Demo Application UI Guide (Audio Player)

Version 0.4.0

Display Audio

Solution Team

Release information

The following changes have been made to this document.

Change History

Date	Change
06 Dec 2017	First release for v0.1.0
15 Dec 2017	Update Storage Event v0.2.0
07 Feb 2018	Update scenario v0.3.0
14 Nov 2018	PlayList null point fix v0.4.0 Storage Remove fix v0.4.0. Player library null point fix v0.4.0.

Proprietary Notice

Information in this document is provided solely to enable system and software implementers to use Nexell products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Nexell reserves the right to make changes without further notice to any products herein.

Nexell makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Nexell assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in Nexell data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. Nexell does not convey any license under its patent rights nor the rights of others. Nexell products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Nexell product could create a situation where personal injury or death may occur. Should Buyer purchase or use Nexell products for any such unintended or unauthorized application, Buyer shall indemnify and hold Nexell and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Nexell was negligent regarding the design or manufacture of the part.

Contact us

[11595] BundangYemiji Bldg. 12F, 31 Hwangsaeul-ro 258 beon gil, Bundang-gu, Sungnam-city, Gyeonggi-do, Korea.

TEL: 82-31-698-7400

FAX:82-31-698-7455

http://www.nexell.co.kr

Contents

Chap 1. Overview **1**

1.1 Overview 1

1.2 UI Guide 1

1.2.1 Features 1
1.2.2 Execute 1
1.2.3 UI Details 2
1.2.4 UI Playlist Details 2

1.3 Flow 3

1.3.1 Application Overall 3
1.3.2 Application Switching 3

Chap 2. GUI Guide **4**

2.1 Status Bar 4

2.1.1 Home 4
2.1.2 Back 4

2.2 Media Player Control 4

2.2.1 Initialization 4
2.2.2 Progress Bar 5
2.2.3 Prev Button 5
2.2.4 Play Button 6
2.2.5 Pause Button 6
2.2.6 Next Button 7
2.2.7 Stop Button 7
2.2.8 Playlist Button 8

Chap 3. Storage Event **9**

3.1 OverView 9

3.2 Flow 9

3.2.1 Removing Storage Event 9
3.2.2 Inserting Storage Event 9

Chap 4. Known Issues **10**

4.1 To Do List 10

Chap 1. Overview

1.1 Overview

This document describes how to use NxAudioPlayer application.

NxAudioPlayer makes audio file list by using sqliteutils that reads database written by media scanner.

NxAudioPlayer also makes xml file that stores last playing information to play again from last state.

