Wisenet Open Platform v4.02

Web Page Development Guide

v4.02

2022-12-16

Contact Information

HANWHA TECHWIN Co., LTD.

Hanwhatechwin R&D Center, 701, Sampyeong-dong, Bundang-gu, Seongnam-si, Gyeonggi-do, Korea, 463-400

TEL: +82-70-7147-8740~60 FAX: +82-31-8018-3745

https://step.hanwha-security.com/

HANWHA TECHWIN AMERICA Inc.

100 Challenger Road Ridgefield Park, New Jersey, 07660 U.S.A.

HANWHA TECHWIN EUROPE LTD.

2nd Floor, No. 5 The Heights, Brooklands, Weybridge, Surrey, KT13 0NY, U.K.

Hanwha Techwin

Revision History

The table below provides the version information and revision history of this document.

Please refer to 2.0x documents for checking revision history before 3.00.

Version	Date	Description
4.02	2022-12-16	Added new version number
4.01	2021-05-31	Deleted ChangeApplyState and pluginPlay
4.00	2020-08-31	Added new version number
3.60	2019-12-20	Added new version number
3.52	2019-09-02	Added new version number
3.51	2019-06-10	Added new version number
3.50	2018-10-29	Added new version number
3.00	2017-04-14	New revision

Table of Contents

Requirements

To create an application using the SDK, a developer needs the following software and hardware.

- Linux, or Windows
- Wisenet Open Platform SDK
- Hanwha IP camera that supports Wisenet Open Platform

Functions

These functions are defined in the camera to request information or to control 3rd-party applications. Please download home/js/SDKApi.js for the details on functions.

RequestAjaxMsg

Define

RequestAjaxMsg function

Method	Parameters
RequestAjaxMsg	msg, alertMsg, reqUrl, command, asyncVal

Description

This is the basic function to request/control the functions of the 3rd-party application. This function is defined in the main server of the camera, so it can be used only for starting/stopping the application.

Parameters

Members	Description
Msg	The string to start/stop the application*
alertMsg	To alert when the function call is failed Use of this message is up to developer
reqUrl	The URL processes all request message
Command	SDK/SDK_APP/SDK_APP_DATA See 3. Commands
asyncVal	For most cases, use “undefined”

Return

This function returns no string when the request is processed successfully. If not, it returns the “alertMsg” set by developer.

> **Note**
>
> *The format of “msg” is xml. Please see the following chapter.

startApplication

Define

startApplication function

Method	Parameters
startApplication	paramJSON

Description

This function is used for starting the application. This function is the same as when the SDK command is used.

Parameters

Members	Description
paramJSON	JSON String that contains the following information: appname: Name of the application. success: Callback function when the application starts successfully.

	error: Callback function when the application doesn't start.

Return

paramJSON parameter is updated without returning anything.

stopApplication

Define

stopApplication function

Method	Parameters
stopApplication	paramJSON

Description

This function is used for stopping the application. This function is the same as when the SDK command is used.

Parameters

Members	Description
paramJSON	JSON String that contains the following information: appname: Name of the application. success: Callback function when the application stops successfully. error: Callback function when the application doesn't stop.

Return

paramJSON parameter is updated without returning anything.

getApplicationStatus

Define

getApplicationStatus function

Method	Parameters
getApplicationStatus	paramJSON

Description

This function is used to fetch application status information. This function returns the status of an application, such as running, stopped, etc.

Parameters

Members	Description
paramJSON	JSON String that contains the following information: appname: Name of the application. success: Callback function when the application status information is fetched successfully.

Return

paramJSON parameter is updated without returning anything.

getApplicationSettings

Define

getApplicationSettings function

Method	Parameters
getApplicationSettings	paramJSON

Description

This function is used for fetching application settings information. This function is the same as when the SDK_APP command is used.

Parameters

Members	Description
paramJSON	JSON String that contains the following information: appname: Name of the application.

	success: Callback function that gets the fetched settings information.

Return

paramJSON parameter is updated without returning anything.

updateApplicationSettings

Define

updateApplicationSettings function

Method	Parameters
updateApplicationSettings	paramJSON

Description

This function is used for updating the application settings information. This function is the same as when the SDK_APP command is used.

Parameters

Members	Description
paramJSON	JSON String that contains following information: appname : Name of the application. appconfig : Application configuration data. success: Callback function called when settings information is updated successfully. error : Callback function when there is an error.

Return

paramJSON parameter is updated without returning anything.

sendCommandToServer

Define

sendCommandToServer function

Method	Parameters
sendCommandToServer	paramJSON

Description

This function is used for sending requests to the server. This function is the same as when SDK_APP_DATA command is used.

Parameters

Members	Description
paramJSON	JSON String that contains the following information : appname: Name of the application. requestMessage: message String containing request. <GetSDK_APP_DATA> <AppName>application_name</AppName> <Data>...</Data> </GetSDK_APP_DATA> success: Callback function called with response from the server. error: Callback function in case of error.

