Web Payment App

Purpose

This page app allows a website with permissions to take a card payment via the browser. The app is designed to fully support chip and pin making this ideal for specific kiosk applications. It also has support for printing to a receipt printer.
The app communicates with the SmartSwitch payment gateway to perform all of the card transactions.

Browser Integration

The browser provides three methods for communication allowing you to communicate with the chip-and-pin card reader, print receipts to the receipt printer and receive information about the current system and operational status of the machine.

All of this control is provided via three special Java Script commands that are only available when the WebPayment app is placed on the same page as the ~~CyberBrowser~~ and ~~CyberBrowser~~ has to be configured with *allowed URLs* that may integrate and communicate to be able to use these features.

It is possible to detect if payment is available from within the JavaScript running on a page.

The three functions that are available are as follows:
1. window.external.cb_getstatus()

Purpose:
Returns the status of the printer and card reader connected to this unit.
The return value is an integer bitmask.
Inputs:
None.
Returns:
A bitmask of the status of the system as follows:
If Bit '0' is set then there is a fault with the printer.
If Bit '1' is set then there is a fault with the card reader.

2. window.external.cb_makepayment(sourceid, transnum, amount, resultfield)

Purpose:

Performs the complete payment process and returns the result. Before the function returns, an XML response from SmartSwitch will have been written into the form field specified in the 4th parameter.

Inputs:

Input	Type	Description
SourceID	String	The source ID used by SmartSwitch and eventually Solve
Transnum	String/Integer	The transaction number to be use. This is used by SmartSwitch and Solve and must be unique
Amount	String (eg: 10.20)	The amount to charge the card for.Currently this is assumed to be GBP, and as such the currency symbol should not be supplied
Resultfield	String	An HTML Form Field Name/ID. This is a field (which will probably be hidden) where the browser may write the XML result from the SmartSwitch directly

Returns:

TRUE: if Payment was successful.
FALSE: if Payment failed.

Inputs:

Input	Type	Description
Header	String	Text to display at the top of the receipt. This text will be word-wrapped to fit the width of the printed receipt. New lines can be started as this field is treated as text. Note: The behaviour of this field can be affected by the optional 4th parameter (see XMLMode)
Items	String	Text to display in a table. This field requires the data to be written in a precise format. There are three elements within this field, each is comma separated. *No commas can appear anywhere else*. The first field is left aligned on the receipt, and the remaining two are right-aligned. For example: CD,,£10.00 ,Total:,£20.00

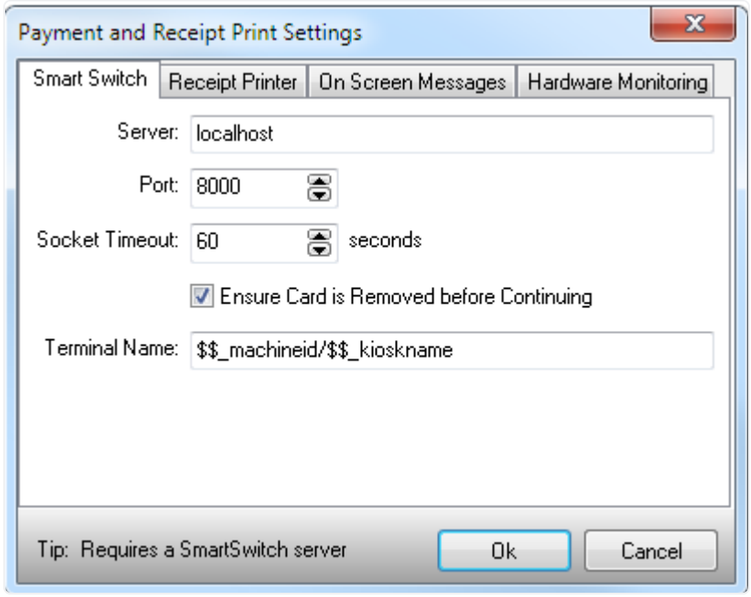
		Note: The behaviour of this field can be affected by the optional 4th parameter (see XMLMode)
Footer	String	Text to display at the bottom of the receipt. This text will be word-wrapped to fit the width of the printed receipt.New lines can be started as this field is treated as text. Note: The behaviour of this field can be affected by the optional 4th parameter (see XMLMode)
XMLMode	Boolean *Optional* Default=False	All fields are treated as plain text unless this field is specified as TRUE. If so, they are added to a special XML file exactly as you supply it. See *Configuration/Receipt Printer*.

Returns:
TRUE: If printing started successfully.
FALSE: If printing failed for some reason

Configuration

This section describes the configuration options within the WebPayment app.

Smart Switch settings tab



Server:
This is the address or IP of the SmartSwitch application service. Usually this will be installed on the same machine.

Port:
The port to communicate with SmartSwitch on. Normally 8000.

Socket Timeout:
The number of seconds to allow for each request to SmartSwitch. If this value is exceeded then payment will fail.

