



Lean Swift eConnect for Infor M3

Test Plan (15.3)

## Revision Sheet

Release no	Date	Description
14.3	2/2/2015	
15.1	12/3/2015	Added Cron, Payment related scenarios.
15.2	3/8/2015	OAUTH security scenario
15.3	12/1/2015	Scenario for Ship complete ,Gift wrap , Address sync

## Contents

1.0	Introduction .....	5
2.0	Review and Sign-Off.....	6
3.0	Available Document .....	6
4.0	Scope .....	6
5.0	Out of scope .....	6
6.0	Testing Approach .....	6
7.0	Test Pass / Fail Criteria .....	7
8.0	Environment Requirements .....	7
9.0	Tools/Knowledge .....	7
10.0	Resource.....	7
11.0	Software Features and Functionality to Be Tested .....	8
11.1	UI Testing .....	8
11.2	Functional Testing .....	12
11.2.1	Product Information Sync .....	13
11.2.2	Price.....	14
11.2.3	Inventory .....	16
11.2.4	Customer Registration .....	17
11.2.5	Order Creation .....	18
11.2.6	Order Status .....	19
11.2.7	Shipment & Tracking .....	20
11.2.8	Order Charges .....	21
11.2.9	Sales .....	22
11.2.10	Order History.....	23
11.2.11	Invoice History .....	24
11.2.12	Payment .....	26
11.2.13	General.....	27
11.2.14	Other Functional Scenario .....	28
11.2.15	Security .....	28
11.2.16	Cron Scheduler .....	29
12.0	Privacy/Security Testing.....	31
13.0	Performance Testing.....	32
14.0	Compatibility Testing .....	32

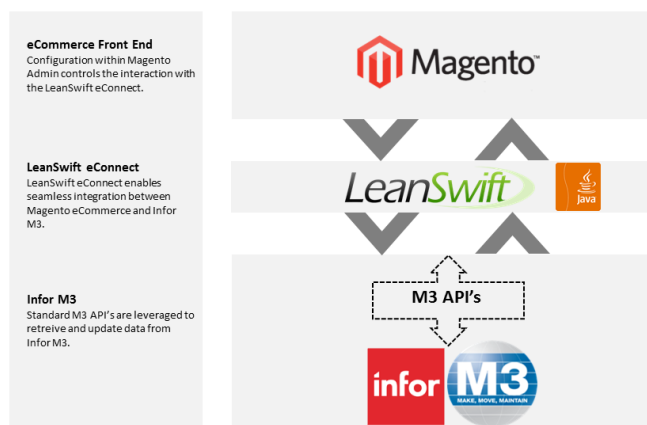
15.0	Localization .....	33
16.0	Automation .....	33
17.0	Appendix A – M3 Programs/setups used.....	34
18.0	Appendix B – MITMAS fields for attribute mapping .....	36

## 1.0 Introduction

This document defines the integration test scenarios of eConnect (LeanSwift product) with Magento and Infor M3. LeanSwift eConnect for Infor M3 provides a seamless integration between Magento and Infor M3 ERP. The product consists of a Magento extension, and a Tomcat based server application that manages the communication with the M3 ERP system.

LeanSwift eConnect for Infor M3 employs a layered architecture to allow more flexibility in supporting different versions of Magento and Infor M3, and to allow independent upgrades. The two components are versioned individually to more easily adapt to different M3 & Magento versions.

### Architecture overview:



## 2.0 Review and Sign-Off

S.NO	TITLE	NAME	SIGN OFF STATUS
1	Product Manager	Robert Bodizs , Max	Signed off
2	Project Delivery manager	Nrithya Rajagopalan	Signed off
3	Project Developer Lead	Binoy Varghese ,Gowtham	Signed off
4	Project QA	Jagadeesan	Signed off

## 3.0 Available Document

LeanSwift user guide documentation is referred for writing the test plan

<https://drive.google.com/a/leanswift.com/#folders/0B2gk1oSJbfXfWDRyUGZWtXZCRjA>

## 4.0 Scope

Scope of the testing is primarily on integration of eConnect with Magento and M3. Bugs would be reported for eConnect related functionality.

## 5.0 Out of scope

Magento and M3 systems are not tested independently as it is out of scope with respect to eConnect integration perspective.

## 6.0 Testing Approach

Tests will be run manually. Manual test case is maintained in excel sheet so that coverage and progress can be monitored by the project team and the stability of the function can be tracked over time.

## 7.0 Test Pass / Fail Criteria

All the issues will be reported in Yodiz under eConnect Project and issues can be tracked here

[Issues](#)

Severity	Comment
<b>Blocker</b>	Feature/Issue does not work and hampers the product shipping to customer
<b>Critical</b>	Feature has critical bug which should not be shipped without fixing it
<b>Major</b>	Feature/Issue has major bug but does not affect requirement/Customer
<b>Normal</b>	Feature/Issue has minor bug when released to customer.

## 8.0 Environment Requirements

- Magento front end set up with 50 sample products with mix of different product types integrated with M3 by eConnect.
- Access/Admin Privileges for all Magento, M3 and eConnect services.
- Connector URL should be up and running.
- SQL read access

## 9.0 Tools/Knowledge

- Magento Community edition and Enterprise edition.
- M3
- SQL/MYSQL
- REST client in chrome
- AOE scheduler(To schedule cron jobs)
- M3 H5 Client, Smart Office, MI Test.

## 10.0 Resource

7 full man days would be needed to run the full regression and test the complete functionality of product.