1.2 UI Guide

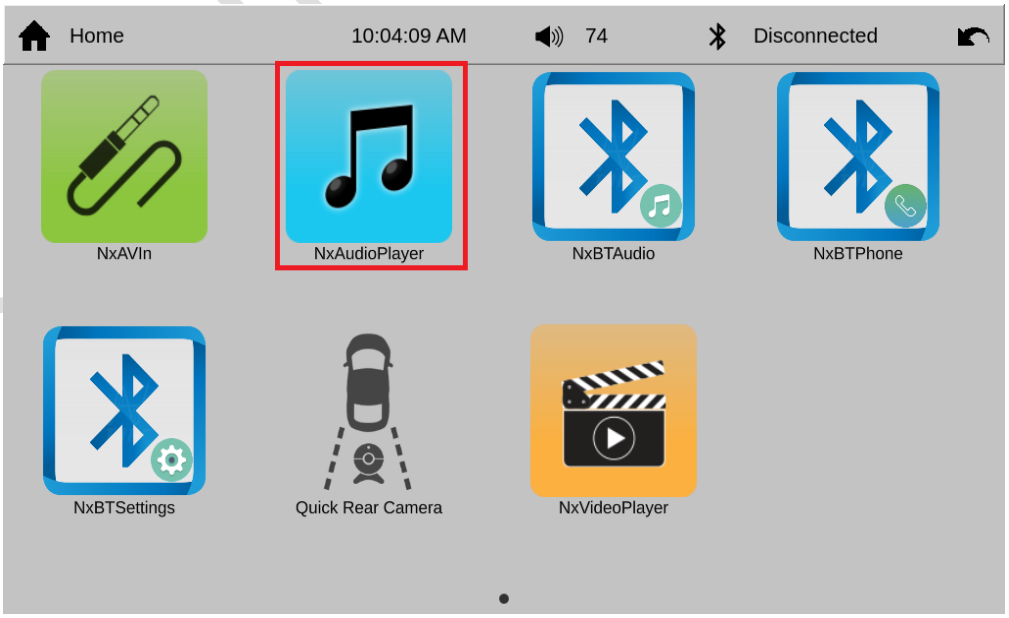
1.2.1 Features

- Audio file play, seek, pause, and resume
- Make audio file list including USB and SD-card.
- Auto play from last state, if available.
- Shows media information such as artist, thumbnail, and so on, if available.

1.2.2 Execute

NxAudioPlayer is executed by clicking icon in the launcher.

When some application is clicked, launcher hides from screen and clicked application appears.

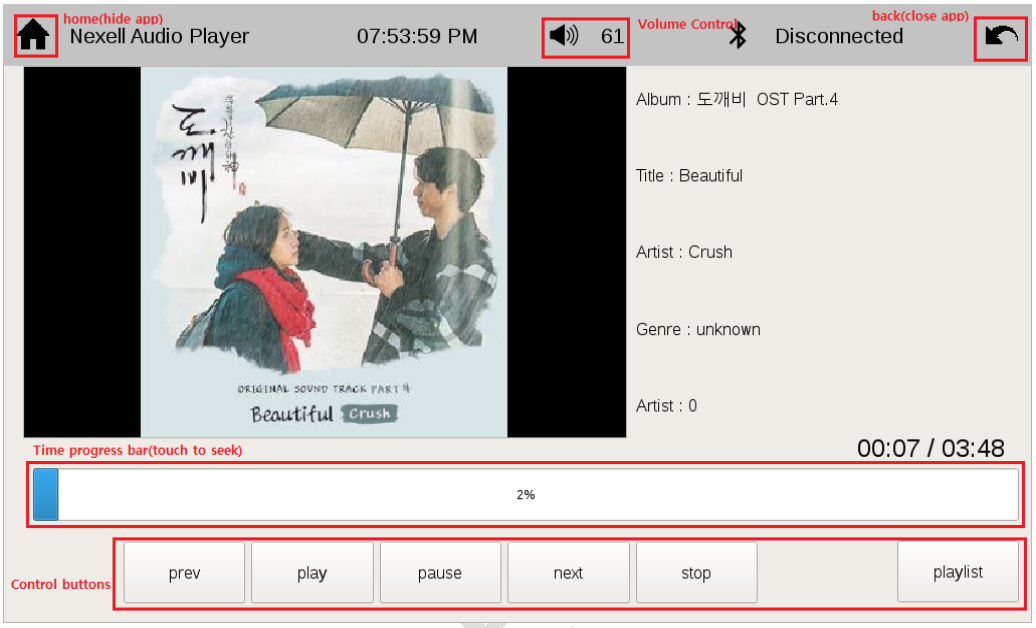


1.2.3 UI Details

GUI is shown like below picture after NxAudioPlayer is executed.

home (hide app) does not mean application window operates hide.