Return

paramJSON parameter is updated without returning anything.

> **Note**
>
> If an error occurred when running sendCommandToServer, change <Data>...</Data> to <data>...</data>.

Commands

The following commands provide a way to communicate with the application and main server.

Parameters

Web communication commands

Command	Description
SDK	To control the application, such as start/stop
SDK_APP	To set/get the setting of the application
SDK_APP_DATA	To communicate with the application, use this command

SDK Command

This command is used to start/stop the application using the 3rd-party application's own web page.

Syntax:

Usage	Syntax
Start the app	<StartSDK> <AppName>application_name</AppName> </StartSDK>
Stop the app	<StopSDK> <AppName>application_name</AppName> </StopSDK>

Description

Syntax is the message used as a parameter("msg") for RequestAjaxMsg function. Developer should use the whole application name for "application_name" as defined in IPCameraManifest.xml file.

Return

After the function RequesAjaxMsg is called, the web page is reloaded regardless of success.

If the function fails to start/stop the application, the web page shows the alert message(“alertMsg”) as a popup.

SDK_APP Command

This command is used to set/get the settings of the application, especially the setting values defined in IPCameraManifest.xml

Syntax:

Usage	Syntax
Set the settings	<SetSDK_APP> <AppName>application_name</AppName> <VideoEncoding>...</VideoEncoding> <rawVideo>...</rawVideo> <appConfigData>...</appConfigData> <vaEvent>...</vaEvent> <rawAudio>...</rawAudio> </SetSDK_APP>
Get the settings	<GetSDK_APP> <AppName>application_name</AppName> </GetSDK_APP>

Description

Syntax is the message used as a parameter for request function(request function will be defined). Developer should put the whole application name in “application_name” as defined in IPCameraManifest.xml file.

Function Define

Request function

Method	Parameters
(up to developer)	msg, alertMsg, reqUrl, command, asyncVal

Function Description

This is the function to set/get the basic information defined in IPCameraManifest.xml file. This function is defined based on RequestAjaxMsg function.

Function Parameters

Members	Description
Msg	The string to set/get the IPCameraManifest.xml
alertMsg	To alert when the function call is failed The use of this message is up to the developer
reqUrl	The URL processes all request messages
Command	SDK_APP
asyncVal	In most cases, use “undefined”

Function Return

Usage	Syntax
Success to set	<Results> <Status>OK</Status> </Results>
Success to get	IPCameraManifest.xml in xml format
Fail to set/get	<Error> <ErrorString>ERROR: Configuration file not found </ErrorString> </Error>

Function Example:

```
function GetSettings()
{
    var msg = “”;
    msg = “<GetSDK_APP>”;
    msg += “<AppName>ServerPushMJPEG</AppName>”;
    msg += “</GetSDK_APP>”;
    SDK_APPGetSettingsAjaxMsg(msg, “”, URL, ‘SDK_APP’, ‘undefined’);
}

function SDK_APPGetSettingsAjaxMsg(msg, alertMsg, reqUrl, command, asyncVal)
{
     if(typeof command == “undefined”) command = “”;
     if(typeof asyncVal == “undefined”) asyncVal = false;

     $.ajax({
          type: “POST”,
          async: asyncVal,
```

```
          cache : false,
          url : reqUrl,
          dataType: “xml”,
          data: encodeURI(msg),
          success: OnSuccessGetSettings
   });
}

function OnSuccessGetSettings(ack)
{
     if($(ack).find(“ErrorString”).text())
     // Request is failed, process error.
     if($(ack).find(“model”).text())
     // Request is success
}
```

> **Note**
>
> Please see the example “ServerPushMJPEG”.