## 11.0 Software Features and Functionality to Be Tested

### 11.1 UI Testing

- Verify LEANSWIFT text is displayed in configuration panel after installation is successful from System -> Configuration.(P0)
- Verify LeanSwift Connector is displayed under LEANSWIFT in configuration panel.(P0)
- Verify RMA is displayed under LEANSWIFT in configuration panel.(P0)
- Verify user is displayed with option of Default Store View in drop down box of Current Configuration scope.(P0)
- Verify users is displayed with other stores in Current Configuration scope drop down box.(P0)
- Verify user is given Save Config button in right corner.(P0)
- Verify all the text in LeanSwift Connector page is aligned properly.(P0)
- Verify the theme of LeanSwift Connector page is uniform with respect to Magento.(P0)
- Verify the font is uniform all throughout the LeanSwift Connector page.(P0)
- Verify hide/unhide (^) option is available for all the available section.(P0)
- Verify there is no spelling mistake on the text displayed on LeanSwift Connector page.(P1)
- Verify the message displayed after pressing the Save Config button is proper.(P0)
- Verify error message is color coded properly and captures end users attention.(P2)
- Check scroll bar is displayed if the screen is zoomed in and zoomed out.
- Verify Buttons used in LeanSwift connector page are proper.(P1)
- Verify Text displayed in LeanSwift connector page conveys message to external customers.(P0)
- Verify user is displayed with below sections in LeanSwift Connector page.(P0)
  - General
  - Basic data
  - Authentication
  - Shipping Method
  - Product Synchronization
  - Customer General Configuration
  - Customer Synchronization
  - Sales
  - Order Charges
  - Payments
  - Price Synchronization
  - Inventory Synchronization
  - Cron



- Verify below options are available under General section .(P0)
  - Email
  - Service url
  - Use erp order history
  - Test mode
  - Debug/log data
  
- Verify Basic Data section has below options. (P0)
  - Company
  - Division
  - Facility
  - Warehouse
  - Price List
  - Currency Code
  - Order Type
  
- Verify Authentication section has below option (P0)
  - Enable OAuth
  - Web Service Username
  - Web Service Password
  - Client Id
  - Client Secret
  
- Verify Order Charges has below section (P0)
  - TAX Transfer
  - Tax charge code
  - Shipping or Freight charge Type
  - Shipping or Freight fee charge code
  - Gift Wrap Transfer
  - Gift Wrap Charge Type
  - Gift Wrap Order Charge code
  - Gift Wrap Line Charge code
  
- Verify Shipping Method section has Delivery Method Mapping.(P0)
  - Shipping Method
  - M3 Delivery Method
  - M3 Delivery term
  
- Verify Product Synchronization section has below options.(P0)
  - Super Attribute Name
  - Website Name

- Attribute Set Name
  - M3 Attribute Mapping
- Verify Customer General Configuration section has below options.(P0)
  - Enable Registration
  - Create New Customer
  - Common Customer id/Customer Template id
- Verify Customer Synchronization section has below options.
  - Update Customer Data on Login
  - Address Type Id
  - Billing Address mapping
  - Shipping Address mapping
  - Customer Master mapping
- Verify Sales section has below options.(P0)
  - Enable order comments
  - Send confirmation when external order placed
  - Display Invoice
  - Capture Payment Online
- Verify price synchronization section has below options.(P0)
  - Enable
  - Cache Hours
- Verify Inventory synchronization section has below options (P0)
  - ERP Inventory Attribute
  - Sync on Product Load
  - Sync on Add to Cart
  - Sync on Checkout
- Verify Cron section has below options (P0)
  - Cron settings for sending orders to ERP
  - Cron settings for getting ERP Orders
  - Cron settings for getting ERP Invoice
  - Cron settings for stock update
  - Cron settings for ERP order synchronization
  - Cron settings for ERP return order sync
  - Cron settings for sync product
  - Cron settings for customer sync
  - Cron settings for config data

- Verify Payment section has below options (P0)
  - Re-Authorization Amount
  - Payment provider
    - Magento Payment Method
    - M3 Payment Term
    - M3 Payment Method
    - M3 credit card Provider
  - Invoice fee charge code
  - Invoice Fees
    - Payment Method
    - Invoice Fee Amount, Invoice Fees Charge code
    - Invoice Fees Charge Item no

## 11.2 Functional Testing

LeanSwift eConnect has wide variety of below functionality that can be offered to customer

- Product Information Sync
- Price Sync
- Inventory
- Customer Registration
- Customer Sync
- Order Creation
- Order Status
- Shipment & Tracking Sync
- Charges
- Invoice Sync
- Order history
- Invoice history
- Payments
- General

Verify LEANSWIFT text is displayed in configuration panel after installation is successful from System -> Configuration. (P0)

- Verify user can hide and unhide all the available sections like General, Basic User Data displayed to user in LeanSwift Connector page.(P0)
- Verify LeanSwift\_Connector option is enabled in advanced option of Configuration set up.
- Verify cache is enabled in test environment and disabled in production environment.
- Verify erp\_item\_nr (Attribute set) is displayed in Attribute ->manage Attribute once eConnect is installed.
- Verify Erp item nr is present for any product in Catalog -> Manage Products.

### 11.2.1 Product Information Sync

Prerequisite: Products to be displayed in Magento should be available in M3.

- Verify 'Add M3 mapping' button will add fields for Magento attribute, M3 attribute and Default fields together - P0
- Verify there is a delete button which is functional and can delete the entire row of attributes - P0
- Magento attribute has a dropdown with options related to magento - P0
- Verify that all the attribute values in the dropdown are functional - P0
- Verify value syncs with M3 only when the item has a Ext item number defined in magento, else throws an error - P0
- Verify that correct values gets synced from M3 (MMS001) - P0
- Verify that unless ERP item number attribute is defined in product sync config, attributes do not sync to the product upon syncing - P0
- Verify that even after deleting few attributes, product sync works with remaining attributes - P0
- Verify the value that syncs when same attribute is mapped twice - P1
- Verify that performing a sync without selecting any product throws the message 'Please select item' - P2
- Verify sync works with Cron job, Cron config setting and manual sync from Manage products page - P0
- Verify that attributes that are not defined in product sync doesn't get synced to the product upon syncing - P2
- Verify any changes made in magento side gets reflected to product provided the item is not synced - P0
- Verify that if any changes made to the product attributes in M3 side gets reflected only when product is synced and not otherwise - P2
- Verify that when ERP item number of a product is changed to any other existing ERP number, details related to the new ERP item number are synced to that product -P1
- Verify that product details sync for items with different item types(M3) - P0
- Ensure the above scenarios works well with all types of products - P0

### 11.2.2 Price

Price function offers two different features

- Price from a single M3 price list
- Customer specific price for certain product

#### *Single Price List*

This feature is mostly applicable for B2C scenarios.

