eNBSP - *NBioBSP*

NITGEN&COMPANY Biometric Service Provider SDK

Programmer's Manual COM

SDK version 4.8x

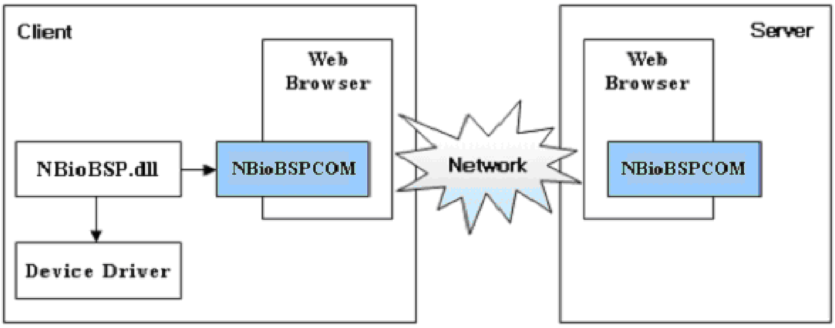
Serial Number:

Specifications subject to change without notice.

INDEX

Chapter 1. ASP Programming

The **NBioBSP COM** module runs on Web environments without any additional modification due to the fact that it is built on the Microsoft COM architecture. It can also be used simultaneously on Servers and Clients because it includes the Server Component and Client ActiveX functions.



[Module structure on Web environments]

1.1 Registration

User’s fingerprints must be registered through the Web Browser for user enrollment. This fingerprint data will be saved either in a file or DB on the Server.

1.1.1 Code for setting object

The NBioBSP COM module should be set as an object for each HTML page to be used on the Web Browser.

```
<OBJECT classid="CLSID:F66B9251-67CA-4d78-90A3-28C2BFAE89BF"
        height=0 width=0
        id="objNBioBSP"
        name="objNBioBSP">
</OBJECT>
```

The name shown here will be used as the object’s name in Javascript.

1.1.2 Form for transferring the fingerprint information

Fingerprint data registered in Javascript is transferred to the Server in this form.

```
<form action='regist.asp' name='MainForm' method='post' OnSubmit='return regist();'>
<input type=hidden name='FIRTextData'>
User ID : <input type=text name=UserID size=20><p>
<input type=submit value=' Click here to register your fingerprint '>
</form>
```

Upon selecting the form, Javascript calls the **regist()** function and performs the registration. Fingerprint data registered in this way will be transferred to the texts of **FIRTextData**.

1.1.3 Javascript code for fingerprint registration

Javascript code will be used to communicate between the Web Browser and the **NBioBSP COM** module.
The **Enroll** method also can be used.

The example above shows how functions can be composed. Note that Javascript is case sensitive.

```
<script lang='javascript'>
function regist()
{
        var err, payload

        // Check ID is not NULL
        if ( document.MainForm.UserID.value == '' )
        {
                alert('Please enter user id !');
                return(false);
        }

        try // Exception handling
        {
                // Open device. [AUTO_DETECT]
                // You must open device before enroll.
                DEVICE_FDU01 = 2;
                DEVICE_FDU11 = 4;
                DEVICE_AUTO_DETECT = 255;

                var objDevice = document.objNBioBSP.Device;
                var objExtraction = document.objNBioBSP.Extraction;

                objDevice.Open(DEVICE_AUTO_DETECT);

                err = objDevice.ErrorCode;        // Get error code

                if ( err != 0 )          // Device open failed
                {
                        alert('Device open failed !');
                        return(false);
                }

                // Enroll user's fingerprint.
                objExtraction.Enroll(payload);
                err = objExtraction.ErrorCode;    // Get error code

                if ( err != 0 )          // Enroll failed
                {
                        alert('Registration failed ! Error Number : [' + err + ']');
                        objDevice.Close(DEVICE_AUTO_DETECT);
                        return(false);
                }
                else    // Enroll success
                {
                        // Get text encoded FIR data from NBioBSP module.
                        document.MainForm.FIRTextData.value = objExtraction.TextEncodeFIR;
                        alert('Registration success !');
                }

                // Close device. [AUTO_DETECT]
                objDevice.Close(DEVICE_AUTO_DETECT);

                objExtraction = 0;
                objDevice = 0;
        }
        catch(e)
        {
                alert(e.message);
                return(false);
        }

        // Submit main form
        document.MainForm.submit();
        return(true);
}
</script>
```

1.1.4 Storing the fingerprint information

Perform verification by bringing fingerprint data forth using ASP code. Saving it in either a file or DB on the server.

```
<%
UserID     = Request.Form("UserID")
FIRTextData   = Request.Form("FIRTextData")

` Write UserID and FIRTextData to File or DB.
%>
```

1.2 Verification

Fingerprints captured for verification will be transferred and compared to the fingerprint stored on the Server.

