UG05 Dynamic Library Interface Description

theme	**UG05 Dynamic Library Interface Description**
version	V1.1.0.1
content	**UG05 Dynamic Library Interface Description**
creation time	February 21, 2020
founder	**Leon**
turnover time	July 9, 2024

Document change record

Changed by	date	Change content
Leon		

Main review opinions of documents

Product group

reviewers	date	idea

QA group

reviewers	date	idea

development platform

Development environment and tools

tool	function
C++	Description of dynamic library file interface

Dependency library description

Document name	explain
UG05Sign.dll	Signature dynamic library

UG05Sign dynamic library environment support

system	Windows XP、Windows 7、Windows 8、Windows 10

Description of UG05Sign dynamic library interface

----------1.

```
/*********************************
* Parameter: None
* Return value: 0: successful opening; Other values: Failed to open.
**********************************/
int UgeeOpenDevice();
```

----------2.

```
/********************************  *
parameters:
    Parameter 1: Signature Pen Width
    Parameter 2: Signature image path (never put it in the system packing path)
* Return value: 0: success; Other values: failed
**********************************/
int UgeeStartSign(int penWidth, const char* szSignPath);
```

----------3

```
/********************************  *
parameters:
    Parameter 1: Fingerprint quality
    Parameter 2: fingerprint image path (never put it in the system packing path)
* Return value: 0: success; Other values: failed
**********************************/
int UgeeStartFinger(int quality, const char* szFingerPath);
```

----------4

```
/*********************************
* parameters:
    Parameter 1: Signature Pen Width
    Parameter 2: Signature image path (never put it in the system packing path)
    Parameter 3: Fingerprint image quality
    Parameter 4: fingerprint image path (never put it in the system packing path)
* Return value: 0: success; Other values: failed
*********************************/
int UgeeStartFinger(int penWidth,const char* SignPath,int quality, const
char* szFingerPath);
```

----------5

```
/*********************************
* Parameter: Callback function pointer
* Return value: 0: success; Other values: failed
*********************************/
int UgeeGetBase64CallBack(BASE64_FUNC func);
 typedef int(__stdcall* BASE64_FUNC)(const char*
szBase64);
```

----------6

```
/*********************************
* Parameter: None
* Return value: 0: success; Other values: failed
*********************************/
int UgeeCloseDevice();
```

----------7

```
/*********************************
* Parameter: None
* Return value: Finger ANSITemplate
*********************************/
Const char* UgeeGetANSITemplate();
```

----------8

```
/*********************************
* Parameter: None
* Return value: Finger ISOTemplate
**********************************/
Const char* UgeeGetISOTemplate();
```

----------9

```
/*********************************
Parameter:
Parameter 1: template1
Parameter 2: template2
*
* Return value: Compare Value
**********************************/
int UgeeCompareTemplate(const char*
tempalte1, const char* tempalte2);
```

----------10

```
/*********************************
Parameter:
Parameter 1:  pointer
* Return value:
**********************************/
void UgeeFreeString(char* str);
```

Example of C# calling c++Dll interface

1、Declaration of function

```
//OpenDevice(0:success;other:failed)
[DllImport("UG05Sign.dll", CallingConvention = CallingConvention.StdCall, EntryPoint = "UgeeOpenDevice")]
public extern static int UgeeOpenDevice();


//StartSign
[DllImport("UG05Sign.dll", CallingConvention = CallingConvention.StdCall, EntryPoint = "UgeeStartSign")]
public extern static int UgeeStartSign(int penWidth, string SignPath);


//GetBase64CallBack private delegate int
UgeeGetBase64CallBack_delegate(string Base64);


[DllImport("UG05Sign.dll", CallingConvention = CallingConvention.StdCall, CharSet = CharSet.Ansi, EntryPoint = "UgeeGetBase64CallBack")]
private extern static int UgeeGetBase64CallBack(UgeeGetBase64CallBack_delegate callback);
//CloseDevice(0:success;other:failed)
[DllImport("UG05Sign.dll", CallingConvention = CallingConvention.StdCall, EntryPoint = "UgeeCloseDevice")]
public extern static int UgeeCloseDevice();

// UgeeStartFinger
[DllImport("UG05Sign.dll", CallingConvention = CallingConvention.StdCall, EntryPoint = "UgeeStartFinger")]
 public extern static int UgeeStartFinger(int quality, string szFingerPath);
// UgeeStartSignFinger
[DllImport("UG05Sign.dll", CallingConvention = CallingConvention.StdCall, EntryPoint = "UgeeStartSignFinger")
]
public extern static int UgeeStartSignFinger(int penWidth, string SignPath,int quality, string szFingerPath);
```

2. Register the callback function

```
//Register the callback function
UgeeGetBase64CallBack_delegate callback = new
UgeeGetBase64CallBack_delegate(BASE64_FUNC);
UgeeGetBase64CallBack(callback);
```

3. The implementation of callback function.

```
/// <summary>
///response function
/// </summary>
/// <param name="status"></param>
/// <returns></returns>
private int BASE64_FUNC(string Base64)
{ try
  {
   pbSignImg.BackgroundImage = ConvertBase64ToFile(Base64);
   pbSignImg.Refresh();      return 0;
    }     catch
(Exception ex)
    {
   MessageBox.Show(ex.ToString());
return -1;
  }
}
```