

ÍNDICE DE DOCUMENTACIÓN

Job 2026-01-23-174258-PROYECTOS-INBOUND · 108 documentos

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
1	PROYECTOS INBOUND/Inbound_CL/Client/ LMRY_CL_Inbound_CLNT.pdf	LMRY_CL_Inbound_CLNT.pdf		10-11
2	PROYECTOS INBOUND/Inbound_CL/Client/ LMRY_CL_Inbound_Detail_CLNT.pdf	LMRY_CL_Inbound_Detail_CLNT.pdf		12-16
3	PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_CLNT_HNDL.pdf	LMRY_CL_Inbound_CLNT_HNDL.pdf		17-20
4	PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_Detail_STLT_HNDL.pdf	LMRY_CL_Inbound_Detail_STLT_HNDL.pdf		21-29
5	PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_STLT_HNDL.pdf	LMRY_CL_Inbound_STLT_HNDL.pdf		30-36
6	PROYECTOS INBOUND/Inbound_CL/ LMRY_CL_Inbound_XML_STLT.pdf	LMRY_CL_Inbound_XML_STLT.pdf		37-38
7	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_Detail_STLT.pdf	LMRY_CL_Inbound_Detail_STLT.pdf		39-40
8	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_Log_Detail_STLT.pdf	LMRY_CL_Inbound_Log_Detail_STLT.pdf		41-43
9	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_Log_STLT.pdf	LMRY_CL_Inbound_Log_STLT.pdf		44-45
10	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_MPRD.pdf	LMRY_CL_Inbound_MPRD.pdf		46-50
11	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_STLT.pdf	LMRY_CL_Inbound_STLT.pdf		51-53
12	PROYECTOS INBOUND/Inbound_CL/Transaltor/ LMRY_CL_Inbound_Translator.pdf	LMRY_CL_Inbound_Translator.pdf		54-55
13	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Client/ LMRY_AR_InboundDetail_CLNT.pdf	LMRY_AR_InboundDetail_CLNT.pdf		56-60
14	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Client/ LMRY_AR_Inbound_CLNT.pdf	LMRY_AR_Inbound_CLNT.pdf		61-62
15	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Handlers/ LMRY_AR_InboundDetail_STLT_HNDL.pdf	LMRY_AR_InboundDetail_STLT_HNDL.pdf		63-69
16	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Handlers/ LMRY_AR_Inbound_CLNT_HNDL.pdf	LMRY_AR_Inbound_CLNT_HNDL.pdf		70-75
17	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Handlers/ LMRY_AR_Inbound_STLT_HNDL.pdf	LMRY_AR_Inbound_STLT_HNDL.pdf		76-81

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
18	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Handlers/ LMRY_AR_VendorBillInbound_URET_HNDL.pdf	LMRY_AR_VendorBillInbound_URET_HNDL.pdf		82-84
19	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundAFIP_MPRD.pdf	LMRY_AR_InboundAFIP_MPRD.pdf		85-87
20	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundAFIP_RSLT.pdf	LMRY_AR_InboundAFIP_RSLT.pdf		88-89
21	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundAFIP_SCHDL.pdf	LMRY_AR_InboundAFIP_SCHDL.pdf		90-91
22	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundPdf_STLT.pdf	LMRY_AR_InboundPdf_STLT.pdf		92-93
23	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundXML_STLT.pdf	LMRY_AR_InboundXML_STLT.pdf		94-96
24	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_InboundDetail_STLT.pdf	LMRY_AR_InboundDetail_STLT.pdf		97-98
25	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_InboundLogDetail_STLT.pdf	LMRY_AR_InboundLogDetail_STLT.pdf		99-101
26	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_InboundLog_CLNT.pdf	LMRY_AR_InboundLog_CLNT.pdf		102-102
27	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_InboundLog_STLT.pdf	LMRY_AR_InboundLog_STLT.pdf		103-105
28	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_Inbound_MPRD.pdf	LMRY_AR_Inbound_MPRD.pdf		106-110
29	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_Inbound_STLT.pdf	LMRY_AR_Inbound_STLT.pdf		111-112

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
30	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Plugin/ LMRY_AR_Inbound_Email_Capture.pdf	LMRY_AR_Inbound_Email_Capture.pdf		113-116
31	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Plugin/ LMRY_AR_Inbound_PLGNIMP.pdf	LMRY_AR_Inbound_PLGNIMP.pdf		117-118
32	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Plugin/ LMRY_AR_Inbound_PLGNTYPE.pdf	LMRY_AR_Inbound_PLGNTYPE.pdf		119-119
33	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Router/ LMRY_AR_Inbound_ROUT.pdf	LMRY_AR_Inbound_ROUT.pdf		120-121
34	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Setup/ LMRY_AR_InboundSetup_CLNT.pdf	LMRY_AR_InboundSetup_CLNT.pdf		122-124
35	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Setup/ LMRY_AR_InboundSetup_STLT.pdf	LMRY_AR_InboundSetup_STLT.pdf		125-128
36	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ Translator/LMRY_AR_Inbound_Translator.pdf	LMRY_AR_Inbound_Translator.pdf		129-129
37	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Client/ LMRY_CO_Inbound_CLNT.pdf	LMRY_CO_Inbound_CLNT.pdf		130-131
38	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Client/ LMRY_CO_Inbound_Detail_CLNT.pdf	LMRY_CO_Inbound_Detail_CLNT.pdf		132-137
39	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Handlers/ LMRY_CO_Inbound_CLNT_HNDL.pdf	LMRY_CO_Inbound_CLNT_HNDL.pdf		138-143
40	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Handlers/ LMRY_CO_Inbound_Detail_STLT_HNDL.pdf	LMRY_CO_Inbound_Detail_STLT_HNDL.pdf		144-153
41	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Handlers/ LMRY_CO_Inbound_STLT_HNDL.pdf	LMRY_CO_Inbound_STLT_HNDL.pdf		154-160

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
42	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Handlers/ LMRY_CO_VendorBillInbound_URET_HNDL.pdf	LMRY_CO_VendorBillInbound_URET_HNDL.pdf		161-164
43	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/ LMRY_CO_Inbound_Pdf_STLT.pdf	LMRY_CO_Inbound_Pdf_STLT.pdf		165-166
44	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/ LMRY_CO_Inbound_XML_STLT.pdf	LMRY_CO_Inbound_XML_STLT.pdf		167-169
45	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_InboundLogDetail_STLT.pdf	LMRY_CO_InboundLogDetail_STLT.pdf		170-172
46	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_InboundLog_CLNT.pdf	LMRY_CO_InboundLog_CLNT.pdf		173-174
47	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_InboundLog_STLT.pdf	LMRY_CO_InboundLog_STLT.pdf		175-177
48	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_Inbound_Detail_STLT.pdf	LMRY_CO_Inbound_Detail_STLT.pdf		178-179
49	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_Inbound_MPRD.pdf	LMRY_CO_Inbound_MPRD.pdf		180-183
50	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_Inbound_STLT.pdf	LMRY_CO_Inbound_STLT.pdf		184-185
51	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Plugin/ LMRY_CO_Inbound_Email_Capture.pdf	LMRY_CO_Inbound_Email_Capture.pdf		186-189
52	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Plugin/ LMRY_CO_Inbound_Email_Capture_STLT.pdf	LMRY_CO_Inbound_Email_Capture_STLT.pdf		190-191
53	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Plugin/ LMRY_CO_Inbound_PLGNIMP.pdf	LMRY_CO_Inbound_PLGNIMP.pdf		192-193

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
54	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Plugin/ LMRY_CO_Inbound_PLGNTYPE.pdf	LMRY_CO_Inbound_PLGNTYPE.pdf		194-194
55	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Router/ LMRY_CO_Inbound_ROUT.pdf	LMRY_CO_Inbound_ROUT.pdf		195-196
56	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Setup/ LMRY_CO_InboundSetup_CLNT.pdf	LMRY_CO_InboundSetup_CLNT.pdf		197-199
57	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Setup/ LMRY_CO_InboundSetup_STLT.pdf	LMRY_CO_InboundSetup_STLT.pdf		200-203
58	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Translator/ LMRY_CO_Inbound_Translator.pdf	LMRY_CO_Inbound_Translator.pdf		204-205
59	PROYECTOS INBOUND/sdf-latam-inbound/ com.latamready.Imrylataminbound/src/FileCabinet/ SuiteApps/com.latamready.Imrylataminbound/ Handlers/LMRY_Inbound_Uret_HNDL.pdf	LMRY_Inbound_Uret_HNDL.pdf		206-207
60	PROYECTOS INBOUND/sdf-latam-inbound/ com.latamready.Imrylataminbound/src/FileCabinet/ SuiteApps/com.latamready.Imrylataminbound/Urets/ LMRY_Inbound_URET.pdf	LMRY_Inbound_URET.pdf		208-209
61	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Client/ LMRY_MX_InboundDetail_CLNT.pdf	LMRY_MX_InboundDetail_CLNT.pdf		210-215
62	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Client/ LMRY_MX_Inbound_CLNT.pdf	LMRY_MX_Inbound_CLNT.pdf		216-217
63	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/ Handlers/LMRY_MX_InboundDetail_STLT_HNDL.pdf	LMRY_MX_InboundDetail_STLT_HNDL.pdf		218-225
64	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/ Handlers/LMRY_MX_Inbound_CLNT_HNDL.pdf	LMRY_MX_Inbound_CLNT_HNDL.pdf		226-233
65	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/ Handlers/LMRY_MX_Inbound_STLT_HNDL.pdf	LMRY_MX_Inbound_STLT_HNDL.pdf		234-241

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
66	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ Handlers/ LMRY_MX_VendorBillInbound_URET_HNDL.pdf	LMRY_MX_VendorBillInbound_URET_HNDL.pdf		242-245
67	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundPdf_STLT.pdf	LMRY_MX_InboundPdf_STLT.pdf		246-247
68	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundSAT_MPRD.pdf	LMRY_MX_InboundSAT_MPRD.pdf		248-250
69	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundSAT_RSLT.pdf	LMRY_MX_InboundSAT_RSLT.pdf		251-252
70	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundSAT_SCHDL.pdf	LMRY_MX_InboundSAT_SCHDL.pdf		253-254
71	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundXML_STLT.pdf	LMRY_MX_InboundXML_STLT.pdf		255-257
72	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_InboundDetail_STLT.pdf	LMRY_MX_InboundDetail_STLT.pdf		258-259
73	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_InboundLogDetail_STLT.pdf	LMRY_MX_InboundLogDetail_STLT.pdf		260-262
74	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_InboundLog_CLNT.pdf	LMRY_MX_InboundLog_CLNT.pdf		263-264
75	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_InboundLog_STLT.pdf	LMRY_MX_InboundLog_STLT.pdf		265-267
76	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_Inbound_MPRD.pdf	LMRY_MX_Inbound_MPRD.pdf		268-271
77	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_Inbound_STLT.pdf	LMRY_MX_Inbound_STLT.pdf		272-274

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
78	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Plugin/ LMRY_MX_Inbound_Email_Capture.pdf	LMRY_MX_Inbound_Email_Capture.pdf		275-278
79	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Plugin/ LMRY_MX_Inbound_PLGNIMP.pdf	LMRY_MX_Inbound_PLGNIMP.pdf		279-281
80	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Plugin/ LMRY_MX_Inbound_PLGNTYPE.pdf	LMRY_MX_Inbound_PLGNTYPE.pdf		282-282
81	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Router/ LMRY_MX_Inbound_ROUT.pdf	LMRY_MX_Inbound_ROUT.pdf		283-284
82	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Setup/ LMRY_MX_InboundSetup_CLNT.pdf	LMRY_MX_InboundSetup_CLNT.pdf		285-289
83	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Setup/ LMRY_MX_InboundSetup_STLT.pdf	LMRY_MX_InboundSetup_STLT.pdf		290-293
84	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/ Translator/LMRY_MX_Inbound_Translator.pdf	LMRY_MX_Inbound_Translator.pdf		294-295
85	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Client/ LMRY_PE_InboundDetail_CLNT.pdf	LMRY_PE_InboundDetail_CLNT.pdf		296-303
86	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Client/ LMRY_PE_Inbound_CLNT.pdf	LMRY_PE_Inbound_CLNT.pdf		304-305
87	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Handlers/ LMRY_PE_InboundDetail_STLT_HNDL.pdf	LMRY_PE_InboundDetail_STLT_HNDL.pdf		306-313
88	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Handlers/ LMRY_PE_Inbound_CLNT_HNDL.pdf	LMRY_PE_Inbound_CLNT_HNDL.pdf		314-318
89	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Handlers/ LMRY_PE_Inbound_STLT_HNDL.pdf	LMRY_PE_Inbound_STLT_HNDL.pdf		319-327

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
90	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Handlers/ LMRY_PE_VendorBillInbound_URET_HNDL.pdf	LMRY_PE_VendorBillInbound_URET_HNDL.pdf		328-331
91	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/ LMRY_PE_InboundPdf_STLT.pdf	LMRY_PE_InboundPdf_STLT.pdf		332-333
92	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/ LMRY_PE_InboundXML_STLT.pdf	LMRY_PE_InboundXML_STLT.pdf		334-335
93	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Libraries/ LMRY_PE_Message_LBRY.pdf	LMRY_PE_Message_LBRY.pdf		336-337
94	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Module/ LMRY_PE_InboundDetail_STLT.pdf	LMRY_PE_InboundDetail_STLT.pdf		338-339
95	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Module/ LMRY_PE_InboundLogDetail_STLT.pdf	LMRY_PE_InboundLogDetail_STLT.pdf		340-342
96	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Module/ LMRY_PE_InboundLog_CLNT.pdf	LMRY_PE_InboundLog_CLNT.pdf		343-347
97	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Module/ LMRY_PE_InboundLog_STLT.pdf	LMRY_PE_InboundLog_STLT.pdf		348-350
98	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Module/ LMRY_PE_Inbound_MPRD.pdf	LMRY_PE_Inbound_MPRD.pdf		351-354
99	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Module/ LMRY_PE_Inbound_STLT.pdf	LMRY_PE_Inbound_STLT.pdf		355-356
100	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Plugin/ LMRY_PE_Inbound_PLGNIMP.pdf	LMRY_PE_Inbound_PLGNIMP.pdf		357-359
101	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Plugin/ LMRY_PE_Inbound_PLGNATYPE.pdf	LMRY_PE_Inbound_PLGNATYPE.pdf		360-360

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
102	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Plugin/ LMRY_PE_Inbound_mail_PLGIMP.pdf	LMRY_PE_Inbound_mail_PLGIMP.pdf		361-363
103	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Router/ LMRY_PE_Inbound_ROUT.pdf	LMRY_PE_Inbound_ROUT.pdf		364-365
104	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Setup/ LMRY_PE_InboundSetup_CLNT.pdf	LMRY_PE_InboundSetup_CLNT.pdf		366-370
105	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Setup/ LMRY_PE_InboundSetup_STLT.pdf	LMRY_PE_InboundSetup_STLT.pdf		371-374
106	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/ Translator/LMRY_PE_Inbound_Translator.pdf	LMRY_PE_Inbound_Translator.pdf		375-376
107	audit_report.pdf	audit_report.pdf		377-640
108	index_with_pages.pdf	index_with_pages.pdf		641-649

GENERADO AUTOMÁTICAMENTE 2026-01-26 14:22 UTC

CLIENTSCRIPT · GESTIÓN DE INBOUND CLIENTE CON HANDLER

SuiteScript 2.1 · Módulos: 3

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Facilita la gestión de registros inbound al delegar la lógica al handler y permite marcar o desmarcar rápidamente todas las líneas de una sublista.

