as possible.
2. Avoid the classic UIx issues with too many **heterogeneous js frameworks**, inconsistent CSS rules requiring ticks all over the place and overuse of the “! Important”
3. Does **not require anything** other infrastructure than the **VERA** itself and a simple plugin. No PHP, No DB, No additional server running. Just plain VERA
4. Really use the **power of the client side** machine (big processors & memory) and far less the Vera side.
5. Works well on all screen size, **full responsive design** using bootstrap
6. Really use [bootstrap](#) facilities for responsiveness, should work on Phone 4S as well as tablets, as well as desktop / large screens. Same app, same code, same access url
7. Dashboards should be optimized in screen real estate. **Undo the UI5 design decision** which links the Scene editor with the dashboard. You can only put in scene what is in the dashboard (unless you use the advanced feature). Dashboard requires maximum use of the screen real estate, Scene editor is something else.
8. Plugin authors should be able to control the display of their device using a **simple javascript functions** , even on the dashboard page. Should not be limited to a VERA Box api or any complex undocumented json behaviors. Just a dynamically loaded javascript which can make full use of bootstrap & useful libraries provided
9. **Full reuse of dynamic icons** (don t want to recreate icons or each plugin logics here) from the json descriptions
10. **Dashboards should customizable by the end user**, he choses the pages and the devices he wants to see (not done yet at this point !)
11. Later on , more features, reuse of UI7 json descriptions for dashboards, control buttons etc if possible. To be investigated
12. Works on UI5 and UI7 with minor degradation on UI5 (housemode for instance)

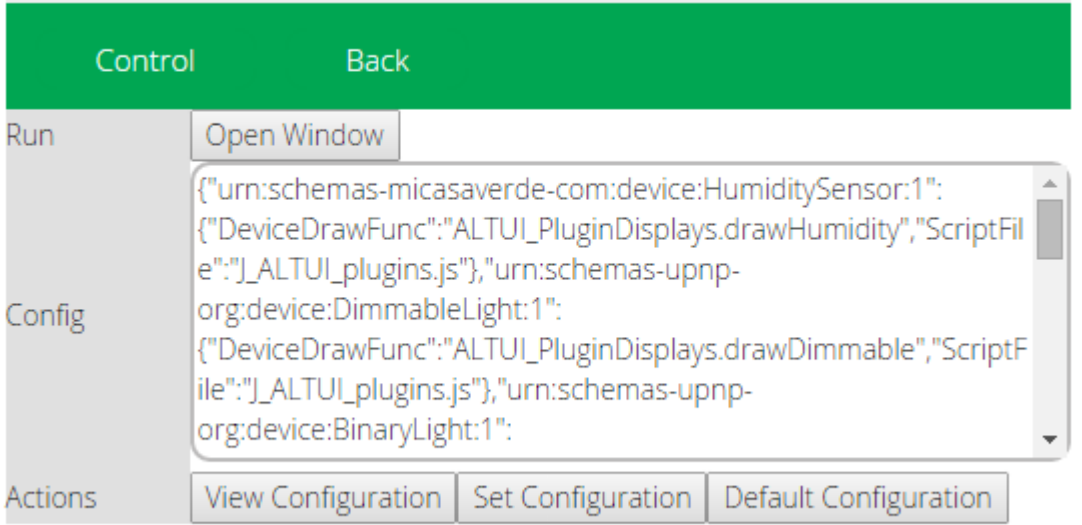
DONE and functional so far – VERSION 0.29

I already uses this more than UI7 on my ipad/phone and desktop at home. So far achievements are the following:

- Single plugin to install. Then access via the setting tab, or directly via :
http://<veraip>/port_3480/data_request?id=lr_ALTUI_Handler&command=home#
- **Remote access login** via a PHP page located at <http://amg0.site11.com/Veralogin.php> or <http://www.amg0.890m.com/Veralogin.php>
 - Unlike UI5 UI7, All plugins **custom ICONS are working** on remote access screen ! The backend servers of VERA (MMS) are not transporting/caching custom icons. This application uses a different technique, the plugin handler transmit necessary images as data URI and the client caches them
- Full responsive design, bootstrap & jquery based . check on your iPad when you rotate from portrait to landscape, it is quite fun , it just adds a column
- HouseMode view & change on ui7
- Footer display of VERA parameters (serial etc)
- List / Delete Rooms
- List devices with ultra simple and small dashboard
 - Device drawing can be customized by an individual js file. List of files to load is hard coded in the .LUA plugin code file but I ll change that later
 - Device Variable display : alphabetically sorted, htmlEncoded (so xml & json appear properly) , **humanly readable timestamps**.
 - Device variable Edit capability : click on it, modify, click out (it will save it automatically)
 - Device UPNP action dialog to dynamically retrieve actions & parameters names and enable the user to trigger action from this dialog.
 - Device Control Panel for a full page drawing customizable by device type dynamically. By default if no plugin is provided to customize the control panel , **the control panel page will display the same “control” tab as the plugging is doing on the VERA box UI5 UI7 (the “flash” tab in the JSON device settings)**. It will try to emulate the vera device control placement rules as much as possible so the look & feel is similar to what the plugin author intended. If that is not enough or if a finer grained customization is wished in the user interface, the full ability to write a custom UI in a dynamically loaded javascript module is possible
 - Ability to filter devices on the device page by
 - per Room of devices
 - per Category of Device
 - per Battery device or not
 - per Visible /Invisible status
 - by name (filter “as you type” a name to search)
 - Display / color of Battery level
 - Room filtering selection in the left bar

- List Delete , Edit, Run Scene
 - Full display of the scene parameters & lua code
 - Display of last run, next rune timestamp
 - Room filtering selection in the left bar
 - Edit features :
 - enable/disable & delete for triggers/timers/actions/groups and room assignment.
 - Delete individual triggers/timers/actions/groups
 - Add/edit Lua code for event triggers (UI7 does not have this capability, UI5 used to have it)
 - Wip...
- 2 fully asynchronous engines : user_data/l_data processing engine and UI refresh engine.
- See device dashboard and can filter by room
 - Implemented :
 - powerswitch, humidity, temperature, dimmers, door lock, door sensor, window covers
 - example of custom 3rd party plugin display: iphonelocator, cplus,
 - Motion sensor (arm / trip status)
 - all device icons are working (even if dynamic) including 3rd party plugin, including old UI5 mode ones
 - Not yet :
 - camera,
 - 3rd party plugin (but if authors are interested it is very easy, checkout the IphoneLocator .JS file)
- Plugins
 - Icon, name and version
 - List of installed 3rd party plugins with version and button to go directly to
 - Author help page
 - MIOS App Store page for the plugin
 - Single button to update to latest version
- Lua restart command in the menu
- Lua startup code edition
- Lua test code dialog box (broken feature on UI7 chrome)
- User control for the cache of icon & files in persistent storage (local storage HTML5 so persistent on a given machine). User can save or clear the cache. It will avoid redownloading icons unnecessarily
- Credit page

Screen shots:

LUA plugin	<p>Plugin Setting for configuration :</p> <p>Open => launch the ALTUI window</p> <p>Dynamic configuration for additional modules/plugin display functions</p> <p>Reset config to default</p> <p>Open & View configuration in a online json viewer</p> 
------------	--

JSON Editor Online

NewOpenSaveHelp

1

```
{
  "urn:schemas-micasaverde-com:device:HumiditySensor:1": {
    "DeviceDrawFunc": "ALTUI_PluginDisplays.drawHumidity",
    "ScriptFile": "J_ALTUI_plugins.js"
  },
  "urn:schemas-upnp-org:device:DimmableLight:1": {
    "DeviceDrawFunc": "ALTUI_PluginDisplays.drawDimmable",
    "ScriptFile": "J_ALTUI_plugins.js",
    "urn:schemas-upnp-org:device:BinaryLight:1": {
      "StyleFunc": "ALTUI_PluginDisplays.getStyle",
      "DeviceDrawFunc": "ALTUI_PluginDisplays.drawBinaryLight",
      "ControlPanelFunc": "ALTUI_PluginDisplays.drawBinLightControlPanel",
      "ScriptFile": "J_ALTUI_plugins.js",
      "info": {
        "ui7Check": "true",
        "urn:schemas-micasaverde-com:device:TemperatureSensor:1": {
          "DeviceDrawFunc": "ALTUI_PluginDisplays.drawTempSensor",
          "ScriptFile": "J_ALTUI_plugins.js",
          "urn:schemas-upnp-org:device:cplus:1": {
            "DeviceDrawFunc": "ALTUI_IPhoneLocator.drawCanalplus",
            "ScriptFile": "J_ALTUI_iphone.js",
            "urn:schemas-micasaverde-com:device:MotionSensor:1": {
              "DeviceDrawFunc": "ALTUI_PluginDisplays.drawMotion",
              "ScriptFile": "J_ALTUI_plugins.js"
            },
            "urn:schemas-upnp-org:device:IPhoneLocator:1": {
              "StyleFunc": "ALTUI_IPhoneLocator.getStyle",
              "DeviceDrawFunc": "ALTUI_IPhoneLocator.drawIPhone",
              "ControlPanelFunc": "ALTUI_IPhoneLocator.drawControlPanel",
              "ScriptFile": "J_ALTUI_iphone.js",
              "urn:schemas-micasaverde-com:device:WindowCovering:1": {
                "DeviceDrawFunc": "ALTUI_PluginDisplays.drawWindowCover",
                "ScriptFile": "J_ALTUI_plugins.js"
              }
            }
          }
        }
      }
    }
  }
}
```

object {9}

urn:schemas-micasaverde-com:device:HumiditySensor:1 {2}

urn:schemas-upnp-org:device:DimmableLight:1 {2}

urn:schemas-upnp-org:device:BinaryLight:1 {4}

StyleFunc : ALTUI_PluginDisplays.getStyle

DeviceDrawFunc : ALTUI_PluginDisplays.drawBinaryLight

ControlPanelFunc : ALTUI_PluginDisplays.drawBinLightControlPanel

ScriptFile : J_ALTUI_plugins.js

info {1}

urn:schemas-micasaverde-com:device:TemperatureSensor:1 {2}

urn:schemas-upnp-org:device:cplus:1 {2}

urn:schemas-micasaverde-com:device:MotionSensor:1 {2}

urn:schemas-upnp-org:device:IPhoneLocator:1 {4}

urn:schemas-micasaverde-com:device:WindowCovering:1 {2}

Remove
Access via
MMS servers
for UI7 boxes

Step 1:

VERA AltUI

VERA AltUI

Détails

VERA AltUI

Alexis

www.amg0.890m.com/Veralogin.php

VERA™

Smarter Home Control

Remote Login to VERA Alternate UI

Enter login

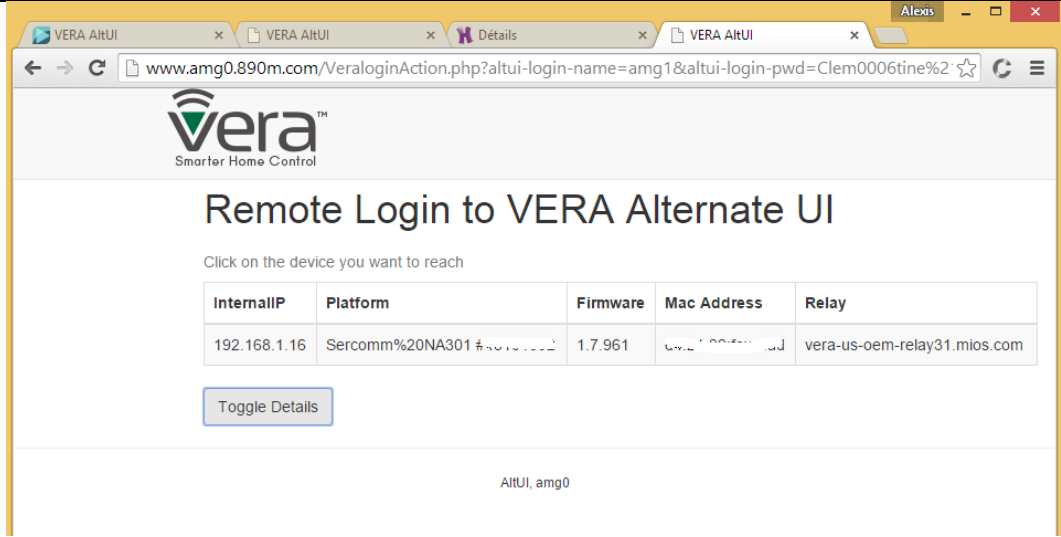
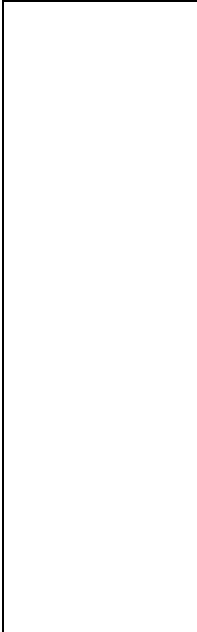
Password

Remember me

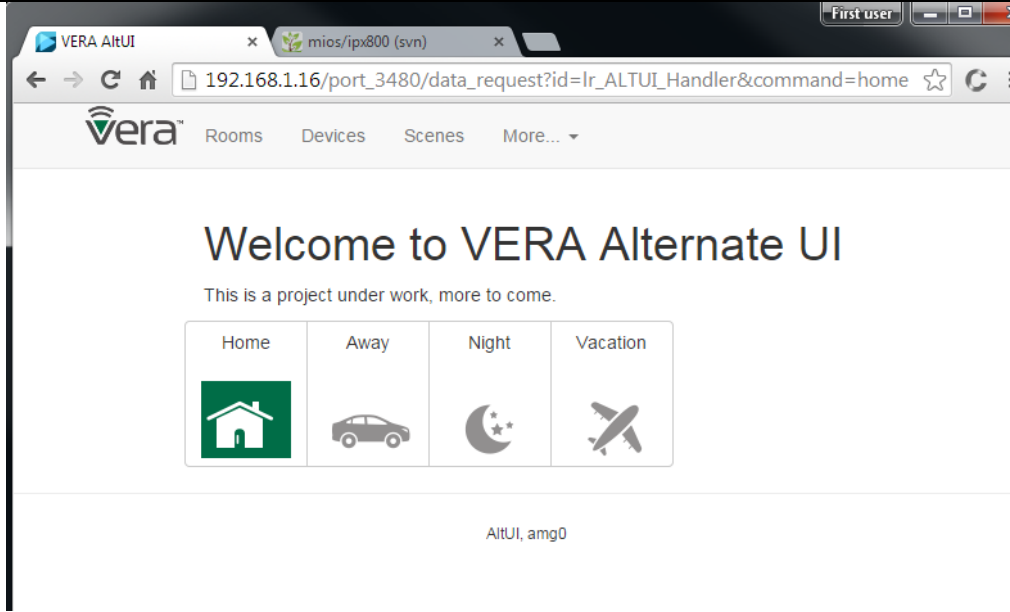
Sign in

AltUI, amg0

Step2:



Home mode selection



Room list and
create/delete
actions

VERA AltUI

← → ↺ 🏠

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=ho

☆ ↺ ☰

vera™

Rooms

Devices

Scenes

Custom Pages

Plugins

More... ▾

Rooms

Room name...

Create!

