



Credit Card Processing Landing Page – Write Your Own Page

General

This document will take you through a short & easy to implement "How-To" Manual to create your own Landing Page. As it is now a mandatory standard, PCI requires all online transactions to go through a dedicated Landing Page hosted on a secure & authorized hosting server. ARKOM provides you with a simple ready to use Landing Page so you can implement the PCI requirements immediately without the need to go through lengthy, tedious & expensive process to PCI compliance.

For those who wish to use their own page that will maintain their UI & design ARKOM offers to host their own built Landing Page.

Basically there are two different Landing Page options:

1. Full Credit Card transaction

This page is most commonly used and is meant for any normal CC transaction

2. Token Generation

This page is used to collect the CC number from the cardholder and generate a Token that will be returned to the calling application to be used in future transactions without storing the CC number.

A few words about design

You are free to use any design you wish to suite your web application, however there a few rules to follow:

- FULL source code will be submitted for our inspection and authorization
- A credit placement to ARKOM must be embedded in the page including a live hyperlink to ARKOM's website (credit can be something like "Powered by ARKOM")
- Your Landing Page will contain one page only
- Note – our hosting environment is Windows based (IIS)



Development approach

Normally your regular web application should do all data collection from the end user and upon Credit Card payment selection it will redirect to your Landing Page for the card number entry.

Your calling page will use our web service to store the transaction basic data field like transaction sum etc. in our backend database and get a unique transaction ID as a reply.

This transaction ID is passed to the Landing Page and used through our web service to retrieve the basic transaction fields.

These fields should be ReadOnly to prevent modifications by end user.

Once the user entered his card number the Submit method will execute a Credit Card transaction through the web service and the results should be redirected back to your normal web application's return page.

Here is a simple Pseudo flow illustration for your convenience

- your web application does what it should do normally, the user chose to pay with his CC
your web application calls our WS function to store transaction basic info fields

```
iRet = WS.MTS_Redirect_GetTransID(TerminalNUM, _  
TerminalPassword, _  
TransSUM, _  
TransInstallments, _  
TransREF, _  
TransCurrency, _  
CreditType, _  
CustomerEmail, _  
ReturnURL, _  
HeaderLine1, _  
HeaderLine2, _  
FooterLine, _  
Lang, _  
TransID)
```

- if return error code = 0 (iRet) then it can call the Landing Page and pass the Transaction ID



- Landing Page loads and use WS function to load stored transaction basic info fields

```
iRet = WS.MTS_Redirect_GetTransData(TransID, _  
    TerminalNum, _  
    TerminalPassword, _  
    TerminalName, _  
    TransSUM, _  
    TransInstallments, _  
    TransRef, _  
    TransCurrency, _  
    TashType, _  
    CustomerEmail, _  
    ReturnURL, _  
    Header1, _  
    Header2, _  
    Footer, _  
    Lang)
```

- Once the user is done with card number entry and clicked Submit your Landing Page performs a regular web service CC transaction

```
iRet = WS.MTS_CC_Transaction(TerminalNUM, _  
    TerminalPassword, _  
    TransID, _  
    CardNum, _  
    CardExpiry, _  
    TransSum), _  
    TransPoints, _  
    CarnNumLast4Digits, _  
    CardCVV2, _  
    ID, _  
    Currency, _  
    ISO_Currency, _  
    CreditType, _  
    "", _  
    0, _  
    0, _  
    Installments, _  
    PrivRef.Text, _  
    0, _  
    "", _  
    "", _  
    "", _  
    "", _  
    TransAnswer,  
    MerchantNote,  
    ClientNote)
```



14/12/2011

- Next step – do not forget to send transaction acknowledge

WS.MTS_PutTransactionAcknowledge(TerminalNUM, TerminalPassword, TransID)

- Now store transaction results in our backend database

WS.MTS_Redirect_SetTransStatus(TransID, TransAnswer)

- Redirect to your return page

- Finally use WS to retrieve transaction result

iRet = WS.MTS_Redirect_GetTransResult(TerminalNUM, TerminalPASSWORD, TransID, TransRESULT)

Note:

- ALL calling samples are cuts from our Landing Page
- For FULL function documentation please refer to the MTS Web Service API Documentation
- Returning parameter TransResults holds the full transaction results as it normally returns from the web service
- Once the function is executed there will be no trace in our DB, so in case of system crash, communications failure or any other flow breaks you can use the normal WS GetResult function with the transaction ID you got in the first step.
- Header & Footer lines as well as Language parameters are used in our Landing Page to pass relevant page fields and setting the page's language – you can use them for any other purpose.
- To distinguish between TEST & Production environment you must support both environments in your page. We recommend a simple mechanism of adding a parameter to the query string or add a sign to the TransID, remember to handle the same mechanism in both Landing Page & Token Page and of course in your Return Page. Also remember to use the appropriate web service URL as there are 2 different URLs – for the URLs refer to the MTS WebService API documentation.

If you need a Landing Page for Token generation you will need to create a second Landing Page or to have a structure change in your Landing Page to accommodate both pages in one container.

We strongly recommend that you use our sample web pages to see the structure and behavior of the Landing Page, this should help you design & develop your own Landing Page.

You can find our samples in downloads page in our website or just use:

<http://apps.arkom.co.il/lp-demo> for CC transaction demo

<http://apps.arkom.co.il/lp-demo/TokenDemo.aspx> for Token generation demo