Lifecycle Hooks

pageInit

Crea una instancia del handler y delega la inicialización del formulario

Lógica: instancia Clnthandler, guarda referencia global y llama a su método pageInit

Salida: handler inicializado y posible configuración del formulario

fieldChanged

Reenvía el evento de cambio de campo al handler

Lógica: llama a handler.fieldChanged con el mismo contexto

saveRecord

Ejecuta la lógica de guardado definida en el handler y devuelve su resultado

Lógica: retorna el valor devuelto por handler.saveRecord

Salida: true para permitir guardado, false para bloquearlo

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
apply	sublist dinámico (identificado en tiempo de ejecución)	escritura	establece el valor true o false en todas las líneas de la sublista indicada

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: El script delega la mayor parte de la lógica al handler externo; aquí solo se controla la marcación masiva de la casilla 'apply' en sublistas

Dependencias

NOMBRE	PROPÓSITO
./LMRY_CL_Inbound_CLNT_HNDL	maneja la lógica de negocio del cliente inbound
N/currentRecord	acceso al registro activo para manipular sublistas

Resumen Técnico

Instancia un handler en pageInit, reenvía eventos de campo y guardado, y ofrece una función para aplicar masivamente el valor de la casilla 'apply' en sublistas mediante N/currentRecord.

CLIENTSCRIPT · DETALLE DE INBOUND EN CLIENTE

SuiteScript 2.1 · Módulos: 6

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

El Client Script valida y controla la información de ítems, impuestos y facturas antes de aprobar o rechazar una nota de crédito en el detalle de inbound.

Lifecycle Hooks

pageInit

Inicializa la página, carga datos de configuración, listas de ítems y códigos de impuesto, y pre-puebla la sublista de ítems cuando corresponde.

Lógica: Obtiene valores de campos ocultos, determina el idioma del usuario, carga configuración por subsidiaria, obtiene ítems y códigos de impuesto, asigna opciones al campo de ítem y, si el XML contiene ítems, los inserta en la sublista con cálculos de precios e impuestos.

Salida: Modifica valores y opciones de campos en el formulario; devuelve true

validateField

Controla la habilitación del checkbox "apply" en la sublista de facturas según el estado del checkbox "assign_flag".

Lógica: Si el campo modificado es assign_flag, recorre la sublista custpage_bill_list y desactiva o activa el campo apply en cada línea según el valor del checkbox.

Salida: Devuelve true

fieldChanged

Responde a cambios en varios campos para buscar ítems, gestionar selección exclusiva de órdenes de compra, completar datos de ítem y recalcular totales.

Lógica: Ejecuta búsqueda de ítems cuando cambia custpage_search_items; gestiona exclusividad de po_checked; al cambiar custpage_item completa descripción, cantidad y valores de impuesto; al cambiar tax_code asigna la tasa correspondiente; al cambiar amount o tax_rate recalcula impuesto y total.

Salida: Devuelve true

validateLine

Guarda los datos de la línea actual de la sublista de ítems en un arreglo interno para su posterior uso.

Lógica: Si la sublista es custpage_items_list, llama a getPreviousLineData para almacenar los valores de la línea en itemValues.

Salida: Devuelve true

validateDelete

Elimina del arreglo interno la información de la línea que se está borrando en la sublista de ítems.

Lógica: Obtiene el índice de la línea actual y elimina el elemento correspondiente de itemValues.

Salida: Devuelve true

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
assign_flag	-	validación control UI	Al marcar o desmarcar habilita o deshabilita el checkbox apply en cada línea de la sublista custpage_bill_list
custpage_bill_list.apply	custpage_bill_list	control UI	Se desactiva cuando assign_flag está sin marcar y se activa cuando está marcado
custpage_search_items	-	búsqueda	Al cambiar dispara la búsqueda de ítems por nombre y actualiza las opciones del campo de ítem en la línea actual
po_checked	custpage_pos_list	validación control UI	Permite seleccionar solo una orden de compra; al marcar una desmarca y desactiva las demás
custpage_item	custpage_items_list	cálculo	Al seleccionar un ítem completa descripción, cantidad, código de impuesto y valores predeterminados según configuración
tax_code	custpage_items_list	cálculo	Al cambiar asigna la tasa de impuesto correspondiente al campo tax_rate
amount	custpage_items_list	cálculo	Al cambiar recalcula tax_amount y total_amount
tax_rate	custpage_items_list	cálculo	Al cambiar recalcula tax_amount y total_amount

Validaciones

UBICACIÓN	CAMPO	CRITERIO	ACCIÓN	BLOQUEA
updateParent	bill selection	Solo se puede asignar una factura a la nota de crédito	Muestra alerta "Only 1 bill can be assigned to a credit note" y bloquea la operación	Sí
updateParent	bill amount vs due	El monto total de la nota de crédito no puede ser mayor al saldo pendiente de la factura asignada	Muestra alerta "You cannot generate the credit note because its amount is greater than the amount due of the assigned bill" y bloquea la operación	Sí
updateParent	date	La fecha es obligatoria cuando la nota está aprobada y no se asignó factura manualmente	Muestra alerta "You must complete the date" y bloquea la operación	Sí
updateParent	total amount	La suma de los totales de línea debe coincidir con el monto total ingresado	Muestra alerta "You must complete the total amount indicated in the Total Amount field" y bloquea la operación	Sí
updateParent	currency	El campo moneda es obligatorio cuando la nota está aprobada	Muestra alerta "You must complete the Currency field" y bloquea la operación	Sí

Interacción con Usuario

alert: Ocurrió un error

Trigger: Excepción en pageInit, validateField, fieldChanged, validateLine, validateDelete o updateParent

alert: Only 1 bill can be assigned to a credit note

Trigger: Intento de asignar más de una factura en updateParent

alert: You cannot generate the credit note because its amount is greater than the amount due of the assigned bill

Trigger: Monto de la nota mayor al saldo de la factura en updateParent

alert: You must complete the date

Trigger: Falta la fecha requerida en updateParent

alert: You must complete the total amount indicated in the Total Amount field

Trigger: Desbalance entre suma de líneas y total en updateParent

alert: You must complete the Currency field

Trigger: Moneda no especificada en updateParent

Comportamiento del Script

VALIDA DATOS

Sí

BLOQUEA GUARDADO

Sí

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

Sí

Notas: El script controla la lógica de ítems, impuestos y validaciones de facturas antes de aprobar o rechazar la nota de crédito.

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener la preferencia de idioma del usuario
N/currentRecord	Acceder al registro actual del formulario
N/url	No utilizado directamente en la lógica mostrada
N/search	Buscar ítems, códigos de impuesto y configuraciones por subsidiaria
N/format	No utilizado directamente en la lógica mostrada
N/ui/dialog	Mostrar diálogos de alerta al usuario

Resumen Técnico

Implementa validaciones en eventos de página, campo, línea y borrado; gestiona listas dinámicas, cálculos de totales e impuestos, y bloquea la operación mediante alertas cuando alguna regla de negocio no se cumple.

INTERNALMODULE · MANEJADOR CLIENTE PARA INBOUND DE FACTURAS ELECTRÓNICAS

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador cliente para inbound de facturas electrónicas

Dependencias: N/RUNTIME N/SEARCH N/URL N/RECORD N/FORMAT N/CURRENTRECORD
N/QUERY N/CURRENCY N/UI/MESSAGE ./LMRY_CL_INBOUND_TRANSLATOR
/SUITEBUNDLES/BUNDLE 37714/LATAM_LIBRARY/LMRY_LOG_LBRY_V2.0

Resumen Ejecutivo

{'business_summary': Markup('Este módulo gestiona la pantalla cliente para la importación de facturas electrónicas, validando la información ingresada, permitiendo la visualización de detalles y registrando la selección para su procesamiento posterior. '), 'technical_summary': Markup('Expone una clase Clnthandler con métodos de ciclo de vida (pageInit, fieldChanged, saveRecord) y una función back; utiliza APIs de NetSuite para búsquedas, consultas, creación de registros y manejo de UI, y mantiene traducciones según el idioma del usuario. ')}

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener información del entorno y del usuario actual
N/search	Ejecutar búsquedas guardadas y consultas dinámicas
N/url	Construir URLs a Suitelets y scripts
N/record	Crear y guardar registros personalizados de log
N/format	Parsear y formatear fechas
N/currentRecord	Acceder al registro de cliente activo en la UI
N/query	Realizar consultas tipo SQL sobre entidades NetSuite
N/currency	Manejo de información de moneda (no usado directamente en lógica principal)
N/ui/message	Mostrar mensajes informativos al usuario
./LMRY_CL_Inbound_Translator	Obtener traducciones de textos según idioma
/SuiteBundles/Bundle 37714/Latam_Library/LMRY_Log_LBRY_V2.0	Biblioteca de logging (comentada en código)

API pública

ClntHandler

Clase que controla la lógica de la página cliente para inbound de facturas electrónicas

back

Navega de regreso a la suitelet principal, preservando parámetros de estado

Efectos colaterales

- Redirige la ventana del navegador a una URL de suitelet

Funciones internas

NOMBRE	ROL	USADO POR
validate_open_window	Verifica si la ventana de detalle ya está abierta para evitar duplicados	fieldChanged
validateMandatoryFields	Comprueba que los campos obligatorios tengan valor	saveRecord
validateDates	Valida que la fecha de fin no sea anterior a la de inicio	saveRecord
validateRangeDates	Asegura que inicio y fin pertenezcan al mismo mes y año	saveRecord
validateSublist	Confirma que al menos un ítem del sub-listado esté seleccionado	saveRecord
validateExecution	Detecta si ya existe una ejecución programada del proceso de inbound	saveRecord

Modelos de datos / estructuras

SetupConfig

Objeto que contiene la configuración de la subsidiaria para inbound

CAMPO	TIPO	DESCRIPCIÓN
taxcode0	Código de impuesto 0%	
taxcode19	Código de impuesto 19%	
department	Id de departamento predeterminado	
class	Id de clase predeterminada	
location	Id de ubicación predeterminada	
item	Id de ítem predeterminado	
client	Id de cliente para conexión	
password	Contraseña para conexión	
mode	Modo de operación	

LogRecord

Registro personalizado que almacena la información de los ítems seleccionados

CAMPO	TIPO	DESCRIPCIÓN
custrecord_lmry_cl_log_employee	Id del usuario que genera el log	
custrecord_lmry_cl_log_subsidiary	Subsidiaria asociada	
custrecord_lmry_cl_log_rut	RUT del cliente	
custrecord_lmry_cl_log_startdate	Fecha de inicio del rango	
custrecord_lmry_cl_log_enddate	Fecha de fin del rango	
custrecord_lmry_cl_log_data	JSON con detalle de los ítems	
custrecord_lmry_cl_log_status	Estado del log (1 = pendiente)	

INTERNALMODULE · MANEJADOR DE POPUP INBOUND CL

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador de Popup Inbound CL

Dependencias: N/URL N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET
/SUITEBUNDLES/BUNDLE 37714/LATAM_LIBRARY/LMRY_LIBSENDINGEMAILSLBRY_V2.0 N/HTTPS
N/FORMAT /SUITEBUNDLES/BUNDLE 37714/LATAM_LIBRARY/LMRY_MD5_ENCRYPT_LBRY_V2.0
N/XML N/QUERY N/CONFIG N/ENCODE ./LMRY_CL_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo muestra un popup con la información de un documento inbound chileno, permite al usuario aprobar, rechazar o crear la factura y asignar facturas manualmente cuando corresponde.'), 'technical_summary': Markup('Expone la clase LibraryHandler que, al instanciarse, parsea el XML inbound, consulta datos de proveedores, PO y periodos, y construye dinámicamente un formulario Suitelet con sublistas y botones según el estado del documento.')}

Dependencias

NOMBRE	PROPÓSITO
N/url	Construir URLs a registros y scripts
N/search	Ejecutar búsquedas guardadas y dinámicas
N/runtime	Obtener información del script, usuario y características activas
N/redirect	Redirigir a otro Suitelet
N/ui/serverWidget	Crear formularios, sublistas y campos de UI del servidor
/SuiteBundles/Bundle 37714/Latam_Library/LMRY_libSendingEmailsLBRY_V2.0	Obtener licencias de correo para envíos
N/https	Descargar el XML remoto
N/format	Parsear y formatear fechas
/SuiteBundles/Bundle 37714/Latam_Library/LMRY_MD5_encrypt_LBRY_V2.0	Encriptar/descifrar datos (no usado directamente en este módulo)
N/xml	Parsear el contenido XML del documento inbound
N/query	Obtener la lista de monedas del proveedor
N/config	Leer preferencias contables (p.ej. UNRECEIVEDBILLS)
N/encode	Convertir codificaciones de texto
./LMRY_CL_Inbound_Translator	Proveer traducciones de textos según idioma

API pública

LibraryHandler

Clase que encapsula la lógica de visualización y gestión de un documento inbound en un popup Suitelet

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto con propiedades params (parámetros del request) y method (GET o POST)	

Retorna

Instancia de LibraryHandler

createForm

Construye y devuelve el formulario del popup con campos, sublistas y botones según el estado del documento

Efectos colaterales

- log.error
- log.debug

Retorna

Objeto serverWidget.Form

setFormValues

Actualiza valores por defecto del formulario (subsidiaria, fechas, estado) y carga datos de la subsidiaria cuando corresponde

Efectos colaterales

- search.lookupFields

getOpenPeriodIds

Devuelve un arreglo con los IDs de periodos contables abiertos y no ajustados

Efectos colaterales

- search.create

Retorna

Array de strings

getDocumentType

Obtiene el catálogo de tipos de documento configurado para Chile

Efectos colaterales

- search.create

Retorna

Objeto clave-valor (código → nombre)

getEntities

Construye un mapa de proveedores activos en Chile (RUT → nombre)

Efectos colaterales

- search.create

Retorna

Objeto clave-valor

toLogSuitelet

Redirige al Suitelet de registro de logs de pagos

Efectos colaterales

- redirect.toSuitelet

getTranslations

Obtiene el diccionario de traducciones para el idioma indicado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código ISO de dos letras (p.ej. 'es')	

Retorna

Objeto con pares clave-texto

getText

Devuelve la cadena traducida para la clave solicitada

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
key	Identificador de texto en el diccionario	

Retorna

String

getDataXML

Descarga, decodifica y parsea el XML inbound, extrayendo información de cabecera, impuestos y detalle de ítems

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
xmlData	URL del archivo XML	