- Ensure that Price list field should have a text box that should accept only number - P0
- Ensure that Currency field should have a text box that accepts only alphabet - P0
- When Price list and Currency are defined in config and product is synced to M3, verify price of product is fetched from M3 (OIS017) - P0
- Ensure that when Price of the item is changed in magento and saved but not synced with M3, price from Magento is only displayed in frontend - P1
- Ensure that when Price of the item is changed and saved in M3 but product is not synced with M3, price from Magento is only displayed in frontend - P1
- Ensure Price sync works with product sync independent of price sync - P1
- Ensure only if price field is added as attribute in Product Sync, price gets synced from M3 upon syncing - P0
- Ensure whenever there is any discounts, special price for a product, it is displayed along with regular price with a strike on regular price - P0
- Ensure the price displayed in product detail page remains the same throughout the entire checkout process - P2
- Verify the regular price is displayed when the customer is not logged in - P0
- When Price list and Currency is not defined in config and product is synced to M3, verify a default value is set for the product - P0
- Verify the price displayed in front end when Price list is not defined but Currency is defined and vice versa - P1
- Verify the price that is displayed in front end when Price list and currency has invalid data -P1
- Enter special price for that item with date validation and ensure special price gets reflected only for the set date - P1
- When special price for item is set and date is not mentioned, verify that the special price should be displayed throughout - P1

- Enable catalog price rule and set a special price and verify the price that gets displayed for that product - P1
- Verify that on changing special price more than once, the updated SP gets reflected - P2

### *Customer Specific Price*

This feature is applicable for B2B customer mostly (Can be used in B2C scenario also).

- Ensure Cache hours has a text field which accepts only numbers and decimal values - P0
- Verify the price displayed when Special price is set in magento and when Price synchronization is enabled - P0
- Verify if Special price displayed in magento front end is same as Special price mentioned in M3 (OIS320)for that customer - P0
- Verify the price that gets displayed in magento when Customer specific price is lesser than the regular price - P1
- Verify the price that gets displayed in magento when Customer specific price is higher than the regular price - P1
- Set cache hours to a specific number of hours and verify price reflected in the front end is fetched from magento for the mentioned hours after which price should get updated from M3 - P0
- Verify that the number of hours for which the cached price is fetched from magento is calculated from the time the customer loads the specific product - P1
- Enable catalog price rule and set a special price and verify the price that gets displayed for the customer - P1
- Ensure the price displayed in product detail page remains the same throughout the entire checkout process - P2
- Change the customer specific price in M3 and verify if it gets reflected to the customer - P1
- Verify that if Cache hours is set to null, price is fetched from magento for 8 hours - P0
- Ensure CSP is displayed for all types of products for all types of customer – P0

### 11.2.3 Inventory

Inventory sync works in two ways

- Background Sync
- Real Time Sync

#### *Background Sync*

- Run the **Cron settings for stock update** and verify the inventory is updating for products accordingly in Magento end from M3.(P0)

#### *Real Time Sync*

- Use MMS310 program for inventory related changes and MWS060 for Balance Identity.

Ensure the following behavior based on the selected option

- On hand balance : updates the overall stock available in the inventory - P0
- Allocatable net : updates the stock that has been allocated till date - P0
- Available inventory : updates the available stock after stock has been allocated - P0
- Verify that stock update fetched from M3 (MWS060) is synced correctly - P0
- Verify that choosing yes updates the quantity and choosing no doesn't update the quantity for the following scenarios - P0

Real Time Sync has three different options for customer

#### *Sync On Product Load:*

- Ensure that stock gets updated from M3 on loading the product detail page, in both front end and magento side

#### *Sync on Add to Cart*

- Ensure that stock gets updated from M3 on adding product to cart, in both front end and magento side

#### *Sync on Checkout:*

- Ensure that stock gets updated during checkout from M3, in both front end and magento side
- Verify system behavior on enabling all the options to yes - P1
- Verify user is given Yes/No options for all the three options - P1



#### 11.2.4 Customer Registration

##### *ERP Registration (B2B only)*

- Place an order by creating a new customer and verify that order cannot be placed and only items can be added to cart - P0
- Verify a new customer gets added in magento customer and ERP customer number gets generated - P0
- Ensure the customer is created with status 10 (CRS610) - P0
- Verify order can be placed after changing customer status to 20 - P0
- Change the status from 20 to 10 for an existing account and verify if order can be placed - P1

##### *Customer General Configuration*

###### *Customer Template Id*

Pre-requisite: Set create new customer to YES and enter a Customer template id

- Ensure you have a text field against customer template id that accepts alphanumeric char - P0
- Place an order by creating a new customer and verify order can be placed successfully and customer status is 20 in M3 - P0
- Verify the order can be synced to M3 without errors and final order number is Generated - P0
- Verify the final order number in M3 is created with new customer id as mentioned in template and a new ERP customer number is generated for the registered user once order is placed - P0
- Verify order can be placed using an existing customer - P1

###### *Common Customer Id*

Pre-requisite: Set create new customer to NO and enter a common customer id

- Ensure you have a text field against Common customer id that accepts alphanumeric Char - P0
- Placing an order by creating a new customer should be possible - P0
- Verify the order can be synced to M3 without errors and final order number gets generated in magento - P0
- Verify the final order number in M3 (OIS300) is created with the customer id that is mentioned in Common customer id - P0
- Enter an invalid common customer id and try to create a new customer and place order and verify that order and customer gets added in magento but throw's error upon syncing the order - P1
- Once order is synced verify the customer list in magento to ensure that a new customer gets added - P1
- Verify the customer gets added in M3 (OIS300) and status is 20 - P0

- Verify order can be placed using an existing customer - P1

Pre-requisite: Set create new customer to NO and do NOT enter a common customer id