ID	Name	Actions
11	0-iPhones	<div>🗑</div>
4	Bureau	<div>🗑</div>
10	Ch Amis	<div>🗑</div>
8	Ch Clementine	<div>🗑</div>
7	Ch Parent	<div>🗑</div>
5	Cine	<div>🗑</div>
3	Cuisine	<div>🗑</div>
13	Dressing	<div>🗑</div>
6	Exterieur	<div>🗑</div>
2	Hall Etage	<div>🗑</div>
12	RFX	<div>🗑</div>
1	Salon	<div>🗑</div>

Scene list and execution

The screenshot displays the Vera! AIUI web interface in a browser. The address bar shows the URL `192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#`. The interface features a top navigation bar with the Vera! logo and links for Rooms, Devices, Scenes, Custom Pages, Plugins, and More... A left sidebar contains a list of rooms: All, Salon, Hall Etage, Cuisine, Bureau, Cine, Exterieur, Ch Parent, Ch Clementine, Ch Amis, 0-iPhones, RFX, and Dressing. The main content area is titled "Scenes" and lists ten scenes in a grid. Each scene entry includes a "Run" button, a settings icon, and a timestamp indicating when the scene was last executed.

Scene Name	Scene ID	Last Execution
Alexis 1km	#57	2015/02/22 17:05:56
Alexis 3km	#56	2015/02/19 20:23:00
Battery is Low	#8	2015/01/31 18:52:28
CanalPlus	#53	2015/02/22 15:09:43
Carillon	#27	
Cine_PS3_OFF	#55	
Cine_PS3_On	#54	
Cine_TV_On	#58	2015/02/21 19:05:26
Cine_Tv_Off	#59	
Film-Regarder	#4	2015/02/21 21:33:46

Device Main page

color coding of headers according to device state.

State icons & dynamic display icon logic completely reused from the Vera files JSON description files of vera plugins

The screenshot displays the Vera Home Automation web interface. The browser address bar shows the URL: `192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#`. The interface features a sidebar on the left with a list of rooms: All, Salon, Hall Etage, Cuisine, Bureau, Cine, Extérieur, Ch Parent, Ch Clementine, Ch Amis, 0-iPhones, RFX, and Dressing. The main content area is titled 'Devices' and includes a search bar labeled 'Device Name'. Below the search bar, there is a grid of device cards. Each card displays the device name, a status icon (e.g., a green dot for 'ON' or a light bulb for 'OFF'), and a control toggle. The devices are organized into rows and columns, with some cards showing additional information like distance (e.g., '0.026 Km') or temperature (e.g., '19°C'). The interface uses a color-coded system for headers and status indicators, as mentioned in the text.

Tooltips with
device
attributes

Devices

ZWave



Prise Groupe Clim



800 Watts

ON



Jardin



OFF



Bouteille



2 Watts

ON



Volet Clementine

_Scene Controller



Caves Vin



187 w

Lamp



Jardin



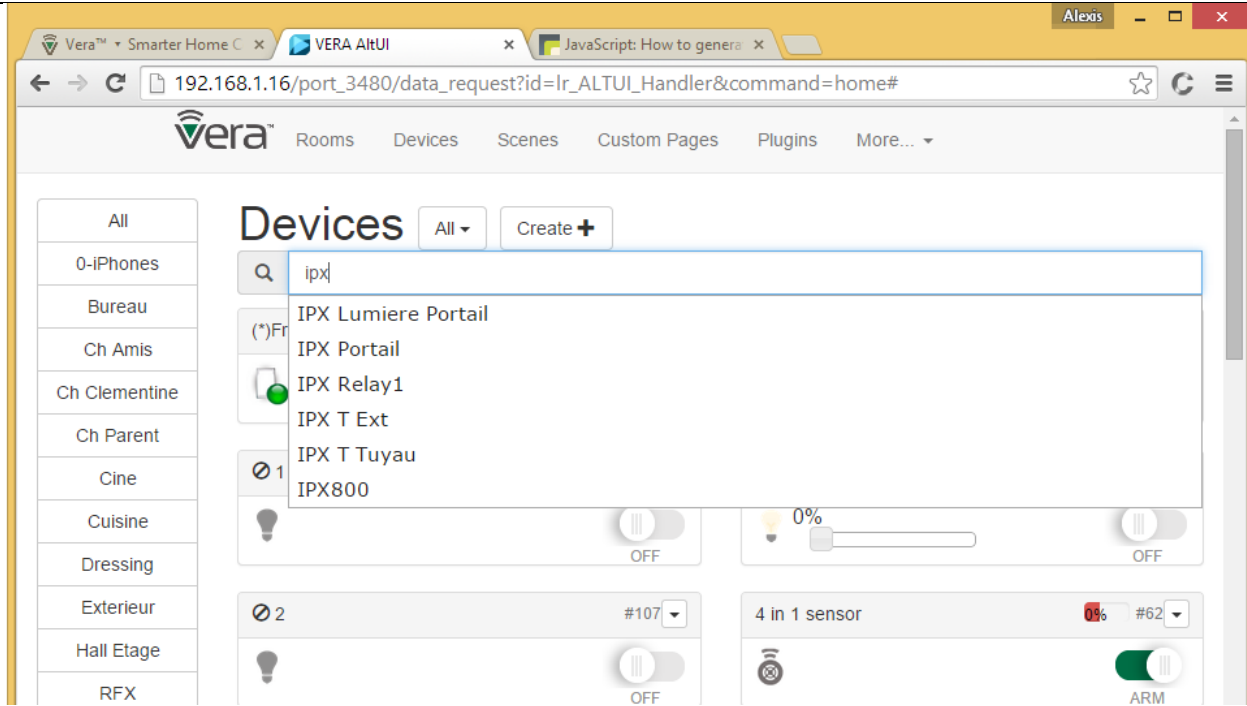
Volet



Store

```
id: 6
device_type: urn:schemas-upnp-org:device:BinaryLight:1
id_parent: 1
embedded: 0
disabled: 0
device_file: D_BinaryLight1.xml
impl_file:
model:
altid: 4
ip:
mac:
time_created: 1331907034
category_num: 3
subcategory_num: 0
room: 1
name: Lampadaire
onDashboard: 0
device_json: D_BinaryLight1.json
manufacturer: Everspring
local_udn: uuid:4d494342-5342-5645-0006-000002b03150
dirty: false
```

autocomplete
filter box



Ability to
filter on
device
Battery,

Display of
battery levels

Vera™ Smarter Home C x VERA AltUI x



192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

Vera™ Rooms Devices Scenes Custom Pages Plugins More... ▾


Devices All ▾

☒ Invisible ☒ Battery Devices **Category:** All ▾



4 in 1 sensor 0% #62 ▾

  ARM


RFX Temp Ext Oregon 10% #71 ▾

 5.6°C


Motion Ext 50% #156 ▾

  ARM

RFX Temp TH1/31489 100% #110 ▾

 17.4°C


RFX Hum TH1/31489 100% #111 ▾


 48%

- All
- Salon
- Hall Etage
- Cuisine
- Bureau
- Cine
- Exterieur
- Ch Parent
- Ch Clementine
- Ch Amis
- 0-iPhones
- RFX
- Dressing


Click on
device title to
rename


- Exterieur
- Ch Parent
- Ch Clementine
- Ch Amis
- 0-iPhones
- RFX
- Dressing

 159 Watts


ON 


Lampadaire

 0 Watts


OFF 


Jardin|




OFF 


Jardin #2




OFF 

 1'


Volet C



Jardin

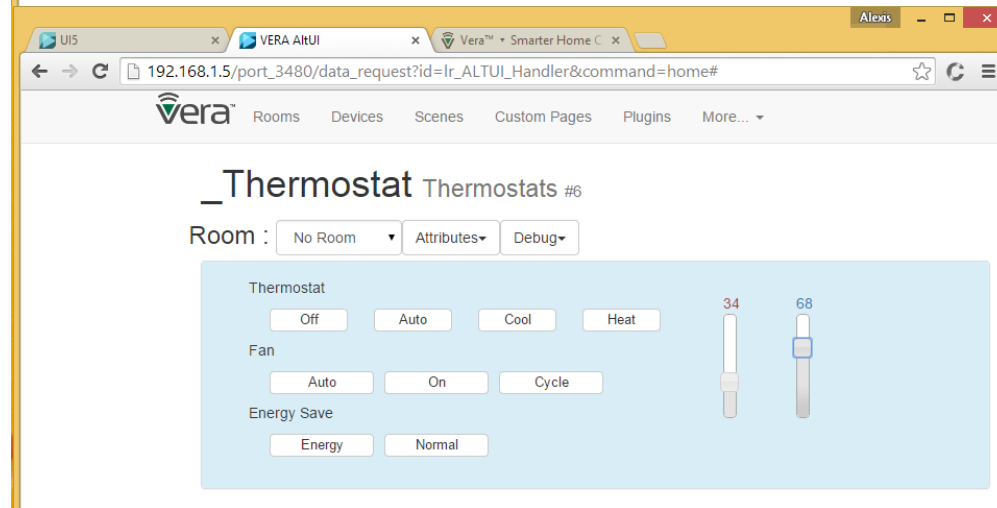
 0

Bouteil



Device Control Panel screen emulate VERA and display the same control panel as the “flash” tab of the device on VERA

Buttons are functional and trigger UPNP actions.



Camera support.

Click on thumbnail to view

Or go into the device control panel

In REMOTE mode: get snapshot of images

In LOCAL mode: get **direct video** stream

VERA AltUI

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

vera Rooms Devices Scenes Plugins Custom Pages More...

Devices

Camera ▼ Create +

☒ Invisible ☐ Battery Devices Category: Camera ▼

Q Device Name

Camera PoE #58 ▼

Camera Wifi

Camera

Close Save changes

VERA AltUI

First user

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

vera

Rooms

Devices

Scenes

Plugins

Custom Pages

More...


Camera PoE Cameras #58

Room :

Salon

Attributes

Debug



Horizontal Patrol

Vertical Patrol

Stop

Camera preset positions

1

2

3

4

5

6

7

8

Optional display of device attributes &

In DEBUG mode only (flag on the LUA device)
We can see the Control tab json definition (for debug)

Vera™ Smarter Home C x

VERA AltUI x

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

vera

Rooms

Devices

Scenes

Custom Pages

Plugins

More...

IPX800 #125

Room :

No Room

Attributes

Debug

id	125	device_type	urn:schemas-upnp-org:device:IPX800:1
room	0	id_parent	0
embedded	0	disabled	0
device_file	D_IPX800.xml	altid	
time_created	1416757981	plugin	7426
impl_file	I_IPX800.xml	name	IPX800
manufacturer	GCE Electronic	model	IPX800 v3
mac	00:04:A3:93:A2:7B	user	
pass		ip	192.168.1.10
device_json	D_IPX800_UI7.json	local_udn	uuid:4d494342-5342-5645-007d-000002b03150
dirty	false		

Firmware 3.05.46

Version v0.41

UNDEBUG

Refresh

IPX Names

{ "ControlGroup": "1", "ControlType": "label", "top": "1", "left": "0", "text_align": "left", "Label": { "lang_tag": "ipx_firmware", "text": "Firmware" }, "Display": { "Top": 1, "Left": 10, "Width": 50, "Height": 20 } }

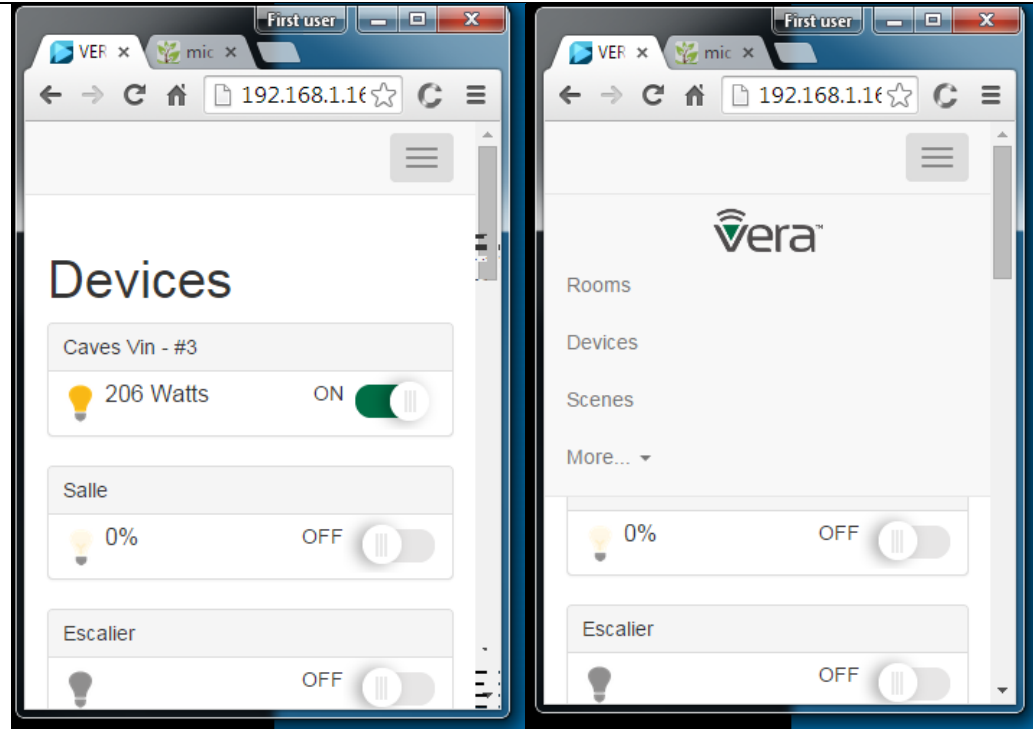
{ "ControlGroup": "1", "ControlType": "variable", "top": "1", "left": "1", "x": "2", "text_align": "left", "Display": { "Service": "urn:upnp-org:serviceId:IPX8001", "Variable": "FirmwareVersion", "Top": 1, "Left": 80, "Width": 50, "Height": 20 } }

{ "ControlGroup": "1", "ControlType": "button", "top": "2", "left": "0", "Label": { "lang_tag": "ipx_refresh", "text": "Refresh" }, "Display": { "Service": "urn:upnp-org:serviceId:IPX8001", "Variable": "Present", "Value": "1", "Top": 80, "Left": 50, "Width": 75, "Height": 20 }, "Command": { "Service": "urn:upnp-org:serviceId:IPX8001", "Action": "Refresh", "Parameters": [] } }

{ "ControlGroup": "1", "ControlType": "button", "top": "2", "left": "1", "Label": { "lang_tag": "ipx_getname", "text": "IPX Names" }, "Dir

Fully
Responsive
design that
works on iPad
or even
iPhone 4S
small screen

On iPad for
instance, it
adds columns
when you
rotate the
iPad



Device variables presented. Timestamp presented as dates

All

Salon

Hall Etage

Cuisine

Bureau

Cine

Exterieur

Ch Parent

Ch Clementine

Ch Amis

0-iPhones

RFX

Dressing

De

ZWave

Prise

Jardin

Bouteille

Volet

Vacances

Exterieur

World

VERA AltUI

CSS - Bootstrap

VERA™ - Smarter Home C

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

VERA™

Rooms

Devices

Scenes

Custom Pages

Plugins

More

Bouteille #24 - Variables

Variable	Value
AllRoutesFailed	0
AssociationNum	0
AssociationSet	
AutoRoute	0-5,2-15,6-15,3-19
Capabilities	210,148,0,3,16,1,L,R,B,[37,39,114,115,117,134,
CommFailure	0
CommFailureTime	0
Configured	1
ConfiguredAssoc	
ConfiguredName	
ConfiguredVariable	
ConfiguredWakeupInterval	
FirstConfigured	28/4/2012 14:25:41
Health	
HealthDate	30/12/2014 21:37:02
LastReset	0
LastRouteUpdate	1/2/2015 17:04:49
LastUpdate	31/1/2015 16:59:25
Log	2,2,2,1422826200,1
ManufacturerInfo	96,4,1
MeterScale	
MeterType	
ModeSetting	1;;2;;3;;4;
MultiChCapabilities	
MultiChEndpoint	

jQuery UI Touch Punch

Afficher tous les téléchargements...