Efectos colaterales

- https.get
- log.debug

Retorna

Objeto con campos fechaEmision, fechaVenc, tipodocumento, rateIVA, IVA, purchaseorder, itemreceipt, items

convertDate

Convierte una cadena de fecha en formato YYYY-MM-DD a objeto Date

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
date	Cadena con formato 'YYYY-MM-DD'	

Retorna

Objeto Date o cadena vacía

getPeriods

Llena un campo de selección con los periodos contables abiertos y marca el periodo actual como predeterminado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
periodField	Campo serverWidget.Field de tipo SELECT	

Efectos colaterales

- search.create
- format.parse

Funciones internas

NOMBRE	ROL	USADO POR
addVendor	Helper que busca datos del proveedor y llena el campo SELECT correspondiente	createForm
createPOSublist	Construye la sublista de órdenes de compra y muestra enlaces a PO y XML/PDF	createForm
createSublist	Genera sublistas de PO, ítems y facturas según el estado del documento	createForm
createAssignManually	Crea la pestaña y sublista para asignar facturas manualmente	createForm
fillManualBill	Llena el campo SELECT de facturas manuales con facturas elegibles	createAssignManually
fillPoSublist	Pobla la sublista de PO con los resultados de la búsqueda	createSublist
getPurchaseOrder	Ejecuta la búsqueda de órdenes de compra que cumplen los filtros y devuelve un mapa de resultados	fillPoSublist
setStatus	Determina el estado final del documento a partir del XML (presencia de PO) y lanza la búsqueda de PO si corresponde	constructor
searchPO	Busca la orden de compra asociada al folio y actualiza statusFinal, idPO y totales	setStatus
setMessagePO	Genera el mensaje HTML que se muestra en la sublista PO según el estado	createPOSublist
addOptions	Agrega opciones a un campo SELECT a partir de una lista de resultados	createPOSublist , createSublist , getTaxCodes , getDepartments , getClasses , getLocations , getCurrencies , getPeriods
getTaxCodes	Obtiene los códigos de impuesto de compra activos para la subsidiaria	createSublist
getDepartments	Obtiene la lista de departamentos activos para la subsidiaria	createSublist
getClasses	Obtiene la lista de clases contables activas para la subsidiaria	createSublist
getLocations	Obtiene la lista de ubicaciones activas para la subsidiaria	createSublist

NOMBRE	ROL	USADO POR
getCurrencies	Recupera la moneda primaria y secundarias del proveedor mediante query	createSublist

Modelos de datos / estructuras

InboundXMLData

Representa la información extraída del XML inbound

CAMPO	TIPO	DESCRIPCIÓN
fechaEmision	Date	
fechaVenc	Date	
tipodocumento	String (código DTE)	
rateIVA	Number (tasa de IVA)	
IVA	Number (monto de IVA)	
purchaseorder	String (folio de PO asociado)	
itemreceipt	String (folio de receipt asociado)	
items	Array de objetos con name, quantity, price, discount, vendorCode (opcional)	

Generado automáticamente 2026-01-23 17:53 UTC

INTERNALMODULE · MANEJADOR DE SUITELET INBOUND CHILE

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador de Suitelet Inbound Chile

Dependencias: N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET
/SUITEBUNDLES/BUNDLE 37714/LATAM_LIBRARY/LMRY_LIBSENDINGEMAILSLBRY_V2.0 N/HTTPS
N/FORMAT /SUITEBUNDLES/BUNDLE 37714/LATAM_LIBRARY/LMRY_MD5_ENCRYPT_LBRY_V2.0
N/XML N/URL N/TASK ./LMRY_CL_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza la obtención y visualización de documentos tributarios inbound de Chile, facilitando a los usuarios la revisión y conciliación de facturas y notas de crédito recibidas del SII.'), 'technical_summary': Markup('Implementa un Suitelet que construye formularios dinámicos, invoca un SOAP al SII, parsea la respuesta XML y muestra los datos en una sublista con enlaces a PDFs, XML y registros de NetSuite, además de lanzar procesos Map/Reduce para procesamiento en segundo plano.')}

Dependencias

NOMBRE	PROPÓSITO
N/search	Ejecutar búsquedas y obtener datos de NetSuite
N/runtime	Obtener información del entorno, parámetros y características activas
N/redirect	Redirigir a otros Suitelets o scripts
N/ui/serverWidget	Construir formularios y sublistas en la UI del Suitelet
/SuiteBundles/Bundle 37714/Latam_Library/LMRY_libSendingEmailsLBRY_V2.0	Gestionar licencias y autorización para el envío de correos
N/https	Realizar llamadas SOAP al servicio web del SII
N/format	Parsear y formatear fechas y valores
/SuiteBundles/Bundle 37714/Latam_Library/LMRY_MD5_encrypt_LBRY_V2.0	Generar token MD5 para autenticación del web service
N/xml	Parsear y escapar XML de la respuesta del SII
N/url	Construir URLs a scripts, registros y recursos
N/task	Ejecutar procesos Map/Reduce en segundo plano
./LMRY_CL_Inbound_Translator	Obtener traducciones de textos según idioma

API pública

LibraryHandler

Clase que encapsula toda la lógica del Suitelet inbound para Chile, incluyendo generación de formularios, consumo del web service del SII y presentación de resultados.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto con propiedades params (parámetros de la petición) y method (GET o POST).	

Retorna

Instancia de LibraryHandler

areThereSubsidiaries

Verifica si existen subsidiarias habilitadas y con licencia válida; rellena la lista interna de subsidiarias.

Retorna

Booleano indicando si hay al menos una subsidiaria activa

getSubsidiaries

Obtiene la lista de subsidiarias activas en Chile que poseen RUT y dígito verificador.

Retorna

Arreglo de objetos {value, text, active}

createFilterForm

Construye el formulario de filtros del Suitelet, incluyendo campos de subsidiaria, fechas y contadores de documentos.

Retorna

Objeto serverWidget.Form

createApplySublist

Agrega la sublista de resultados al formulario, definiendo columnas según el estado y filtro seleccionado.

Retorna

Objeto serverWidget.Sublist

loadSublist

Pobla la sublista con los documentos obtenidos del web service, enlaza facturas o notas de crédito y genera enlaces a PDF/XML.

fillSubsidiaries

Rellena el campo SELECT de subsidiarias con las opciones activas.

getRedirectParams

Construye el objeto de parámetros para redirigir al Suitelet con estado 1.

Retorna

Objeto con keys subsidiary, startdate, enddate, status

setFormValues

Establece los valores por defecto del formulario (subsidiaria, fechas, RUT) a partir de los parámetros recibidos.

getDocumentType

Obtiene el catálogo de tipos de documento SII para Chile y devuelve un mapa código->nombre.

Retorna

Objeto map con códigos como claves y nombres como valores

getEntities

Construye un mapa de proveedores (RUT+digito) a su internalId y nombre, filtrado por subsidiaria y país.

Retorna

Objeto map {RUT+digito: {id, name}}

toLogSuitelet

Redirige al Suitelet de registro de logs.

Efectos colaterales

- **Type:** redirect; **Details:** Redirección a script customscript_lmry_cl_inbound_log_stlt

executeMapReduce

Inicia la ejecución del proceso Map/Reduce asociado al inbound, pasando el ID de log como parámetro.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
logIdparam	ID del registro de log a procesar	

Efectos colaterales

- **Type:** task; **Details:** Creación y envío de tarea Map/Reduce

getTranslations

Obtiene el diccionario de traducciones para el idioma especificado.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código ISO de dos letras (es, en, ...)	

Retorna

Objeto con pares clave->texto traducido

getText

Devuelve el texto traducido para la clave indicada.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
key	Clave del diccionario de traducciones	

Retorna

String con el texto traducido

getResponse

Construye la petición SOAP al SII, la envía, parsea la respuesta XML y devuelve los documentos encontrados.

Efectos colaterales

- **Type:** https; **Details:** Llamada POST al endpoint del SII
- **Type:** log; **Details:** Registro de depuración de la respuesta XML

Retorna

Objeto {results: [], request: string, response: string}

getBills

A partir de la lista de documentos, busca facturas vinculadas y devuelve un mapa folio->proveedor->detalle factura.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
data	Arreglo de resultados del web service	

Retorna

Objeto map {folio: {vendorId: {id, name}}}

getCredits

Similar a getBills pero para notas de crédito; devuelve un mapa folio->proveedor->detalle crédito.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
data	Arreglo de resultados del web service	

Retorna

Objeto map {folio: {vendorId: {id, name}}}

convertDate

Convierte una fecha en formato DD-MM-YYYY a la representación de NetSuite.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
date	String con formato DD-MM-YYYY	

Retorna

String con fecha formateada según NetSuite

Modelos de datos / estructuras

InboundResponse

Resultado de la llamada al web service del SII.

CAMPO	TIPO	DESCRIPCIÓN
results	Arreglo de objetos documento con campos issuer_name, issuer_rut, folio, total_amount, document_type, date, see_pdf, see_xml, status, statustype	
request	String XML de la petición SOAP (escapado)	
response	String XML de la respuesta (escapado)	

VendorMap

Mapa de proveedores usado para relacionar RUT con internalId y nombre.

CAMPO	TIPO	DESCRIPCIÓN
key	RUT + dígito verificador	
value	Objeto {id, name}	

Generado automáticamente 2026-01-23 17:54 UTC

SUITELET · FORMATO Y VISUALIZACIÓN DE XML INBOUND

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a usuarios obtener y visualizar de forma legible XML inbound desde una URL

Flujo de Ejecución

GET

POST

No implementado

Flujo: N/A

Genera: N/A

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_id_message	Message	INLINEHTML	Muestra mensaje de error cuando ocurre una excepción

Lógica POST

Procesa el XML recibido, normaliza la codificación, lo formatea y corrige caracteres especiales antes de enviarlo al cliente

Operaciones

TIPO	TARGET	PROPÓSITO
https.get	URL especificada en el parámetro xml	Obtener el contenido bruto del XML
encode.convert	Cadena XML	Convertir de Base64 a UTF-8 cuando la codificación no es ISO-8859-1
xml.Parser.fromString / xml.Parser.toString	Documento XML	Parsear el XML y volver a generar una cadena con sangrado

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Muestra un formulario de error con mensaje inline cuando ocurre una excepción

Dependencias

NOMBRE	PROPÓSITO
N/https	Realizar la petición GET al recurso XML externo
N/encode	Convertir la codificación del contenido XML cuando sea necesario
N/ui/serverWidget	Crear el formulario de error y el campo INLINEHTML
N/url	No utilizado directamente en la lógica actual
N/xml	Parsear y formatear el documento XML

Resumen Técnico

Recupera el XML vía HTTPS, normaliza la codificación, lo parsea con N/xml y lo devuelve como texto plano

SUITELET · DETALLE DE INBOUND CL

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar el detalle de los inbound de clientes en NetSuite

Flujo de Ejecución

GET

POST

Parámetros de Entrada

Lógica POST

La lógica GET delega la creación del formulario y sublistas al manejador externo; en caso de error genera un formulario de mensaje y envía una notificación por correo

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

Sí

Notas: Muestra formulario en GET y envía email si ocurre un error

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Crear formularios, campos y sublistas
N/runtime	Obtener información del script en ejecución
N/redirect	Disponible pero no utilizado en este fragmento
./LMRY_CL_Inbound_Detail_STLT_HNDL	Manejar la lógica de creación del formulario y sublistas
/SuiteBundles/Bundle 37714/Latam_Library/LMRY_libSendingEmailsLBRY_V2.0	Enviar notificaciones por correo electrónico en caso de error

Resumen Técnico

Genera dinámicamente un formulario y sublistas mediante un manejador externo y muestra errores con notificación por correo

SUITELET · DETALLE DE INBOUND CL - LOG

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar de forma detallada los documentos procesados en un log inbound de Chile, con enlaces directos a los registros originales.

Flujo de Ejecución

GET

POST

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
count	Contador	INTEGER	valor incremental, solo lectura
subsidiary	Subsidiaria	TEXT	solo lectura
typedocument	Tipo de Documento	TEXT	solo lectura
po	Orden de compra	TEXT	enlace HTML a la orden de compra, solo lectura
bill	Factura de compra	TEXT	enlace HTML a la factura, solo lectura
vendorcredit	Nota de crédito	TEXT	enlace HTML a la nota de crédito, solo lectura
status	Estado Enternet	TEXT	muestra éxito o error traducido, solo lectura
description	Descripción	TEXT	mensaje descriptivo del proceso, solo lectura

Lógica POST

Recupera los registros de detalle del log inbound, interpreta la respuesta XML para obtener el estado y la descripción, traduce los tipos de documento y construye la sublista con enlaces a los registros relacionados.

Búsquedas

Obtener los detalles del log inbound filtrado por idLog

Filtros:

Retorna: internalid, subsidiaria, folio, PO, factura, crédito, datos JSON y respuesta XML

Obtener la tabla de códigos de documento para traducir el tipo de documento

Filtros:

Retorna: código de documento y nombre descriptivo

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Solo lectura y presentación de datos; maneja errores mostrando un formulario informativo

Dependencias

NOMBRE	PROPÓSITO
N/xml	Parsear la respuesta XML del log para extraer estado y mensaje
N/url	Construir URLs de registro para los enlaces de PO, factura y crédito
N/ui/serverWidget	Crear el formulario y la sublista que se muestra al usuario
N/search	Ejecutar búsquedas de registros de detalle y de tipos de documento
N/redirect	Importado pero no utilizado en la lógica actual
N/runtime	Obtener la preferencia de idioma del usuario actual
./LMRY_CL_Inbound_Translator	Módulo personalizado, importado pero no usado en este script

Resumen Técnico

Construye un formulario con sublista usando N/ui/serverWidget, consulta registros mediante N/search, interpreta respuestas XML y traduce textos según la preferencia de idioma del usuario.

SUITELET · REGISTRO DE INBOUND LOG

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Este suitelet permite a los usuarios visualizar y acceder a los registros de inbound log de forma centralizada.

Flujo de Ejecución

GET

POST

Redirige al suitelet principal después de presionar el botón de envío

Flujo: Al recibir una solicitud POST se ejecuta una redirección a otro suitelet definido por scriptId y deploymentId

Genera: Redirección al suitelet principal

Lógica POST

En modo GET se recuperan los registros de log mediante una búsqueda paginada y se rellenan los campos de la sublista; en modo POST se redirige al suitelet principal

Búsquedas

Obtener los registros activos del customrecord_lmry_cl_inbound_log

Filtros:

Retorna: internalid, lastmodified, created, empleado, subsidiaria, fechas de inicio y fin, estado y datos del log

Operaciones

TIPO	TARGET	PROPÓSITO
search	customrecord_lmry_cl_inbound_log	Recuperar los logs para mostrarlos en la sublista
redirect	suitelet principal (customscript_lmry_cl_inbound_stlt)	Volver al menú principal después de enviar el formulario

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: Solo muestra datos y permite navegación; no modifica registros

Dependencias

NOMBRE	PROPÓSITO
N/url	Construir la URL del suitelet de detalle
N/ui/serverWidget	Crear el formulario y la sublista
N/search	Ejecutar la búsqueda de registros de log
N/redirect	Redirigir al suitelet principal en POST
N/runtime	Obtener la preferencia de idioma del usuario
./LMRY_CL_Inbound_Translator	Proveer traducciones de textos según el idioma

Resumen Técnico

Construye un formulario con sublista usando serverWidget, recupera datos mediante una búsqueda paginada y genera enlaces a detalle mediante url.resolveScript; el envío del formulario redirige al suitelet principal.