- Verify if order can be synced by placing an order using existing customer - P2
- Place an order by creating new customer and verify that order gets placed in magento and customer gets added but cannot be synced with M3 and throws an error as failed to send to ERP - P2

### 11.2.5 Order Creation

- Verify that different product types can be placed in a single order and the order can be synced to M3 without issues - P0
- Verify individual product types can be placed in an order and the order can be synced to M3 without issues - P0
- Verify that order can be synced to M3 for orders placed with different order type - P0
- Verify order can be synced to M3 for items from different item types - P0
- Verify the order status updates correctly from M3 for each order status (33,44,66,77) and finally Complete state in both magento admin and in My Orders page - P0
- Add a tracking number detail panel F of the delivery in 'Delivery. Open Toolbox' (MWS410) and verify tracking number is synced in Magento. - P1
- "Verify that different product types can be placed in a single order and the order can be synced to M3 without issues for different types of customers and invoices are generated for all types of customers" P0
- Verify the order is placed based on the config setting (Basic) - P0
- Verify order sync with the following scenarios:
  - Enter a name that includes alphanumeric and special characters in all the fields such as first name,middle name and last name - P2
  - Enter gmail id with small and capital letters - P2
  - Enter invalid data for zipcode, address, city, fax - P2
  - Enter maximum values in address fields and create customer - P2

### 11.2.6 Order Status

Order status verifies synchronization of the status of the customer order within M3 with the corresponding order in Magento.

Verify When the *Highest status* is changed in M3 – the Magento order status is updated as per the following table:

M3 Highest Status	Should update Magento status to	Magento State	Status Text
22	erp_sts22	Processing	22-Order sent to warehouse
23	erp_sts23	Processing	23-Order sent to warehouse
24	erp_sts24	Processing	24-Order being packed
26	erp_sts26	Processing	26-Order partially shipped
27	erp_sts27	Processing	27-Order partially invoiced
33	erp_sts33	Processing	33-Order sent to warehouse
34	erp_sts34	Processing	34-Order being packed
36	erp_sts36	Processing	36-Order partially shipped
37	erp_sts37	Processing	37-Order partially invoiced
44	erp_sts44	Processing	44-Order being packed
46	erp_sts46	Processing	46-Order partially shipped
47	erp_sts47	Processing	47-Order partially invoiced
66	erp_sts66	Processing	66-Order shipped
67	erp_sts67	Processing	67-Order partially invoiced
77	erp_sts77	Processing	77-Order invoiced
		Complete	Complete

### 11.2.7 Shipment & Tracking

- Verify that Shipping method has a dropdown with options defined in Magento and all options are functional - P0
- Verify M3 Delivery method and M3 delivery term have a text box that accepts alphanumeric characters - P0
- Verify that the rate defined in magento applies to shipping fee correctly (OIS300) - P0
- Verify that Delivery method and Delivery term mapped to a order is based on the information provided in the config - P0
- Verify the Delivery method and Delivery term mapped to a order, when the fields are empty - P2
- Verify the Delivery method and Delivery term mapped to a order, when invalid values are entered in the fields - P2
- Verify the value that gets mapped when Delivery method has a valid value and Delivery term has a invalid value and vice versa - P2
- Verify the value that gets mapped when Delivery method is empty and Delivery term has a valid value and vice versa - P2
- Verify the value that gets mapped when Delivery method is empty and Delivery term has a invalid value and vice versa - P2
- Verify that clicking the add button will add all the fields necessary for shipment method and delete button will delete the entire row added - P0
- Verify the shipping charges that applies when more than one shipping method is defined - P1

### 11.2.8 Order Charges

#### *Tax charge*

- Verify that tax transfer has options yes/no and are functional - P0
- Verify that choosing yes will sync the tax charge to M3 and choosing no, will not sync the tax to the order in M3 - P0
- Verify that on choosing yes, an additional option called Tax charge code appears - P0
- Verify that Tax charge code has options fetched from M3 and are functional - P0
- Verify the chargers when charge code is Tax1 - P0
- Verify the chargers when charge code is other than Tax1 - P1
- Verify that tax is applied as order line charge in M3(OIS300) - P0

#### *Shipping charge*

- Verify Shipping/Freight fee charge type has options charges and non stock items - P0
- Verify Shipping/Freight fee charge code options vary based on the option chosen in charge type - P0
- Verify that when Shipping/Freight fee charge type is charges, the shipping charge is applied at order header level and choosing non stock items applies charge at order line level - P0

#### *Gift wrap*

- Verify Gift wrap transfer has dropdown with yes/no values and are functional - P0
- Verify choosing yes will show additional fields, Gift wrap charge type, gift wrap order charge code, gift wrap line charge code - P0
- Verify Gift wrap charge type has dropdown with options charges and non stock items and are functional - P0
- Verify that based on the option charges or non stock, values in Gift wrap order charge code and line charge code vary with respective values fetched from M3 - P0
- Verify that choosing charges in charge type will apply charges at order header level and choosing non stock item will apply charge at order line level - P0
- Verify the value that is applied when charge code is chosen as GIFT1,2 - P0
- Verify the value that is applied when charge code other than GIFT is chosen - P1
- Ensure the above scenarios works well with all products types- P0

### 11.2.9 Sales

- Ensure that Enable order comments has options yes/no and are functional - P0
- Verify that Enable order comments set to yes will enable user to add header level comments while placing the order in the front end and choosing no will disable the option for entering comments - P0
- Ensure that Enable ship complete has options yes/no and are functional P0
- Verify that choosing Enable ship complete to yes, will enable a checkbox in the front end to the customer and choosing no will disable that option - P0
- Verify that when customer checks the Enable ship complete while placing the order in the frontend, all the products get delivered at a single delivery and unchecking it will enable multiple deliveries based on the number of orders - P0
- Verify the functionality on M3 when ship complete is enabled/disabled - P1
- Ensure that display invoice has options yes/no and are functional - P0
- Verify that setting Display invoice to yes will fetch updated invoice from M3, and setting it to no will not display any invoice in the frontend - P0
- Ensure Send confirmation when external order placed has options yes/no and are functional - P0
- Verify that confirmation mail is sent when order is placed and when order is shipped - P0