Edit device variable by click into , then click out

Smarter Home C x VERA AltUI x CSS - Bootstrap x .on() | jQuery API Docume x

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

vera Rooms Devices Scenes Custom Pages Plugins More...

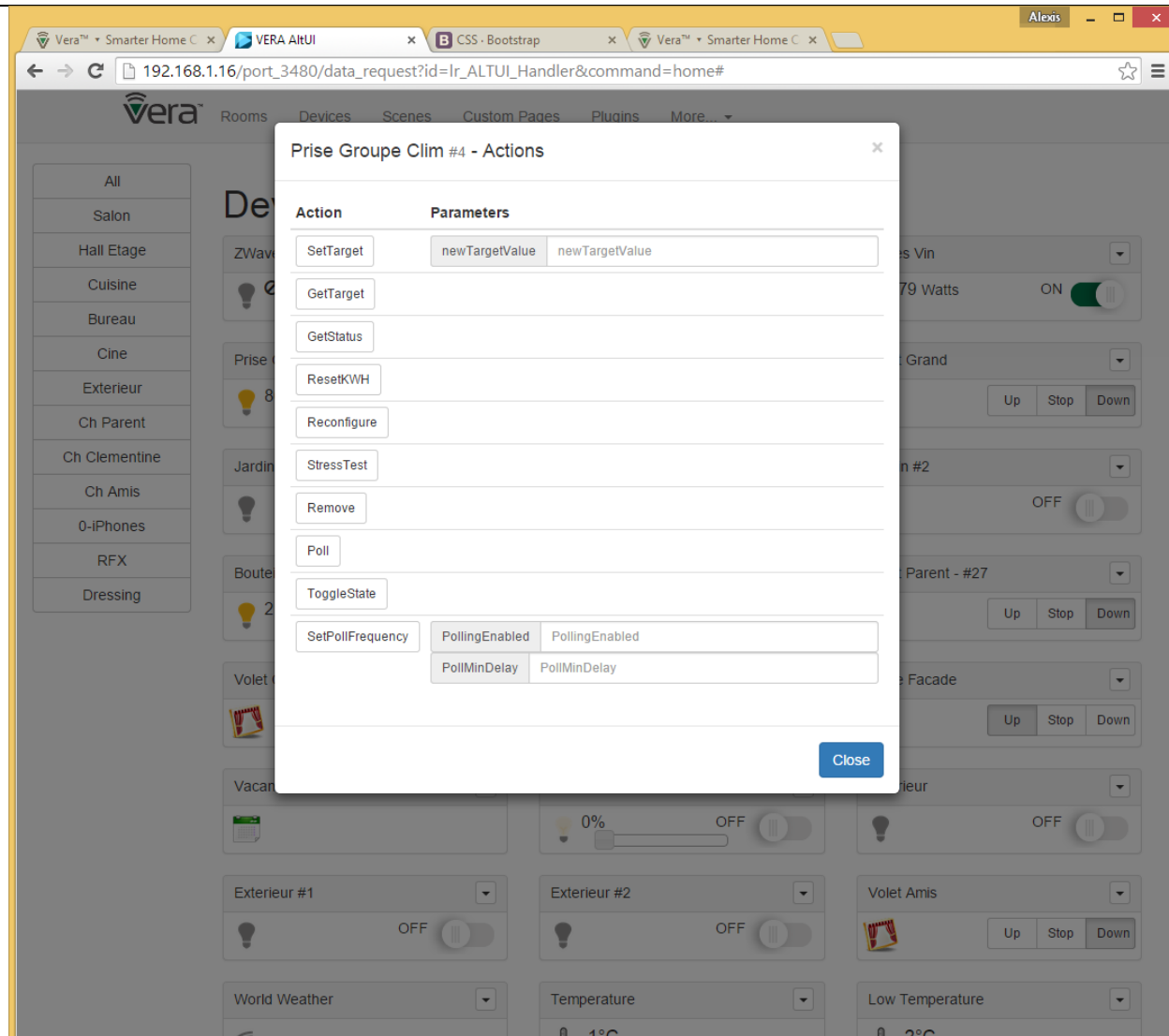
4 in 1 sensor (temp #63 - Variables

Variable	Value
ArmedTripped	1
CurrentTemperature	<input type="text" value="19"/>
LastTrip	2013/08/10 13:28:13
ManufacturerInfo	0,0,0
SensorMIScale	3
SensorMIType	1
Tripped	1
VariablesGet	

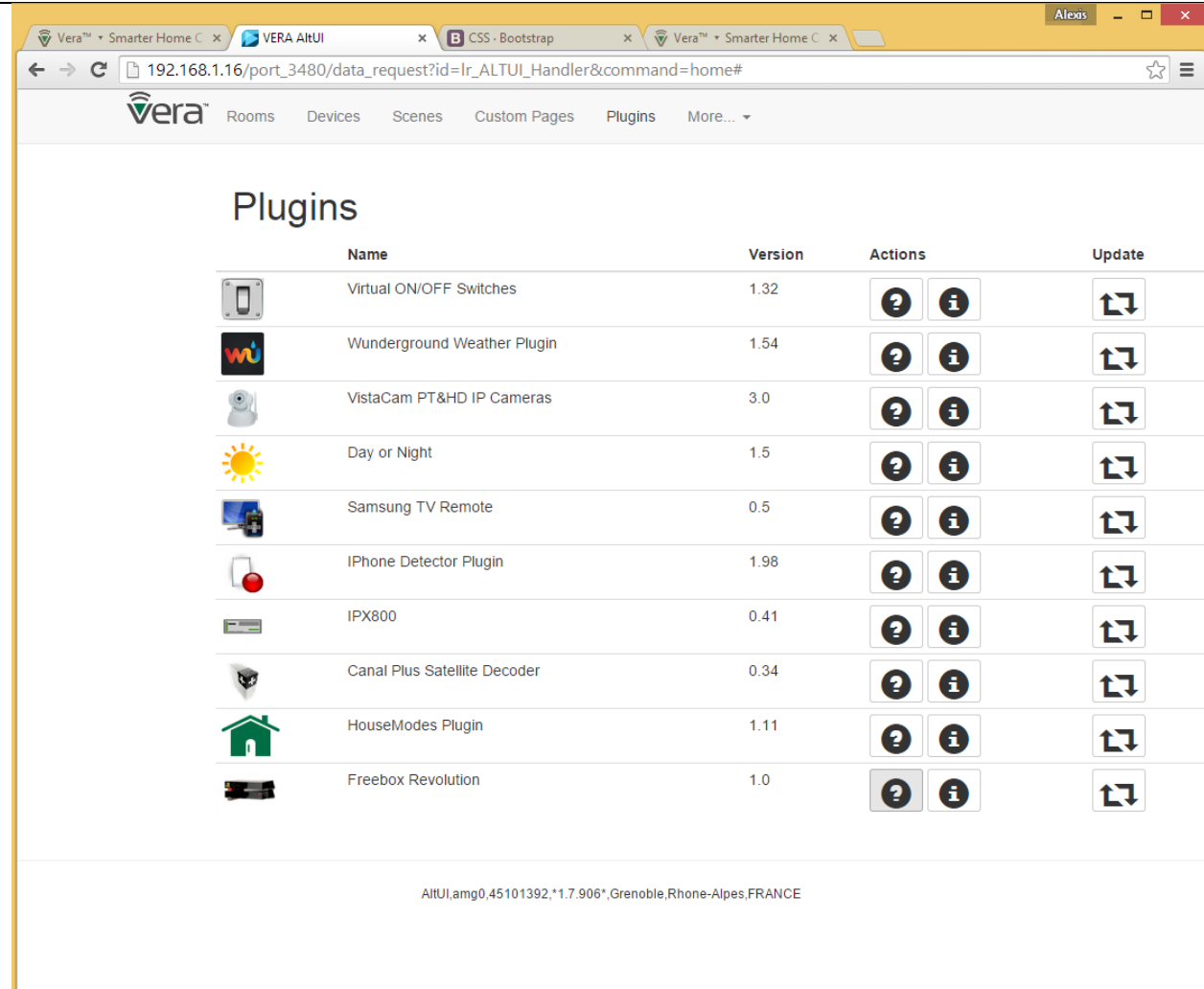
Close Save changes

Routeille #24 CPLUS #140 Caisson Basse #105 Cam PoF









































Device UPNP
action &
parameters
callable from
the user
interface.
UPnp
definitions
dynamically
read from the
D_ & S_xx
files



Installed
Plugin screen
and Update
with a button



The screenshot shows the Vera UI Plugins page in a web browser. The browser's address bar displays the URL `192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#`. The page features a navigation bar with links for Rooms, Devices, Scenes, Custom Pages, Plugins, and More... The main heading is "Plugins". Below it is a table listing various plugins with their icons, names, versions, and action buttons. The table has four columns: Name, Version, Actions, and Update. Each plugin row includes an icon, the plugin name, its version number, and two buttons in the Actions column (a question mark and an 'i' icon) and an Update button (a circular arrow icon).

	Name	Version	Actions	Update
	Virtual ON/OFF Switches	1.32	 	
	Wunderground Weather Plugin	1.54	 	
	VistaCam PT&HD IP Cameras	3.0	 	
	Day or Night	1.5	 	
	Samsung TV Remote	0.5	 	
	iPhone Detector Plugin	1.98	 	
	IPX800	0.41	 	
	Canal Plus Satellite Decoder	0.34	 	
	HouseModes Plugin	1.11	 	
	Freebox Revolution	1.0	 	

AltUI,amg0.45101392,*1.7.906*,Grenoble,Rhone-Alpes,FRANCE

Click on “i” directly brings to Apps Store

Click on “update” triggers an update of the plugin from the apps store

Click on “?” opens the developer HELP page

Message Box
for messages.

Badge for
repeated
messages.

Vera™ · Smarter Home C x

VERA AltUI x

B CSS · Bootstrap x

Vera™ · Smarter Home C x

Alexa

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

vera™

Rooms

Devices

Scenes

Custom Pages

Plugins

More... ▾

All

Salon

Hall Etage

Cuisine

Bureau

Cine

Exterieur

Ch Parent

Ch Clementine

Ch Amis

0-iPhones

RFX

Dressing

Devices

#7:Bouteille:SUCCESS! Transmit was OK

#6:Sapin:SUCCESS! Transmit was OK

Lampadaire

0 Watts

OFF

Store Facade

Up

Stop

Down

Cam Wifi

Unarmed

Ampli Onkyo

83 Watts

ON

Bouteille

2 Watts

ON

Camera Wifi

Cam PoE

Unarmed

Caisson Basse

10 Watts

ON

Store Cote

Up

Stop

Down

Camera PoE

Samsung TV Remote

Sapin

0 Watts

ON

AltUI,amg0,45101392,*1.7.906*,Grenoble,Rhone-Alpes,FRANCE


Example of
“grouped”
error
message with
a badge
number while
LUA is
restarted for
instance

The screenshot displays the Vera UI interface in a web browser. The address bar shows the URL `192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#`. The page header includes the Vera logo and navigation tabs: Rooms, Devices, Scenes, Custom Pages, Plugins, and More.... A sidebar on the left lists various rooms: All, Salon, Hall Etage, Cuisine, Bureau, Cine, Extérieur, Ch Parent, Ch Clementine, Ch Amis, 0-iPhones, RFX, and Dressing. The main content area is titled "Devices" and features a red error banner at the top that reads "5 Request failed: error". Below this banner, there are eight device control cards arranged in a 4x2 grid. The first row contains "Lampadaire" (0 Watts, OFF) and "Bouteille" (2 Watts, ON). The second row contains "Store Cote" and "Store Facade", each with "Up", "Stop", and "Down" buttons. The third row contains "Camera Wifi" and "Camera PoE". The fourth row contains "Cam Wifi" (Unarmed) and "Cam PoE" (Unarmed). On the right side of the browser window, the "Developer Tools" panel is open, showing the "Network" tab with a list of data requests to `/port_3480`.

Modify Lua Startup editor

VERA AltUI

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

RoomsDevicesScenesCustom PagesPluginsMore...

LUA Startup

Lua startup code:

```
MYIP = nil
-- luup.attr_set("invisible","",20)

--- DEVICES
DEVID_CAVE = 3
DEVID_CLIM = 4
DEVID_WND_BUREAUPETIT = 26
DEVID_WND_STORECOTE = 29
DEVID_WND_STOREFACADE = 30
DEVID_VACATION = 31
DEVID_MOTION_ENTREE = 156
DEVID_MOTION_4I1 = 62
DEVID_LIGHT_LAMPADAIRE = 6
DEVID_LIGHT_BOUTEILLE = 24
DEVID_LIGHT_CINE = 32
DEVID_LIGHT_JARDIN = 20
DEVID_LIGHT_ENTREE = 41
DEVID_LIGHT_ESCALIER = 51
DEVID_WEATHER = 45
DEVID_TEMP_EXT = 46
DEVID_TEMP_INT = 63
```

Submit

Lua test code

The screenshot shows a web browser window with the address bar displaying `192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#`. The page title is "LUA Code Test". A green notification bar at the top states "Test code succeeded". Below this, a message explains: "This test code will succeed if it is syntactically correct and does not return false. an error in the code or a return false will trigger a failure". The section "Lua Test Code:" contains a text area with the code `return true`. A "Submit" button is located below the text area.

Vera™ Smarter Home C x VERA AltUI x

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

Vera™ Rooms Devices Scenes Custom Pages Plugins More...

LUA Code Test

Test code succeeded

This test code will succeed if it is syntactically correct and does not return false. an error in the code or a return false will trigger a failure

Lua Test Code:

```
return true
```

Submit

Edit Scene #1

test

#1

Run



NaN

Room : No Room

Trigger Submit

<input checked="" type="checkbox"/>	blw 2	(*)iPhone de Alexis (#5)	Distance goes below	Distance < 2	
<input checked="" type="checkbox"/>	abv 35	_Thermostat (#8)	Ambient temperature goes above	CurrentTemperature > 35	

Timers Submit

<input checked="" type="checkbox"/>	Interval	interval: 3h	
<input checked="" type="checkbox"/>	Day of Week	day of week: 2,4 h:m:s= [7:0:0]	
<input checked="" type="checkbox"/>	Absolute	absolute:	
<input checked="" type="checkbox"/>	Day of Month	day of month: 5 h:m:s= [-1:0:0R]	
<input checked="" type="checkbox"/>	Interval	interval: 2h	

Actions Submit

0 sec	(*)iPhone de Alexis (#5)	ForceRefresh (PresentStatus: 0)	
	ALTUI (#4)	Refresh ()	
30 sec	(*)iPhone de Alexis (#5)	SetPresent (newPresentStatus: 0)	

Lua

Lua scene code: Submit

enter code here

Submit

Lua event trigger (does not exist any more in UI7 but it works fine) so I added it back with a test code button right there

The screenshot displays the Vera UI interface for configuring a trigger. At the top, a 'Trigger' dropdown is set to 'Submit'. Below it, a checkbox is checked, and the trigger is named 'Below 1km' for the device '(*)iPhone de Alexis (#94)'. The condition is 'Distance goes below' with a value of 'Distance < 1'. A '⚡ Lua' button is circled in orange, with a blue arrow pointing to the 'Lua Editor' dialog box.