MAP/REDUCE · LMRY - CL INBOUND MPRD

Pipeline de procesamiento por lotes (SuiteScript Map/Reduce).

Procesos masivos, búsquedas y operaciones sobre registros.

Metadatos

Tipo: MapReduce

SuiteScript: 2.1

LMRY - CL Inbound MPRD

Resumen Ejecutivo

{'business_summary': Markup('Procesa los documentos de un log inbound de CL, genera facturas o notas de crédito según la respuesta del cliente y registra el resultado en detalle y en el log principal.'), 'technical_summary': Markup('Lee un registro de log, itera cada documento en la fase map, crea/transforma registros de NetSuite (vendorbill, vendorcredit), realiza llamadas SOAP y actualiza registros de detalle; la fase summarize marca el log como completado o con error.')}

getInputData

Obtiene el registro de log inbound mediante runtime parameter custscript_lmry_cl_log, realiza un lookupFields sobre customrecord_lmry_cl_inbound_log y devuelve cada elemento del campo data (JSON) como objeto del arreglo, añadiendo información de subsidiary, employee, fechas y rut.

Columnas

NOMBRE	ETIQUETA
custrecord_lmry_cl_log_employee	
custrecord_lmry_cl_log_subsidiary	
custrecord_lmry_cl_log_startdate	
custrecord_lmry_cl_log_enddate	
custrecord_lmry_cl_log_data	
custrecord_lmry_cl_log_status	
custrecord_lmry_cl_log_rut	

map

Parsea el documento, crea un registro detalle customrecord_lmry_cl_inbound_detail, determina la acción (aprobado o rechazado) y según filtros crea o transforma facturas de proveedor, notas de crédito o envía solicitud SOAP de aceptación de reclamo. Actualiza el detalle con los IDs generados y escribe en la salida la clave "1" con el id del log.

Campos utilizados

- subsidiary
- action
- statusFinal
- filter
- bill_manually
- doctype
- folio
- data.idPO
- data.dateBill
- data.fechaVenc
- data.currency
- data.items
- vendor
- rut
- issuer_rut
- total_amount
- dateOriginal

Operaciones en registros (map)

OPERACIÓN	RECORD	CAMPOS
create	customrecord_lmry_cl_inbound_detail	custrecord_lmry_cl_inbnd_deta_subsidary, custrecord_lmry_cl_inbnd_deta_log, custrecord_lmry_cl_inbnd_deta_folio, custrecord_lmry_cl_inbnd_deta_data, custrecord_lmry_cl_inbnd_deta_po, custrecord_lmry_cl_inbnd_deta_status_ini
submitFields	customrecord_lmry_cl_inbound_log	custrecord_lmry_cl_log_status
submitFields	customrecord_lmry_cl_inbound_detail	varios campos según proceso (bill, vc, process, error, request, response, manual, etc.)
transform	purchaseorder -> vendorbill	memo, custbody_lmry_created_by_inbound, custbody_lmry_created_from, custbody_lmry_num_preimpreso, approvalstatus, custbody_lmry_document_type, duedate, cseg1 (custom segment)
transform	vendor -> vendorbill	subsidiary, trandate, currency, memo, custbody_lmry_created_by_inbound, custbody_lmry_created_from, custbody_lmry_num_preimpreso, custbody_lmry_document_type, duedate
create	vendorcredit	varios campos para NC con o sin bill asociado, incluyendo apply line cuando corresponde
transform	vendorbill -> vendorcredit	trandate, custbody_lmry_created_by_inbound, custbody_lmry_created_from
load	inventoryitem	actualiza vendorcode en sublista itemvendor
https.post		

summarize

Actualiza el registro de log inbound a estado 2 (finalizado) y captura errores cambiando el estado a 4 con detalle del error.

Errores

- Errores capturados en try/catch del summarize, enviados a
customrecord_lmry_cl_inbound_log.custrecord_lmry_cl_log_error

Acciones finales

- Actualización de estado del log
- Registro de errores en el log

Búsquedas utilizadas

ID	FILTROS	COLUMNAS
setupSearch	[Markup('isinactive'), Markup('is'), Markup('F')], AND, [Markup('custrecord_lmry_cl_setup_subsidiary'), Markup('anyof'), Markup('subsidiary')]	custrecord_lmry_cl_setup_clientid, custrecord_lmry_cl_setup_password, custrecord_lmry_cl_setup_mode, custrecord_lmry_cl_setup_user_rut, custrecord_lmry_cl_setup_url, custrecord_lmry_cl_setup_approval_status, custrecord_lmry_cl_setup_item
documentTypeSearch	[Markup('isinactive'), Markup('is'), Markup('F')], AND, [Markup('custrecord_lmry_country_applied'), Markup('is'), Markup('45')], AND, [Markup('custrecord_lmry_tipo_transaccion'), Markup('anyof'), Markup('17')]	name, custrecord_lmry_codigo_doc
searchBill	{'name': Markup('subsidiary'), 'operator': Markup('is'), 'values': Markup('values.subsidiary')}, {'name': Markup('entity'), 'operator': Markup('is'), 'values': Markup('values.vendor')}, {'name': Markup('custbody_lmry_num_preimpreso'), 'operator': Markup('is'), 'values': Markup('values.folio')}	

Operaciones con registros

OPERACIÓN	RECORD	CAMPOS
customrecord_lmry_cl_inbound_log		
customrecord_lmry_cl_inbound_detail		
vendorbill		
vendorcredit		
purchaseorder		
vendor		
inventoryitem		

Side effects

- Creación y actualización de registros customrecord_lmry_cl_inbound_detail
- Creación de facturas de proveedor (vendorbill)
- Creación de notas de crédito (vendorcredit)
- Actualización de campos en items de inventario (vendorcode)
- Envío de solicitud SOAP a servicio externo
- Envío de correos electrónicos de notificación de error mediante LibraryMail

SUITELET · SUITELET INBOUND CL - FORMULARIO Y PROCESAMIENTO

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios filtrar, revisar y aplicar acciones a transacciones inbound, facilitando su gestión y seguimiento.

Flujo de Ejecución

GET

POST

Procesa el envío del formulario, redirige o ejecuta el proceso Map/Reduce

Flujo: Lee el parámetro `custpage_status` de la solicitud POST. Si no tiene valor, redirige al mismo Suitelet con los parámetros generados por el handler. Si tiene valor, obtiene el ID de log (`custpage_log_id`), ejecuta el proceso Map/Reduce a través del handler y luego redirige a la Suitelet de logs.

Genera: Redirección a otro Suitelet o ejecución del proceso Map/Reduce

Parámetros de Entrada

Lógica POST

Delegación de la lógica de creación de formularios, sublistas y ejecución de procesos a `LibraryHandler`, con manejo de redirecciones y envío de correos en caso de error

Operaciones

TIPO	TARGET	PROPÓSITO
redirect	Suitelet (mismo script)	Redirigir al usuario con parámetros actualizados cuando no se ha definido status
executeMapReduce	Proceso Map/Reduce definido en el handler	Procesar registros de log según el ID proporcionado
redirect	Suitelet de logs	Mostrar resultados del proceso de log al usuario
sendEmail	Módulo de envío de correos	Notificar al equipo de soporte cuando ocurre un error

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

Sí

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

Sí

Notas: En caso de error muestra un formulario de mensaje y envía un correo electrónico al equipo de soporte

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construcción y renderizado de formularios y campos
N/runtime	Obtención de información del script y contexto de ejecución
N/redirect	Redirección a otros Suitelets
./LMRY_CL_Inbound_STLT_HNDL	Manejo de la lógica de formularios, sublistas y ejecución de Map/Reduce
/SuiteBundles/Bundle 37714/Latam_Library/LMRY_Log_LBRY_V2.0	Gestión de logs y generación de identificadores de registro
/SuiteBundles/Bundle 37714/Latam_Library/LMRY_libSendingEmailsLBRY_V2.0	Envío de correos electrónicos para notificaciones de error

Resumen Técnico

Construye dinámicamente un formulario y sublista mediante un handler, gestiona redirecciones y ejecuta un proceso Map/Reduce, enviando notificaciones por correo en caso de error.

Generado automáticamente 2026-01-23 17:52 UTC

INTERNALMODULE · TRADUCTOR DE ETIQUETAS PARA CL INBOUND

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: utils

Traductor de etiquetas para CL Inbound

Resumen Ejecutivo

Este módulo centraliza todas las etiquetas de texto usadas en la funcionalidad Inbound de Chile, permitiendo obtenerlas en español o inglés según el idioma solicitado.

API pública

getTranslations

Devuelve un objeto con todas las etiquetas traducidas al idioma solicitado; si el idioma no está soportado se devuelve el conjunto en inglés.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código del idioma (por ejemplo, "es" o "en")	

Retorna

Objeto cuyas claves son identificadores de etiqueta y cuyos valores son los textos traducidos

Modelos de datos / estructuras

Translations

Mapa de claves de etiqueta a su texto traducido

CAMPO	TIPO	DESCRIPCIÓN
clave	string que identifica la etiqueta (por ejemplo, msg_validatorangedate)	
valor	texto traducido correspondiente al idioma	

Detalles de implementación

La función getTranslations selecciona el conjunto de etiquetas según el código de idioma recibido y lo devuelve; si el idioma no está definido en el mapa interno, se utiliza el conjunto en inglés como valor por defecto.

Generado automáticamente 2026-01-23 17:52 UTC

CLIENTSCRIPT · DETALLE AR INBOUND - VALIDACIONES Y CÁLCULOS

SuiteScript 2.1 · Módulos: 6

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

El script valida datos críticos de facturación y órdenes de compra antes de aprobar una factura, evitando inconsistencias de montos y fechas.

Lifecycle Hooks

pageInit

Inicializa variables, determina el idioma del usuario y muestra datos del XML cuando el estado es 3 o 4.

Lógica: Lee valores de campos ocultos, carga el XML, establece el idioma y, si el estado es 3 o 4, muestra información del XML en la consola.

Salida: Ninguna modificación visible del formulario

validateField

Habilita o deshabilita el campo de facturación manual según el valor de la casilla asignación.

Lógica: Si el campo es assign_flag, cambia la propiedad isDisabled de custpage_bill_manually y lo reinicia a 0.

Salida: Modifica la habilitación del campo custpage_bill_manually

fieldChanged

Gestiona cambios en campos y sublistas, actualiza valores de PO y recalcula impuestos, montos y totales.

Lógica: Cuando se marca po_checked deshabilita otras casillas; al cambiar purchase_order completa datos de PO; al modificar amount, tax_rate, tax_amount, rate o quantity recalcula los valores correspondientes.

Salida: Actualiza valores de sublista y campos calculados

sublistChanged

Recalcula el total de líneas del sublist de items cada vez que cambia la sublista.

Lógica: Si la sublista es custpage_items_list, llama a recalculateTotalLineAmount para comparar con el total del XML y muestra el resultado en custpage_lines_amount.

Salida: Actualiza custpage_lines_amount con estilo verde o rojo

validateInsert

Placeholder que permite la inserción de nuevas líneas sin restricciones.

Lógica: Lee datos del XML y del sublist pero no impone reglas.

validateDelete

Placeholder que permite la eliminación de líneas sin restricciones.

Lógica: No ejecuta lógica adicional.

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
assign_flag	-	validación	Al marcarla habilita el campo custpage_bill_manually; al desmarcarla lo deshabilita.
custpage_bill_manually	-	lectura/ escritura	Se habilita o deshabilita según assign_flag y se reinicia a 0 en ambos casos.
po_checked	custpage_pos_list	validación	Al seleccionar una casilla, deshabilita las casillas de las demás líneas; al deseleccionar, vuelve a habilitarlas.
purchase_order	custpage_pos_list	cálculo	Al cambiarlo completa los campos date, total_amount y document_type con datos del PO seleccionado.
amount	custpage_items_list	cálculo	Al modificarse recalcula tax_amount y total_amount.
tax_rate	custpage_items_list	cálculo	Al modificarse recalcula tax_amount y total_amount.
tax_amount	custpage_items_list	cálculo	Al modificarse recalcula total_amount.
rate	custpage_items_list	cálculo	Al modificarse recalcula amount (neto) y total_amount.
quantity	custpage_items_list	cálculo	Al modificarse recalcula amount (neto) y total_amount.
total_amount	custpage_items_list	cálculo	Se actualiza automáticamente tras cambios en amount, tax_amount, rate o quantity.
custpage_lines_amount	-	visual	Muestra el total de líneas con color verde si coincide con el total del XML, rojo en caso contrario.

Validaciones

UBICACIÓN	CAMPO	CRITERIO	ACCIÓN	BLOQUEA
updateParent (aprobación)	date	Debe existir una fecha cuando se aprueba y no se factura manualmente	Muestra alerta y bloquea la operación	Sí
updateParent (aprobación)	total de líneas	La suma de los totales de línea debe coincidir con el total de la factura	Muestra alerta y bloquea la operación	Sí
updateParent (aprobación)	currency	Debe estar definido cuando se aprueba y no se factura manualmente	Muestra alerta y bloquea la operación	Sí
updateParent (aprobación con PO)	purchase_order amount	El monto del PO debe coincidir con el total sin percepción	Muestra alerta y bloquea la operación	Sí
updateParent (aprobación sin PO)	total de líneas vs total factura	Diferencia mayor a 0.01 o total de líneas cero	Muestra alerta y bloquea la operación	Sí

Interacción con Usuario

alert: Debe completar el campo Periodo Contable

Trigger: No implementado en este script

alert: Debe completar el monto total indicado en el campo Monto Total

Trigger: Validación de total de líneas en updateParent

alert: Debe completar el campo Currency

Trigger: Validación de moneda en updateParent

alert: Debe completar la fecha

Trigger: Validación de fecha en updateParent

alert: Ocurrió un error

Trigger: Excepciones capturadas en cualquier método

alert: El monto de purchase order no coincide con el monto de la factura emitida, modifique su purchase order

Trigger: Validación de monto PO en updateParent

alert: El número de líneas del XML es diferente al número de líneas del purchase order.

Trigger: No implementado en este script

alert: El total de líneas ingresadas es diferente al total de la Factura.

Trigger: Validación de total de líneas en updateParent

Comportamiento del Script

VALIDA DATOS

Sí

BLOQUEA GUARDADO

Sí

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

Sí

Notas: Deshabilita campos dinámicamente, recalcula impuestos y totales, y muestra alertas cuando las validaciones fallan.

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener la preferencia de idioma del usuario
N/currentRecord	Acceder al registro actual del formulario
N/url	No utilizado directamente en la lógica mostrada
N/search	No utilizado directamente en la lógica mostrada
N/format	No utilizado directamente en la lógica mostrada
N/ui/dialog	No utilizado directamente; se emplea alert() en su lugar

Resumen Técnico

Implementa validaciones en eventos de campo y sublista, recalcula impuestos y totales en tiempo real, y bloquea la aprobación mediante alertas cuando las reglas de negocio no se cumplen.

