

# DS Audio Voice control (Tasker; Autoinput; Autovoice)

Version 0.3 - 02/02/2019

Here is a first version of scripts for voice control of the DS Audio application. I made them for my own needs, but I hope they can serve others. Even better, some of you may be able to improve them (I'm not a top programmer !).

## **First :**

These scripts require the installation of the Tasker application and the AutoVoice and AutoInput plugins. In addition, "AutoVoice Google Now Integration" must be enabled in the accessibility settings of Android.

DS Audio connection settings must already be populated with the "remember me" option checked.

Then import the "DS Audio EN" Profile and the "DS Audio Config EN", "DS Audio Start", "Detect End Loading Text", "Detect End Loading Popup" and "Audio Request" tasks in Tasker.

## **How to use :**

Voice commands are retrieved by the Google Assistant and forwarded to Tasker through the AutoVoice Plugin. So you have to say "ok google" before each request. Here are the commands that can be used:

- Ask autovoice to play the song <song>
- Ask autovoice to play the album <album>
- Ask autovoice to play the artist <artist>
- Ask autovoice to play the song <song> from the artist <artist>
- Ask autovoice to play the song <song> from the album <album>
- Ask autovoice to play the song <song> from the album <album> from the artist <artist>
- Ask autovoice to play the radio <radio\*>
- Ask autovoice to play a random playlist
- Ask autovoice to play the next song
- Ask autovoice to play the previous song

*\* Limited to the radios from the first page because i don't need more. Tell me if you need more.*

To note, the DS audio application presents several peculiarities which did not facilitate the task:

- (1) At each launch, we do not always arrive on the same page (because that application is not stable and often crashes)
- (2) The audio output tends to reset itself, and the setup popup appears randomly at each start.
- (3) All the "Back" navigation buttons are not well identified in the application, and are not necessarily recognized by "Autoinput"

- (4) Searching for a song requires clicking on the "Enter" button on the keyboard after the entry, or automation of validation is not handled in the AutoInput Plugin on non-rooted phones

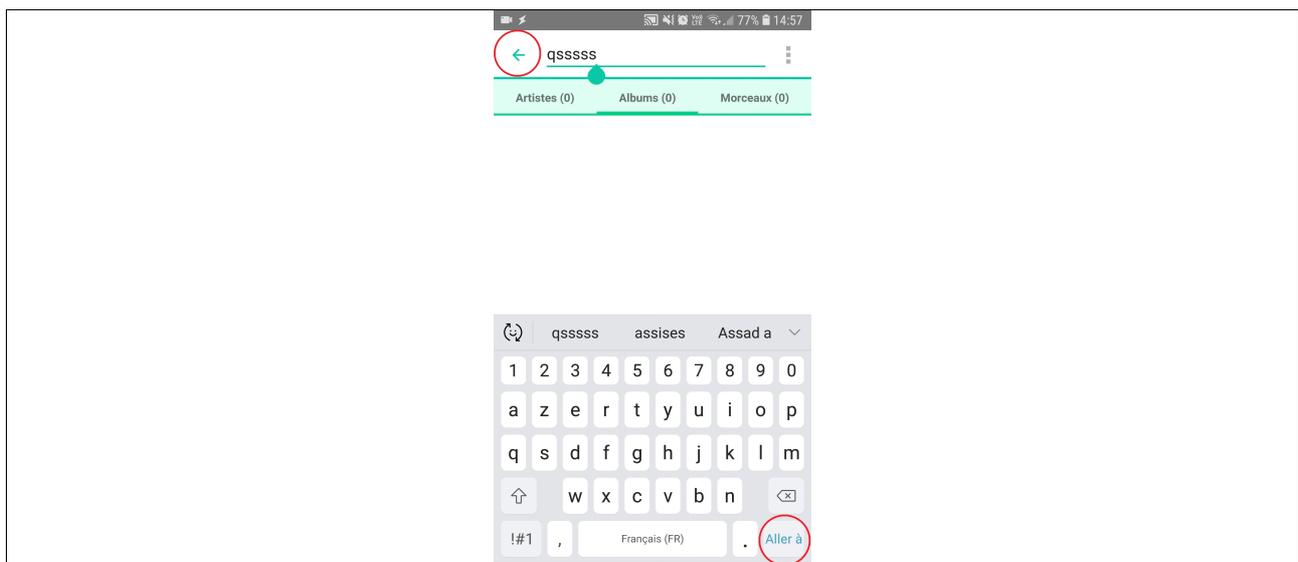
To solve the point (1), I started on the following way (quite heavy): At each launch of a user request, the script starts the application and place always it at the same page: Current song page with a menu accessible directly when you click on the back button.