The 'Lua Editor' dialog box is open, showing a text area for 'Lua Code' with the placeholder 'enter code here'. At the bottom of the dialog are three buttons: 'Close', 'Test Code', and 'Save changes'.

Below the dialog, the 'Actions' section is visible, with a 'Submit' button. Under '0 sec', the 'Lua' section shows a 'Lua scene code:' field with a 'Submit' button. The code area contains the following Lua script:

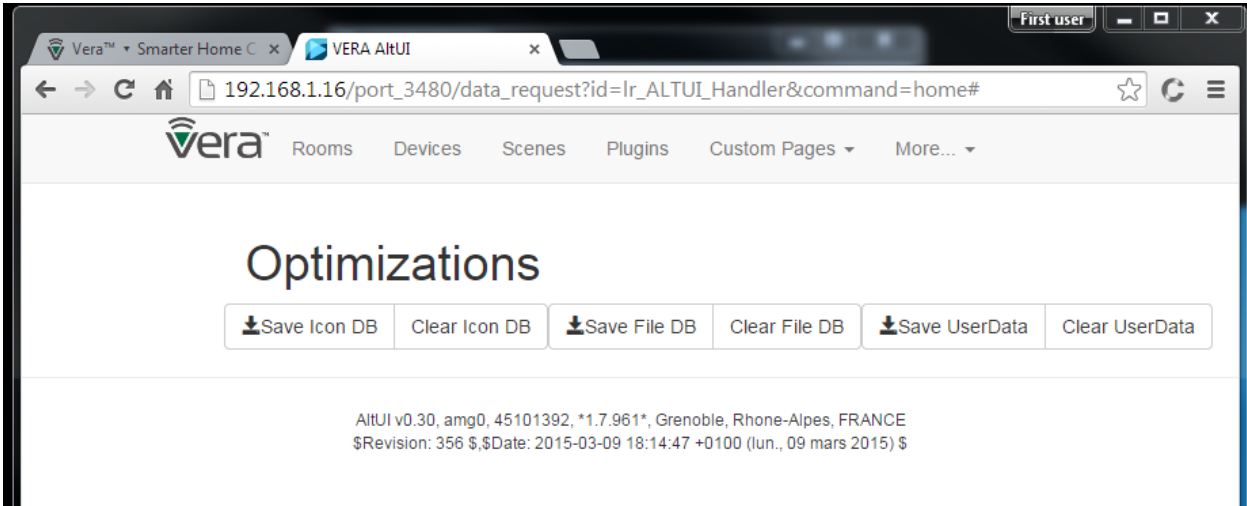
```
--- message
local current = os.time()
local message = "\nBelow 1km, \n Heure:" .. os.date("%c", current) .. "\n"
```

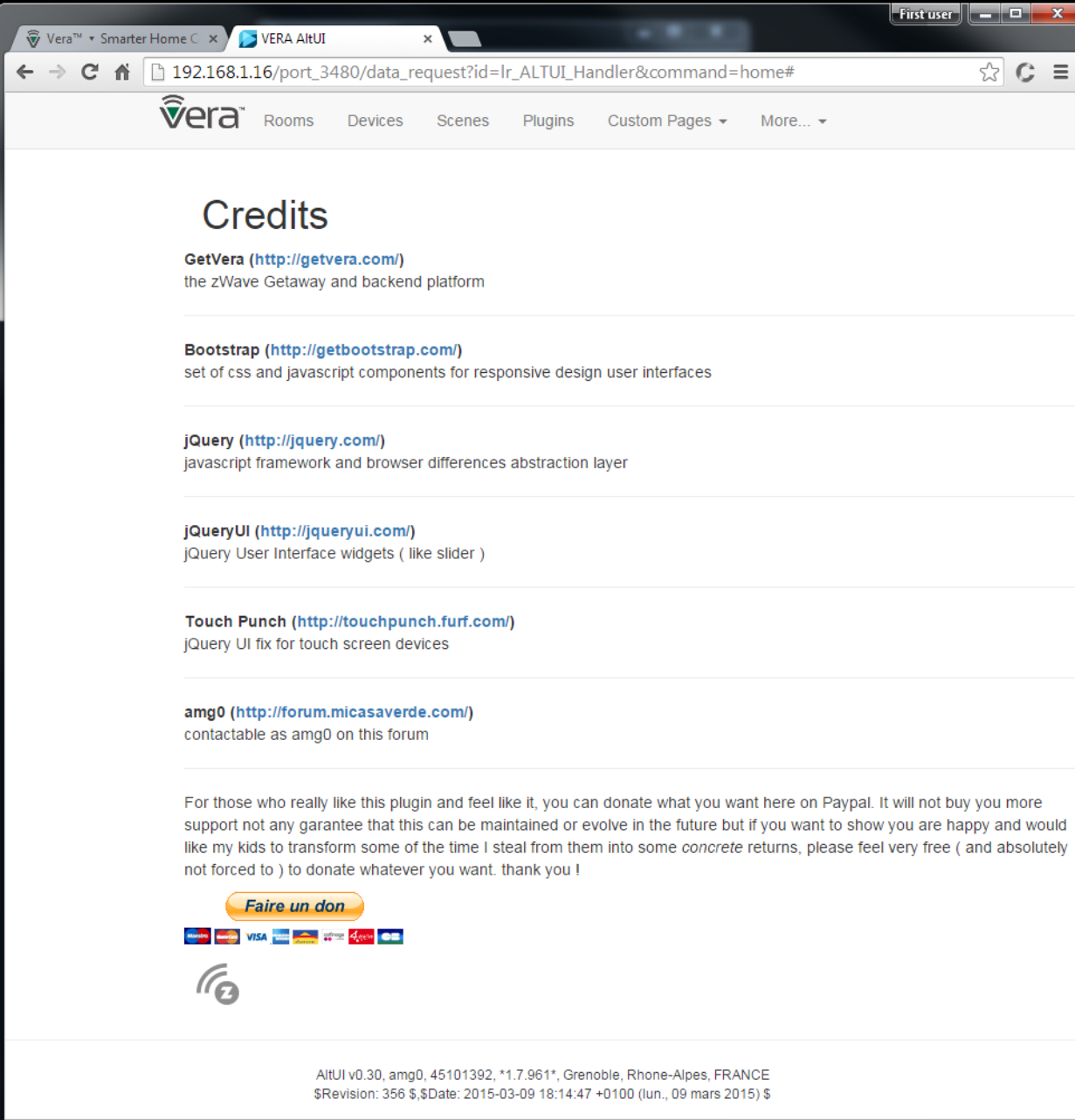
User
controllable
Cache

Cache for Icons (in remote access, icons are delivered as data uri , base64 and can be cached by the app)

Cache for device pnp files (D_xx S_xx) to avoid reloading when not needed.

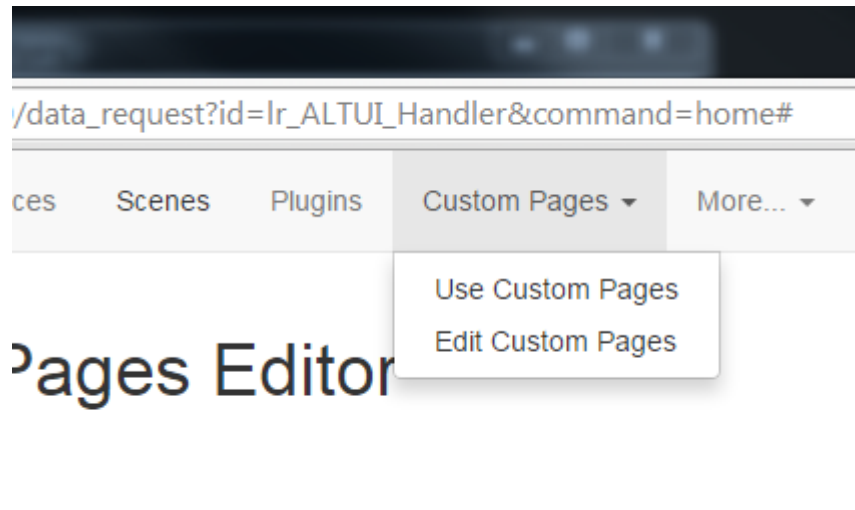
Cache for last user_data to optimize useage from remote location.



Credits	
Plugins / Custom device	

Custom Pages

The following below explains the concept around custom pages. You can basically create your own panels and retrieve these panels whenever you want. For this you have 2 new Menu commands

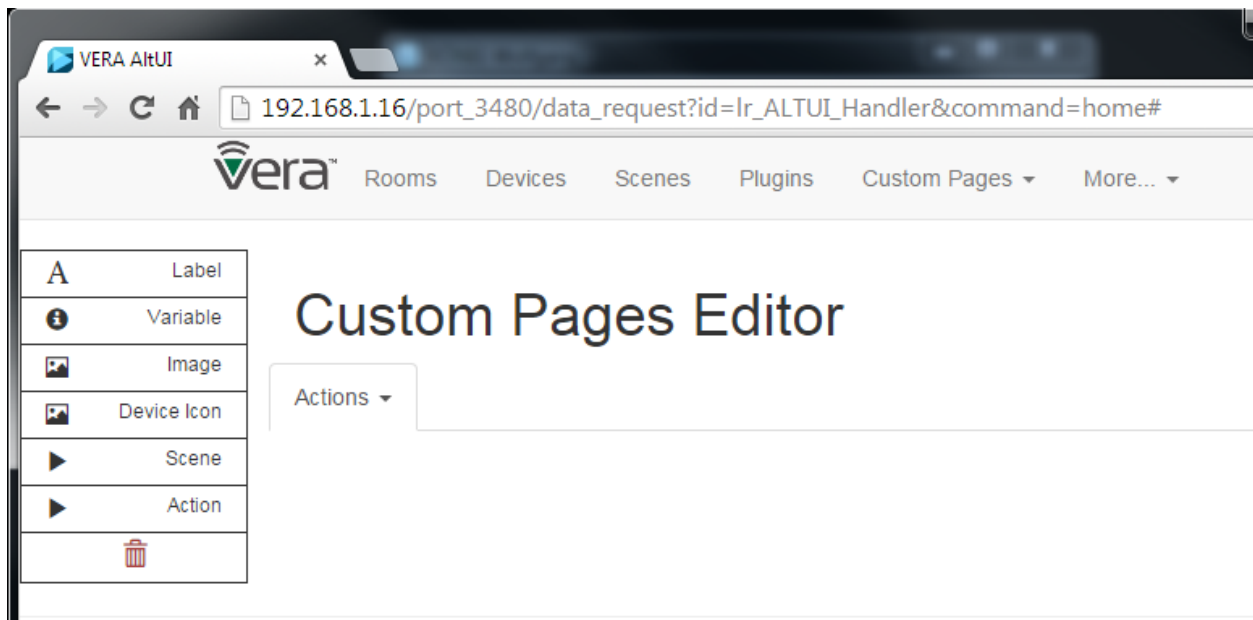


- Use Custom pages : just for readonly use of a custom panel you have built
- Edit Custom pages : to edit the panel.

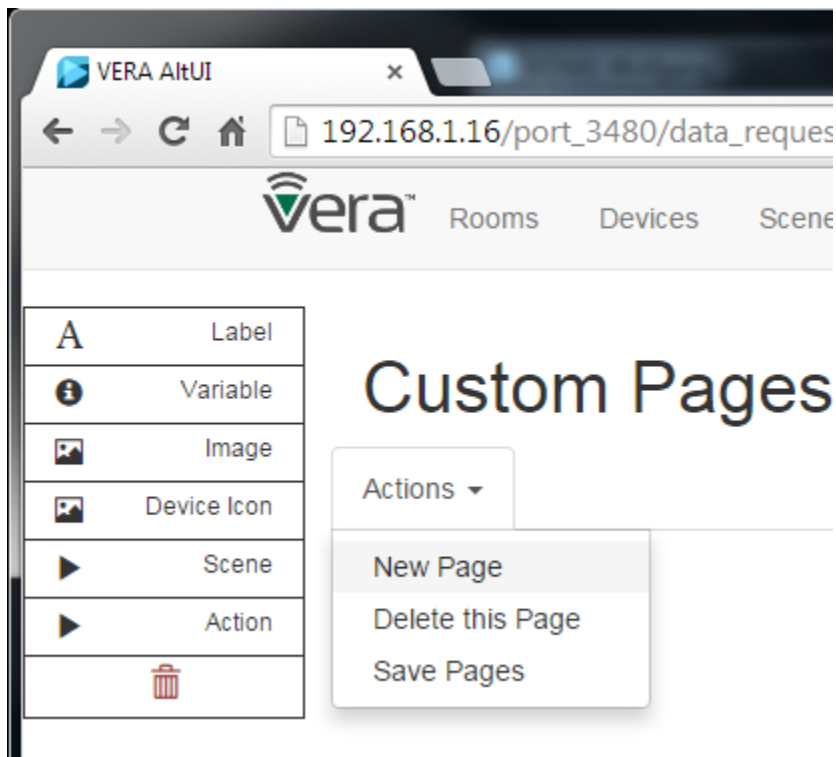
The first time you go there, you will not see any custom pages as you have not created any yet. So let's start by going into Edit mode first.

On the left, you have a list of tools in a toolbox. For now there are 3 tools:

- The Label one : to show a static label
- The Variable one : to display a current device variable value
- The trashcan : to delete a widget from a panel screen by drag and drop.

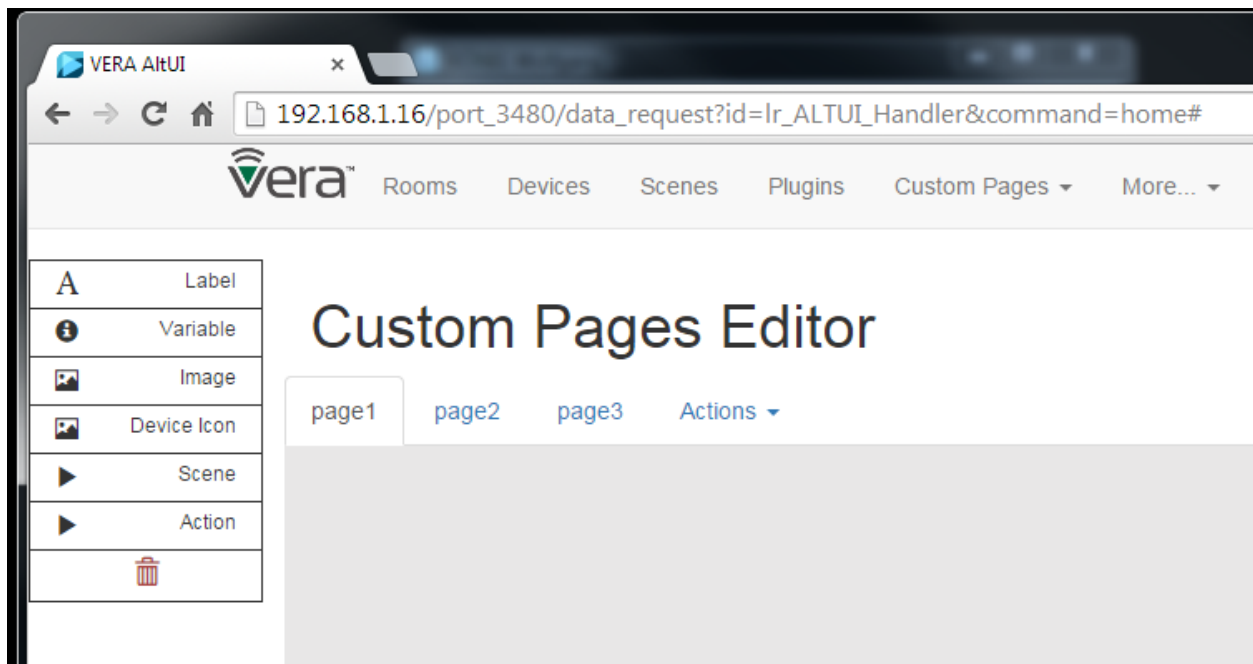


For now, you do not see any page , so let’s create one by going into the menu Actions.