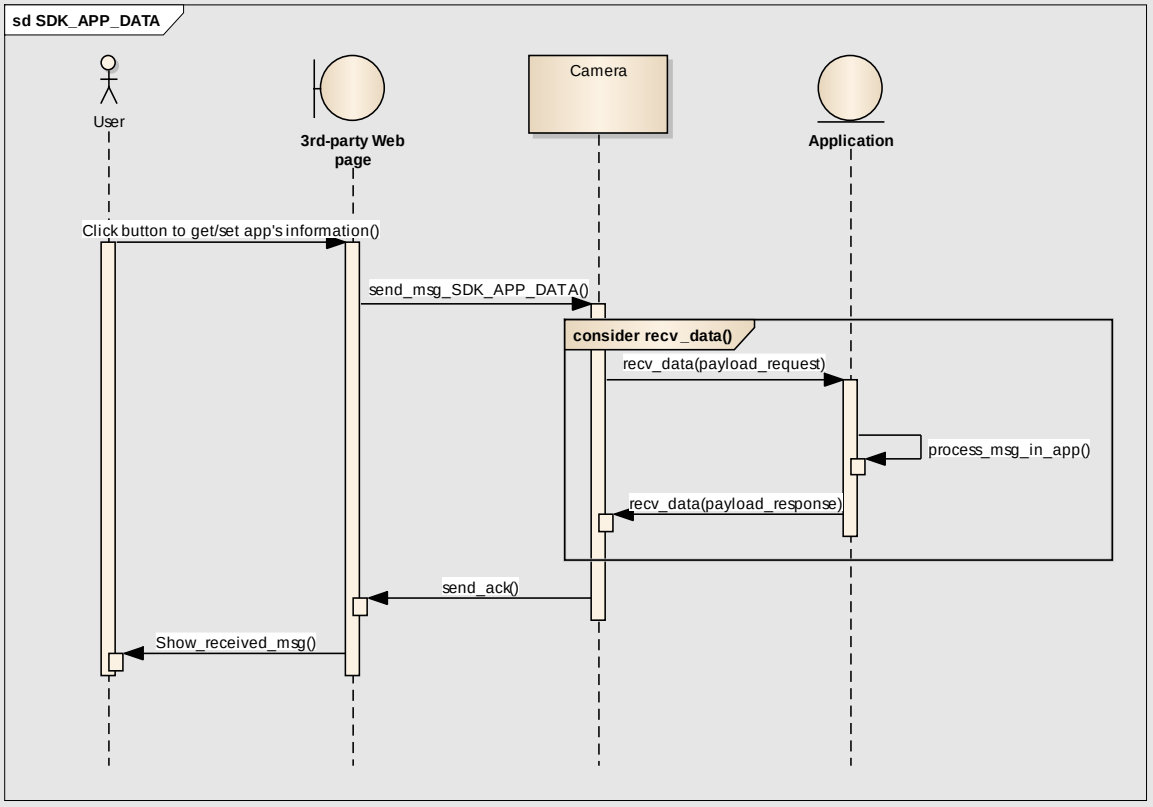
SDK_APP_DATA Command

This command is used to communicate with the application. The sequence diagram of this command is as shown in the picture below.

When the camera receives “SDK_APP_DATA” messages from the web, it sends the message to the application, especially “recv_data(void*, void*)”.

The function(recv_data()) has 2 parameters to receive and send messages; the first parameter (void *payload_request) is to receive the message from the camera(from the web) and the second (void *payload_response) is to send the message to the camera(to the web).

See the main source code when you make a project in Docker.



Syntax:

Usage	Syntax
Send the message to application	<SetSDK_APP_DATA> <AppName>application_name</AppName> <Data>...</Data> </SetSDK_APP_DATA> <GetSDK_APP_DATA> <AppName>application_name</AppName> <Data>...</Data> </GetSDK_APP_DATA>

Description

Syntax is the message used as a parameter for request function (see the example below). The developer should put the whole application name in “application_name” as defined in IPCameraManifest.xml file.

Note

If an error occurred when running SDK_APP_DATA Command, change <Data>...</Data> to <data>...</data>.

Function Example

```
function GetSettings()
{
    var msg = “”;
    msg = “<GetSDK_APP>”;
    msg += “<AppName>ServerPushMJPEG</AppName>”;
    msg += “<Data>GetInitData</Data>”
    msg += “</GetSDK_APP>”;
    SDK_APP_DataGetAjaxMsg(msg, “”, URL, ‘SDK_APP_DATA’, ‘undefined’);
}

function SDK_APP_DATAGetMsg(msg, alertMsg, reqUrl, command, asyncVal)
{
     if(typeof command == “undefined”) command = “”;
     if(typeof asyncVal == “undefined”) asyncVal = false;

     $.ajax({
          type: “POST”,
          async: asyncVal,
          cache : false,
          url : reqUrl,
          dataType: “xml”,
          data: encodeURI(msg),
          success: OnSuccessGet
     });
}

function OnSuccessGet(ack)
{
    if($(ack).find(“Success”).text())
     // Request is success, the message is from the application
}
```

```
OPENSDK_ERR_CODE recv_data(void *payload_request, void *payload_response)
{
    OPENSDK_PAYLOAD_REQUEST *req =
    (OPENSDK_PAYLOAD_REQUEST *)payload_request;
    OPENSDK_PAYLOAD_REQUEST *rsp =
    (OPENSDK_PAYLOAD_REQUEST *)payload_response;
    if(strstr(req_payload->pBuff, “GetInitData”) != NULL)
```

```
    {
        strcpy(res_payload->pBuff, “Success”);
    }
    Return OPENSDK_APP_OK;
}
```

All messages to get/set data (SDK_APP_DATA command) will be sent to recv_data(void *, void *) in the application.

References

Please refer to ServerPushMJPEGApp/html/index.html for examples on the use of functions.

Limitations

Recommended to use the functions of 3.50 or higher version.