### 11.2.10 Order History

- Verify My order page has 2 views, ERP orders and Magento orders respectively - P0
- Ensure the following columns are displayed in ERP order page - P0
  - Magento order number
  - ERP order number
  - Order date
  - Customer PO
  - Status
- Ensure the following columns are displayed in Magento order page - P0
  - Order number
  - Date
  - Ship to
  - Order Status
  - Order total
- Verify Sort by and Show options are available in ERP orders page and not in Magento orders page - P0
- Verify the Sort by has the following options and are functional: - P0
  - Magento order number
  - ERP order number
  - Order date
  - Customer PO
  - Status
  - Arrow mark which arranges orders in increasing or decreasing value, which by default is set to ascending order
- Verify Show has options 50,100 and all and are functional - P0
- Verify there is Loadmore and Filter options in ERP orders page and not in Magento orders page - P0
- Verify that clicking loadmore option will fetch updated orders from M3 - P0
- Verify that clicking Filter will display the following options and are functional: - P0
  - Magento order number
  - ERP order number
  - Order date: From, To
  - Customer PO

Status with options: None,Pending,Processing,Shipped,Invoiced,Cancelled

- Reset and search button at the end and are functional - P0
- Verify there is an Export to CSV option in ERP order page to export the orders - P0
- Verify there is a View order hyperlink against every order in magento order page - P0
- Verify that clicking view order redirects user to order detail page with 2 tabs: Order information and Shipments - P0

- Verify order information tab has information about the following - P0
  - Order date
  - Shipping address
  - Billing address
  - Items ordered
  - Back to my orders hyperlink
- Verify shipment tab has information about the following - P0
  - Order date
  - Shipping address
  - Billing address
  - Items shipped
  - Shipment number
  - Print Shipment and Print all shipments hyperlink
  - Back to my orders hyperlink
- Verify that clicking ERP order number displays the following information: - P0
- Final order number as order number
  - Order date
  - Shipping address
  - Billing address
  - Shipping method
  - Payment method
- Order items which have the following information:
  - Line number
  - Item number
  - Name
  - Quantity
  - Sales price
  - Line Amount
  - Grand total
  - Back to ERP order hyperlink

#### 11.2.11 Invoice History

- Ensure My Invoice page has the following - P0
  - Loadmore hyperlink
  - Export to CSV
  - Filter option that can be expanded or collapsed



- Ensure the Filter option has the following: - P0
  - Invoice number
  - Our order number
  - Invoice date : From, To
  - Customer PO/ref
  - Search button
- Verify Sort by and Show options are available - P0
- Verify the Sort by has the following options and are functional: - P0
  - Invoice number
  - Our date
  - Customer PO
  - Invoice date
  - Invoice amount
  - Outstanding amount
  - Arrow mark which arranges orders in increasing or decreasing value, which by default is set to ascending order
- Ensure the following columns are displayed - P0
  - Invoice number
  - Our date
  - Customer PO
  - Invoice date
  - Invoice amount
  - Outstanding amount
- Verify Show has options 50,100 and all and are functional - P0
- Verify that clicking loadmore option will fetch updated invoices from M3 - P0
- Verify each invoice in magento corresponds to invoice in M3 - P0

### 11.2.12 Payment

- Verify that Reauthorization amount has a text field that should accept only numbers and decimal - P0
- Verify that the amount mentioned here is the amount considered for authorizing credit card (OSI300) - P0
- Verify that the payment methods specified in Magento payment method, are the options for payment of order by the customer in the front end - P0
- Verify that clicking the add button will add all the fields necessary for payment provider and delete button will delete the entire row added - P0
- Verify that M3 payment terms and M3 payment methods have dropdowns with all options from M3 and are functional - P0
- Verify that M3 credit card provider has a text field which accepts alphanumeric characters - P0
- Verify the payment process for different combinations of payment terms and Methods - P1
- Verify the payment options when Payment provider is not configured - P1
- Ensure that in Invoice fee charge type both the options charges/non stock items can be chosen and are functional - P0
- Ensure that choosing charge will apply invoice fee at order header charge level and choosing non stock item will apply charge at the order line level (OIS300) - P0
- Verify that the Invoice fee amount is based on payment methods specified in payment method - P0
- Ensure that Invoice fee amount has a text field that should accept only numbers and decimal - P0
- Ensure that Invoice fee charge code and Invoice fee charge item no has a dropdown with invoice options from M3 which are functional - P0
- Verify the invoice fees mentioned in Invoice fee amount applies for invoice when the option INVF1 is chosen in Invoice fee charge code and mapped with NS-INVOICEFEE in Invoice fee charge item no - P0
- Verify the invoice fee charge when invoice fee is not configured - P1
- Verify that clicking the add button will add all the fields necessary for Invoice fee and delete button will delete the entire row added - P0
- Verify the amount that applies for invoice when mapping apart from the above values are done - P1
- Ensure that Credit card refund type has dropdown with options manual/automatic which are functional - P0

### 11.2.13 General

#### *Email*

- Verify that if email is mentioned in the field, any errors are sent to the email address mentioned in the field and not to any other email - P0
- Verify that if email is not mentioned in the field, errors are not mailed - P0
- Verify errors are mailed only when the email address exist - P1
- Ensure the field accepts alphanumeric characters - P2
- Ensure that only letters, number, period and underscore and @ should be accepted. - P2
- Entering any sequence of underscore or periods, any other special characters should throw an error - P2
- Ensure that only valid email formats are accepted - P2

#### *Service URL*

- Verify that service URL has a text field that accepts alphanumeric characters and special characters - P0
- When URL is empty or invalid/wrong, verify that trying to sync anything from magento to M3 throws an error - P1
- Verify that connection works and options from M3 are displayed only on running the cron job Leanswift\_config - P0
- Verify if values get changed, when URL is edited and saved - P1