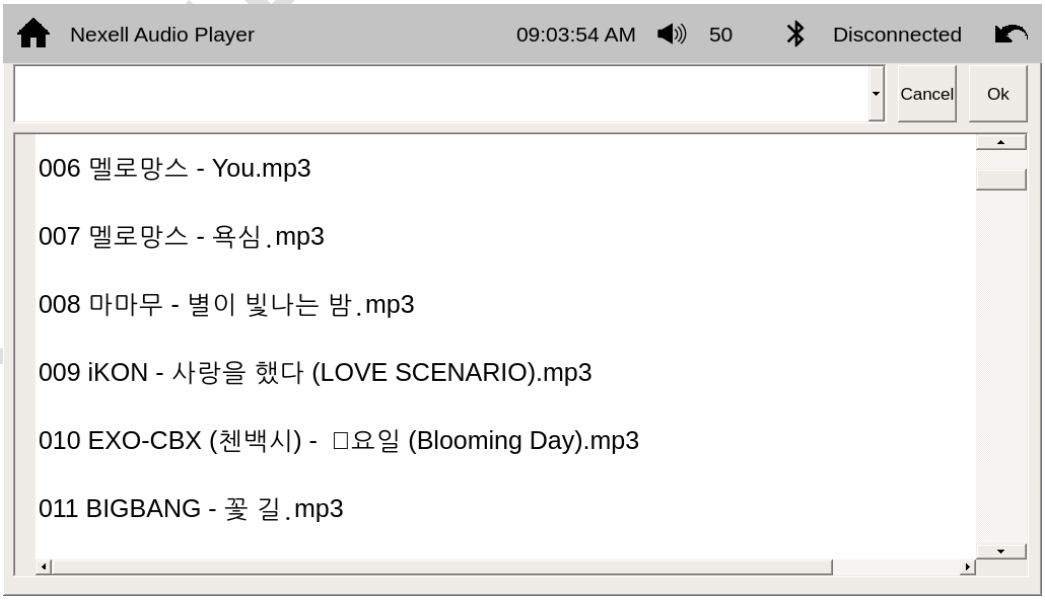
NxAudioPlayer disappears behind other application by changing Z-order.



1.2.4 UI Playlist Details

Playlist UI is shown by clicking playlist button.

List can be scrolled down, and items in list can be played by double-clicking or select(item click) and Ok button.



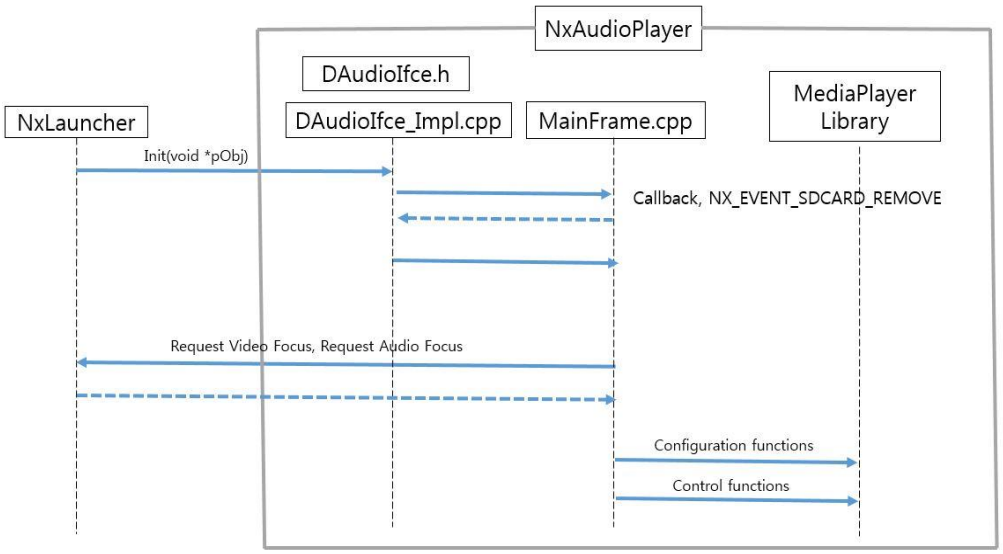
1.3 Flow

1.3.1 Application Overall

NxAudioPlayer communicates through NxLauncher.

RequestVideoFocus is needed, when application wants to appear on screen.

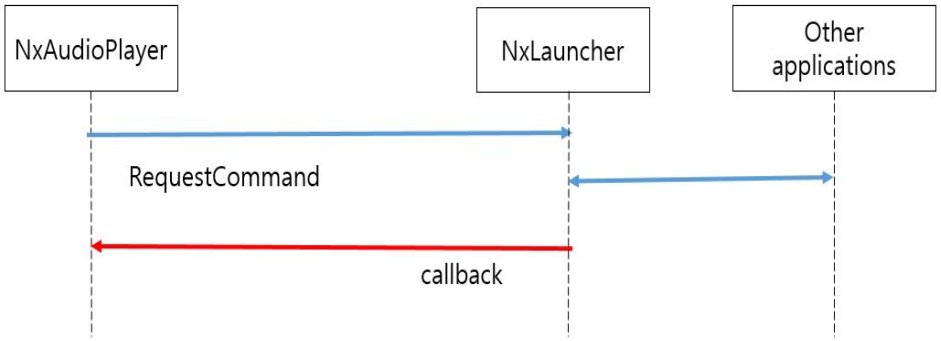
In particular, to use an Audio Device, must return RequestAudioFocus.