CLIENTSCRIPT · CONTROL DE INBOUND AR: MANEJO DE EVENTOS Y TOGGLE DE CASILLAS

SuiteScript 2.1 · Módulos: 3

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Facilita la interacción del usuario en la pantalla de Inbound AR, permitiendo marcar todas las líneas y gestionando la lógica de guardado.

Lifecycle Hooks

pageInit

Instancia el handler personalizado y delega la lógica de inicialización del formulario

Lógica: Crea una instancia de Clnthandler, la almacena en una variable global y ejecuta su método pageInit pasando el contexto

Salida: Inicializa valores y configuraciones del formulario a través del handler

fieldChanged

Reenvía el evento de cambio de campo al handler para su procesamiento

Lógica: Llama al método fieldChanged del handler pasando el contexto

Salida: Puede modificar valores o mostrar mensajes según la lógica del handler

saveRecord

Solicita al handler la validación final antes de guardar el registro

Lógica: Ejecuta handler.saveRecord y devuelve su resultado (true permite guardar, false lo bloquea)

Salida: Bloquea o permite el guardado según la respuesta del handler

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
apply	sublist dinámico especificado en la llamada	escritura	Marca o desmarca la casilla apply en todas las líneas del sublist indicado

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

Sí

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: La mayor parte de la lógica (validaciones, cálculos y navegación) se delega al handler externo

Dependencias

NOMBRE	PROPÓSITO
../Handlers/ LMRY_AR_Inbound_CLNT_HNDL	Proporciona la clase ClntHandler que contiene la lógica de negocio del formulario
N/url	Importado pero no utilizado directamente en este script
N/currentRecord	Obtiene la instancia del registro actual para manipular valores del sublist

Resumen Técnico

Inicializa un handler personalizado y delega los eventos `pageInit`, `fieldChanged` y `saveRecord`; incluye una función utilitaria `toggleCheckBoxes` que actualiza el campo `apply` en cada línea del sublist.

INTERNALMODULE · MANEJADOR DE POPUP INBOUND AR

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador de Popup Inbound AR

Dependencias: N/URL N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET
N/SUITEAPPINFO N/FORMAT N/XML N/QUERY N/CONFIG
../TRANSLATOR/LMRY_AR_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo muestra el detalle de facturas inbound argentinas y permite a los usuarios aprobar, rechazar o crear facturas a partir de la información del XML y de PO pendientes.'), 'technical_summary': Markup('Expone la clase LibraryHandler que, mediante búsquedas, parsing de XML y generación dinámica de sublistas, construye un formulario Suitelet con botones de acción y enlaces a PO, XML y PDF.')}

Dependencias

NOMBRE	PROPÓSITO
N/url	Construir URLs a registros y scripts
N/search	Ejecutar búsquedas guardadas y dinámicas
N/runtime	Obtener contexto de ejecución y parámetros de script
N/redirect	Redirigir a otro Suitelet
N/ui/serverWidget	Crear formularios y sublistas en Suitelet
N/suiteAppInfo	Detectar si el bundle de localización está instalado
N/format	Parsear y formatear fechas
N/xml	Parsear contenido XML del registro inbound
N/query	Obtener monedas del vendor mediante Query API
N/config	Leer preferencias de contabilidad
../Translator/LMRY_AR_Inbound_Translator	Obtener traducciones según idioma del usuario

API pública

LibraryHandler

Clase que genera y controla el formulario de detalle inbound AR

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto con propiedades params (parámetros del Suitelet) y method (GET o POST)	

Efectos colaterales

- Accede a configuraciones de cuenta
- Realiza búsquedas y lecturas de registros

Retorna

Instancia de LibraryHandler

createForm

Construye el formulario principal con campos de información básica y botones de acción

Efectos colaterales

- Registra depuración en log
- Establece valores por defecto en campos ocultos

Retorna

Objeto serverWidget.Form

createPOSublist

Agrega la sublista de órdenes de compra y muestra enlaces a PO y XML

Efectos colaterales

- Resuelve URLs a registros de PO

createSublist

Crea sublistas de PO y de items según el estado final del proceso

fillItemsSublist

Pobla la sublista de items calculando precios, impuestos y totales

setFormValues

Actualiza los valores del formulario con datos de subsidiaria y rango de fechas

toLogSuitelet

Redirige al Suitelet de registro de pagos

Efectos colaterales

- Redirección a otro script

getTranslations

Obtiene el objeto de traducciones para el idioma especificado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código ISO de dos letras	

Retorna

Objeto con pares clave-texto

getDataXML

Parsea el XML almacenado en el registro inbound y devuelve un objeto con información de la factura y sus items

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
xmlData	Id del registro customrecord_psg_ei_inbound_edoc	

Efectos colaterales

- Lectura de registro mediante lookupFields
- Parseo de XML

Retorna

Objeto con campos purchaseorder, items, perceptionAmount, etc.

searchPO

Busca una orden de compra por número y monto para determinar el estado del proceso

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
folioPO	Número de PO recibido en el XML	

Efectos colaterales

- Ejecuta búsqueda de purchaseorder

addItemValues

Llena un campo SELECT con los items disponibles para la subsidiaria

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
field	Campo serverWidget.FieldType.SELECT	
subsidiary	Id de la subsidiaria	

convertDate

Convierte una cadena en formato YYYY-MM-DD a objeto Date

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
date	Cadena con fecha	

Retorna

Objeto Date o cadena vacía

Funciones internas

NOMBRE	ROL	USADO POR
getSetup	Obtiene configuración de taxcode, department, class, location e item para la subsidiaria	constructor
getTaxRates	Calcula la distribución de montos entre dos tasas de IVA que rodean al rate total	getDataXML
addOptions	Agrega opciones a un campo SELECT a partir de una lista de resultados de búsqueda	createPOSublist , createSublist , createItemSublist , createAssignManually , getCurrencies , getJuridictions , getperceptionTypes , getPeriods
getTaxCodes	Recupera la lista de códigos de impuesto de compra activos en Argentina	createSublist , createItemSublist
getDepartments	Obtiene departamentos activos (filtrados por subsidiaria si corresponde)	createSublist , createItemSublist
getClasses	Obtiene clasificaciones activas	createSublist , createItemSublist
getLocations	Obtiene ubicaciones activas	createSublist , createItemSublist
getCurrencies	Recupera la moneda primaria y secundarias del vendor	createSublist
fillPoSublist	Pobla la sublista de PO con datos obtenidos de getPurchaseOrder	createPOSublist
fillManualBill	Llena el campo de selección de facturas manuales pendientes	createAssignManually
getItemDetailsByVendorCodes	Busca items por código de vendor y devuelve un mapa id-item	fillItemsSublist

Modelos de datos / estructuras

DataXML

Representa la información extraída del XML de la factura

CAMPO	TIPO	DESCRIPCIÓN
purchaseorder	string	
itemreceipt	string	
rateIVA	number	
items	array de objetos con name, quantity, price, discount, rateIvaItem, vlrCodigo, isItemPercep	
perceptionAmount	number	

PurchaseOrderResult

Objeto con datos de una orden de compra encontrada

CAMPO	TIPO	DESCRIPCIÓN
entity	string	
status	string	
transactionname	string	
approvalstatus	string	
trandate	date	
amount	number	
otherrefnum	string	
tranid	string	

Generado automáticamente 2026-01-23 17:47 UTC

INTERNALMODULE · MANEJADOR CLIENTE PARA INBOUND AR

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador cliente para Inbound AR

Dependencias: N/RUNTIME N/RECORD N/SEARCH N/URL N/HTTPS N/FORMAT
N/CURRENTRECORD N/QUERY N/CURRENCY N/UI/MESSAGE
../TRANSLATOR/LMRY_AR_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza la lógica de validación y procesamiento de facturas inbound AR, facilitando la selección y registro de documentos para su posterior conversión.'), 'technical_summary': Markup('Expone una clase ClntHandler con métodos de manejo de eventos de cliente, validaciones de campos, carga dinámica de listas y creación de registros de log mediante la API de SuiteScript 2.1.')}}

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener información del entorno y preferencias del usuario
N/record	Crear y guardar registros personalizados
N/search	Ejecutar búsquedas guardadas y consultas dinámicas
N/url	Construir URLs a Suitelets y scripts
N/https	Realizar llamadas HTTP externas
N/format	Parsear y formatear fechas
N/currentRecord	Acceder al registro activo en la página cliente
N/query	Obtener datos de entidades mediante la API de consultas
N/currency	Manejo de conversiones y formatos de moneda
N/ui/message	Mostrar mensajes informativos al usuario
../Translator/LMRY_AR_Inbound_Translator	Proveer traducciones de textos según el idioma del usuario

API pública

ClntHandler

Clase que controla la lógica de la página cliente para la importación de facturas AR

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto de configuración opcional	

Retorna

Instancia de ClntHandler

ClntHandler.pageInit

Inicializa la página, deshabilita campos según el estado y carga la configuración de la subsidiaria

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
context	Objeto de contexto que contiene el registro actual	

Efectos colaterales

- Modifica propiedades de campos en el sublist
- Establece valores en el registro actual

Retorna

true si la inicialización se completa sin errores

ClntHandler.fieldChanged

Maneja cambios en campos del sublist, abre una ventana popup con detalle del documento y controla la selección única

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
context	Objeto que indica el sublist, campo y línea modificados	

Efectos colaterales

- Abre ventana del navegador
- Actualiza valores de sublist

Retorna

true

ClntHandler.saveRecord

Valida los datos antes de guardar; si el estado indica procesamiento, crea un registro de log y verifica ejecuciones concurrentes

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
context	Objeto que contiene el registro actual	

Efectos colaterales

- Crea registro de log inbound
- Muestra mensajes de alerta

Retorna

true si la validación pasa, false en caso de error

ClntHandler.validateExecution

Comprueba si existe una ejecución de Map/Reduce en curso y muestra un mensaje informativo

Efectos colaterales

- Actualiza campo custpage_is_processing
- Muestra mensaje UI

Retorna

false si hay una ejecución activa, true en caso contrario

ClntHandler.createLogInbound

Genera un registro personalizado con la información de los documentos seleccionados para procesamiento

Efectos colaterales

- Crea y guarda registro customrecord_lmry_ar_inbound_po
- Actualiza campo custpage_log_id en el registro cliente

Retorna

false (valor de retorno no utilizado)

ClntHandler.fillPeriod

Carga los periodos contables activos en el campo de selección del formulario

Efectos colaterales

- Modifica opciones del campo custpage_period

ClntHandler.fillDepartments

Pobla el sublist de departamentos según la subsidiaria y la configuración del sistema

Efectos colaterales

- Modifica opciones del campo custpage_department

ClntHandler.fillClasses

Pobla el sublist de clases según la subsidiaria

Efectos colaterales

- Modifica opciones del campo custpage_class

ClntHandler.fillLocations

Pobla el sublist de ubicaciones según la subsidiaria

Efectos colaterales

- Modifica opciones del campo custpage_location

ClntHandler.fillCurrenciesByEntity

Obtiene y muestra las monedas asociadas al proveedor seleccionado

Efectos colaterales

- Modifica opciones del campo custpage_currency

ClntHandler.getSetup

Recupera la configuración de impuestos y cuentas para la subsidiaria indicada

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
subsidiary	Id interno de la subsidiaria	

Retorna

Objeto con la configuración encontrada

ClntHandler.getText

Devuelve la cadena traducida correspondiente a la clave indicada

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
key	Clave del texto a traducir	

Retorna

Texto traducido

back

Navega a la Suitelet principal con los parámetros de búsqueda actuales

Efectos colaterales

- Redirige la ventana del navegador a una URL de Suitelet

Funciones internas

NOMBRE	ROL	USADO POR
handleError	Registra errores en la consola y, opcionalmente, en un log de biblioteca	ClntHandler.pageInit , ClntHandler.fieldChanged , ClntHandler.saveRecord

INTERNALMODULE · MANEJADOR DE SUITELET PARA INBOUND AR

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador de Suitelet para Inbound AR

Dependencias: N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET N/RECORD
N/FORMAT N/XML N/URL N/TASK ../TRANSLATOR/LMRY_AR_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza la visualización y procesamiento de documentos inbound para Argentina, facilitando a los usuarios filtrar por fechas, subsidiaria y estado, y lanzar la conversión masiva.'), 'technical_summary': Markup('Implementa una clase LibraryHandler que construye formularios Suitelet, consulta registros mediante búsquedas paginadas, genera sublistas dinámicas y coordina la ejecución de un script Map/Reduce para la conversión de documentos.')}

Dependencias

NOMBRE	PROPÓSITO
N/search	Ejecutar búsquedas y obtener resultados de NetSuite
N/runtime	Acceder a información de ejecución, parámetros y características de la cuenta
N/redirect	Redirigir a otros scripts o suitelets
N/ui/serverWidget	Construir formularios y sublistas en la interfaz del suitelet
N/record	Crear y actualizar registros personalizados
N/format	Formatear y parsear fechas y valores
N/xml	Parsear contenido XML de documentos inbound
N/url	Construir URLs a registros y scripts
N/task	Crear y lanzar tareas de Map/Reduce
../Translator/ LMRY_AR_Inbound_Translator	Obtener traducciones según el idioma del usuario

API pública

LibraryHandler

Clase que genera y controla el suitelet de inbound AR, incluyendo filtros, sublistas y procesos de conversión

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto con propiedades params (parámetros del request) y method (GET o POST)	

Efectos colaterales

- Crea formularios y sublistas en la UI
- Realiza búsquedas y crea registros

Retorna

Instancia de LibraryHandler

Funciones internas

NOMBRE	ROL	USADO POR
areThereSubsidiaries	Determina si la cuenta tiene subsidiarias activas y las carga en la propiedad subsidiaries	constructor , createFilterForm
getSubsidiaries	Obtiene la lista de subsidiarias activas en Argentina mediante una búsqueda	areThereSubsidiaries
createFilterForm	Construye el formulario de filtros del suitelet y agrega campos dinámicos	setFormValues
createApplySublist	Crea la sublista de resultados y define sus columnas según el estado del proceso	loadSublist
loadSublist	Pobla la sublista con datos obtenidos del método getResponse y ajusta contadores	
fillSubsidiaries	Agrega opciones de subsidiarias al campo SELECT del formulario	createFilterForm
getRedirectParams	Construye un objeto con los parámetros necesarios para redirigir al suitelet en modo filtro	
setFormValues	Establece valores por defecto en el formulario a partir de los parámetros recibidos	
toLogSuitelet	Redirige al suitelet de registro de logs	
executeMapReduce	Inicia una tarea Map/Reduce pasando datos de conversión y la subsidiaria	
createLogInbound	Crea un registro personalizado de log de inbound con la información de la conversión	
getTranslations	Obtiene el objeto de traducciones para el idioma especificado	constructor
getText	Devuelve la cadena traducida para una clave dada	createFilterForm , createApplySublist , loadSublist
getResponse	Construye la lista de resultados a partir de los registros inbound y sus archivos XML/PDF	loadSublist
getInbounStandarData	Recupera el XML consolidado y metadatos de los registros inbound dentro del rango de fechas	getResponse
getInboundRecordsIds	Obtiene los IDs internos de los registros inbound que cumplen con los filtros de fecha y subsidiaria	getInbounStandarData
getBillsByinboundRecord	Mapea facturas vendorbill asociadas a documentos inbound convertidos	loadSublist