#### *ERP order history*

- Verify that choosing yes in ERP order history, displays order history from ERP in the frontend - P0
- Verify that choosing no in ERP order history, displays order only from magento in the frontend - P0
- Test Mode
- Verify when test mode is set to yes and there are any issues with M3 connectivity, errors are displayed in log - P0
- Verify when test mode is set to no, Connectivity issues are not captured in logs - P0

#### *Debug/log data*

- Verify that logs get captured when debug/log data is set to yes - P0
- Verify logs doesn't get captured when debug/log data is set to no - P0
- 
- Ensure that pressing save, saves the config and the message 'The Configuration has been saved' message displays - P0

#### 11.2.14 Other Functional Scenario

- Verify system behavior when Magento server is down and when it recovers all the data are detained and eConnect functionality is working fine.(P1)
- Verify browser crashes/closed when manual sync is initiated from Magento and verify how system behaves.(P1)
- Verify user is notified when Connector is down with email id specified in Email field.(P0)  
Verify logs are formatted properly and readable. (P1)

#### 11.12.15 Security

- Fill in the Authentication section with valid data in Web Service Username, web Service Password, Client ID, Client Password and then perform any eConnect operation like Sync order with M3. Verify that the sync works fine without issue and log has token id generated. - P0
- Verify all the options have a text box that can accept alphanumeric characters - P0
- Verify that communication to M3 happens when authentication details are filled correctly - P0
- Verify if M3 values get populated when either of the fields is wrong or empty - P1
- Verify if sync works when either of the fields is wrong or empty - P1
- Verify that logs do not display the credentials - P0
- Verify when access token expires, 'Access token invalid' message is displayed followed by 'Initialize new access token in logs - P1
- Verify new access token is initialized and connectivity works fine, when access token expires - P0

### 11.12.16 Cron Scheduler

- Verify Cron scheduler executes in correct time interval by verifying in System -> Scheduler -> List view.

Below are list of scenarios for verifying Cron job scheduler

Cron Job	Scheduler Job	Scenario
<b>Cron settings for orders sent to ERP</b>	leanswift_send_orders_to_erp	1) Verify cron expression value set in eConnect is reflected in scheduler leanswift_send_order_to_erp
		2) Run the scheduler job and verify Temp Order#, Final Order # is updated for sales order in Order grid.
		3) Create a new order in M3 and run the cron "leanswift_send_orders_to_erp" and verify the order is not reflected in Magento end
<b>Cron settings for getting ERP Orders</b>	leanswift_get_orders_from_erp	1) Verify cron expression value set in eConnect is reflected in leanswift_get_orders_from_erp.
		2) Create a new order in M3 and run the cron "leanswift_send_orders_to_erp" and verify the order is reflected in Magento end with Temp order# and Final order No #
		3) Run the cron "leanswift_send_orders_to_erp" and verify Temp Order#, Final Order # is updated for sales order in Order grid and not for the order created in M3.
<b>Cron settings for getting ERP Invoice</b>	leanswift_get_invoice_from_erp	1) Verify cron expression value set in eConnect is reflected in leanswift_get_invoice_from_erp.
		2) Run the scheduler cron and verify ERP invoice is reflected in front end under My Invoice
<b>Cron settings for stock update</b>	leanswift_update_stock	1) Verify cron expression value set in eConnect is reflected in leanswift_update_stock.
		2) Update quantity in Magento/M3 side and run the scheduler job and verify stock is updated with correct numbers

<b>Cron settings for ERP order synchronization</b>	leanswift_send_orders_to_erp	1) Verify cron expression value set in eConnect is reflected in scheduler "leanswift_sheduler_order_sync"
		2) Run the scheduler job "leanswift_send_orders_to_erp" and verify Temp Order# , Final Ord # is updated for sales order in Order grid
	leanswift_sheduler_order_sync	3) Update the status of order , tracking info , Shipping details and run the cron "leanswift_sheduler_order_sync" to verify the status is updated in Magento sales grid
		4) Update the status of order, tracking info, Shipping details and run the cron "leanswift_return_order_sync" and check the status is not updated in Magento in sales grid.
<b>Cron settings for ERP return order sync</b>	leanswift_return_order_sync	1) Verify cron expression value set in eConnect is reflected in leanswift_return_order_sync
		2) Run the scheduler "leanswift_return_order Sync" and verify RMA order and their status is updated.
		2) Run the scheduler leanswift_send_order_to_erp and verify the cron is not updating any data in return order
<b>Cron settings for sync product</b>	leanswift_sync_product	1) Verify cron expression value set in eConnect is reflected in leanswift_sync_product
		2) Run the scheduler "leanswift_sync_product" and verify if the product attributes are updated properly

## 12.0 Privacy/Security Testing

- Edit HTML/JAVA script source code from client side and verify the change made should not get reflected in client side production environment.
- Edit some configuration in LeanSwift connector page, Copy and paste the url in different browser, verify user is shown login Admin panel screen.
- Verify on clicking on different Admin menu option user is displayed different Key in url
- Try passing some queries in URL and do URL manipulation and verify the system behavior.
- Try entering special characters in text field of LeanSwift connector page and verify the system behavior if it crashes DB.
- Identify user roles and the functionality is restricted/usable as per roles.
- Test for password quality wherever applicable.
- Check for browser cache or cookies if it stores user critical information like credit card.
- Check what happens if the request times out response from server.