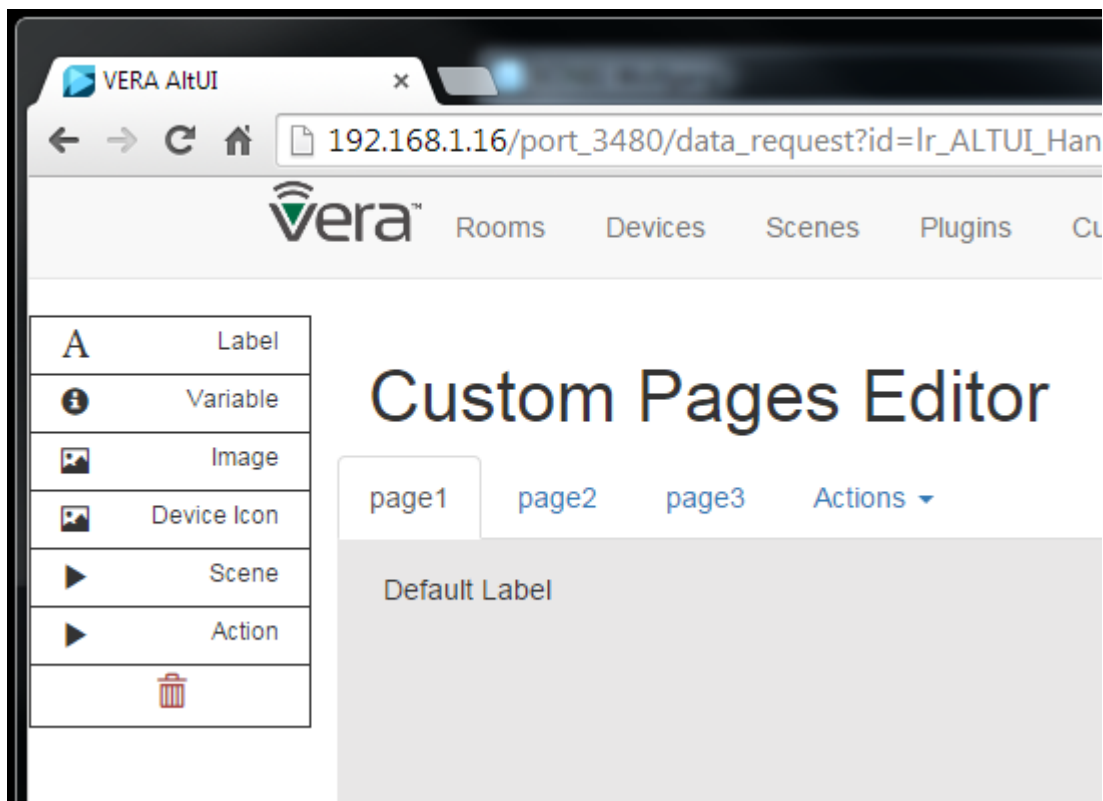


Click New page and your first page is created and is empty for now, but you see a grey canvas where you are going to position your controls.

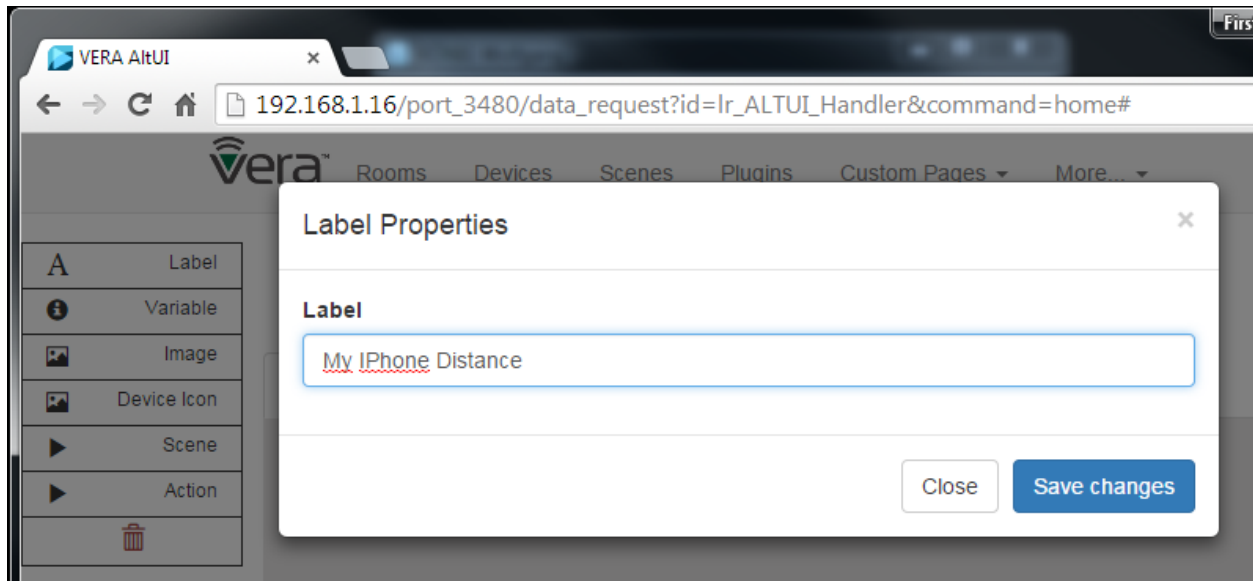
You can create several pages, they will be displayed as “Tabs” you can select to move from one page to the other.



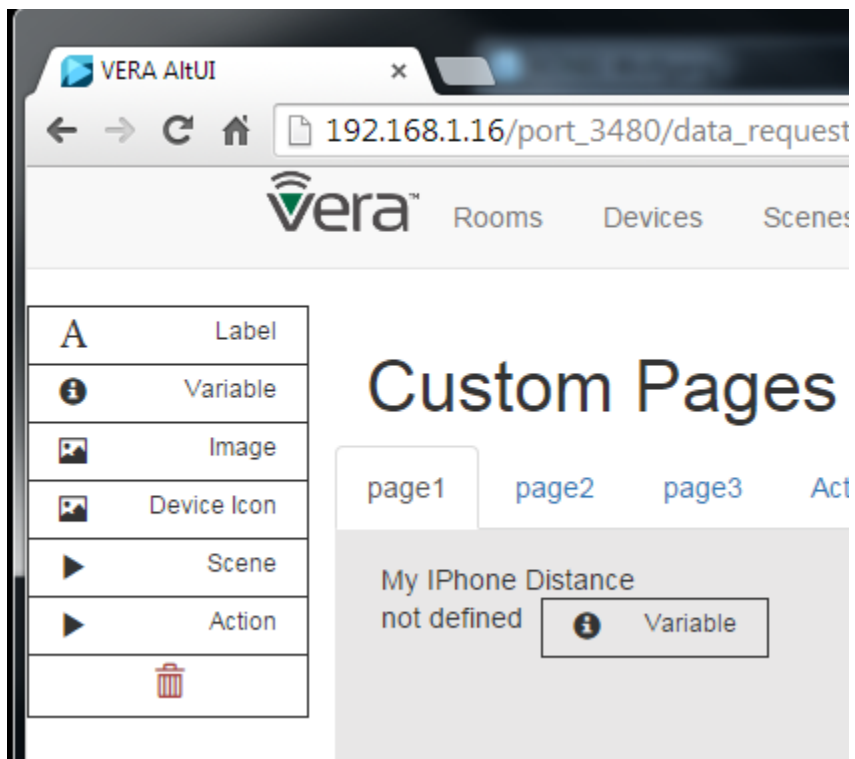
Now, let's position a few controls on the panel. I have an iPhoneLocator plugin in my Vera and I want to display the distance and the unit it is reporting. So you are going to select the first tool (the A for labels) and drag it into the canvas area. It will create a default label right at the position you left it.



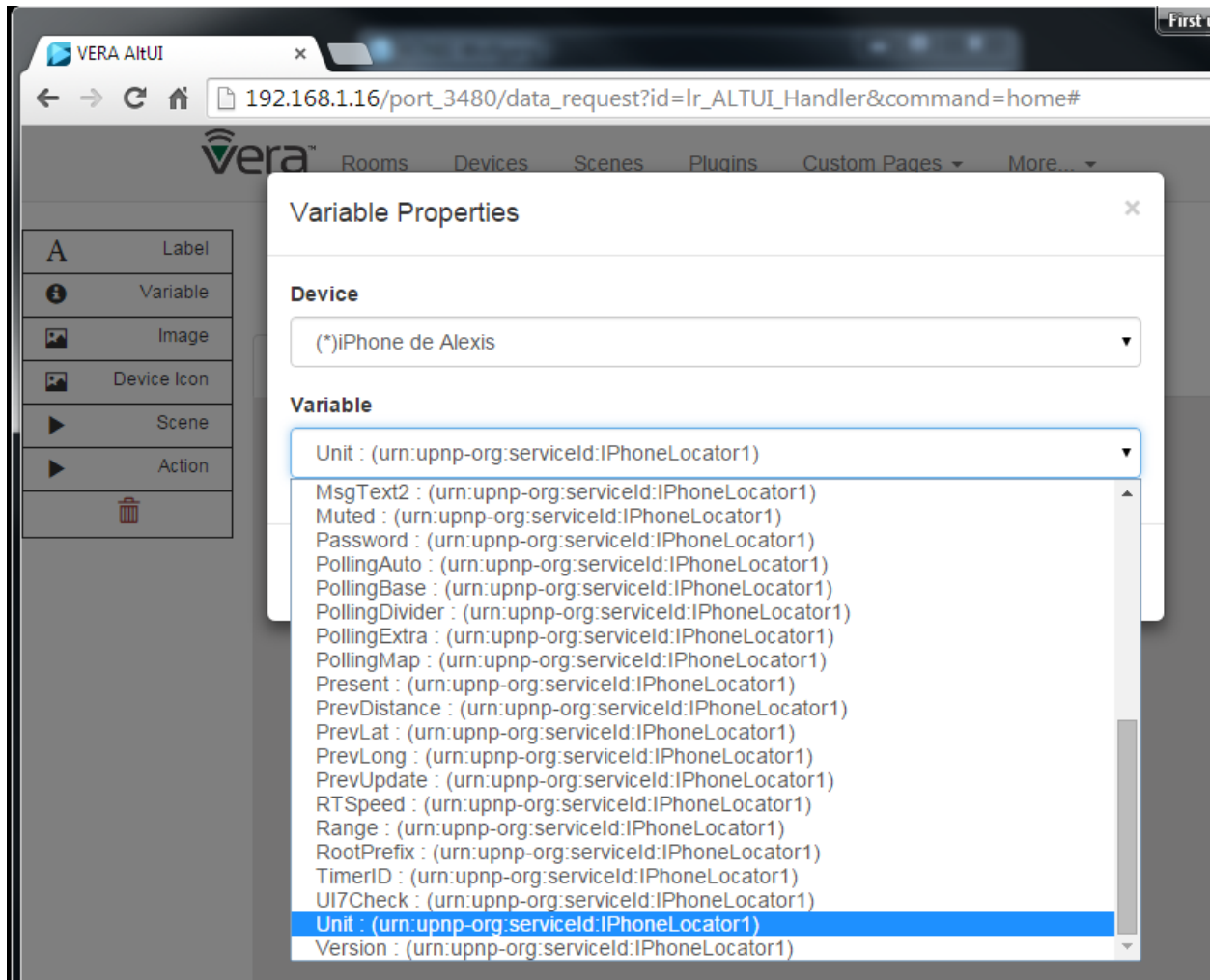
By clicking on it you can change this Default Label.



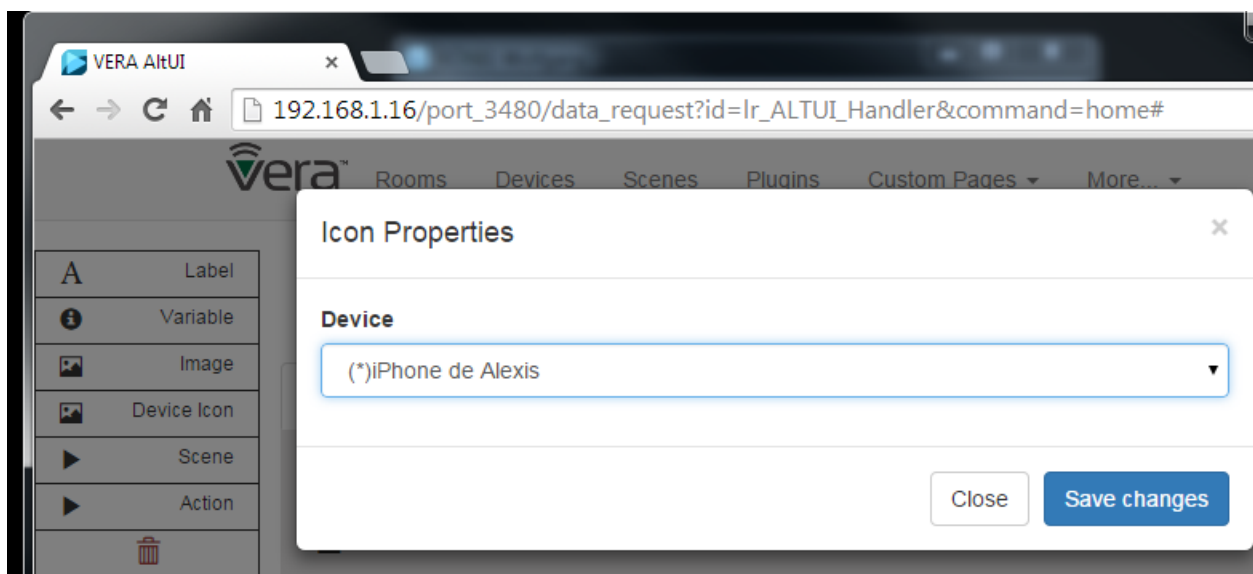
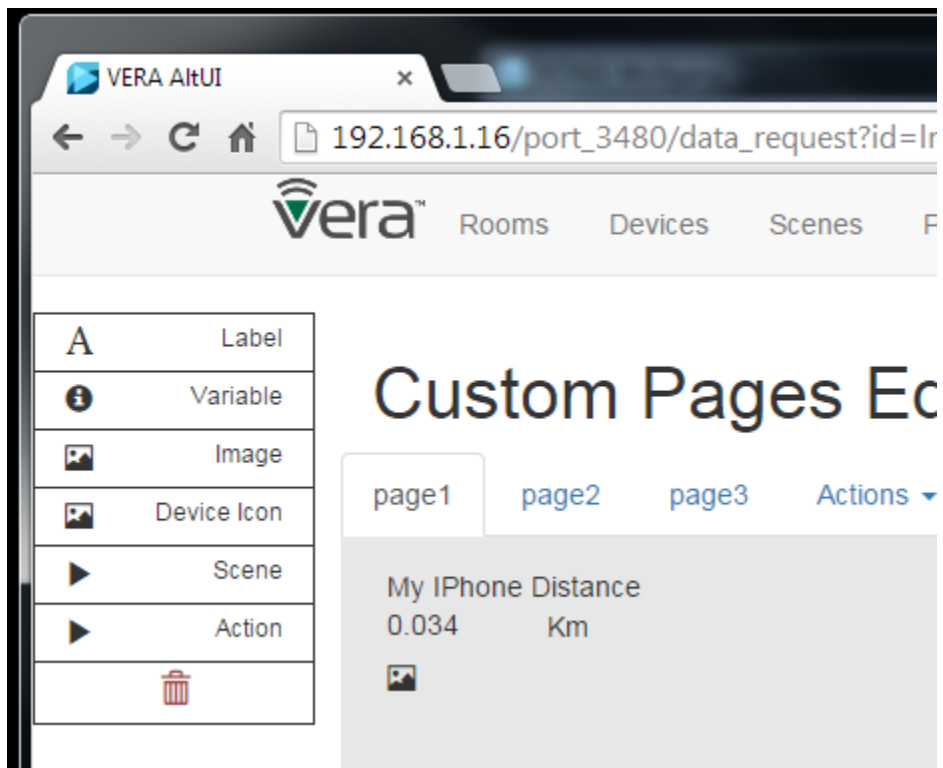
As expected the label has changed on the canvas. I now want to report a dynamic value coming from the device variable, I will use the second tool from the toolbox (the I for Info, which is a variable). I want the distance and the unit which are 2 different variables on this plugin so I will drag and drop 2 “Info” controls. You can move around a control after you have dropped it on the canva surface, just move them around as you want.