NOMBRE	ROL	USADO POR
getLatestAuditTrialByState	Obtiene el último registro de auditoría para documentos en un estado específico	loadSublist
getDateFromXml	Extrae la fecha del contenido XML del documento	
getUuidInExpense	Mapea UUIDs de documentos inbound que ya están asociados a informes de gastos	
convertDate	Convierte una cadena de fecha en formato YYYY-MM-DD a un objeto Date formateado	

Modelos de datos / estructuras

InboundResult

Objeto que representa un documento inbound procesado

CAMPO	TIPO	DESCRIPCIÓN
issuer_name	Nombre del emisor	
issuer_rut	RUT del emisor	
folio	Número de folio del documento	
total_amount	Monto total del documento	
document_type	Tipo de documento (Factura, etc.)	
date	Fecha del documento	
see_pdf	URL al PDF del documento	
see_xml	ID del registro XML del documento	
status	Código de estado interno	
statustype	Estado legible (Pending, Converted, etc.)	
uuid	Código UUID del documento	

InboundStandardData

Estructura que agrupa datos comunes de los registros inbound

CAMPO	TIPO	DESCRIPCIÓN
inboundRecordsXml	XML concatenado de todos los documentos	
vendorList	Lista de nombres de proveedores	
statusList	Lista de códigos de estado	
xmlList	Lista de IDs internos de los registros XML	
pdfList	Lista de IDs de archivos PDF asociados	

INTERNALMODULE · GESTIÓN DE CAMPOS EXTRA PARA FACTURAS DE PROVEEDOR INBOUND

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: userEvent

Gestión de campos extra para facturas de proveedor inbound

Dependencias: N/LOG N/RUNTIME N/SEARCH N/RECORD

Resumen Ejecutivo

{'business_summary': Markup('Este módulo procesa facturas de proveedor inbound, extrae la fecha del XML y mantiene actualizados los campos extra en un registro personalizado para su uso en otros procesos.'), 'technical_summary': Markup('En afterSubmit lee el contenido XML del campo custrecord_psg_ei_inbound_content, extrae la fecha mediante expresión regular y crea o actualiza el registro customrecord_lmry_inbound_fields con la fecha y el vendor asociado.')}

Dependencias

NOMBRE	PROPÓSITO
N/log	Registrar errores y eventos en el log de NetSuite
N/runtime	Obtener información del entorno de ejecución
N/search	Buscar registros personalizados de campos inbound
N/record	Crear, cargar y guardar registros personalizados

API pública

beforeLoad

Función de evento que se ejecuta antes de cargar el registro; actualmente no realiza ninguna acción

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene el registro y la información del contexto	

beforeSubmit

Función de evento que se ejecuta antes de enviar el registro; actualmente no realiza ninguna acción

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene el registro nuevo y el anterior	

afterSubmit

Función de evento que se ejecuta después de enviar el registro; extrae la fecha del XML y crea o actualiza los campos extra inbound

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene el registro nuevo y el anterior	

Efectos colaterales

- Creación o actualización del registro customrecord_lmry_inbound_fields
- Escritura de logs en caso de error

Funciones internas

NOMBRE	ROL	USADO POR
getXmlData	Extrae la fecha del contenido XML del campo custrecord_psg_ei_inbound_content y la convierte a objeto Date	afterSubmit
createOrUpdateInboundExtraFields	Busca el registro customrecord_lmry_inbound_fields asociado y lo crea o actualiza con la fecha extraída y el vendor relacionado	afterSubmit

Modelos de datos / estructuras

XmlData

Objeto que contiene la fecha extraída del XML

CAMPO	TIPO	DESCRIPCIÓN
xmlDate	Objeto Date con la fecha del documento o null si no se encontró	

Generado automáticamente 2026-01-23 17:46 UTC

MAP/REDUCE · IMPORTACIÓN DE DOCUMENTOS AFIP Y CREACIÓN DE REGISTROS INBOUND

Pipeline de procesamiento por lotes (SuiteScript Map/Reduce).

Procesos masivos, búsquedas y operaciones sobre registros.

Metadatos

Tipo: MapReduce

SuiteScript: 2.1

Importación de documentos AFIP y creación de registros inbound

Resumen Ejecutivo

```
{'business_summary': Markup('Importa facturas electrónicas recibidas de AFIP, evita duplicados y genera registros inbound para su posterior procesamiento.'), 'technical_summary': Markup('Consume un servicio externo vía http.post, filtra documentos existentes, busca o asigna vendors, crea registros customrecord_psg_ei_inbound_edoc con contenido XML y actualiza datos de configuración.')}
```

getInputData

Obtiene documentos desde el servicio externo mediante http.post, filtra los que ya existen en el registro inbound y agrega datos de configuración (vendor genérico, paquete genérico y tipo de documento).

map

Parsea el documento recibido, busca el vendor asociado al RFC del emisor, crea un registro `customrecord_psg_ei_inbound_edoc` con los datos del documento, genera XML a partir del JSON y asigna valores de estado y plantilla según corresponda.

Campos utilizados

- Cód. Autorización
- Nro. Doc. Emisor
- `genericVendor`
- `genericPackage`
- `edocType`
- Fecha
- Tipo
- Punto de Venta
- Número Desde
- Número Hasta
- Tipo Doc. Emisor
- Denominación Emisor
- Tipo Cambio
- Moneda
- Imp. Neto Gravado
- Imp. Neto No Gravado
- Imp. Op. Exentas
- Otros Tributos
- IVA
- Imp. Total

Operaciones en registros (map)

OPERACIÓN	RECORD	CAMPOS
create	customrecord_psg_ei_inbound_edoc	name, custrecord_psg_ei_inbound_vendor, custrecord_psg_ei_inbound_status, custrecord_psg_ei_inbound_source, custrecord_psg_ei_inbound_tran_type, custrecord_psg_ei_inbound_template, custrecord_psg_ei_inbound_content
submitFields	vendor	custentity_psg_ei_entity_edoc_standard
lookupFields	vendor	custentity_psg_ei_entity_edoc_standard

summarize

Registra en log audit la cantidad total de documentos procesados y el número de errores en la fase map.

Acciones finales

- log.audit con resumen de importación

Búsquedas utilizadas

ID	FILTROS	COLUMNAS
buscarVendorPorRFC	formulatext: CONCAT({vatregnumber}, {custentity_lmry_digito_verificador}) is RFC	internalid
getTemplate	custrecord_psg_ei_template_edoc_standard is edocPackage, custrecord_psg_ei_template_trans_type is tranType, custrecord_edoc_template_xsd noneof @NONE@	custrecord_edoc_template_xsd, internalid
findFolderByName	name is folderName	internalid
existingDocuments	name contains each folio (OR concatenado)	name
getSetupData	custrecord_lmry_ar_setup_afip_cuit is subsiCUIT	custrecord_lmry_ar_setup_vendor, custrecord_lmry_ar_setup_package

Operaciones con registros

OPERACIÓN	RECORD	CAMPOS
customrecord_psg_ei_inbound_edoc		
vendor		
folder		
file		

Side effects

- Creación de registros inbound para cada documento nuevo
- Actualización del campo custentity_psg_ei_entity_edoc_standard en vendor cuando no está definido
- Posible creación de carpeta y archivo PDF (aunque no se usa en la lógica actual)
- Logs de depuración y auditoría

RESTLET • RESTLET QUE LANZA UN PROCESO MAP/REDUCE PARA AFIP CON EL ID DE PROCESO RECIBIDO.

SuiteScript 2.1

Documentación técnica del servicio RESTlet.

Metadatos

Tipo: RESTlet

SuiteScript: 2.1

RESTlet que lanza un proceso Map/Reduce para AFIP con el ID de proceso recibido.

Resumen Ejecutivo

El RESTlet expone únicamente el método POST; recibe un JSON con el campo processID, lanza una tarea Map/Reduce configurada con el script y deployment especificados, pasa el ID del proceso como parámetro y devuelve un objeto con el estado, mensaje, ID de la tarea y el registro del trabajo enviado.

Endpoints

GET

Definición inválida o no estructurada.

POST

Recibe un cuerpo JSON con el campo processID, crea y lanza una tarea Map/Reduce con los parámetros indicados y devuelve información del lanzamiento.

Body

NOMBRE	DESCRIPCIÓN	TIPO
processID		

Lógica

Construye un objeto con processID, lo serializa a JSON, crea una tarea Map/Reduce asignando scriptId y deploymentId, establece el parámetro custscript_lmry_afip_processid con el JSON, envía la tarea y registra el ID en el log.

Salida

Objeto JSON con status, message, taskId y jobRecord (cadena JSON del proceso).

PUT

Definición inválida o no estructurada.

DELETE

Definición inválida o no estructurada.

Errores

UBICACIÓN	DESCRIPCIÓN	EXCEPCIÓN
post	Error capturado en bloque catch y rethrow	error rethrow

Side effects globales

- **Type:** log; **Level:** debug; **Message:** MapReduce lanzado
- **Type:** log; **Level:** error; **Message:** Error en RESTlet SATComplete
- **Type:** task; **Action:** submit; **Tasktype:** MAP_REDUCE; **Scriptid:** customscript_lmry_ar_inb_afip_mprd; **Deploymentid:** customdeploy_lmry_ar_inb_afip_mprd

SCHEDULED SCRIPT · DESCARGA DE DATOS DEL SAT MEDIANTE SERVICIO EXTERNO

SuiteScript 2.1

Procesos periódicos y lotes programados.

Metadatos

Tipo: ScheduledScript

SuiteScript: 2.1

Descarga de datos del SAT mediante servicio externo

Resumen Ejecutivo

{'business_summary': Markup('Inicia la descarga de información del SAT para la subsidiaria especificada'), 'technical_summary': Markup('Ejecuta una llamada HTTP POST al endpoint configurado usando credenciales obtenidas de un registro de configuración y registra la respuesta')}

Entrada

Obtiene el CUIT de la subsidiaria a través del parámetro de script custscript_subsiary_cuit y busca las credenciales SAT en el registro customrecord_lmry_ar_setup_inbound filtrado por ese CUIT

Parámetros

PARÁMETRO	DESCRIPCIÓN
custscript_subsiary_cuit	

Filtros

CAMPO	OPERADOR	VALOR
custrecord_lmry_ar_setup_afip_cuit	is	subsiParam

Columnas

NOMBRE	ETIQUETA
custrecord_lmry_ar_setup_afip_user	
custrecord_lmry_ar_setup_afip_pass	

Lógica

1. Obtener el parámetro de script con el CUIT de la subsidiaria. 2. Calcular la fecha actual (año, mes, día). 3. Recuperar usuario y contraseña SAT mediante una búsqueda en customrecord_lmry_ar_setup_inbound y desencriptar la contraseña. 4. Construir el cuerpo JSON con fecha, credenciales y CUIT. 5. Realizar una petición HTTP POST al endpoint configurado. 6. Registrar la respuesta: debug si código 200, error en caso contrario. 7. Capturar y registrar cualquier excepción del proceso.

Condiciones

- **Condition:** response.code === 200; **True Action:** log.debug con mensaje de inicio de descarga; **False Action:** log.error con código HTTP y cuerpo de respuesta

Búsquedas

TIPO	FILTROS	COLUMNAS
search	{'field': Markup('custrecord_lmry_ar_setup_afip_cuit'), 'operator': Markup('is'), 'value': Markup('subsiParam')}	custrecord_lmry_ar_setup_afip_user, custrecord_lmry_ar_setup_afip_pass
search	{'field': Markup('custentity_mx_rfc'), 'operator': Markup('is'), 'value': Markup('rfc')}	internalid

Governance

Errores

ESCENARIO	MANEJO

Side effects

- **Type:** http request; **Description:** Envía una petición POST al endpoint http://149.130.171.192:3006/receptor/download con datos de fecha y credenciales
- **Type:** logging; **Description:** Genera entradas de log de tipo debug y error según la respuesta del servicio

SUITELET · MOSTRAR PDF O TEXTO DE ARCHIVO INBOUND

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a usuarios autorizados visualizar o descargar archivos inbound directamente desde NetSuite.

Flujo de Ejecución

GET

Parámetros de Entrada

Lógica POST

Carga el archivo indicado, determina su tipo y codificación y entrega el contenido o redirige al recurso

Operaciones

TIPO	TARGET	PROPÓSITO
load	archivo del File Cabinet	obtener contenido y metadatos del archivo
redirect	URL del archivo en el File Cabinet	redirigir cuando el archivo es de tipo texto con codificación ISO-8859-1

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: No se crean ni modifican registros, solo se entrega contenido de archivo

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	estructura base del Suitelet (aunque no se construye formulario en este caso)
N/search	no utilizado directamente, incluido por convención
N/runtime	obtener información del usuario y validar la característica de subsidiarias
N/record	no utilizado directamente, incluido por convención
N/redirect	realizar redirección a la URL del archivo cuando corresponde
N/file	cargar el archivo del File Cabinet

Resumen Técnico

Carga el archivo mediante N/file, verifica tipo y codificación y responde con PDF, texto UTF-8 o redirección según corresponda.

SUITELET · VISUALIZADOR DE XML INBOUND AR

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar de forma legible el XML asociado a un registro inbound de AR.

Flujo de Ejecución

GET

POST

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_id_message	Message	INLINEHTML	Muestra mensaje HTML estático dentro del formulario de error

Lógica POST

En modo GET, recupera y formatea el XML del registro inbound; en caso de excepción, muestra un formulario con mensaje de acceso no permitido

Búsquedas

Obtener el contenido XML del registro inbound

Filtros: Tipo customrecord_psg_ei_inbound_edoc, id inboundRecordId, columna custrecord_psg_ei_inbound_content

Operaciones

TIPO	TARGET	PROPÓSITO
lookupFields	customrecord_psg_ei_inbound_edoc	Extraer el campo XML del registro
parse	XML document	Convertir la cadena XML en un objeto de documento
format	XML document	Aplicar indentación y saltos de línea para legibilidad
writeResponse	context.response	Entregar el XML formateado como texto plano al cliente
createForm	error form	Mostrar mensaje de error cuando ocurre una excepción

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Entrega XML formateado o formulario de error; no modifica registros ni envía notificaciones

Dependencias

NOMBRE	PROPÓSITO
N/search	Realizar lookupFields para obtener el contenido XML del registro
N/encode	Módulo importado pero no utilizado en la lógica actual
N/ui/serverWidget	Construir el formulario de error con campo INLINEHTML
N/url	Módulo importado pero no utilizado en la lógica actual
N/xml	Parsear y serializar el documento XML para formatearlo

Resumen Técnico

Recupera el XML mediante lookupFields, lo parsea y formatea usando N/xml, y lo entrega como texto plano; en caso de error muestra un formulario con mensaje informativo.