1.3.2 Application Switching

Each application communicates with NxLauncher.

NxLauncher is used for changing Z-order of application



Chap 2. GUI Guide

2.1 Status Bar

NxStatusBar is used for Status Bar.

2.1.1 Home

When Home button clicked, current application disappears by changing launcher's Z-order to top.

In the home button callback function of NxStatusBar, NxAudioPlayer uses NxLauncher to request to display the program. .

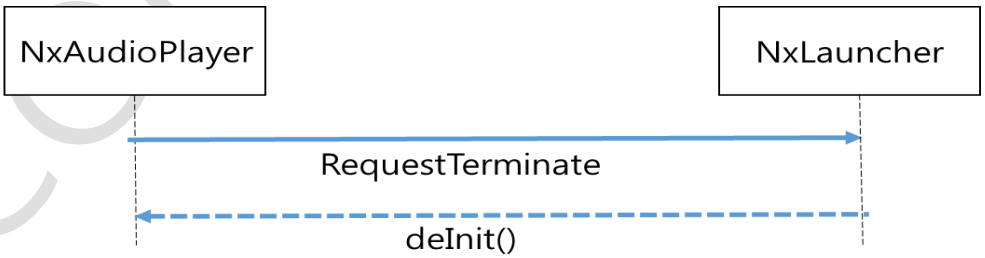
2.1.1.1 Flow

2.1.2 Back

When Back button clicked, NxAudioPlayer closes, and other application(previously top) appears.

In the nearest routine, NxAudioPlayer uses NxLauncher to signal loss of audio focus, loss of video focus, and process removal.

2.1.2.1 Flow



2.2 Media Player Control

In NxAudioPlayer application, Media Player library is wrapped by CNX_MoviePlayer class.

2.2.1 Initialization

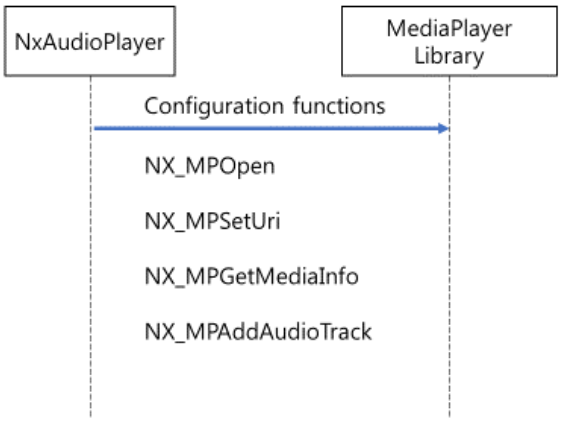
NxAudioPlayer must get Audio Focus successfully before initialize Media Player lib.

Media Player Initialization is done by CNX_MoviePlayer->InitMediaPlayer

InitMediaPlayer function uses Media Player library's Configuration functions in following order, NX_MPOpen, NX_MPSetUri, NX_MPGetMediaInfo, and NX_MPAddAudioTrack.

All Media Player library functions must return MP_ERR_NONE.

2.2.1.1 Flow

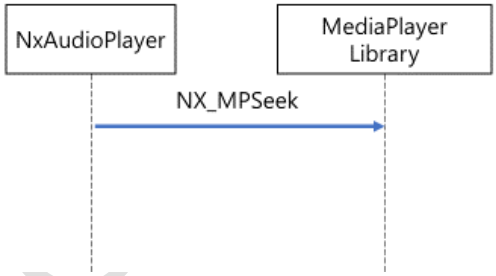


2.2.2 Progress Bar

When Progress Bar is clicked, Bar UI and audio position are changed to corresponding value.

