

Remotenet API Documentation

Djf 6/17/16 – updated 10/22/18

Overview:

We offer multiple methods for integration (depending on the modules and options that Client chooses)

1. The most common is EDI – or Electronic Data Interchange, while it's text file based and requires some format changes typically it's the most widely used.
2. We Also offer our Remotenet Web Interface which has an API for building a web site as well as a customizable UI, but for your purposes It would expose the ability to create sales orders in real-time or View new Orders placed into the system via any method.
3. We also offer CXML – Eprocurement integration – this is for creating sales orders typically
4. And for those who want it –we offer spreadsheet integration - i.e. upload a spreadsheet to create an order (this can also be done via a web interface with remotenet if they want it)

This document explained and discusses how to access and use the Remotenet API (application Programming interface) to interact with the MDS system.

The API is a toolset to allow you to create customers, products and pricing files in MDS.

IT allow allows you to create Web Orders and Send them to MDS as Sales Orders.

All functions of the website are exposed via the API as well and a full list of functions is outlined in the second section.

Remotenet API Via Http/Https Post

The Remotenet API via HTTP Post and Get is the simplest to implement for most clients. Each interaction requires you to authenticate initially and get a login token known as a unique Orderno. – You do not have to reauthenticate once you get an orderno and the orderno will not expire if there are products added to your order. If no products are added to the order it will be cleaned up after 24hrs.

The method of communication for remotenet is via http post and get commands –

And requires the use of two variables

LOGINPATH and PAGEID

The pageid tells remotenet what page you are coming from (this is a number from the list below)

The loginpath tells you what page/function to call

The above example starts are pageid=0 which is a login

You can pass credentials via this url/link and it will go to the PS page (product search)

Currently the Remotenet API has about 180 pages or paths you can provide.

Once logged in you are assigned a unique order number (6 DIGIT ALPHA NUMERIC)
And that will store your credentials on the server, so you don't need to authenticate again
If you pass a valid web order#

Each customers URL is unique for Live and Test Database

Typically

<http://hosting.tshinc.com/MDSWEB/login012> - would translate to

<http://servername/applicationname/loginXXX> - where

servername - is the name of the server hosting remotenet

Application name – MDSWEB for remotenet

LoginXXX = database to connect to – loginXXX would be your customer suffix 3 digit and for test your suffix with a B for beta – example 989 and 989B

Example Auto Login/Passing of Credentials via URL

<http://hosting.tshinc.com/MDSWEB/login012?CustomerID=DEMO&PASSWORD=DEMO&LOGINPATH=OOH&PAGEID=0>

This example would bring up a web page with a unique orderno embedded in it.

To parse the data, you would just look for the field called orderno –

Example using jQuery to Get Login info and pass to another page.

This assumes you have a login form with an id of StandardLoginForm and a CustomerID and password.

```
/// NEW FUNCTION TO LOGIN AND GO TO SEARCH PAGE AFTER ADDING ITEMS TO CART
function AddRnItem_Search( pstring ) {
  // added options to get all qty on page here..
  var content = $("#StandardLoginForm").serialize();
  // Send the data using post
  $("#StatusMsg").html("Getting Order Number...") ;
  if(content == null) {
    $("#StatusMsg").html("Could not get Login Information...") ;
    return ;
  } else {
    // we have data so send it.
    $("#StatusMsg").html("Sending Login Information...") ;
  }
  var posting = $.post( url, content );
  // Put the results in a div
  posting.done(function( data ) {
```

```

// var content = $( data ).find( "#content" );
var ordeno = $( data ).find( "input[name='orderno']" ).val() ;

// $( "#result" ).empty().append( "Ordeno is: " + ordeno );
if(ordeno) {
    $("#StatusMsg").html("Order Number " + ordeno) ;
    // Assume we have a good order now so login add item to order
    var content = $("#StandardLoginForm").serialize();
    $("#StatusMsg").html("Creating Order " + ordeno + "...") ;
//PS is the product search page , assumes you have all content you want to
//pass as a hidden variable in the form.
    var ncontent = "&PAGEID=7&LOGINPATH=PS&STARTPTR=1&ORDERNO=" + ordeno
    var content = content + ncontent ;
    $("#StatusMsg").html("Processing Search...") ;
    submitOtherFormUnique(content);
    $("#StatusMsg").html("Logged in Successfully" ) ;
    $('#SessLoader').fadeOut(2000) ;
    $('#mask').fadeOut(1000) ;
} else {
    // $("#StatusMsg").html("Invalid Login Information, Please try again...")
;
    var lockmsg = $( data ).find( "input[name='lockmsg']" ).val() ;
    if(lockmsg) {    $("#StatusMsg").html(lockmsg) ; } else {
        $("#StatusMsg").html("Invalid Login Information, Please try
again...") ;};
    }
});
} ;