SUITELET · DETALLE INBOUND AR

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar el detalle de inbound AR con sus ítems y sublistas.

Flujo de Ejecución

GET

Lógica POST

Construye y muestra el formulario usando la librería handler; en caso de excepción muestra un formulario de error con mensaje estático.

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Muestra formulario y gestiona errores mediante un formulario de mensaje

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construir formularios y campos
N/runtime	Obtener información del script y despliegue actual
N/redirect	Módulo importado pero no utilizado en este Suitelet
../Handlers/LMRY_AR_InboundDetail_STLT_HNDL	Lógica de creación y llenado del formulario y sublistas

Resumen Técnico

Genera el formulario mediante una librería handler y lo muestra en respuesta GET; en caso de error muestra un formulario con mensaje estático.

Generado automáticamente 2026-01-23 17:44 UTC

SUITELET · DETALLE DE REGISTRO INBOUND EI

SuiteScript 2.x

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar de forma detallada el estado y la auditoría de los documentos inbound asociados a un registro de log, facilitando la identificación de errores y la trazabilidad de facturas.

Flujo de Ejecución

GET

POST

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
id_internalid	Contador	TEXT	solo lectura, muestra el número de fila
id_tran_id	ID Interno	TEXT	solo lectura, muestra el internalid del registro inbound
id_name	Nombre Archivo	TEXT	solo lectura
id_vendor	Proveedor	TEXT	solo lectura
id_subsidiary	Subsidiaria	SELECT	inline, muestra la subsidiaria del log cuando la característica está habilitada
id_doc_status	Estado de Documento EI	TEXT	solo lectura, muestra el estado del documento inbound
id_tran_ur	Transacción	TEXT	solo lectura, muestra enlace a la factura de proveedor o mensaje de error
audit_trial	Auditoria	TEXT	solo lectura, muestra el detalle del último audit trail con estado 'failed'

Lógica POST

Recupera los inbound IDs asociados al log, los busca en el registro customrecord_psg_ei_inbound_edoc y obtiene el último audit trail con estado fallido. Completa cada fila del sub-listado con la información obtenida y genera enlaces a facturas cuando existen.

Búsquedas

Obtener datos del log inbound

Filtros:

Retorna: subsidiaria, JSON con transacciones inbound y JSON con facturas

Buscar los registros inbound edoc por internalid

Filtros:

Retorna: internalid, name, referencia, estado, proveedor

Obtener el último audit trail con estado 'failed' para cada inbound edoc

Filtros:

Retorna: detalle del audit trail más reciente

Operaciones

TIPO	TARGET	PROPÓSITO
lookupFields	customrecord_lmry_ar_inbound_po	extraer la subsidiaria y los JSON de transacciones y facturas
search.create	customrecord_psg_ei_inbound_edoc	recuperar información de cada documento inbound
search.create	customrecord_psg_ei_audit_trail	obtener el detalle del último audit trail por inbound
url.resolveRecord	vendorbill	generar enlace a la factura de proveedor asociada
setSublistValue	sublista estática	poblar cada columna del sub-listado con los valores obtenidos

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Muestra una lista estática; no persiste cambios ni envía correos

Dependencias

NOMBRE	PROPÓSITO
N/runtime	obtener parámetros del script y verificar la característica de subsidiarias
N/log	registrar información de depuración
N/search	realizar búsquedas y lookupFields sobre registros personalizados
N/record	no se usa directamente para crear/actualizar, pero está disponible
N/ui/serverWidget	construir el formulario y el sub-listado estático
N/url	generar enlaces a facturas de proveedor

Resumen Técnico

Construye un formulario con sub-listado estático usando serverWidget, recupera datos mediante lookupFields y búsquedas paginadas, y genera enlaces a registros de facturas mediante url.resolveRecord.

CLIENTSCRIPT · PLANTILLA DE CLIENT SCRIPT PARA MASS INBOUND

SuiteScript 2.1

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Este script no implementa lógica; sirve como plantilla para futuros desarrollos de Client Script en Mass Inbound.

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

No

MUESTRA ALERTAS

No

Notas: No contiene lógica de validación, ocultación o alertas

Resumen Técnico

Actualmente solo expone una función de cancelación y una validación de guardado vacía.

SUITELET · LOG DE INBOUND AR

SuiteScript 2.0

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar el estado de los procesos inbound de Argentina y acceder al detalle de los que ya fueron completados.

Flujo de Ejecución

GET

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_hidden	Hidden	TEXT	oculto
custpage_id_message	Message	INLINEHTML	texto informativo
id_internalid	Internal ID	TEXT	solo lectura
id_subsidiary	Subsidiary	SELECT	valor inline, solo si la característica de subsidiarias está activa
id_datefrom	Date From	TEXT	solo lectura
id_dateto	Date To	TEXT	solo lectura
id_employee	Employee	TEXT	solo lectura
id_total	# Transactions	TEXT	solo lectura
id_transactions	Transactions	TEXTAREA	muestra detalle o enlace según estado
id_state	State	TEXTAREA	muestra estado del proceso

Lógica POST

Ejecuta una búsqueda de registros de log inbound, recorre los resultados y asigna valores a la sublista; según el estado (0 pendiente, 1 error, 2 completado) muestra texto o enlace al detalle.

Búsquedas

Obtener los logs de procesos inbound con estado definido

Filtros:

Retorna: internalid, usuario, subsidiaria, transacciones, estado, fecha de creación, conteo, fecha desde, fecha hasta

Operaciones

TIPO	TARGET	PROPÓSITO
populate sublist	sublista id_sublista	llenar cada fila con los valores obtenidos de la búsqueda
generate url	script detalle LMRY_AR_InboundLog_DT_STLT	crear enlace al detalle cuando el estado es completado

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Muestra información de logs y permite acceso al detalle mediante enlace

Dependencias

NOMBRE	PROPÓSITO
N/runtime	verificar si la característica de subsidiarias está activa y obtener parámetros del script
N/log	registrar información de depuración y errores
N/search	consultar los registros de log inbound
N/record	no se utiliza directamente en este Suitelet, pero está disponible para futuras extensiones
N/ui/serverWidget	construir el formulario y la sublista que se muestra al usuario
N/url	generar la URL del Suitelet de detalle para los registros completados

Resumen Técnico

Construye un formulario con una sublista estática mediante N/ui/serverWidget, consulta los logs con N/search y muestra enlaces a un Suitelet de detalle usando N/url.

Generado automáticamente 2026-01-23 17:46 UTC

MAP/REDUCE · LMRY AR INBOUND MAP/REDUCE

Pipeline de procesamiento por lotes (SuiteScript Map/Reduce).

Procesos masivos, búsquedas y operaciones sobre registros.

Metadatos

Tipo: MapReduce

SuiteScript: 2.0

LMRY AR Inbound Map/Reduce

Resumen Ejecutivo

{'business_summary': Markup('Convierte documentos electrónicos entrantes en facturas de proveedor y actualiza el registro de proceso de inbound.'), 'technical_summary': Markup('Lee IDs de documentos inbound, procesa cada uno en la fase map creando o transformando facturas, y guarda los resultados en la fase summarize.')}

getInputData

Array of inbound EDoc record IDs obtenidos del registro customrecord_lmry_ar_inbound_po indicado por el parámetro custscript_lmry_ar_inb_log_id.

Filtros

CAMPO	OPERADOR	VALOR
	anyof	paramIdDocuments

Columnas

NOMBRE	ETIQUETA
custrecord_lmry_ar_inb_transacts_po	

map

Para cada elemento se carga el registro `customrecord_psg_ei_inbound_edoc`, se obtienen datos del vendor y del PO (si existe), se llama a `convertInboundEDoc` que procesa el XML mediante una plantilla, ejecuta un plugin de validación y crea o transforma una factura de proveedor. Finalmente se escribe en la salida del MR la clave = id del registro inbound y el valor = objeto de resultado de la conversión.

Campos utilizados

- `custrecord_psg_ei_inbound_content`
- `custrecord_psg_ei_inbound_vendor`
- `custrecord_psg_ei_inbound_template`
- `tranid`
- `custrecord_psg_ei_inbound_parsed_content`
- `custrecord_psg_ei_inbound_refnum`
- `custrecord_psg_ei_inbound_po`

Operaciones en registros (map)

OPERACIÓN	RECORD	CAMPOS
load	<code>customrecord_psg_ei_inbound_edoc</code>	
submitFields	<code>customrecord_psg_ei_inbound_edoc</code>	<code>custrecord_psg_ei_inbound_status</code>
load	<code>vendor</code>	
load	<code>customrecord_psg_ei_template</code>	
pluginLoadImplementation		
create	<code>vendorbill</code>	
transform		
save	<code>vendorbill</code>	
submitFields	<code>customrecord_psg_ei_inbound_edoc</code>	<code>custrecord_psg_ei_inbound_status</code>
auditLog		
auditLog		

summarize

Itera sobre context.output, construye un objeto JSON con los resultados de cada registro inbound y lo guarda en el registro customrecord_lmry_ar_inbound_po.

Errores

- Registro de errores mediante log.error en caso de excepción.

Acciones finales

- **Type:** load; **Recordtype:** customrecord_lmry_ar_inbound_po; **Fields:** {}
- **Type:** setValue; **Recordtype:** customrecord_lmry_ar_inbound_po; **Fields:** {'custrecord_lmry_ar_inbound_bill_po': Markup('JSON string con resultados'), 'custrecord_lmry_ar_inbound_state_po': Markup('2')}
- **Type:** save; **Recordtype:** customrecord_lmry_ar_inbound_po; **Fields:** {}

Búsquedas utilizadas

ID	FILTROS	COLUMNAS
InboundIDSearch	{'name': Markup('internalid'), 'operator': Markup('anyof'), 'value': Markup('paramIdDocuments')}	custrecord_lmry_ar_inb_transacts_po
SetupSearch	{'name': Markup('custrecord_lmry_ar_setup_subsidiary'), 'operator': Markup('equalto'), 'value': Markup('subsidiaryParam')}, {'name': Markup('isinactive'), 'operator': Markup('is'), 'value': Markup('F')}	custrecord_lmry_ar_setup_item, custrecord_lmry_ar_setup_taxcode, custrecord_lmry_ar_setup_taxcode.taxt, custrecord_lmry_ar_setup_taxcode0, custrecord_lmry_ar_setup_taxcode0.tax, custrecord_lmry_ar_setup_taxcode10, custrecord_lmry_ar_setup_taxcode10.ta, custrecord_lmry_ar_setup_taxcode21, custrecord_lmry_ar_setup_taxcode21.ta, custrecord_lmry_ar_setup_taxcode27, custrecord_lmry_ar_setup_taxcode27.ta
CreatedFromPOsearch	{'name': Markup('tranid'), 'operator': Markup('is'), 'value': Markup('createdFromParam')}	internalid, entity
BillsByReferenceSearch	[Markup('entity'), Markup('is'), Markup('vendorId')], and, [Markup('tranid'), Markup('is'), Markup('refNum')], and, [Markup('mainline'), Markup('is'), Markup('T')]	tranid
ItemDetailsByVendorCodeSearch	{'name': Markup('vendorcode'), 'operator': Markup('is'), 'value': Markup('vendorCodeArray')}, AND, {'name': Markup('othervendor'), 'operator': Markup('is'), 'value': Markup('entityId')}	vendorcode
TaxTypeSearch	{'name': Markup('internalid'), 'operator': Markup('anyof'), 'value': Markup('taxCodeArray')}	internalid, taxtype
ItemRecordTypeSearch	{'name': Markup('internalid'), 'operator': Markup('is'), 'value': Markup('itemId')}	itemid

Operaciones con registros

OPERACIÓN	RECORD	CAMPOS
	customrecord_psg_ei_inbound_edoc	
	customrecord_lmry_ar_inbound_po	
	vendorbill	
	purchaseorder	
	vendor	
	customrecord_psg_ei_template	
	salestaxitem	
	item	

Side effects

- Creación o transformación de facturas de proveedor (vendorbill).
- Actualización del estado y contenido de los registros inbound EDoc.
- Actualización del registro de proceso LMRY AR Inbound PO con resultados JSON.
- Registro de auditoría mediante auditManager (logConverted, logConversionFailed).
- Ejecución de plugin personalizado de validación de datos.

Generado automáticamente 2026-01-23 17:47 UTC

SUITELET · AR INBOUND SUITELET - GESTIÓN DE APLICACIONES

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios filtrar y aplicar transacciones AR inbound, ejecutando procesos de integración y mostrando resultados en un sublist.

Flujo de Ejecución

GET

POST

Procesa la sumisión del formulario; redirige o ejecuta Map/Reduce

Flujo: Lee custpage_status; si es falsy redirige al mismo suitelet con parámetros de filtro; si es truthy obtiene logId y subsidiary, ejecuta el proceso Map/Reduce y redirige al suitelet de logs

Genera: Redirección a suitelet o ejecución de proceso y posterior redirección

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_id_message	Message	INLINEHTML	muestra mensaje de error, layout fuera de la tabla

Lógica POST

Coordina la lógica del suitelet delegando en el handler para crear formularios, cargar sublistas y ejecutar procesos Map/Reduce

Operaciones

TIPO	TARGET	PROPÓSITO
redirect	suitelet	redirige al mismo suitelet con parámetros cuando no hay status
executeMapReduce	handler	ejecuta proceso Map/Reduce con logId y subsidiary
redirect	log suitelet	redirecciona al suitelet de logs después de ejecutar el proceso

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

Sí

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Ejecuta proceso Map/Reduce que puede actualizar registros; redirige al suitelet de logs

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	crear formularios y campos
N/runtime	obtener IDs de script y despliegue, acceder a parámetros de ejecución
N/redirect	realizar redirecciones a suitelets
../Handlers/ LMRY_AR_Inbound_STLT_HNDL	lógica de negocio para crear formularios, sublistas y ejecutar Map/Reduce

Resumen Técnico

Construye dinámicamente un formulario con N/ui/serverWidget, gestiona la lógica mediante un handler personalizado y controla flujos GET/POST con redirecciones y ejecución de Map/Reduce.