(2) After each start, a check of the selected audio output is made.

For points (3) and (4), I had to use the x, z coordinates of the buttons to click on and not their identifier. As a result, it will be necessary to adjust some coordinates in the parameters of the application if it is necessary (see below).

### **"DS Audio Config" task configuration:**

If the script does not work, make sure the x, y coordinates of the "Back" and "Enter" buttons are corrects in the "DS Audio Request" task. These are the variables identified as "**PosBtnEnter**" and "**PosBtnReturn**". Their coordinates must correspond to the buttons circled in red below (it is possible to obtain them by creating an "AutoInput Action" task in Tasker and by clicking on it).



Put the audio output to use in the variable "**AudioOutput**"

### **Customizing the language used, or commands**

It is possible to adapt the script for other languages.

- Modify the first part of the text in the "DS Audio" profile: Replace the "play" text with the equivalent of another language
- Change the variables used for speech recognition in the "DS Audio Config" task:

> Ask autovoice to play [the song] <song>

The vocal identifiant [the song] is associated with %IdVocalSong variable

> Ask autovoice to play [the album] <album>

The vocal identifiant [The album] is associated with %IdVocalAlbum1 variable

> Ask autovoice to play [the artist] <artist>

The vocal identifiant [the artist] is associated with %IdVocalArtist1 variable

> Ask autovoice to play [the song] <song> [from the artist] <artist>

The vocal identifiant [from the artist] is associated with %IdVocalArtist2 variable

> Ask autovoice to play [the song] <song> [from the album] <album>

The vocal identifiant [from the album] is associated with %IdVocalAlbum2 variable

> Ask autovoice to play [a random playlist]

The vocal identifiant [a random playlist] is associated with %IdVocalRandom variable

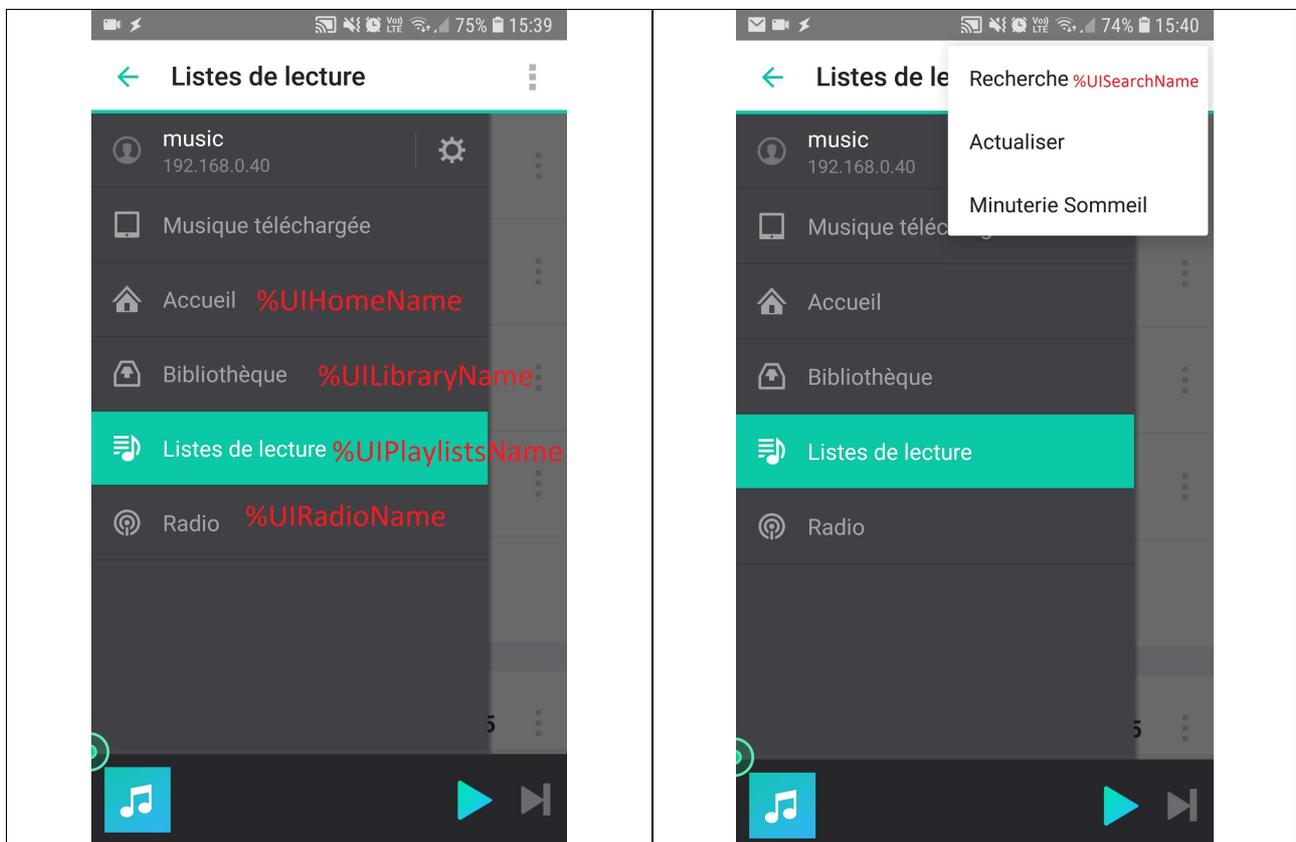
> Ask autovoice to play [the next song]

