

MedicSDKJ

Medical SDK Library Java Wrapper Description

Document Version 2.1

November 2010

Table of Contents

Introduction	4
Scanning and processing medical insurance	5
Functions	5
Functions Description	7
InitMedSdk	7
ProcessMedical	7
ProcessMedicalSide	8
getMedFrontSide	9
getMedPlanProvider	9
getMedMemberName	9
getMedMemberID	9
getMedGroupNumber	10
getMedPayerID	10
getMedCopayOV	10
getMedCopaySP	10
getMedCopayUC	11
getMedCopayER	11
getMedEffectiveDate	11
getMedExpireDate	11
getMedDateOfBirth	11
getMedTelTotalItems	13
getMedTelLabel	13
getMedTelValue	13
getMedEmailTotalItems	14
getMedEmailLabel	14
getMedEmailValue	14
getMedWebTotalItems	14
getMedWebLabel	14
getMedWebValue	15
getMedAddressTotalItems	15
getMedAddressFull	15
getMedAddressStreet	15
getMedAddressCity	17
getMedAddressState	17
getMedAddressZip	17
getMedOther	17
getMedRawText	18
getMedContractCode	18
getMedPlanType	18
getMedDeductible	18
getMedRxBin	18
getMedRxPCN	18
getMedRxGroup	19
getMedEmployer	19
getMedCoverage	20

getMedPlanCodeTotalItems	20
getMedPlanCode	20
getMedRxId	20
getMedPlanAdmin	20
getMedGroupName	20
getMedIssuerNumber	21
getMedNameFirst	21
getMedNameMiddle	21
getMedNameLast	21
getMedNamePrefix	21
getMedNameSuffix	21
getMedDeductibleTotalItems	21
getMedDeductibleLabel	22
getMedDeductibleValue	22

Introduction

MedicSDKJ is a Java wrapper for MedicSDK that encapsulates all the functionalities needed to scan, process, parse and export medical insurance documents.

Scanning and processing medical insurance

Scanning medical insurance and analyzing its content is done in the following automated sequence:

Detect document placement on the scanner tray: MedicSDK automatically monitor the scanner paper sensor and send event to the application once a document is placed on the scanner's tray.

Scan the document: MedicSDK automatically starts the document scan and send event to the application once the scan ends. When a duplex scanner is used, both document sides may be scanned in a single document pass.

Process the document: MedicSDK automatically analyze the scanned document as soon as the scan ends. If a duplex image is acquired, the processing algorithm processes both sides simultaneously. The result data is loaded MedicSDK properties.

MedicSDK is defaulted to perform steps 1-4 automatically without the application intervention. However, the application can also customize MedicSDK to trigger each step by the application.

Functions

MedicSDK.ocx communicates with the hosting application using functions events and properties.

Function Name	Operation
InitMedSdk	Initialize the SDK with the license key
ProcessMedical	Extract the information from the medical insurance document
ResetMedFields	Reset all the fields
getMedFrontSide	boolean
getMedPlanProvider	string
getMedMemberName	string
getMedMemberID	string
getMedGroupNumber	string
getMedPayerID	string
getMedCopayOV	string
getMedCopaySP	string
getMedCopayUC	string
getMedCopayER	string
getMedEffectiveDate	string
getMedExpireDate	string
getMedDateOfBirth	string

getMedOther	string
getMedTelTotalItems	short
getMedTelLabel	string
getMedTelValue	string
getMedEmailTotalItems	short
getMedEmailLabel	string
getMedEmailValue	string
getMedWebTotalItems	short
getMedWebLabel	string
getMedWebValue	string
getMedAddressTotalItems	short
getMedAddressFull	string
getMedAddressStreet	string
getMedAddressCity	string
getMedAddressState	string
getMedAddressZip	string
getMedRawText	string
getMedContractCode	string
getMedPlanType	string
getMedDeductible	string
getMedRxBin	string
getMedRxPCN	string
getMedRxGroup	string
getMedEmployer	string
getMedCoverage	string
getMedPlanCodeTotalItems	short
getMedPlanCode	string
getMedRxId	string
getMedPlanAdmin	string
getMedGroupName	string
getMedIssuerNumber	string
getMedNameFirst	string
getMedNameMiddle	string
getMedNameLast	string
getMedNamePrefix	string
getMedNameSuffix	string
getMedDeductibleTotalItems	short
getMedDeductibleLabel	string
getMedDeductibleValue	string

Functions Description

InitMedSdk

Format

```
InitSdk (License As String) As Long
```

Parameters

[in] **License** – Null terminated string that holds license key value.