NX_MPSeek in Media Player library is used.

2.2.2.1 Flow



2.2.3 Prev Button

When prev button is clicked, NxAudioPlayer stops playing and plays previous audio file.

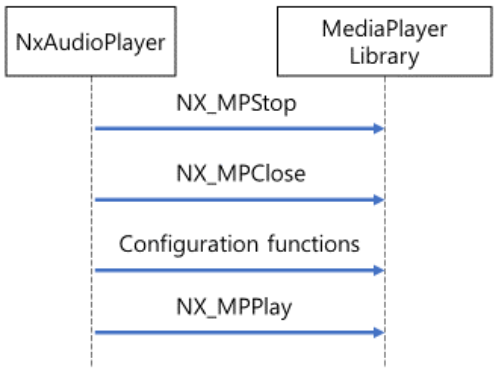
Index of audio file list is set to previous audio file.

NX_MPStop and NX_MPClose in Media Player library are used.

Initialization with previous audio file like above explanation.

NX_MPPlay in Media Player library is used after Initialization is succeed.

2.2.3.1 Flow



2.2.4 Play Button

When play button is clicked, NxAudioPlayer plays audio file.

In NxAudioPlayer application, play button sequence contains Initialization of Media Player library.

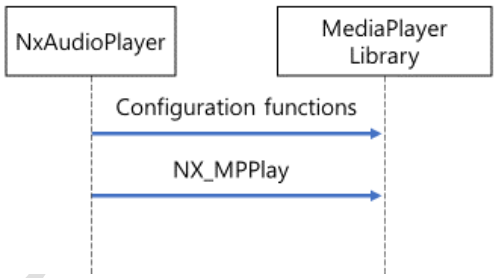
Just after Initialization of Media Player lib is done, NxAudioPlayer updates UI information such as, progress bar, thumbnail, and so on.

If Initialization is failed, NX_MPClose in Media Player library is called and Initialization is tried again with next audio file in list.

When NxAudioPlayer is already playing audio file, play button does nothing.

When NxAudioPlayer is in pause, play button just call NX_MPPlay without Initialization.

2.2.4.1 Flow

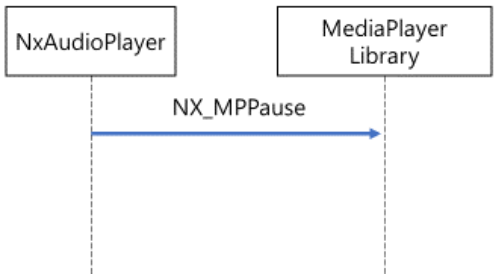


2.2.5 Pause Button

When pause button is clicked, NxAudioPlayer pauses audio playing.

NX_MPPause in Media Player library is used.

2.2.5.1 Flow



2.2.6 Next Button

When next button is clicked, NxAudioPlayer stops playing and plays next audio file.

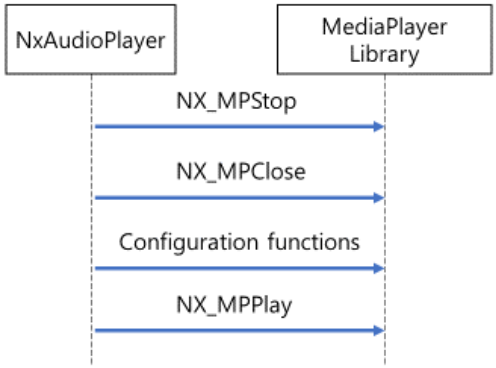
Index of audio file list is set to next audio file.

NX_MPStop and NX_MPClose in Media Player library are used.

Initialization is done with next audio file like above explanation.

NX_MPPlay in Media Player library is used.

2.2.6.1 Flow

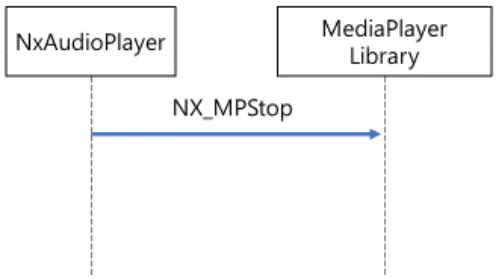


2.2.7 Stop Button

When stop button is clicked, NxAudioPlayer stops playing audio.