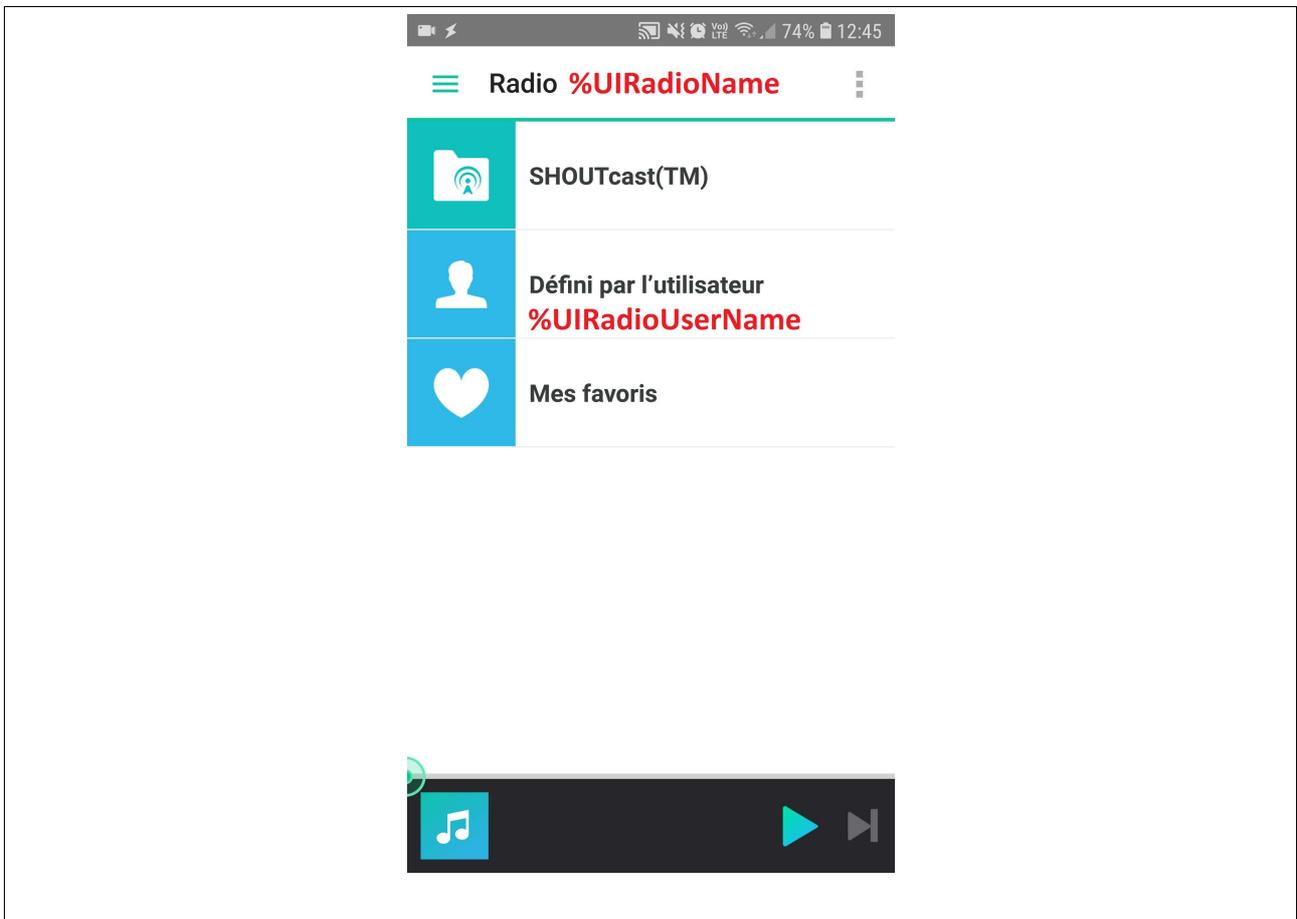
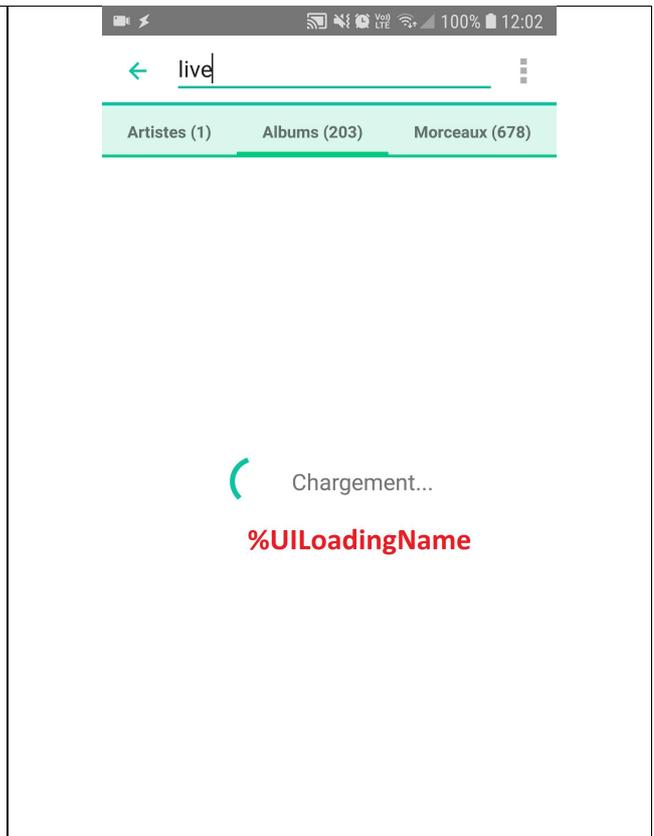
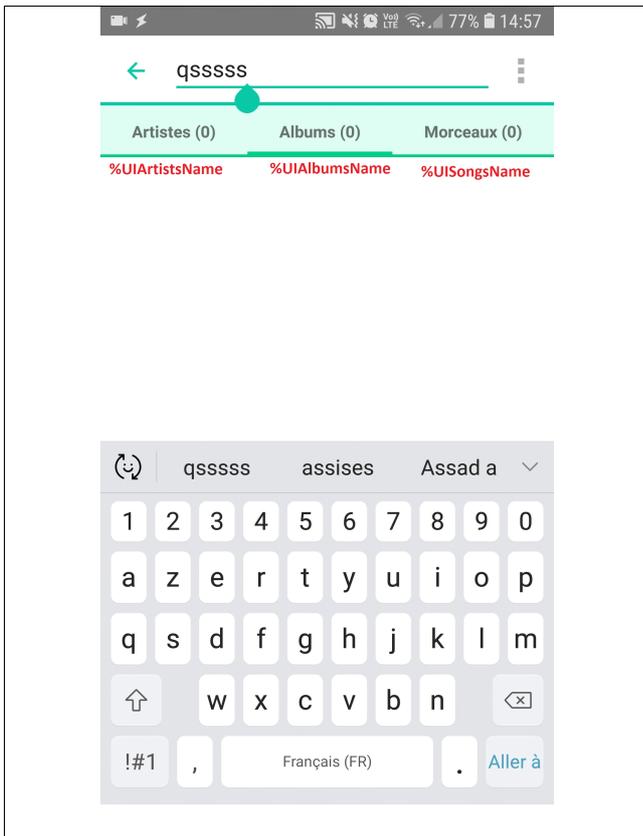
The vocal identifiant [the next song] is associated with %IdVocalNextSong variable

> Ask autovoice to play [the previous song]

The vocal identifiant [the previous song] is associated with %IdVocalPreviousSong variable

- Modify all the texts that appear in the application, for that modify the variables starting with "UI". Here below are the correspondences.





## Tests

The proper functioning of the scripts can be tested by adding the desired query in the variable %request in the task "DS Audio Request"

Exemple :

%request = the song [sweet child](#) from the artist [guns](#)