## 13.0 Performance Testing

- Check for time taken to populate **M3 Batch Order#** for 200 orders by Manual sync , performing **Send Order to M3**
- Check for time taken to populate M3 data for 200 orders by Manual sync , performing **Sync with M3**
- Check for time taken to populate Tracking info for 200 orders by Manual sync , performing **Sync Tracking info**
- Update 5000 product in M3 side and run the cron, Cron settings for stock update and verify time taken for inventory to get updated in Magento side.
- Order more than 5000 products from client side and run the **Cron settings for orders sent to ERP** in LeanSwift Connector page and verify time for order to get processed and updated in M3 side.
- Generate 5000 invoices in M3 side and run the **Cron settings for getting ERP Invoice**. Verify time taken for invoices to get updated in Magento side.
- Complete the order for 5000 products in M3 side and run the **Cron settings for getting ERP Orders**. Verify time taken to update the order in M3 side.
- Run all the above mentioned cron at same time and verify the system behavior and their functionality.
- Verify all the request and response threads are handled properly for multiple request.
- Enable out of stock for 1000 products in M3 side and verify time take for Out of Stock to replicate in Magento front end when **Sync on Add to Cart** is enabled for 1000 products.
- Enable out of stock for 1000 products in M3 side and verify time take for Out of Stock to replicate in Magento front end when **Sync on Product Load** for 1000 products.
- Enable out of stock for 1000 products in M3 side and verify time take for Out of Stock to replicate in Magento front end when **Sync on Checkout** for 1000 products.

## 14.0 Compatibility Testing

Verify eConnect integrates seamlessly with different version of Magento and M3 after upgrade or downgrade and works properly.

Test environment has only latest version of Magento and M3. Depending on client set up appropriate version of M3 and Magento will be used for testing.



## 15.0 Localization

Verify LeanSwift Connector page is localized with different language and all the functionality is working as expected when connected to Magento and M3 system.

- Verify My Order, My Invoice in Magento front end is localized with respect to language selected in Magento platform.

## 16.0 Automation

Automation would be initiated once the product is stable and cases would be cases in such a way that would cover end customer scenario. Feasibility of different automation tools in the market would be studied up on and appropriate tools would be chosen based on parameter that would add value to the product.

**Note: Depending on different stores configured (B2B, B2C) in Magento site all the scenarios would be executed across all the available stores.**

## 17.0 Appendix A – M3 Programs/setups used

Program	Description	Setting level
CRS055	M3 Currencies are defined here. Used to define what currency M3 orders are created in	Store view
CRS076	M3 Payment methods are defined here. Used to define the Payment method used with a Magento Payment method	Website
CRS075	M3 Payment terms are defined here. Used to define the Payment term used with a Magento Payment method	Website
CRS070	M3 Delivery methods are defined here. Used to define the Delivery method used with a Magento Shipping method	Website
CRS065	M3 Delivery terms are defined here. Used to define the Delivery term used with a Magento Shipping method	Website
OIS030	M3 Header Charges are defined here. Used to define the charge codes used for freight and invoice fees	Freight = Website; Invoice fee = Website
CRS275	M3 Line Charges are defined here. Used to define the line charge codes used for a discount and taxes	Website
CRS008	M3 facilities are defined here. Used to define what facility M3 orders are created in	Website
MMS005	M3 warehouses are defined here. Used to define what warehouse M3 orders are created in	Website

Program	Description	Setting level
CRS610	M3 customers are defined here. Used to define the customer number used to either create an order or to use as a template to create a new customer	Website
MMS001	M3 items are defined here. Used to define the item number used for freight and invoice fees	Freight = Website; Invoice = Website;
OIS017	M3 Price lists are defined here. Used to define what Price list is used to store the RRP's	Store view
OIS010	M3 Customer Order types are define here. Used to define what order type is used for normal and RMA orders	Orders = Website; Returns = Website

## 18.0 Appendix B – MITMAS fields for attribute mapping

MMAAD0	ExtendedqaUsed
MMAAD1	GenerateTxtFromTextTemplate
MMACCG	AcceptanceGroup
MMACHK	AssortmentCheck
MMACMC	CompatibilityCode
MMACRF	UserdefinedAccountingControlObject
MMACTI	ActiveOrCatchWeightItem
MMAEQY	ExtraQuantity
MMAES2	ExtendedrangeTwinengineOperations
MMAESC	CriticalityClass
MMALUC	AlternateUmInUse
MMALUN	AlternateUm
MMAMOR	MaintenanceoverhaulrepairClass
MMAMPT	RepairTime
MMARPA	RepairSupplier
MMARPR	RepairPlanner
MMARR1	RemovalClass
MMATMN	AttributeManaged
MMATMO	AttributeModel
MMATNR	AttributeNumber
MMAUTC	IssueMessage
MMAWDY	RepairWarrantyDays
MMBACD	LotNumberingMethod
MMBGRP	BonusGroup

MMBOGR	BonusGenerating
MMBPEY	UpdateBuyingPattern
MMBUAR	BusinessArea
MMBYPR	BycoproductCode
MMCAWP	LocationbasedBatchweight
MMCCCM	CostingModelCoreCharge
MMCCI1	CoreChargeItemNumber
MMCCO1	CalculationConstant1
MMCCO2	CalculationConstant2
MMCCO3	CalculationConstant3
MMCETY	LotRestrictedRelease
MMCFI1	UserdefinedField1Item
MMCFI2	UserdefinedField2Item
MMCFI3	UserdefinedField3Item
MMCFI4	UserdefinedField4Item
MMCFI5	UserdefinedField5Item
MMCHCD	ConfigurationCode
MMCHCL	ChargeCalculation
MMCHID	ChangedBy
MMCHNO	ChangeNumber
MMCONC	ContinuousNetChange
MMCONO	Company
MMCPUN	CostUnitOfMeasure
MMCRI1	CoreRemanufacturingItemNumber
MMCUCD	Currency
MMCUCS	CurrencySalesPrice
MMCUEX	CustomerTableExtension

MMCWUN	CatchWeightUnitOfMeasure
MMDCCD	NumberOfDecimalPlaces
MMDIGI	DiscountGroupItem
MMDIM1	Measurement1
MMDIM2	Measurement2
MMDIM3	Measurement3
MMDIRR	DimensionalRemovalRate
MMDOID	DocumentIdentity
MMDPID	PeriodFrame
MMDTID	DataIdentity
MMDWNO	DrawingNumber
MMECAC	EngineeringChangeOrderType
MMECCC	ConsumptionCodeTradeStat(tst)
MMECFL	ActiveEngineeringChanges
MMECMA	EcoManaged
MMECVE	RevisionNumber
MMEPCD	YieldCalculation
MMETRF	ExternalInstruction
MMEVGR	EnvironmentGroup
MMEXCA	Exchangeable
MMEXPD	ExpirationDateMethod
MMFCCM	ForecastLogic
MMFCU1	CapacityFreeCapacityUnit
MMFEBA	ToolReporting
MMFRAG	Fragility
MMFRE3	StatisticsIdentity3Item
MMFRE4	StatisticsIdentity4Item