NX_MPStop in Media Player library is used.

2.2.7.1 Flow



2.2.8 Playlist Button

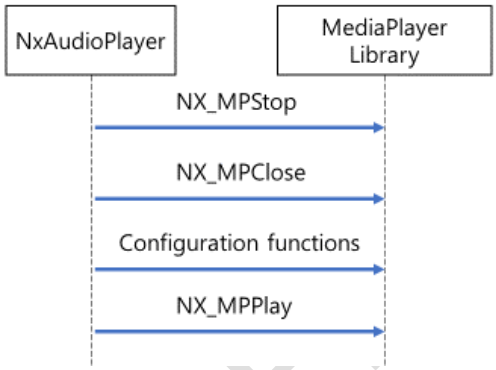
When playlist button is clicked, NxAudioPlayer shows playlist UI.

If NxAudioPlayer was playing audio, it continues until some audio file is selected.

If file is selected, NxAudioPlayer stops playing audio and plays selected audio file.

2.2.8.1 Flow

The case that file is selected.



Chap 3. Storage Event

3.1 OverView

NxAudioPlayer detects storage events.

Used storage events are removing external storage(USB, SD-card) and media scan done.

Media scan done event is received, whenever storage status is changed such as inserting and removing.

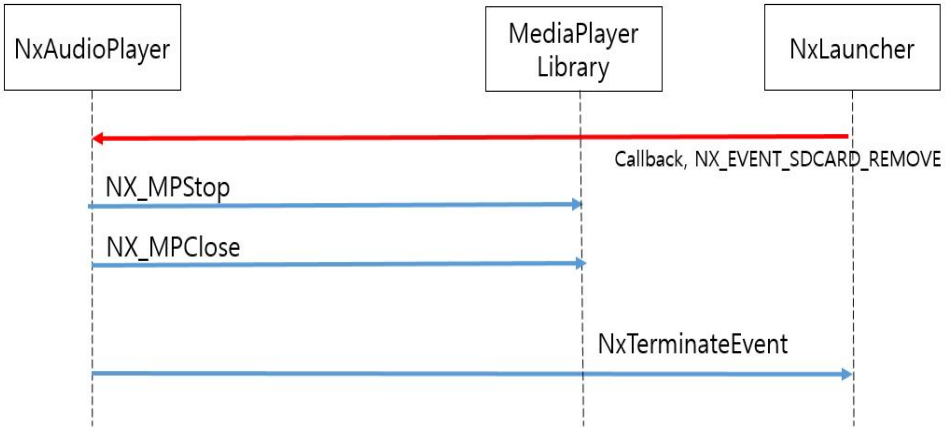
If removing storage event is received, NxAudioPlayer is closed.

If media scan done event is received, NxAudioPlayer refreshes media file list.

3.2 Flow

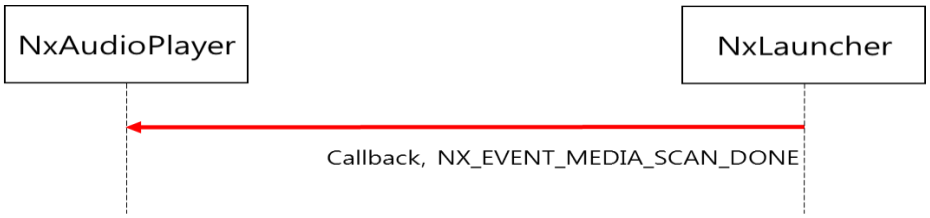
3.2.1 Removing Storage Event

NX_EVENT_SDCARD_REMOVE and NX_EVENT_USB_REMOVE are used.



3.2.2 Inserting Storage Event

When NX_EVENT_MEDIA_SCAN_DONE is received, media file list is refreshed.



Chap 4. Known Issues

4.1 To Do List

Supporting Multi Language for audio information and playlist UI.