```

Here is a simple example of how to check pricing and inventory in Realtime

Example of an INVENTORY AND Pricing call - the PRCX function calls the INVENTORY routine for a product

And returns an html formatted message with price and stock status. note an ORDERNO is required. as is the product code.

http://hosting.tshinc.com/MDSWEB/login012?orderno=xxxxx&CUSTOMERID=DEMO&LOGINPATH=PRCX&APRODNO=MOL_40855&PAGEID=31

Example list of all current functions:

```

SA Stock Alert Stock Alert
SCTX Ajax Product Subcategory Product Subcategory Information
SHP Shipping Confirmation Email Shipping Confirmation Email
SL Salesrep Login
SLS Slsrep User ID
SMX Ajax Ship Method Shipping Method
SO Payment Method Information Payment Method
SOP System Options
SOPS Save Sys Opts
SOX Order Detail via Ajax Web Order Detail
SPC Popular Products Popular Products
SPX Item of the Month Special Item
SR Search Results Search Results

```

SRE Search Results Search Results
SRUG Suggested ReOrder Template Suggested ReOrders
SS Supplier Setup
SSX Shipping Locations Shipto Locations
ST Save as Templ Save as Template
STKX Web Stock Info Stock Info
STM Save Templ Maint
STR Search Results Tab Data Search Results Tab Data
STTX Save as Template Popup Save as Template
STX Shopping Cart Web Order Detail
STX0 Ajax Select Box Data
STXN Shopping Cart Web Order Detail
SUG Suggested Items Template Suggested Items
SUSR Salesman User Login Orders On Hold
UC Update Comments
UD Upload Page
UPI Upload Images
VPI Vendor Page
VU Vend User Setup
VUD Del Vend User
VUM Vend User Select
VUN Del Vend Note
VUS Save Vend User
WD Web Disclaimer Terms and Conditions
WS Web Security
DLA Get a archived pdf Generic Get a File Routine
DO Order Deleted Order Deleted
DON Delete Order - Caution Delete - Caution
DS Our Daily Specials Daily Specials Page
DT Del Ord Templ Template Deleted
DTN
DU Delete User Delete End User
DUN Del User Note Delete - Caution
EDI EDI Order
EGN Generate EDI
EM Email CustomerId
EMH Email Order On Hold Web Order Reminder
EMO Email Order Confirmation Web Order Confirmation
EMR Suggested Re Orders Email Suggested Re Orders Email
EO Order Confirm Thank You
EPR EProcure Response Page EProcure Response Page
ERR Use to Update Error Message
EU End User Setup
EUM End User Maint
EUS Save User Setup
FB Feedback Form Feedback Form
FBN Feedback Form Feedback Form Sent
FTI Feature Products Feature Products
FTX Ajax Footprint Footprint Information
GRID Grid Order Sheet Grid Order Sheet
HO Order Hold Order Hold Notice
HOT Hot Items Template Hot Items
HOTV Hot Items Template(Vertical) Hot Items
IQ Inventory Inq Stock Inquiry
IQ_S Stock Status Inquiry
JS Use to update Javascript Utils
LBL ScanConnect Labels PRODUCT
LIC License Information State License Page
LO Logout
MS MSDS Info
MYA My Account My Account
NOA User Setup
NOB Admin Setup
NTI Product Notes Show Product Notes
NTS Product Notes Show Product Notes
OC Order Cancelled
ODP Order Dtl Prt
OE Order Entry Hdr Order Entry Header
OEM Ord Entry Meth
OOH Orders On Hold Orders On Hold
OOHE ScanConnect Orders On Hold Orders On Hold
ORDX Order Detail via Ajax Web Order Detail
PCS Product Contract Search Product Contract Search
PCSR Product Contract Search Results
PCX Ajax Product Class Product Information
ADM Admin Maint Administration
AI Web Audit Inquiry - Select Order Web Audit Select
AL Activity Log
ALN Activity Log
AO Add to Order Add to Order
AP1 Admin Page