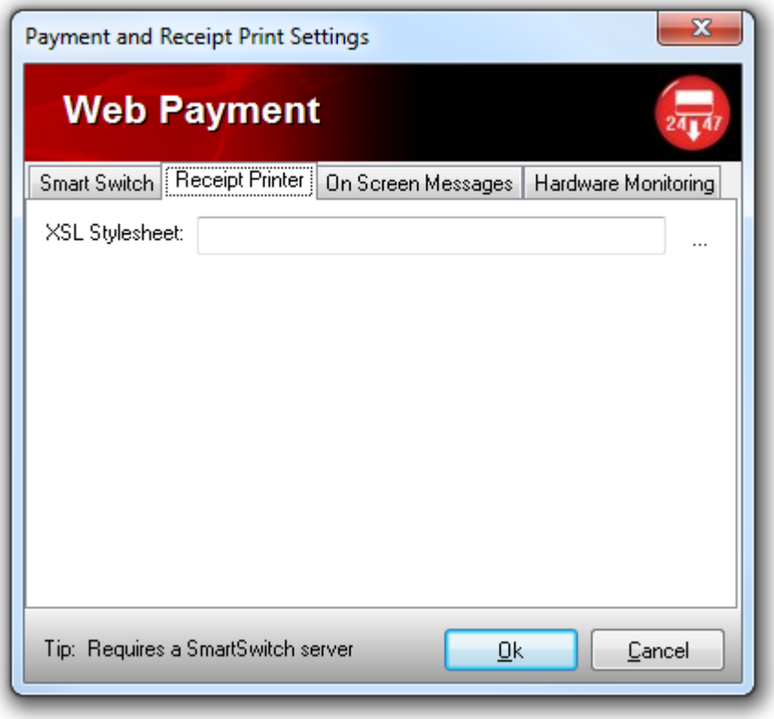
Ensure Card is Removed before Continuing:

If this box is ticked then the system will not exit the payment process until the card has been removed from the reader.

Terminal Name:

The name of the terminal. This name is fed straight into the SmartSwitch application. It defaults to the unique ID of the machine and its name (assuming the unit is connected to the PLUS! Network).

Receipt Printer tab



The XSL style sheet field allows you to choose a .XSL file. The file must be in the media folder. It is recommended that it is contained within an **"ALWAYS SEND"** folder to ensure that it and any graphics/files it uses are also deployed.
The XSLT style sheet is merged with a dynamic payment file to create the receipt which is then printed.

The XML file generated has the following structure:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="name_of_xsl_file.xsl"?>
<receipt>
	<header>
		Header Text Goes Here
	</header>
	<items>
		<item>
			<field1>Item 1 - Field 1</field1>
			<field2>Item 1 - Field 2</field2>
			<field3>Item 1 - Field 3</field3>
		</item>
		<item>
			<field1>Item 2 - Field 1</field1>
			<field2>Item 2 - Field 2</field2>
			<field3>Item 2 - Field 3</field3>
		</item>
	</items>
	<footer>
		Footer Text Goes Here
	</footer>
</receipt>
```

You can test your XSL file by creating an XML file like the above and changing the

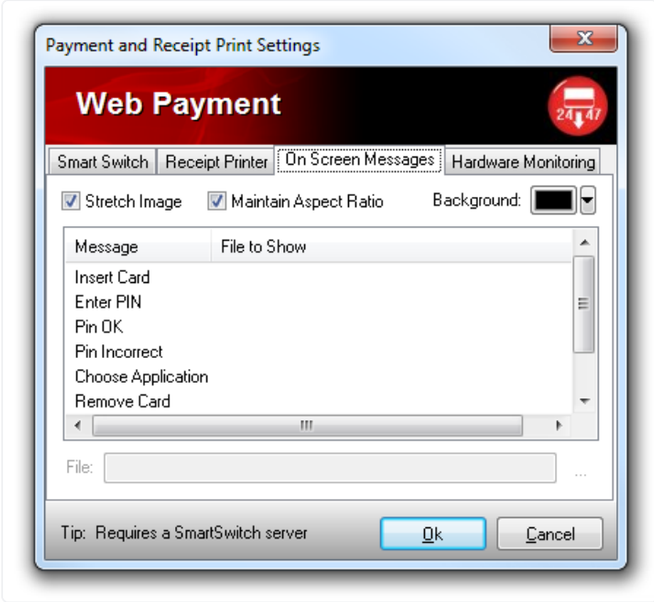
<?xml-stylesheet HREF to point to your XSLT file.
When printing a receipt, there is an optional 4th field XMLMode. If this is set then any text supplied to the Header, Items or Footer areas are treated as XML and are pasted into the relevant fields above without being altered in any way. This could allow more advanced formatting of these fields by specifying formatting using XHTML within the XSLT file. If this parameter is set to false then the fields are first processed to change some characters (e.g. £) into an XML complient format.

An example XSLT file:

```
<!DOCTYPE stylesheet [
<!ENTITY lt    "&#38;#60;">
<!ENTITY gt    "&#62;">
<!ENTITY amp   "&#38;#38;">
<!ENTITY apos  "&#39;">
<!ENTITY quot  "&#34;">
<!ENTITY pound "&#163;">
]>

