EPOD Interface Test Plan

Aptean Ltd

Contents

1 EPOD Interface Test Plan

Test Script / Scenario Reference	*EPOD - Interface*	**Call Number(s): 292887**
Test Script / Scenario Description	*Test the Interface functionality*	**PASS / ISSUES / FAIL**
Menu Access	*EPOD Interface*	
Pre-requisites	*SoapUI*	**Tested By:**
Test Objective	*To ensure that the Interface operates as expected*	**Date:**

Step	Action	Result	Remarks	P/F
1	**General**			
	Open the EPOD Import Project within SOAP UI.			
1.01	Within the IMPORT_COMPLETE, alter the EPL_SITE_ID, EPL_USER_ID, and EPL_PASSWORD to values that are incorrect, run the request and revert the values to correct values.	The system will not allow the message to be processed, and will respond with a 'NAK' message advising of the reason.		
1.02	Within the IMPORT_COMPLETE, set the EPL_LOAD_ID to a value with length of 40.	The system will respond with a 'NAK' message advising of the error.		
1.03	Within the IMPORT_COMPLETE, set the EPL_JOB_TYPE to a value of X.	The system will respond with a 'NAK' message advising of the error.		
1.04	Within the IMPORT_COMPLETE, remove the EPL_JOB_ID tag from the first Job.	The system will respond with a 'NAK' message advising of the error.		
1.05	Run the IMPORT_COMPLETE request.	The system will create a new Load assigned to the EPL_USER_ID and create and assign 4 jobs to it. The jobs will have all fields populated and will match the following characteristics: 1. Collection Job with a Job Address Record, 3 Containers each with 2 products. A new customer record will be created, 2. Delivery Job with a Job Address Record, 3 Containers each with no products and 6 Loose Products. 3. Collection Job with no Job Address Record, 6 Loose Products. 4. Service Job with a Job Address.		

		A 'ACK' will be returned advising of the following: 1. A Load Created 2. A Job Created 3. A Customer Created 4. { A Container ,A Product,A Product } * 3 Created 5. A Job Created 6. A Container * 3 Created 7. A Product * 6 Created 8. A Job Created 9. A Product * 6 Created 10. A Job Created 11. A Service Created		
1.06	Run the UPDATE_COMPLETE request.	The Data imported in the previous test will be updated. The Load will be unassigned from the user. The jobs match the following characteristics: 1. This will be removed from the load. Product 'TESTPROD1' will be removed from the first container. A new Product 'NEW PRODUCT' will be added. Both the Job Address and Customer Address will be updated. 2. Container 1 will be removed and new Container 'NEW CONTAINER' will be added. 3. No Changes should occur. 4. No Changes should occur. 5. New Service Job Added with a Job Address Record. A 'ACK' will be returned advising of the following: 1. A Load Updated 2. A Job Updated		

		3. A Customer Updated 4. { A Container ,A Product,A Product } * 3 Created 5. A Job Updated 6. A Container * 3 Created 7. A Product * 6 Created 8. A Job Created 9. A Service Created		
1.07	Run the IMPORT_MINIMUM request.	Data will be imported with the minimum requirements met. A new load will be created with four jobs assigned to it. The jobs match the following characteristics: 1. Delivery Job with 2 containers each with 2 products 2. Delivery Job with 2 Loose Products. 3. A Service Job 4. A Delivery Job with No Details A 'ACK' will be returned advising of the following: 1. A Load Created 2. A Job Created 3. { A Container ,A Product,A Product } * 2 Created 4. A Job Created 5. A Product * 2 Created 6. A Job Created 7. A Service Created 8. A Job Created		
1.08	Run the IMPORT_LINKED request.	A linked Collection and Delivery Job with the same EPL_JOB_CODE assign to a newly created Load. The jobs will have all fields populated and will match the following characteristics:		

		1. Collection Job with a generated Job ID with 3 Containers each with 2 Products 2. Delivery Job with a generated Job ID with 3 containers each with 2 Products A 'ACK' will be returned advising of the following: 1. A Load Created 2. A Job Created 3. { A Container ,A Product,A Product } * 3 Created 4. A Job Created 5. { A Container ,A Product,A Product } * 3 Created		
1.09	Run the IMPORT_CUSTOMER. Set the third customer to a existing customer ID.	Two new customers will be created: one with the specified Customer Code the other with a generated one. The third customer should be updated. A 'ACK' will be returned advising of the following: 1. Customer Created * 2 2. Customer Updated		
1.10	Run the EPOD_EXPORT_JOB, populate the EPL_JOB_ID tag with the first contents of the EPL_JOB_ID from the Import_COMPLETE test.	The system will export full XML for the specified Job. Check that these details are correct.		
1.11	Run the EPOD_EXPORT_JOB_DATE, populate the EPL_START_PLANNED_DATE tag with the first contents of the EPL_START_PLANNED_DATE tag from the Import_COMPLETE test.	The system will export full XML for the all Jobs with a matching start date. Check that these details are correct.		
1.12	Run the EPOD_EXPORT_LOAD, populate the EPL_LOAD_ID tag with the first contents of the EPL_LOAD_ID from the Import_COMPLETE test.	The system will export full XML for the specified Load. Check that these details are correct.		
1.13	Run the EPOD_EXPORT_LOAD_DATE, populate the EPL_LOAD_START_PLANNED_DATE tag with the first contents of the EPL_LOAD_START_PLANNED_DATE from the Import_COMPLETE test.	The system will export full XML for all Loads with a matching start date. Check that these details are correct.