APS Ajax Product Search Ajax Product Search
AR About RemoteNet
ARC Get an Archived Doc Document Archive Viewer
ARD Show Detailed AR Info Show Detailed AR Info
ARQ Accounts Receivable Inquiry Accounts Receivable
BAE Stock Alert Email Stock Alert Email
BAS Browse Search
BASR Browse Search Results
BLG Blog Our Blog
BO Backorder Inquiry Backorder Inquiry
BOX Backorder Inquiry via Ajax Backorder Inquiry
BRC Browse Class Browse Class
BRCAS Browse Class Browse Class
BRCX Browse Class Ajax Browse Class
BRS Browse Category Browse Category
BRSC Browse Subcategory Browse Subcategory
BRSCX Browse SubCategory Ajax Browse SubCategory
BRSX Browse Category Ajax Browse Category
BRU Search Results Search Results
BRUE Search Results Catalog Search Results
BRX Browse Type Ajax Browse Type
BU Batch Upload
BUN Batch Upload
CC Cash User Setup
CCDI Cust Contract Detail Inq Cust Contract Detail Inq
CCI Customer Contract Inquiry Customer Contract Inquiry
CCL Create Ord Templ
CCS Cash User Setup Save Cash User Setup Save
CIQ Customer/Item Price Inquiry Customer/Item Price Inquiry
CL Order Template Order
CLD
CLM Ord Templ Maint
CLP Export and Print Template Exported File Viewer
CO Current Order Current Order
COO Order Inquiry Order Inquiry
CP Change Password Change Password
CPW Confirm Password Upd Confirm Password Change
CS Continue Shopping
CSS Use to update Style Sheets
CTX Ajax Product Category Product Category Information
CU Contact Us
DD Download Page
DDR Download Results
DL Get a file Generic Get a File Routine

XML/Json Web Service for product information from MDS Via Remotenet

Overview:

Remote net uses the Solr Database for its web search engine. Below are the default detailed fields we have available in the feed for Remote net, multivalued means there may be multiple fields for this “row” i.e. one product may be in multiple categories etc. The “id” field is a unique 6-digit numeric code for each product in the feed. Most are self-explanatory example mfg name is the manufacturers name., price is the list price Etc.

AllText is a virtual field that combines multiple other fields for search (see the copyfield function at the bottom)

We use that for word searches etc.

Customers is a list of customers who bought the product, citems are the customers item number. For example, if they call it a 101 and we called it a blue pen. They can search by 101 and find the correct item.

Aliases are other product aliases for the product examples might be NDC#, NSN#, UPC#, GTIN etc.

You can use the end at

https://customerremotenebsite/solr/customerdbname/select?q=*&fq=allText:gloves*

where the site and db are provided by your support team.

if you need them email support@tshinc.com

our public example is

[http://hosting.tshinc.com/solr/products/select?q=%3A*&fq=allText%3A\(gloves*\)%20OR%20aprodn%3A\(gloves*\)](http://hosting.tshinc.com/solr/products/select?q=%3A*&fq=allText%3A(gloves*)%20OR%20aprodn%3A(gloves*))

To access the feed example above of a search for glove would be.

<http://hosting.tshinc.com/solr/products/select?>

Some options are below but there are literally thousands. – here is a link to a tutorial that will help you get

Started.

https://lucene.apache.org/solr/3_6_2/doc-files/tutorial.html

also, some other sites.