<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
    <xsl:template match="/">
        <html>
            <style>
                body,
                table,
                tr,
                td  {
                      background: #FFFFFF;
                      color: #000000;
                      font-family: Arial, Verdana,  Helvetica, sans-serif;
                      font-size: 14pt;
                      font-weight: bold;
                      line-height: 16px;
                    };
            </style>
            <body leftmargin="0" topmargin="0" marginwidth="0"
                                        marginheight="0" text="black" color="white">
                <table border="0" cellpadding="0" cellspacing="4" width="100%">
                   <tr>
                       <td>
                           <center>
                              <img src="logo.gif" width="160" height="60" />
                           </center>
                       </td>
                   </tr>
                </table>
                <br />
                <p>
                    <xsl:copy-of select="receipt/header" />
                </p>
                <table border="0" cellpadding="2" cellspacing="1" width="100%">
                    <xsl:for-each select="receipt/items/item">
                        <tr>
                            <td align="left">
                                <xsl:copy-of select="field1"/>
                            </td>
                            <td align="right">
                                <xsl:copy-of select="field2"/>
                            </td>
                            <td align="right">
                                <xsl:copy-of select="field3"/>
                            </td>
                        </tr>
                    </xsl:for-each>
                </table>
                <p>
                    <xsl:copy-of select="receipt/footer" />
                </p>
            </body>
        </html>
    </xsl:template>
</xsl:stylesheet>
```

On Screen Messages tab
Use this tab to configure messages to be shown within the plug-in area. They can be a PNG, JPEG or BMP file,



Stretch Image: If ticked, the image is stretched to fill the size of the plug-in.
Maintain Aspect Ratio: If ticked, the image will maintain it's original shape even when resized.
Background: If the image is stretched and the aspect ratio is kept, the edge of the plug-in may not be covered by the image. If this is the case then you can change the background colour at these points.
Messages: Select image files to be shown on the following messages:

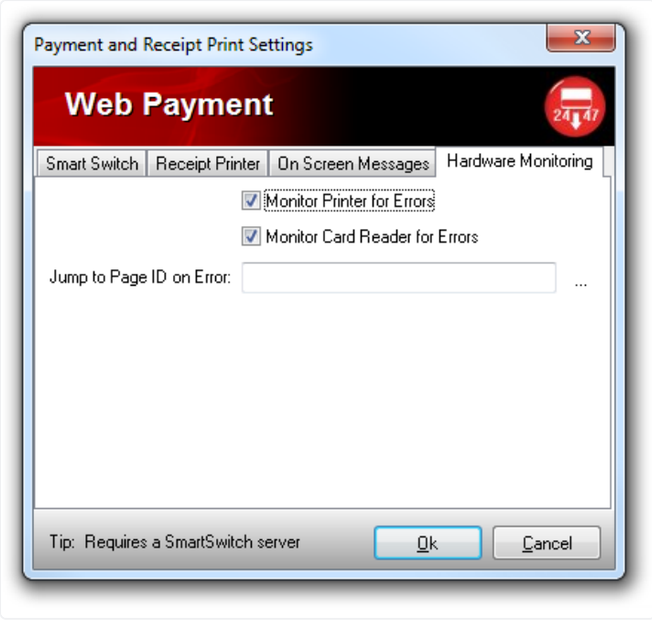
- Insert Card: An image that represents a request for a card to be inserted.
- Enter PIN: An image that represents a request for a PIN for the inserted card
- Pin OK: An image that reflects that the entered PIN was correct.
- Pin Incorrect: An image that reflects that the entered PIN was rejected.
- Choose Application: Represents that the card reader is waiting for the user to choose the payment application (for example if the card supports more than one payment system e.g. Maestro, Solo, MasterCard etc.).
- Remove Card: An image that informs the user they must remove their card from the reader.
- Busy: An image that informs the user that the reader is busy processing.
- Printing Receipt: A message that informs the user that their receipt is being prepared and printed.

If any of these are left empty then no image will be displayed.

File: Use these controls to select a file for the selected message in 4.

Hardware Monitoring tab.
This tab allows you to configure the plug-in to monitor the attached printer and card reader.

This can also be used on a page that does not allow payment. If an error occurs you can make Acquire display a different page, possibly an out of order message as appropriate.



Monitor Printer for Errors:

Tick to periodically monitor the printer for any unusual errors. Note that not all printers report errors like paper out etc, so the effectiveness of this option is based on the printer.

When ticked it will also check the status of each print job to see if there are any faults with them.

Monitor Card Reader for Errors:

Tick to periodically monitor the card reader/SmartSwitch. An error is signalled if either SmartSwitch cannot be contacted, or if the card reader device cannot be communicated with

Jump to Page ID on Error: Select an Acquire Page ID to display when one of the above errors occurs.

The hardware monitors the selected items when the page is loaded, and then periodically every 15 seconds. Reader Monitoring requires all of the settings on the SmartSwitch tab to be filled in.

During the request to SmartSwitch, the plug-in will set the SourceID and Transaction Number of the request to **“READER_PROBE_TEST”** so that it will not conflict with any other requests.