

# **MANTRA**

---

**MFS100 Web SDK**

**Ver. 9.0.2.1**

**Ubuntu Series**

## **MANTRA Softech India Pvt. Ltd.**

Mantra Softech (India) Pvt. Ltd. is a leader in Biometric security and Business solutions. Automated biometric Identification System (ABIS) and software for fingerprint identification search matching / finger print recognition are the primary products of Mantra Technologies. We provide the accuracy criminal and civil fingerprint identification search and authentication systems in the industry.

Company is not only powerful manufacturer of skin biometric time tracker products, but having their own supply chain across India with leading biometrics dealers, distributors and other biometrics suppliers. Company's manufactured biometrics is of high quality with Global standards. Company has been manufacturing biometrics for last eight years. Their Ahmadabad based Biometrics manufacturing centre is having skilled professional staff.

Mantra Softech (India) Pvt. Ltd. designs, provides and manages biometric solutions utilizing its proprietary Fingerprint Technology - designed for the purpose of "1-to-Many" fingerprint matching solutions.

## **Objective**

This document provides a guidance about to capture fingerprint ISO (FMR) data using MFS100 Scanner from web application.

### **1. Requirement Specification**

#### **I. OS Compatibility**

- i. Ubuntu Series

#### **II. Software Specification**

- i. Java Version Jre7 or higher
- ii. MFS100 Redistributable
- iii. Web browsers with java plug-in
  - Mozilla Firefox (Min) > version 3.0
- iv. Enable java plugin in firefox using below commands  
cd /usr/lib/mozilla/plugins/  
ln -fs /usr/java/jre1.x.x\_xx/lib/i386/libnpjp2.so

Please note that the browse must be set to load the applet control

#### **III. System Setting**

- i. Main Class
  - com.mantra.mfs100.mfs100api.class
- ii. Required Jar
  - mantra.mfs100.applet.linux.jar

#### IV. Hardware Specification

- i. USB 2.0 port
- ii. MFS100 (Mantra Fingerprint Scanner)

**Note:** Please attach scanner to system before load the web page.

## 2. Function

### I. Init()

**Type:** Integer

Return 0 if success. If not 0 then check Last error. Init is necessary before accessing any function.

### II. GetSerialNo()

Return device serial number if Init success, else return blank.

### III. Uninit()

**Type:** Integer

Return 0 if success. If not 0 then check Last error. After Uninit no any function can access.

### IV. Capture()

**Type:** void

This function is use to capture the finger print from the scanner.

After successful execution of this function, captured value has been stored in property in capture control.

### V. GetQuality()

**Type:** integer

Get the quality of Last Captured Finger.

### VI. GetNFIQ()

**Type:** integer

Get the NFIQ (NIST Finger Image Quality) of Last Captured Finger.

### VII. GetRawImage()

**Type:** String

Get the Raw Image of Last Captured Finger.

### VIII. GetBitmapImage()

**Type:** String

Get the Bitmap Image of Last Captured Finger.

- IX. GetISOImage()**  
Type: String  
Get the ISO Image of Last Captured Finger.
- X. GetISOTemplate()**  
Type: String  
Get the ISO Template of Last Captured Finger.
- XI. GetANSITemplate()**  
Type: String  
Get the ANSI Template of Last Captured Finger.
- XII. GetLastError()**  
Type: String  
Get the Last Error Generated in SDK/Applet.
- XIII. MatchISO()**  
Type: Integer  
Return the Matching Score 0 to 1,00,000 of two ISO templates. For passing Gallery Template, set the value in property of **GalleryISOTemplate**.  
Recommended score is  $\geq 14000$ .
- XIV. MatchANSI()**  
Type: Integer  
Return the Matching Score 0 to 1,00,000 of two ISO templates. For passing Gallery Template, set the value in property of **GalleryANSITemplate**.  
Recommended score is  $\geq 14000$ .
- XV. RotatImage()**  
Type: Integer  
Return 0 if Success. And it will rotate the image of fingerprint. Possible value is 0 or 180.
- XVI. IsDeviceConnected()**  
Type: Integer  
Return 0, if device connected. For that initialization is necessary.
- XVII. SetKey(String Key)**  
Type: Integer

Param: Key – String (32 chars key contains a-z, A-Z,0-9)

Return 0, if success. Else refer GetLastError().

Key size: 32 Bit

Algorithm-Scheme: DESede

Note: If you have not set Encryption key then default 32bit key “FslwO1CKOCJ166eJVSw0jGQpEZji7F7C” will be used.

### 3. Property

#### I. Quality

**Type:** Integer (input)

**Range:** 50 to 100

**Default:** 55

The quality is the parameter passed to control for capture finger with minimum quality.

#### II. TimeOut

**Type:** Integer (input)

**Range:** 10 to 100 (in Second)

**Default:** 15

**Maximum:** 30

**Minimum:** 10

The TimeOut is the property used for stop the capturing if defined quality will not acquire with in time second.

#### III. InitOnLoad

**Type:** Integer

**Value:** 0 or 1 (Default)

This property is used to define that whether applet will init device on load or not. If this value is set to 0 then you have to manually init the device by calling Init() function. And if it is set to 1 then applet will init device on load event.

#### IV. EnabledConsoleLog

**Type:** Integer

**Value:** 0 (Default) or 1

This property is used to define that log will be write on java console or not. If 0 then log will not write and if 1 then log will be write on console for debut purpose.

#### V. EncMode

**Type:** Integer

**Value:** 1 (Default) or 0

This property is used to define that enable encryption or not. If 1 then encryption will be performed before returning data or passing data. Else encryption will not be performed.

## 4. Code Snippets

### HTML

```
<applet name="ScanFinger" code="com.mantra.mfs100.mfs100api.class"
archive="mantra.mfs100.applet.linux.jar" width="137px" height="187px">
  <param name="Quality" value="60" />
  <!--set minimum quality you want to capture.-->
  <param name="TimeOut" value="15" />
  <!--set time out if minimum quality will not acquire with in time (second)-->
  <param name="InitOnLoad" value="1" />
  <!--InitOnLoad=1 will init scanner on applet load-->
</applet>
```

### Java Script

```
<script type="text/javascript" language="javascript">
  var isCaptureStarted=false;
  function Caputure()
  {
    if(isCaptureStarted==true)
    {
      return false;
    }
    try
    {
      isCaptureStarted=true;
      document.ScanFinger.CaptureFinger();
      var res = document.ScanFinger.FMRValue;

      if(res.length<=100)
      {
        alert("Capture Failed, Please Try Again.");
        return false;
      }
      alert(res); //captured successfully, do stuff with res.
      return false;
    }
    catch(e)
    {
      alert(e);
      return false;
    }
    finally
    {
      isCaptureStarted=false;
    }
  }
}
```

```
</script>
```

### Calling Java Script

```
<input id="btnCapture" type="submit" value="Capture Finger" onclick="return  
Caputure()"/>
```

Note: See sample html file for more functions.

MANTRA