<http://yonik.com/solr/query-syntax/> or <http://www.solrtutorial.com/solr-query-syntax.html>

Standard Simple Options – simple search for item

q=%3A* - Query Filter – we usually leave the query alone and use filters (fq below) as they are faster

&rows=10 – Number of Rows to Return

&sort=hotseller%20desc – Sort of the Data (Asc or Desc afterwards)

&fq=allText%3A(hlo*)%20OR%20aprodn%3A(hlo*)%20OR%20citems%3A(000594%5C*hlo*)

- The Filter Query example above is a looking for hlo in the aprodno, citems or allText fields

&wt=json - wt – writer template output – standard options are xml,json, csv

&json.wrf=jQuery111104170454426202923_1451421046687

Callback function useful if using jquery ajax with callback so you know when completed.

list of fields to follow.

```
<!-- added standard mds data -->
<!-- "closeout|discount|catpage|stock|videorexref|datasheetxref -->
<field name="mfgname" type="string" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="mfgpart" type="string" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="stock" type="string" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="videorexref" type="string" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="datasheetxref" type="string" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="id" type="string" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="aprodno" type="text_aprodno" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="proddesc" type="text_general" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="proddesc2" type="text_general" indexed="true" stored="true"
omitNorms="true" termVectors="true" />
<field name="bestseller" type="int" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="hotseller" type="int" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="image" type="string" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="createdate" type="date" indexed="true" stored="true" omitNorms="true"
termVectors="true" />
<field name="desc" type="text_general" indexed="true" stored="true" omitNorms="true"
termVectors="true" />

<field name="webtypes" type="text_general" indexed="true" stored="true"
multiValued="true" omitNorms="true" termVectors="true" />
<field name="webclasses" type="text_general" indexed="true" stored="true"
multiValued="true" omitNorms="true" termVectors="true" />
<field name="webcatgs" type="text_general" indexed="true" stored="true" multiValued="true"
omitNorms="true" termVectors="true" />
<field name="websubcatgs" type="text_general" indexed="true" stored="true"
multiValued="true" omitNorms="true" termVectors="true" />
```

```
<field name="allText" type="text_partial" indexed="true" stored="true" multiValued="true"
omitNorms="true" termVectors="true" />
<field name="text" type="text_general" indexed="true" stored="true" multiValued="true"
omitNorms="true" termVectors="true" />
<field name="searchcat" type="string" indexed="true" stored="true" multiValued="true"
omitNorms="true" termVectors="true" />
<field name="price" type="float" indexed="true" stored="true" multiValued="false" />
```

```
<field name="aliases" type="text_general" indexed="true" stored="true" multiValued="true"
omitNorms="true" termVectors="true" />
<field name="citems" type="text_general" indexed="true" stored="true" multiValued="true"
omitNorms="true" termVectors="true" />
<field name="customers" type="text_general" indexed="true" stored="true"
multiValued="true" omitNorms="true" termVectors="true" />
```

```
<!-- end added data -->
```

```
<!--- added lot info -->
<field name="lotno" type="text_general" multiValued="true" indexed="true"
stored="true" omitNorms="true" termVectors="true" />
<field name="lotqty" type="float" multiValued="true" indexed="true" stored="true"
omitNorms="true" termVectors="true" />
<field name="lotexp" type="date" multiValued="true" indexed="true" stored="true"
omitNorms="true" termVectors="true" />
<field name="proddesc_sort" type="alphaOnlySort" indexed="true" stored="false"
required="false" multiValued="false" />
<field name="webprodassoc" type="string" multiValued="true" indexed="true"
stored="true" omitNorms="true" termVectors="true" />
```

```
<!-- added demo data -->
```

```
<copyField source="aprodno" dest="allText"/>
<copyField source="desc" dest="allText"/>
<copyField source="proddesc" dest="allText"/>
<copyField source="proddesc2" dest="allText"/>
<copyField source="text" dest="allText"/>
<copyField source="webtypes" dest="allText"/>
<copyField source="webclasses" dest="allText"/>
<copyField source="webcatgs" dest="allText"/>
<copyField source="websubcatgs" dest="allText"/>
<copyField source="aliases" dest="allText"/>
```