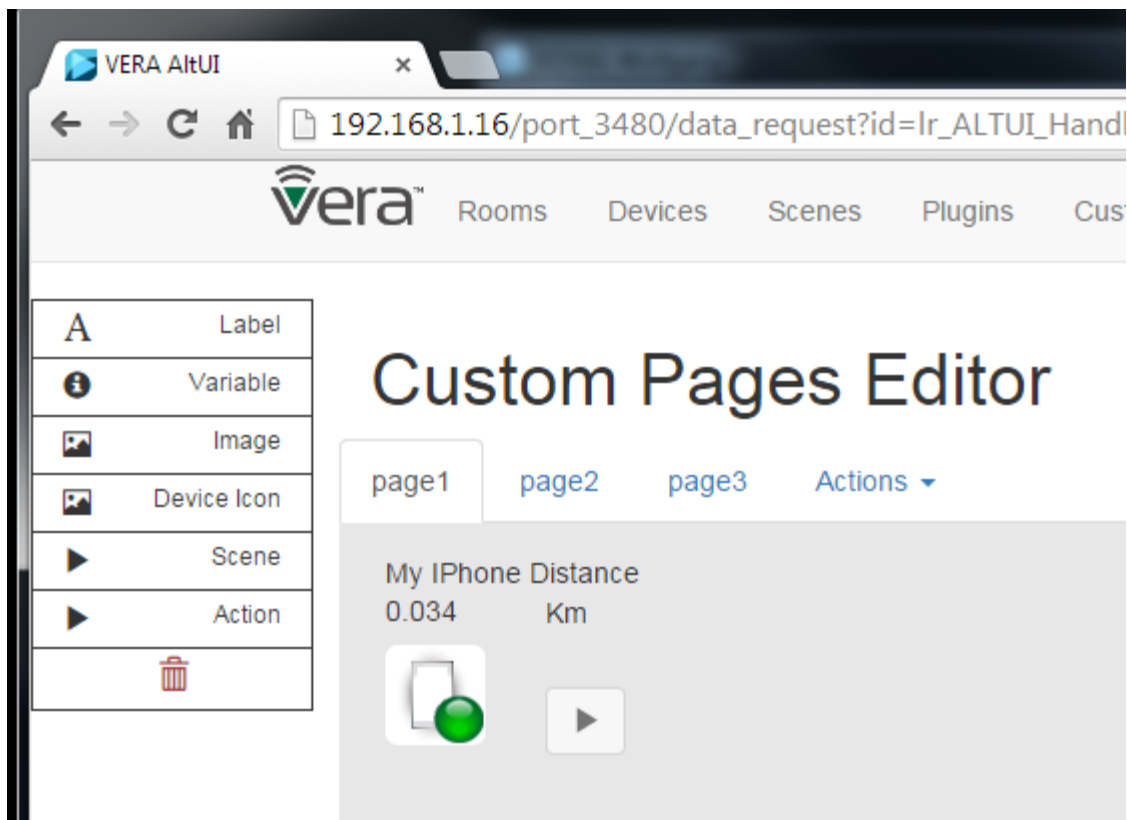
Double clicking on the variable, you can change the parameters so let's now select the right variables.



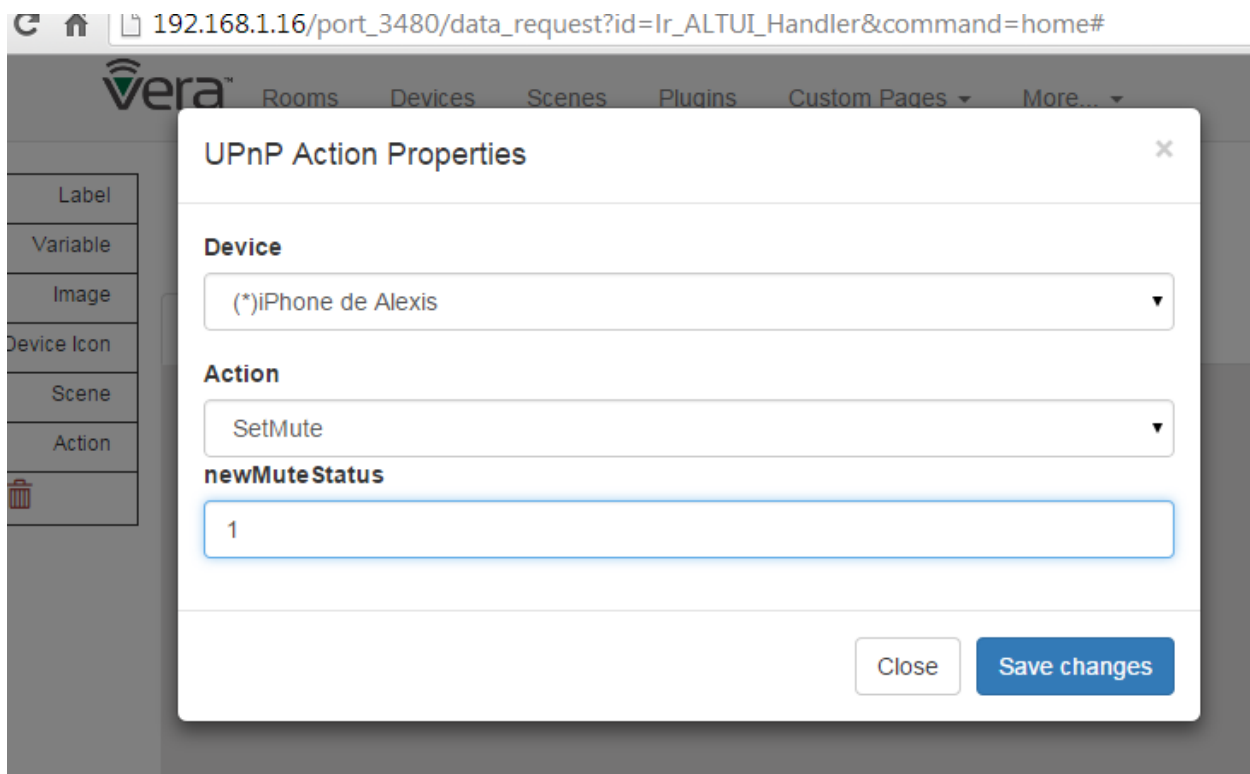
Let's add the icon of the device (which will follow the dynamic states as defined per the plugin author)



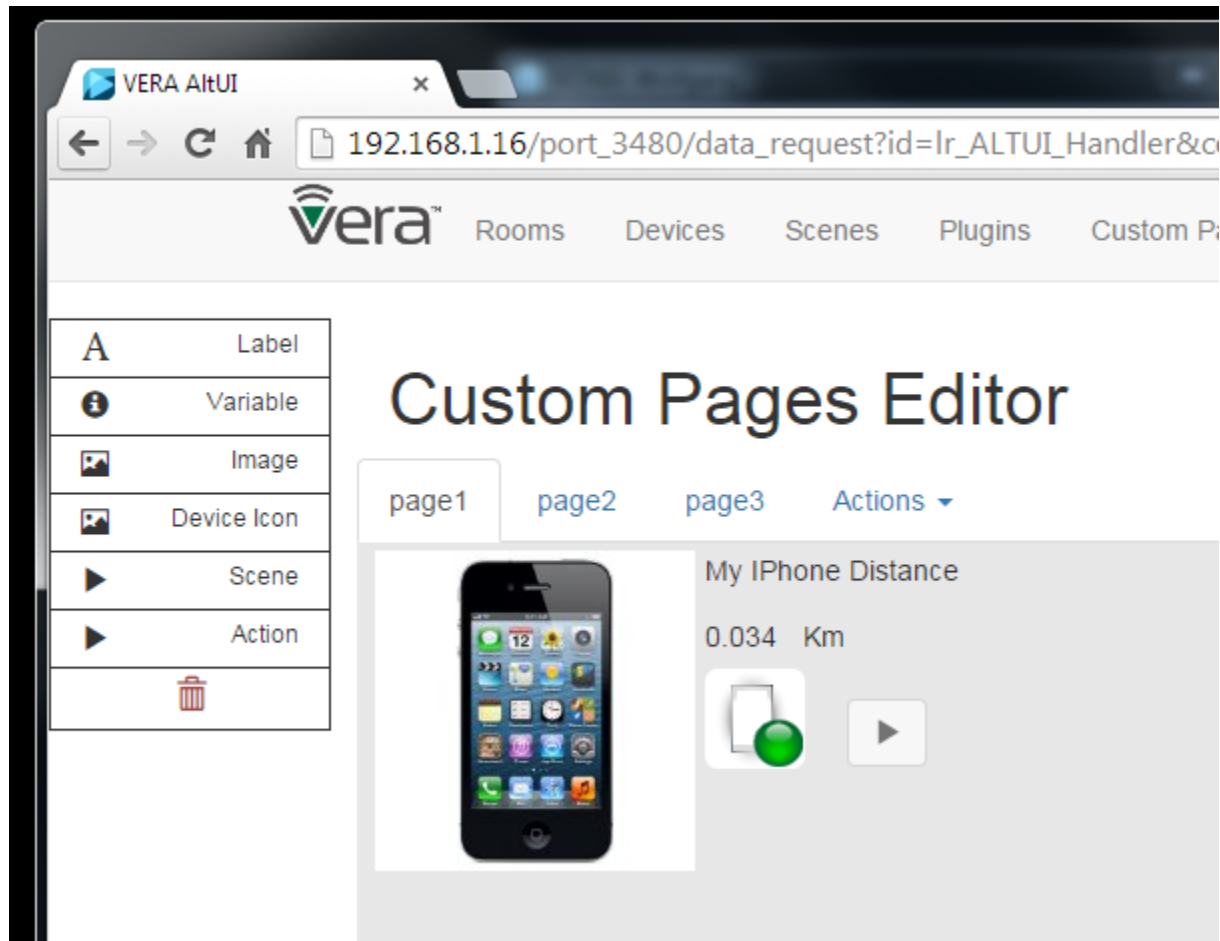
And Let's add a mute button.



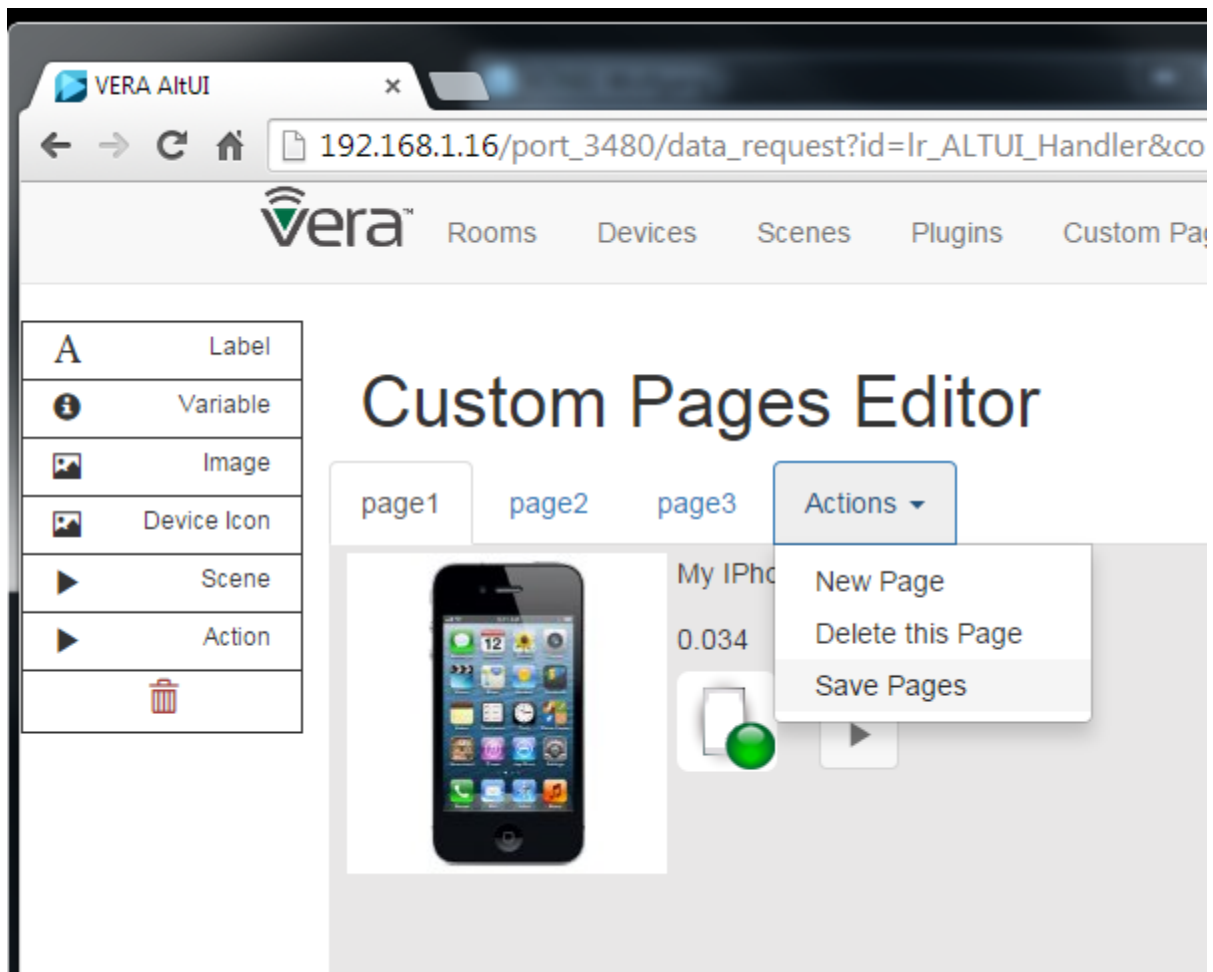
Which we need to configure to run the right UPnP action:



Et voila (with a 3rd tool from the toolbox , image which can be any URL or data uri (for embedded image))

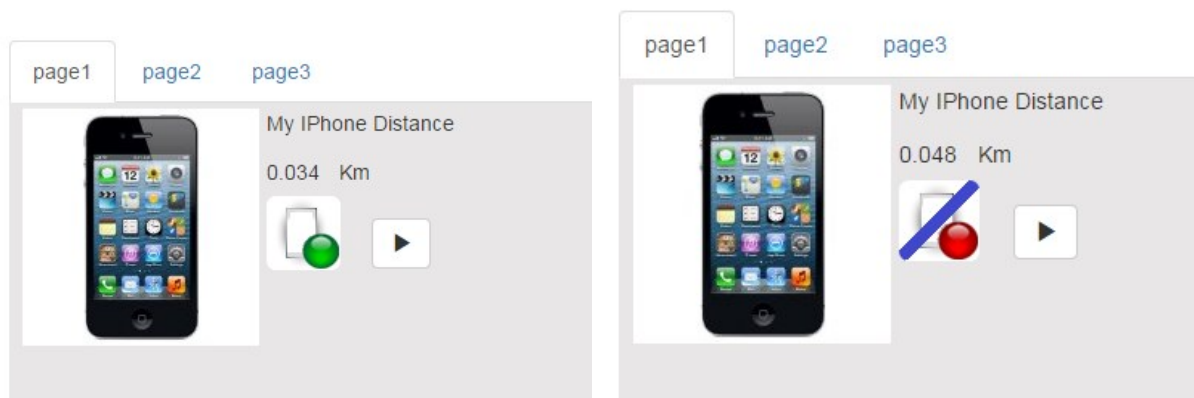


I now want to save it so that it can be persisted and reopened next time so I go into the Actions/Save menu.