Return

LICENSE_VALID: License is valid and the library is ready to be used.

LICENSE_INVALID: The license is invalid. All scanner operations are disabled.

LICENSE_EXPIRED: License has expired. All scanner operations are disabled.

LICENSE_DOES_NOT_MATCH_LIBRARY: The license is invalid for this library. All library operations are disabled.

GENERAL_ERR_PLUG_NOT_FOUND: This error returns if no valid scanner is attached to the PC.

SLIB_LIBRARY_ALREADY_INITIALIZED: The *InitSdk* function call is ignored since the library is already loaded.

Remarks

Use this function to initialize the MedicSdk ActiveX. This function must be called before calling any other function in the library.

ProcessMedical

Format

```
ProcessMedical (ImageFileSideA As String, ImageFileSideB As String)
```

Parameters

[in] **ImageFileSideA** – Null terminated empty string – reserved.

[in] **ImageFileSideB** – Null terminated empty string – reserved.

Remarks

This function process medical insurance image and extract the different fields in the image. The processed image is taken from the internal image buffer in the memory that was loaded in the last scan. If a duplex scanner is used to scan both sides of the insurance card in a single pass, both images (front and back) are saved into two internal image buffers in the memory. These images are processed in a single

function call and the information from both sides is extracted and loaded to the SDK properties.

The two parameters of the functions are ignored and should be empty, null terminated strings.

Return

If function succeeds, the return value is equal or larger than 0 and contain the number of times that the image was rotated in 90 degrees until it was aligned properly in the memory:

- 0: The image of side A was not rotated.
- 1: The image of side A was rotated in 90 degrees by the function.
- 2: The image of side A was rotated in 180 degrees by the function.
- 3: The image of side A was rotated in 270 degrees by the function.

If the function fails, the return value is smaller that 0 and may be one of the following values:

INVALID_INTERNAL_IMAGE – No internal image is loaded. This value return when attempting to use this function without scanning an image first.

GENERAL_ERR_PLUG_NOT_FOUND: This error returns if the image was not scanned by CSSN scanner model.

ProcessMedicalSide

Format

ProcessMedicalSide (**Side** As Int, **ImageFile** As String, **Reserved** As Int)

Parameters

- [in] **Side** – 0 for front side of the image. 1 for back side of the image.
- [in] **ImageFile** – Null terminated empty string "".
- [in] **Reserved** – 0

Remarks

This function process medical insurance image and extract the different fields in the image. The processed image is taken from the internal image buffer in the memory that was loaded in the last scan.

Return

If function succeeds, the return value is equal or larger than 0 and contain the number of times that the image was rotated in 90 degrees until it was aligned properly in the memory:

- 0: The image of side A was not rotated.
- 1: The image of side A was rotated in 90 degrees by the function.

- 2: The image of side A was rotated in 180 degrees by the function.
- 3: The image of side A was rotated in 270 degrees by the function.

If the function fails, the return value is smaller than 0 and may be one of the following values:

INVALID_INTERNAL_IMAGE – No internal image is loaded. This value is returned when attempting to use this function without scanning an image first.

GENERAL_ERR_PLUG_NOT_FOUND: This error is returned if the image was not scanned by the CSSN scanner model.

getMedFrontSide

Type:

Boolean.

Remarks

True if the scanned image is the image of the front side of the medical insurance.

getMedPlanProvider

Type:

String.

Remarks

Plan provider name

getMedMemberName

Type:

String.

Remarks

Member\Card holder name

getMedMemberID

Type:

String.

Remarks

Member\Card holder id number

getMedGroupNumber

Type:

String.

Remarks

Group number value.

getMedPayerID

Type:

String.

Remarks

Payer identification number.

getMedCopayOV

Type:

String.

Remarks

Copay O/V rate.

getMedCopaySP

Type:

String.

Remarks

Copay S/P rate.

getMedCopayUC

Type:

String.

Remarks

Copay U/C rate.

getMedCopayER

Type:

String.

Remarks

Copay E/R rate.

getMedEffectiveDate

Type:

String.

Remarks

Effective date value.

getMedExpireDate

Type:

String.

Remarks

Expiration date value.

getMedDateOfBirth

Type:

String.