MMFUDS	Description2
MMFUNC	FunctionArea
MMGRMT	GoodsReceivingMethod
MMGRP1	SearchGroup1
MMGRP2	SearchGroup2
MMGRP3	SearchGroup3
MMGRP4	SearchGroup4
MMGRP5	SearchGroup5
MMGRTI	GroupTechnologyClass
MMGRTS	DistributionGroupTechnology
MMGRWE	GrossWeight
MMHAC1	DangerClass1
MMHAC2	DangerClass2
MMHAC3	DangerClass3
MMHAZI	DangerIndicator
MMHDPR	MainProduct
MMHIE1	HierarchyLevel1
MMHIE2	HierarchyLevel2
MMHIE3	HierarchyLevel3
MMHIE4	HierarchyLevel4
MMHIE5	HierarchyLevel5
MMHVMT	HarvestMethod
MMIACP	AttributecontrolledItem
MMIEAA	ItemExistsAsAliasIdentity
MMIHEI	Height
MMILEN	Length
MMINDI	LotControlMethod

MMITCL	ProductGroup
MMITDS	Name
MMITGR	ItemGroup
MMITNE	ExtendedItemNumber
MMITNO	ItemNumber
MMITRC	IndividualItemTracing
MMITRF	InternalInstruction
MMITRU	IncludeInTerritorialRollup
MMITTY	ItemType
MMIWID	Width
MMLAMA	MinimumAcceptedContributnMarginRatio
MMLEAI	InternalLeadTime
MMLMDT	ChangeDate
MMLMTS	Timestamp
MMMABU	MakebuyCode
MMMABL	PlannerMaintenance
MMMABN	MaintenancePlanning
MMMBFR	MeterFailure
MMMBSR	MeterScheduledShopVisits
MMMBUR	MeterUnscheduledRemovalRate
MMMDIR	MeterDimensioningRemovalRate
MMMES1	Meter1
MMMES2	Meter2
MMMES3	Meter3
MMMES4	Meter4
MMMOTP	Modelsite
MMMPGM	MaintenanceProgram



MMMVA1	MeterValue1
MMMVA2	MeterValue2
MMMVA3	MeterValue3
MMMVA4	MeterValue4
MMNESA	NextShopActionType
MMNEWE	NetWeight
MMNMIT	NonmaterialType
MMNPTO	ToolQuantity
MMNSUF	NextService
MMORTY	OrderType
MMOTDI	OrderTotalDiscountGenerating
MMPDCC	NumberOfPriceDecimalPlaces
MMPEQ1	QuantityOrTime
MMPEQ2	QuantityOrTime
MMPEQ3	QuantityOrTime
MMPEQ4	QuantityOrTime
MMPET1	TriggerType
MMPET2	TriggerType
MMPET3	TriggerType
MMPET4	TriggerType
MMPLCD	PlanningPolicy
MMPLMZ	PlanningHorizon
MMPOCY	NormalPotency
MMPPDT	PurchasePriceDate
MMPPUN	PurchasePriceUm
MMPRCM	CostingModelProductCosting
MMPRGP	ProcurementGroup

MMPRGR	CommissionGenerating
MMPROD	Manufacturer
MMPRVG	CommissionGroup
MMPUCD	PurchasePriceQuantity
MMPUPR	PurchasePrice
MMPUUN	PurchaseOrderUm
MMQACD	InspectionCode
MMRESP	Responsible
MMRGDT	EntryDate
MMRGTM	EntryTime
MMRIDC	ReferenceIdentityCode
MMRIDE	ReferenceIdentity
MMRMSG	ReturnableMessage
MMRNRI	ReturnablenonreturnableIndicator
MMSACD	SalesPriceQuantity
MMSAFC	SupplierabcCode
MMSALE	SalesItem
MMSAPR	SalesPrice
MMSCGR	ServiceLevel
MMSEPR	SetupPrice
MMSMFI	SingleManufacturer
MMSPAC	StoragePackaging
MMSPDT	SalesPriceDate
MMSPE1	Specification1
MMSPE2	Specification2
MMSPE3	Specification3
MMSPE4	Specification4

MMSPE5	Specification5
MMSPUC	FixedOrDynamicSalesPriceUm
MMSPUN	SalesPriceUnitOfMeasure
MMSTAT	Status
MMSTCD	InventoryAccounting
MMSTCN	StorageRequirements
MMSTCS	DowntimeCost
MMSTNC	StandardNumberingSystem
MMSTUN	StatisticalUm
MMSUNO	Supplier
MMTANK	LocationCharacteristic
MMTAXC	TaxCodeCustomeraddress
MMTBFR	TimeBetweenFailure
MMTBSR	TimeBetweenScheduledShopVisits
MMTBUR	UnscheduledRemovalRate
MMTECR	CoreTerms
MMTOHI	ToolHistory
MMTORE	ToolAllocation
MMTPCD	ItemCategory
MMTPLI	TemplateItemNumber
MMTRPA	TransportationPackaging
MMTXID	TextIdentity
MMUNMS	BasicUnitOfMeasure
MMUNNN	UnNumber
MMUNPA	UnPackCode
MMVOL3	Volume
MMVTCP	VatCodePurchase

MMVTCS	VatCodeSales
MMWADY	ManufacturerWarranty
MMWAL1	WarrantyLimit1
MMWAL2	WarrantyLimit2
MMWAL3	WarrantyLimit3
MMWAL4	WarrantyLimit4
MMWAPC	NormalWastePercentage
MMWSCA	CostingModel