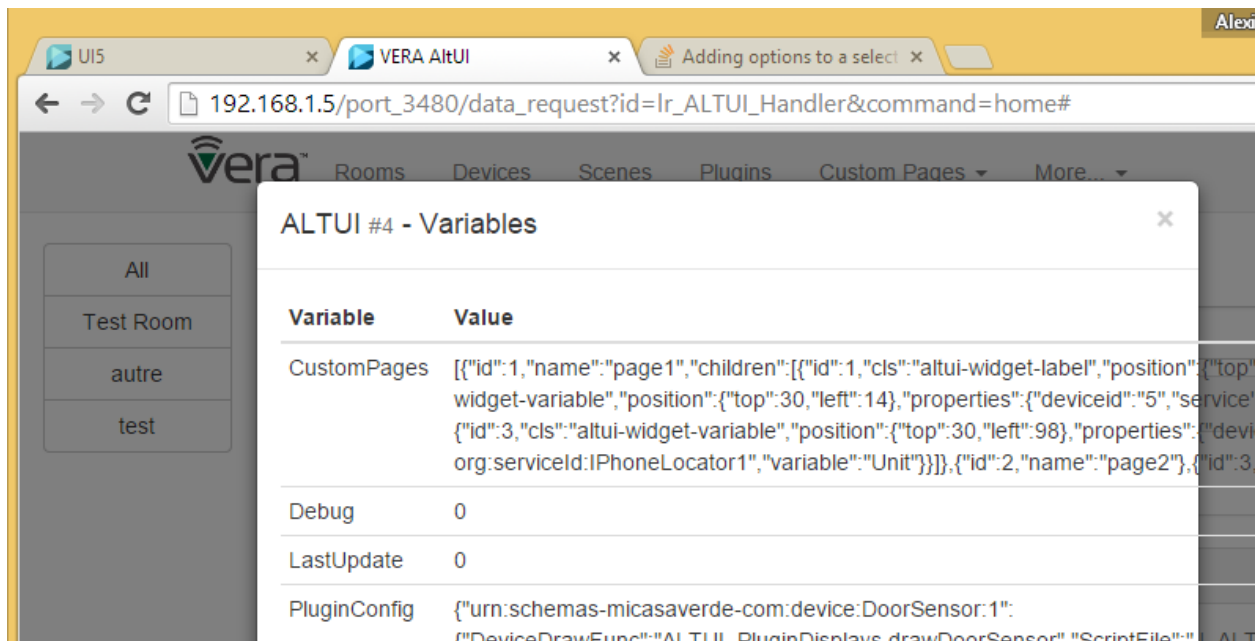


That is it , now the page is visible by the Custom Pages / “Use custom page” menu and you can close your browser and reopen it , it will still be there.

Now I can simply use it in read only mode and the button & icon are functional



All pages definitions are stored in the LUA plugin variable “CustomPages”, you can see it from ALTUI and copy paste in a JSON online viewer if you are interested

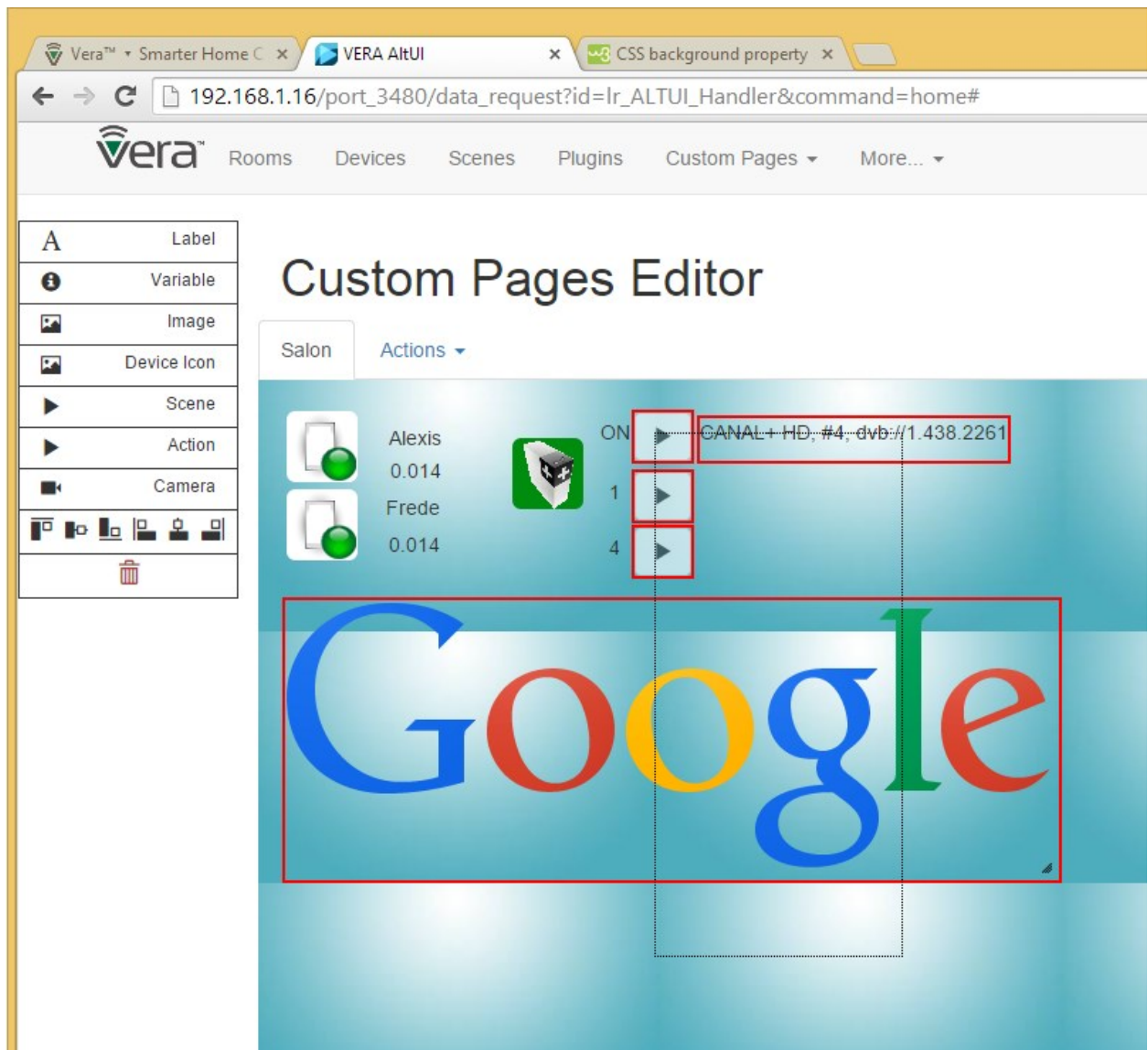


Other capabilities:

- Page Property menu items enables to:
 - Change a page name
 - Change a page background , any valid CSS3 background string is accepted. Solid color, grade, radiants, stripes, url('http://xxxx/image.png') are valid. See the syntax of **"background"** css property

On this Picture you can see various important elements:

- The lasso (dotted line) rectangle enabling the selection of multiple controls. Ctrl+Click is also supported
- The alignments tools in the left tool bar
- The resize handle at the bottom right corner of the image enabling you to size the image.



This will continue to evolve to add some more new tools (which can be VERA related or even something totally different like a google chart gauge or whatever)

- New 2 state button tool
- New Google gauge with customizable min max & color ranges

Vera™ Smarter Home UI5 Alternate UI to UI7 VERA AltUI

192.168.1.16/port_3480/data_request?id=lr_ALTUI_Handler&command=home#

vera Rooms Devices Scenes Plugins Custom Pages Mo


A	Label
i	Variable
🖼️	Image
🖼️	Device Icon
▶	Scene
▶	Action
⏻	Multi State
📹	Camera
📊	Gauge
🗑️	

Custom Pages Editor

Drag and Drop to add/move/delete controls. use Ctrl+Click or lasso

Salon

Actions ▾



Off

Off

Alexis

0.022 Km

▶


On

Frede

20.148Km

▶

On



Ext

10

Piscine

8

Caves

Watts

155

OnOff Button Properties

Device

(*)iPhone de Alexis

Variable

Muted : (urn:upnp-org:serviceId:IPhoneLocator1)

☒ Inverted

OffLabel

Off

Action to switch OFF

SetMute

newMuteStatus

0

OnLabel

On

Action to switch ON

SetMute

newMuteStatus

1

Close

Save changes
















UI5 Installation Instructions (similar for UI7)

PREFERED METHOD:

- Install from store <http://apps.mios.com/plugin.php?id=8246>
- Then override with latest version where xxx is the latest revision number :
http://code.mios.com/trac/mios_alternate_ui/changeset/xxxxx/?old_path=%2F&format=zip

DETAILS

- 1) Upload all these files

Nom	Modifié le	Type
 J_ALTUI_uimgr.js	08/03/2015 17:13	Fichier JS
 J_ALTUI_verabox.js	08/03/2015 16:12	Fichier JS
 L_ALTUI.lua	08/03/2015 15:36	Fichier LUA
 J_ALTUI_utils.js	07/03/2015 00:33	Fichier JS
 J_ALTUI_plugins.js	01/03/2015 19:10	Fichier JS
 J_ALTUI_iphone.js	01/03/2015 15:51	Fichier JS
 D_ALTUI.json	28/02/2015 16:49	Fichier JSON
 J_ALTUI.js	18/02/2015 13:52	Fichier JS
 D_ALTUI_UI7.json	15/02/2015 21:06	Fichier JSON
 I_ALTUI.xml	15/02/2015 18:01	Fichier XML
 S_ALTUI.xml	15/02/2015 18:01	Fichier XML
 J_ALTUI_jquery.ui.touch-punch.min.js	01/02/2015 22:48	Fichier JS
 D_ALTUI.xml	17/01/2015 16:05	Fichier XML
 L_ALTUIjson.lua	17/01/2015 15:41	Fichier LUA
 iconALTUI.png	17/01/2015 15:38	Image PNG

Example:

Upload files

J_ALTUI_utils.js

L_ALTUI.lua

J_ALTUI_plugins.js

J_ALTUI_iphone.js

J_ALTUI.js

I_ALTUI.xml

S_ALTUI.xml

D_ALTUI_UI7.json

D_ALTUI.json

D_ALTUI.xml

☐ Restart Luup after upload

2) Create a device

ONLY DO THIS IF THE DEVICE DOES NOT ALREADY EXIST. If you installed from the store, the device has been created automatically for you