Remarks

Birth date value.

getMedTelTotalItems

Type:

short.

Remarks

This property holds the amount of telephone number fields found on the card.

getMedTelLabel

Type:

String

Remarks

This property holds the label string of the corresponding telephone field index. To retrieve this field you must supply the telephone label index value. This index value can be between 0 to *PropTelTotalItems-1*

getMedTelValue

Type:

String

Remarks

This property holds the telephone number of the corresponding telephone field index . To retrieve this field you must supply the telephone label index value. This value can be between 0 to *PropTelTotalItems-1*.

getMedEmailTotalItems

Type:

short.

Remarks

This property holds the amount of email fields found on the card.

getMedEmailLabel

Type:

String

Remarks

This property holds the label string of the corresponding email field index. To retrieve this field you must supply the email label index value. This index value can be between 0 to *PropEmailTotalItems-1*

getMedEmailValue

Type:

String

Remarks

This property holds the email address of the corresponding email field index. To retrieve this field you must supply the email index value. This index value can be between 0 to *PropEmailTotalItems-1*

getMedWebTotalItems

Type:

short.

Remarks

This property holds the amount of Web fields found on the card.

getMedWebLabel

Type:

String

Remarks

This property holds the label string of the corresponding Web field index. To retrieve this field you must supply the Web label index value. This index value can be between 0 to *PropEmailTotalItem-1*

getMedWebValue

Type:

String

Remarks

This property holds the Web address of the corresponding Web field index. To retrieve this field you must supply the Web index value. This index value can be between 0 to *PropWebTotalItem-1*

getMedAddressTotalItems

Type:

short.

Remarks

This property holds the amount of Address fields found on the card.

getMedAddressFull

Type:

String

Remarks

This property holds the full address string of the corresponding address field index. To retrieve this field you must supply the address index value. This index value can be between 0 to *PropAddressTotalItems-1*

getMedAddressStreet

Type:

String

Remarks

This property holds the street string of the corresponding address field index. To retrieve this field you must supply the address index value. This index value can be between 0 to *PropAddressTotalItems - 1*

getMedAddressCity

Type:

String

Remarks

This property holds the city string of the corresponding address field index. To retrieve this field you must supply the address index value. This index value can be between 0 to *PropAddressTotalItems-1*

getMedAddressState

Type:

String

Remarks

This property holds the state string of the corresponding address field index. To retrieve this field you must supply the address index value. This index value can be between 0 to *PropAddressTotalItems-1*

getMedAddressZip

Type:

String

Remarks

This property holds the zip string of the corresponding address field index. To retrieve this field you must supply the address index value. This index value can be between 0 to *PropAddressTotalItems-1*

getMedOther

Type:

String

Remarks

This property holds all the strings that were not assigned to other fields.

getMedRawText

Type:

String

Remarks

This property holds all the text on the card as bulk text.

getMedContractCode

Type:

String

getMedPlanType

Type:

String

getMedDeductible

Type:

String

getMedRxBin

Type:

String

getMedRxPCN

Type:

String

getMedRxGroup

Type:

String

getMedEmployer

Type:

String

getMedCoverage

Type:

String

getMedPlanCodeTotalItems

Type:

Long

Remarks

This property contains the number of Plan Codes values found on the card.

getMedPlanCode

Type:

String

Remarks

This property holds the Plan code of the corresponding field index. To retrieve this field you must supply the Plan Code index value. This index value can be between 0 to *PropPlanCodeTotalItems-1*.

getMedRxId

Type:

String

getMedPlanAdmin

Type:

String

getMedGroupName

Type:

String

getMedIssuerNumber

Type:

String

getMedNameFirst

Type:

String

getMedNameMiddle

Type:

String

getMedNameLast

Type:

String

getMedNamePrefix

Type:

String

getMedNameSuffix

Type:

String

getMedDeductibleTotalItems

Type:

Long

Remarks

This property contains the number of Deductible values found on the card.

getMedDeductibleLabel

Type:

String

Remarks

This property holds the Deductible label of the corresponding field index. To retrieve this field you must supply the Deductible index value. This index value can be between 0 to *PropDeductibleTotalItems-1*.

getMedDeductibleValue

Type:

String

Remarks

This property holds the Deductible value of the corresponding field index. To retrieve this field you must supply the Deductible index value. This index value can be between 0 to *PropDeductibleTotalItems-1*.