1.2.1 Code for setting object

The **NBioBSP COM** module should be set as an object for each HTML page to be used on the Web Browser.

```
<OBJECT classid="CLSID:F66B9251-67CA-4d78-90A3-28C2BFAE89BF"
        height=0 width=0
        id="objNBioBSP"
        name="objNBioBSP">
</OBJECT>
```

The name shown here will be used for the object in Javascript.

1.2.2 Form for transferring fingerprint information

Fingerprint data captured in Javascript will be transferred to the server in this form.

The Javascript function for capturing fingerprint is called when the form is submitted.

```
<form action='verify.asp' name='MainForm' method='post' OnSubmit='return capture();'>
<input type=hidden name='FIRTextData'>
User ID : <input type=text name=UserID size=20><p>
<input type=submit value=' Click here to verification with your fingerprint '>
</form>
```

Upon selecting the button set in this form, the fingerprint will be captured as if calling the **capture()** function in Javascript and then transferred as **FIRTextData** text.

1.2.3 JavaScript code for capturing fingerprints

The flow chart above shows how functions can be composed. Note that text is case sensitive in Javascript.

```
<script lang='javascript'>

function capture()
{
        var err

        // Check ID is not NULL
        if ( document.MainForm.UserID.value == '' )
        {
                alert('Please enter user id !');
                return(false);
        }

        try // Exception handling
        {
                // Open device. [AUTO_DETECT]
                // You must open device before capture.
                DEVICE_FDU01 = 2;
                DEVICE_FDU11 = 4;
                DEVICE_AUTO_DETECT = 255;

                var objDevice = document.objNBioBSP.Device;
                var objExtraction = document.objNBioBSP.Extraction;

                objDevice.Open(DEVICE_AUTO_DETECT);
                err = objDevice.ErrorCode;       // Get error code

                if ( err != 0 )                  // Device open failed
                {
                        alert('Device open failed !');
                        return(false);
                }

                // Enroll user's fingerprint.
                objExtraction.Capture();
                err = objExtraction.ErrorCode;    // Get error code

                if ( err != 0 )                   // Enroll failed
                {
                        alert('Capture failed ! Error Number : [' + err + ']');
                        objDevice.Close(DEVICE_AUTO_DETECT);
                        return(false);
                }
                else    // Capture success
                {
                        // Get text encoded FIR data from NBioBSP module.
                        document.MainForm.FIRTextData.value = objExtraction.TextEncodeFIR;
                        alert('Capture success !');
                }

                // Close device. [AUTO_DETECT]
                objDevice.Close(DEVICE_AUTO_DETECT);

                objExtraction = 0;
                objDevice = 0;
        } // end try
        catch(e)
        {
                alert(e.message);
                return(false);
        }

        // Submit main form
        document.MainForm.submit();
        return(true);
}
</script>
```

1.2.4 Matching with the existing fingerprint

Use the **VerifyMatch** method to read the existing fingerprint data and compare it to the captured fingerprint data. The result can be found in the **MatchingResult** property.

```
<%
        ` Read FIR data from file or DB.

        Set objNBioBSP = Server.CreateObject("NBioBSPCOM.NBioBSP")
        Set objMatching = objNBioBSP.Matching

        ' Verify Match
        ' FIRTextData is Caputured FIR
        ' fFIRTextData is FIR from file
        Call objMatching.VerifyMatch(CStr(FIRTextData), CStr(fFIRTextData))

        if objMatching.MatchingResult = 0 then
                ` Verification failed !!!
        else
```

```
                ‵ Verification success !!!
        end if

        ' Release NBioBSP object
        Set objMatching = nothing
        Set objNBioBSP = nothing
%>
```