Create device

Device type

Internal ID

Description

Upnp Device Filename

Upnp Implementation Filename

Ip Address

MAC

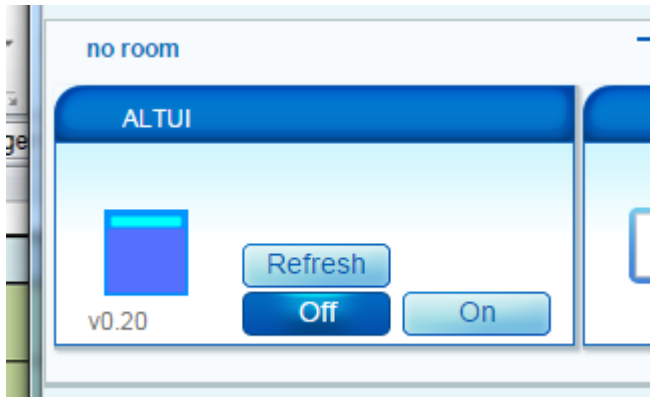
Room

Parent device

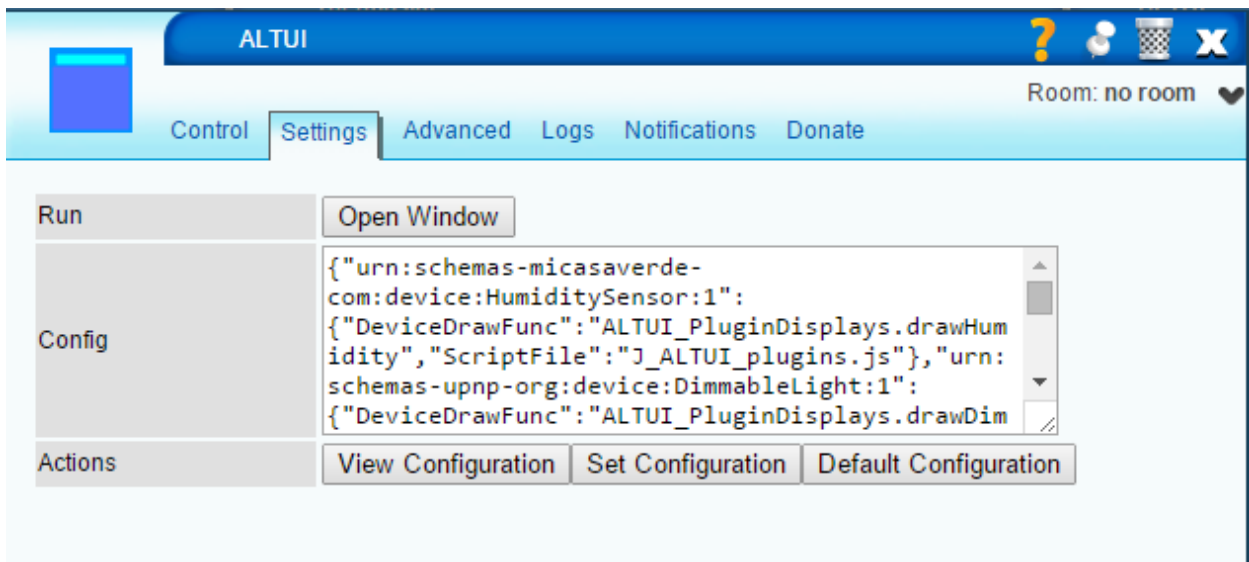
3) Reload lua



4) Find the device in UI5



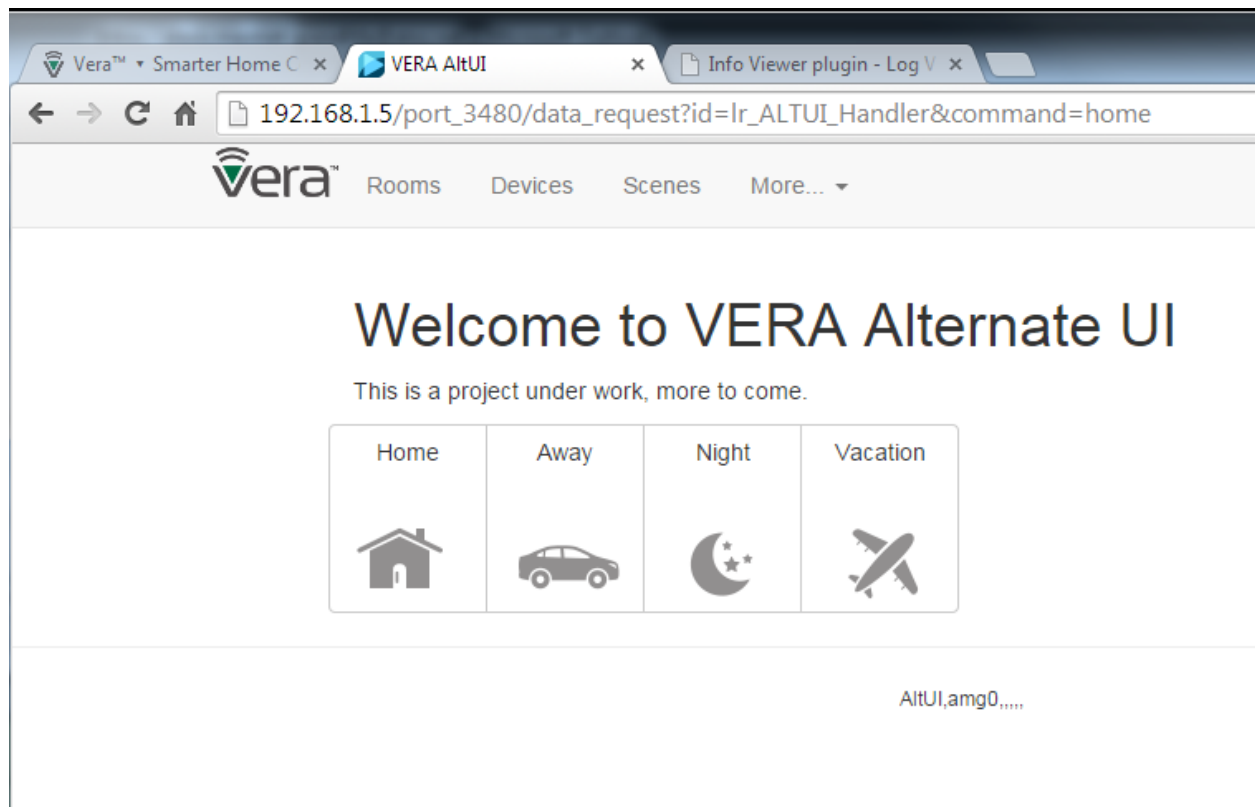
5) Open the settings tab



- View configuration : view the JSON configuration object in a JSON online viewer
- Set configuration : set the ALTUI plugin configuration
- Default : reset to default

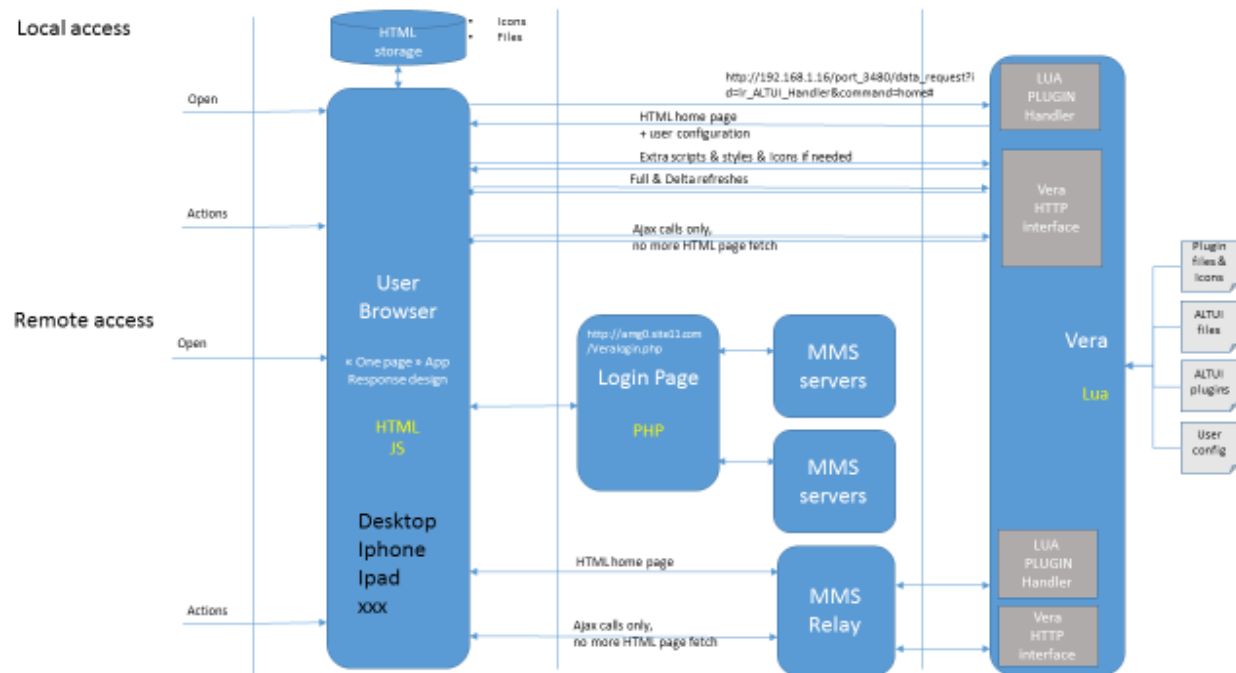
6) Click on open window

- a. "modes" cannot work on UI5 of course but rest should be ok



Architecture and Source Code organization

Data Flows



Extensibility

Mechanisms to extend

- Device dashboard drawing js function
- Device control panel js function
- New pages can be added (Upnp devices, IP devices, Custom user pages , floor plan dashboard , google gauges etc)

Javascript modules for customizable plugins

Plugins drawing are javascript modules providing function code and style css necessary. All the modules are loaded dynamically when needed

Plugins can customize/extend the drawings of device for 2 distinct scenario.

- the small device box on the Device page
- a control panel, dedicated for one device, having almost the full page to play with and display specific device status , controls, drawings etc... (I added this "control panel" feature just in the drop down menu under Variable & Actions items.)

A default implementation is provided for both obviously. Right now the "control panel" one is useless and work in progress but I demonstrate the ability on 2 devices uses a custom control panel function. the Binary Light and the iPhone Locator. The point for me was to explain / demonstrate the extensibility

of the architecture and how it would work. if JS developpers / plugin authors have interest to create a control panel for their device (or some other device), we can integrate their work easily in independent modules

Now a bit on the "how":

- each device type can have a custom javascript file. this is declared in the .LUA file L_ALTUI.lua. The "PluginConfig" LUA device variable contains the JSON object for this configuration and **can be modified to add new plugins.**
- in the configuration, for a given device type you can specify a script file (`["ScriptFile"]="J_ALTUI_plugins.js",`) , a small device box drawing function (`["DeviceDrawFunc"]="ALTUI_PluginDisplays.drawBinaryLight",`) , a full blown control panel drawing function (`["ControlPanelFunc"]="ALTUI_PluginDisplays.drawBinLightControlPanel",`) and a style function for your own CSS (`["StyleFunc"]="ALTUI_PluginDisplays.getStyle",`). All these are optional, default implementation is provided in any case. All these scripts & functions are dynamically loaded and executed when needed by the main page.
- The declared function can be qualified by any number of module name thus enabling to use the javascript module object pattern.
 - o Function can be 'myfunction()'
 - o Or 'myModule.mySubModule.myFunction()' (any depth)
- see examples of Style and drawing functions in J_ALTUI_plugins.js or J_ALTUI_iphone.js

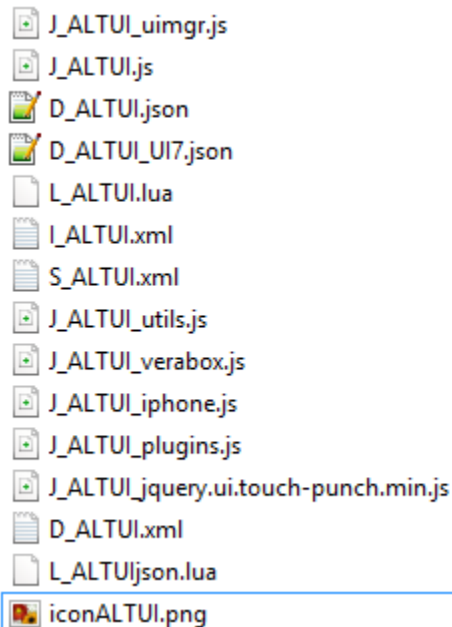
Note:

- the ["DeviceDrawFunc"] takes input parameters like (devid, device) and must return a string which the HTML going into the small device box on the main page.
- the ["ControlPanelFunc"] works slightly different in order to give almost full DOM control to the code writer. It takes input parameters like (devid, device, domparent (jquery based) and must write directly its HTML code into the domparent object (using `$(domparent).append(...)`). it felt more comfortable for the contralPanel function to really write in the DOM as they almost own the full page this time.

Full Source code

all code available on http://code.mios.com/trac/mios_ipx800/browser/trunk/AltUI so feel free to try if you are interested.

Source Files:



- **J_ALTUI_uimgr.js**
 - o Implements the UIManager object. This object is in charge of all drawing on pages
 - Error messages
 - Device Drawing (default & custom) – manages the loading of JS files needed. Evaluate Icon conditions based on existing UI5 or UI7 descriptions.
 - Scene Drawing (and editor)
 - Refresh UI (when new data is arriving)
 - The main entry points (pagesxxx() function per each page of the app)
 - House mode on UI7 only . the LUA plugin tells the application if we are on UI5 or UI7
 - o It maintains internally the cache for device type information (json, Upnp descriptions etc)
- **J_ALTUI.js**
 - o The classical JS for the setting page of the UI5 or UI7 plugin
- **D_ALTUIxx.json**
 - o The classical JSON files for the UI5 or UI7 plugin
- **L_ALTUI.lua**
 - o The main plugin lua code
 - o It is seldomly used, just to register a handler to act as a web server serving initially the first home page layout, and eventually responding to a few (one so far) ajax call from the client. The idea is to not use it as much as possible to offload the application work on the client side as explained in the initial project rules. VERA is small, our PC are big
 - o It will act as the data persistent place where configuration and (in the future) user custom pages descriptions are stored & saved as device variable. UI7 can store and

display JSON in its variable. UI5 has trouble to display it in the advanced tab as the string are not htmlEncoded but this is not a problem; we can manage this display & edit in the plugin JS setting page.

- **I_ALTUI.xml , S_ALTUI.xml**
 - Classical device files.
 - A Reset uPNP action is implemented to restore configuration to default
 - In the future, we may need a few to manage user custom pages, not sure yet
- **J_ALTUI_utils.js**
 - Global utilities like string.format() addition, or string.htmlEncode(), htmlDecode() addition to the string prototype
 - CSS Styles required by the application are managed here and injected dynamically (avoid having to change the .LUA file and reloading every time)
 - It initializes the application by launching the Init() for the UIManager object and the VeraBox object
- **J_ALTUI_verabox.js**
 - Implements the communication with VERA
 - The UPnpHelper module
 - facilitates building of URL (get set variables, run upnp, all the VERA Http calls basically including the HAG SOAP one)
 - Provides facilities for plugin author like simple SetOnOff() , SetArm() methods
 - The FileDB module
 - A cache of dynamically loaded files (D_xx files S_xx files, or whatever). Key is the file name.
 - In the future, I intent to use HTML5 persistent storage to cache content on a even longer term basis (even when user closes the browser)
 - The VeraBox module
 - The core data load engine.
 - Manages getting the user_data and status_data using the loadversion versioning and various optimizations documented
 - Manages all information in a cache to not load it twice
 - Highly asynchronous, code executed in callback methods instead of waiting
 - getWeatherSettings : _getWeatherSettings,
 - getBoxInfo : _getBoxInfo,
 - getLuaStartup : _getLuaStartup,
 - getRooms : _getRooms, // in the future
getRooms could cache the information and only call _getRooms when needed
 - getDevices : _getDevices,
 - getDeviceByID : _getDeviceByID,
 - getScenes : _getScenes,
 - getSceneByID : _getSceneByID,

- getPlugins : _getPlugins,
 - getHouseMode : _getHouseMode,
 - setHouseMode : _setHouseMode,
 - getStatus : _getStatus,
 - getStates : _getStates,
 - evaluateConditions : _evaluateConditions, // evaluate a device condition table (AND between conditions)
 - deleteRoom : _deleteRoom,
 - runScene : _runScene,
 - deleteScene : _deleteScene,
 - reloadEngine : _reloadEngine,
 - setStartupCode : _setStartupCode,
 - setScene : _setScene,
 - getCategoryTitle : _getCategoryTitle,
 - getDeviceTypes
 - initEngine()
- **J_ALTUI_IPhone.js**
 - The custom drawing functions for the IPHONE locator plugin and the French Canal Plus control Plugin
 - Dynamically loaded when/if needed and configured in the LUA "PluginConfig" table to be loaded
 - **J_ALTUI_Plugins.js**
 - Same but for all the out of the box devices provided by VERA (bin lights, motion, temp sensors, etc)
 - **J_ALTUI_jquery.ui.touch-punch.min.js**
 - A small jquery 3rd party to make the ipad/iphone/ touch screen device compatible with the click event () so that touchend event can be used as a mouse click

Basic rules for developers:

I most welcome any programmers help in this project if they are interested in submissions. The rules are simple,

- use bootstrap grid model (row / cols) for full responsive design, I d like to keep it running from desktop to ipad to iPhone 4S screen !
- minimize additional JS framework : I am trying to use bootstrap, jquery, jqueryUI and that's all. probably would like a try to add a lib for graphic widgets (gauges etc) also in the near future, suggestion welcome. must be easy and working with jquery as I am far from a JS guru

- use JS module pattern (same as UI7) see example in the various modules. prefix private function with a '_' and public function with a naming convention
doSomethingToSomethingElse()
- all CSS class: try to always use the prefix : altui-xxx-xxx etc
- avoid synchronous call when possible (always possible almost)