INTERNALMODULE · CAPTURA Y PROCESAMIENTO DE E-DOCUMENTOS ENTRANTES POR EMAIL

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: service

Captura y procesamiento de e-documentos entrantes por email

Dependencias: NLAPI NLOBJSEARCHFILTER NLOBJSEARCHCOLUMN

Resumen Ejecutivo

{'business_summary': Markup('Este módulo captura e-documentos enviados por correo, los registra en NetSuite y notifica a los administradores cuando ocurre algún problema.'), 'technical_summary': Markup('Expone la función process que valida adjuntos XML/PDF, crea un registro personalizado, asocia proveedor o cliente mediante búsquedas, guarda PDF en el File Cabinet y gestiona errores enviando correos de notificación.')}

Dependencias

NOMBRE	PROPÓSITO
nlapi	API global de NetSuite para crear, buscar, actualizar registros, enviar correos y manipular archivos
nlobjSearchFilter	Construir filtros de búsqueda en NetSuite
nlobjSearchColumn	Definir columnas de resultados en búsquedas

API pública

process

Punto de entrada del plugin que procesa un correo entrante, extrae el XML y PDF adjuntos, crea el registro de e-documento, asocia proveedor o cliente y gestiona errores

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
email	Objeto de correo proporcionado por NetSuite	

Efectos colaterales

- crea registro customrecord_psg_ei_inbound_edoc
- crea archivo PDF en el File Cabinet
- envía notificación por email en caso de error

Funciones internas

NOMBRE	ROL	USADO POR
retrieveFile	helper	process
getActiveAdmins	helper	sendEmail
sendEmail	helper	process
identifyVendor	helper	process
identifyVendorLMRY	helper	process
identifyCustomer	helper	process
updateXMLContent	helper	process
createPDFFile	helper	process
getPDFFolder	helper	createPDFFile
getFolderIdByName	helper	getPDFFolder
createPDFFolder	helper	getPDFFolder
getParentSubsidiaryId	helper	process

Modelos de datos / estructuras

InboundEdocRecord

Registro personalizado que almacena el e-documento recibido por email

CAMPO	TIPO	DESCRIPCIÓN
custrecord_psg_ei_inbound_source	Origen del e-documento (valor constante 2 para email)	
custrecord_psg_ei_inbound_content	Contenido XML del e-documento	
custrecord_psg_ei_inbound_transtype	Tipo de transacción (Bill, BillCredit o SalesOrder)	
custrecord_psg_ei_inbound_vendor	Id interno del proveedor asociado	
custrecord_psg_ei_inbound_customer	Id interno del cliente asociado	
custrecord_psg_ei_inbound_template	Id de la plantilla de e-documento seleccionada	
custrecord_psg_ei_attach_pdf	Id del archivo PDF adjunto	

Efectos colaterales

- **Type:** record_creation; **Details:** Crea registros customrecord_psg_ei_inbound_edoc y folder (si no existe)
- **Type:** file_creation; **Details:** Crea archivo PDF en el File Cabinet
- **Type:** email_sending; **Details:** Envía notificaciones de error a administradores o destinatario configurado
- **Type:** logging; **Details:** Registra información de depuración y errores mediante nlapiLogExecution

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
retrieveFile	Lanza error si el número de archivos XML o PDF no cumple con las reglas de negocio	EMAIL_CAPTURE_ERROR_CODE
identifyVendor	Lanza error cuando no se encuentra ningún proveedor para el NIT proporcionado	EMAIL_CAPTURE_ERROR_CODE
identifyCustomer	Lanza error cuando no se puede identificar un cliente único a partir del remitente	EMAIL_CAPTURE_ERROR_CODE
process (catch block)	Captura cualquier excepción, registra el error y envía email de notificación	

Detalles de implementación

El script recibe un correo, extrae los adjuntos XML y PDF mediante `retrieveFile`, valida que exista exactamente un XML y como máximo un PDF, y limpia el XML. Crea un registro `InboundEdocRecord` con el contenido XML y determina el tipo de transacción a partir del asunto del correo. Según el tipo, busca al proveedor (`identifyVendor` / `identifyVendorLMRY`) o al cliente (`identifyCustomer`) y rellena los campos correspondientes. Busca la plantilla de e-documento asociada al paquete del proveedor/cliente y la guarda en el registro. Si hay PDF, lo guarda en el File Cabinet y enlaza el archivo al registro. Finalmente envía el registro. En caso de cualquier error, captura la excepción, genera un mensaje descriptivo y lo envía por email a los administradores configurados.

Generado automáticamente 2026-01-23 17:44 UTC

PLUG-IN IMPLEMENTATION · INJECT CUSTOM DATA SOURCE FOR AR INBOUND ELECTRONIC INVOICING

SuiteScript 2.0

Documentación técnica del plug-in.

Metadatos

Tipo: PlugInImplementation

SuiteScript: 2.0

Inject custom data source for AR inbound electronic invoicing

Resumen Ejecutivo

Provides a custom data source during the generation of Argentine inbound electronic invoices, extracting XML data, subsidiary setup, and currency information.

Deployment / Enablement

Se habilita al asignar este plug-in a la configuración de generación de documentos electrónicos de cuentas por cobrar (AR).

Entry points

NOMBRE	DESCRIPCIÓN	INPUTS	OUTPUTS
inject	Genera y devuelve un DataSource personalizado con información de la factura electrónica, configuración de la subsidiaria y monedas filtradas.	Objeto params que contiene inboundEDocRec (registro de documento entrante), templateRec y entityRec (registro de entidad).	Objeto con la propiedad customDataSources que incluye un DataSource de tipo OBJECT, alias "item_type" y los datos de inyección.

Búsquedas

PROPÓSITO	FILTROS	RETORNO
Obtener la configuración de la subsidiaria para el documento entrante.	custrecord_lmry_ar_setup_subsidiary = ID de la subsidiaria del proveedor.	Departamento, clase, ubicación, ítem y ítem de percepción configurados.
Recuperar monedas filtradas (ARS y USD) para incluirlas en la inyección.	symbol contiene 'ARS' O symbol contiene 'USD'.	Mapa de símbolo de moneda a nombre.

Dependencias

NOMBRE	PROPÓSITO
N/render	Definir el tipo de DataSource devuelto.
N/log	Registrar información y errores.
N/record	Acceder a campos de registros entrantes.
N/search	Ejecutar búsquedas de configuración y monedas.
N/https	Disponible pero no utilizado directamente en este plug-in.
N/runtime	Obtener parámetros de script y usuario actual.
N/xml	Parsear el contenido XML del documento entrante.
/SuiteBundles/Bundle 436209/com.netsuite.electronicinvoicing/src/app/audit/app_audit_manager	Registrar auditoría de conversiones fallidas.
N/error	Crear y lanzar errores estructurados.

Side effects

- Registro de auditoría de conversiones fallidas.
- Lanzamiento de errores que pueden detener la generación del documento.

Errores

UBICACIÓN	DESCRIPCIÓN	EXCEPCIÓN
	Cualquier excepción capturada se registra y se vuelve a lanzar como un error SuiteScript.	

PLUG-IN IMPLEMENTATION · INJECTS CUSTOM DATA SOURCE DURING INBOUND PROCESSING

SuiteScript 2.0

Documentación técnica del plug-in.

Metadatos

Tipo: PlugInImplementation

SuiteScript: 2.0

Injects custom data source during inbound processing

Resumen Ejecutivo

Provides an inject entry point that adds a custom data source to the inbound eDocument processing flow.

Entry points

NOMBRE	DESCRIPCIÓN	INPUTS	OUTPUTS
inject	Adds a custom data source to the inbound eDocument processing context.	Object `params` containing `inboundEDocRec`, `templateRec` and `entityRec` records.	Object with a `customDataSources` array that defines a data source (format OBJECT, alias "custom", data = empty object).

Dependencias

NOMBRE	PROPÓSITO
N/render	Provides DataSource constants used to specify the format of the custom data source.

INTERNALMODULE · ROUTER DE EVENTOS DE USUARIO PARA FACTURAS DE PROVEEDOR

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: factory

Router de eventos de usuario para facturas de proveedor

Dependencias: N/LOG

Resumen Ejecutivo

Este módulo permite a otros scripts obtener el manejador de eventos de usuario adecuado según el tipo de transacción, centralizando la lógica de inbound AR y manejando dinámicamente la carga de los handlers mediante require.

Dependencias

NOMBRE	PROPÓSITO
N/log	Registro de mensajes y errores en NetSuite

API pública

getUserEventRouter

Selecciona y carga dinámicamente el script de evento de usuario correspondiente al tipo de transacción solicitado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
type	Tipo de transacción (por ejemplo, vendorbill)	

Efectos colaterales

- Registra errores en el log mediante N/log.error cuando ocurre una excepción

Retorna

Objeto del handler cargado o cadena vacía si no existe un mapeo para el tipo indicado

Efectos colaterales

- **Type:** log; **Details:** Escribe en el log de NetSuite errores capturados en la función getUserEventRouter

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
getUserEventRouter	Captura cualquier excepción lanzada durante la resolución del módulo y la registra en el log	No lanza, solo registra el error

Detalles de implementación

El módulo mantiene un mapa interno de tipos de transacción a rutas de scripts handler. Cuando se invoca getUserEventRouter con un tipo, verifica el mapa; si existe, usa require de forma asíncrona para cargar el script y devuelve el módulo handler. Si el tipo no está mapeado o ocurre un error, devuelve una cadena vacía y registra el error.

CLIENTSCRIPT · CARGA DE CONFIGURACIÓN INBOUND SEGÚN SUBSIDIARIA

SuiteScript 2.1 · Módulos: 3

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

El script carga automáticamente la configuración de importación (tax codes, departamentos, ítems, etc.) según la subsidiaria elegida, evitando que el usuario tenga que ingresar esos datos manualmente.

Lifecycle Hooks

pageInit

Al iniciar el formulario, verifica si la característica de Subsidiarias está activa; si no lo está, asigna la subsidiaria 1 y busca la configuración inbound asociada para rellenar los campos del formulario.

Lógica: Comprueba la característica SUBSIDIARIES. Si está desactivada, fija la subsidiaria en 1 y ejecuta una búsqueda del registro `customrecord_lmry_ar_setup_inbound` filtrado por esa subsidiaria. Si encuentra un registro, extrae los valores de códigos de impuesto, departamento, clasificación, ubicación, ítems, usuario, contraseña (enmascarada), CUIT, proveedor y paquete, y los asigna a los campos del formulario. Si no hay resultados, limpia los campos relacionados.

Salida: Actualiza los campos `custpage_subsidiary`, `custpage_taxcode`, `custpage_taxcode0`, `custpage_taxcode8`, `custpage_taxcode16`, `custpage_taxcode27`, `custpage_department`, `custpage_class`, `custpage_location`, `custpage_item`, `custpage_item_percep`, `custpage_user`, `custpage_password`, `custpage_cuit`, `custpage_vendor` y `custpage_package`.

fieldChanged

Cuando el usuario cambia la subsidiaria, busca la configuración inbound correspondiente y rellena o limpia los campos del formulario.

Lógica: Si el campo cambiado es `custpage_subsidiary` y tiene un valor, ejecuta una búsqueda del registro `customrecord_lmry_ar_setup_inbound` filtrado por la subsidiaria seleccionada. Si encuentra un registro, asigna los valores a los mismos campos que en `pageInit`. Si no hay resultados o la subsidiaria se deselecciona, limpia los campos relacionados.

Salida: Actualiza los campos `custpage_taxcode`, `custpage_taxcode0`, `custpage_taxcode10`, `custpage_taxcode21`, `custpage_taxcode27`, `custpage_department`, `custpage_class`, `custpage_location`, `custpage_item`, `custpage_item_percep`, `custpage_user`, `custpage_password`, `custpage_cuit`, `custpage_vendor` y `custpage_package`.

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
custpage_subsiary	-	lectura/ escritura	Permite seleccionar la subsidiaria; al cambiar dispara la carga de la configuración inbound.
custpage_taxcode	-	escritura	Se rellena con el código de impuesto configurado para la subsidiaria.
custpage_taxcode0	-	escritura	Se rellena con el código de impuesto 0% configurado para la subsidiaria.
custpage_taxcode8	-	escritura	Se rellena con el código de impuesto 10% configurado para la subsidiaria.
custpage_taxcode16	-	escritura	Se rellena con el código de impuesto 21% configurado para la subsidiaria.
custpage_taxcode27	-	escritura	Se rellena con el código de impuesto 27% configurado para la subsidiaria.
custpage_department	-	escritura	Se rellena con el departamento configurado (valor extraído de JSON).
custpage_class	-	escritura	Se rellena con la clasificación configurada (valor extraído de JSON).
custpage_location	-	escritura	Se rellena con la ubicación configurada (valor extraído de JSON).
custpage_item	-	escritura	Se rellena con el ítem configurado para la subsidiaria.
custpage_item_percep	-	escritura	Se rellena con el ítem de percepción configurado.
custpage_user	-	escritura	Se rellena con el usuario AFIP configurado.
custpage_password	-	escritura	Se muestra la contraseña AFIP enmascarada con asteriscos.
custpage_cuit	-	escritura	Se rellena con el CUIT configurado.
custpage_vendor	-	escritura	Se rellena con el proveedor configurado.
custpage_package	-	escritura	Se rellena con el paquete configurado.

Interacción con Usuario

inline_message: La contraseña se muestra enmascarada con asteriscos.

Trigger: Al cargar la configuración inbound

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: El script solo rellena o limpia campos según la subsidiaria seleccionada; no impide el guardado ni muestra alertas.

Dependencias

NOMBRE	PROPÓSITO
N/search	Ejecutar búsquedas sobre el registro customrecord_lmry_ar_setup_inbound.
N/runtime	Verificar si la característica SUBSIDIARIES está habilitada.
N/currentRecord	Obtener y manipular el registro actual del formulario.

Resumen Técnico

En pageInit y fieldChanged se ejecuta una búsqueda del registro customrecord_lmry_ar_setup_inbound filtrado por subsidiaria; los valores obtenidos se asignan a los campos del formulario, y se limpian cuando no existen resultados.

SUITELET · LR - AR INBOUND SETUP

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los administradores configurar los parámetros de inbound para Argentina, incluyendo códigos de impuesto, ítems de compra, credenciales AFIP y opciones de subsidiaria, de forma centralizada por subsidiaria.

Flujo de Ejecución

GET

POST

Procesa los datos enviados y crea o actualiza el registro de configuración

Flujo: Al recibir una petición POST, el Suitelet lee los parámetros, cifra la contraseña, busca un registro de configuración existente por subsidiaria, lo actualiza o crea uno nuevo con los valores recibidos y finalmente redirige al mismo Suitelet

Genera: Redirección al Suitelet que recarga la página

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_subsiary	Subsidiaria	SELECT	obligatorio, opciones cargadas por búsqueda de subsidiarias AR
custpage_taxcode	Tax Code Not liable	SELECT	opcional, opciones cargadas por búsqueda de códigos de impuesto AR
custpage_taxcode0	Tax Code 0%	SELECT	opcional
custpage_taxcode10	Tax Code 10.5%	SELECT	opcional
custpage_taxcode21	Tax Code 21%	SELECT	opcional
custpage_taxcode27	Tax Code 27%	SELECT	opcional
custpage_department	Department	SELECT	opcional, visible solo si el feature DEPARTMENTS está activo
custpage_class	Class	SELECT	opcional, visible solo si el feature CLASSES está activo
custpage_location	Location	SELECT	opcional, visible solo si el feature LOCATIONS está activo
custpage_item	Generic Purchases Item	SELECT	opcional, origen de lista: item
custpage_item_percep	Perception Item	SELECT	opcional, origen de lista: item
custpage_user	User	TEXT	opcional
custpage_password	Password	TEXT	opcional, se cifra antes de guardarse
custpage_cuit	Cuit	TEXT	opcional
custpage_vendor	Vendor Generic	SELECT	opcional, origen de lista: vendor
custpage_package	Inbound Package	SELECT	opcional, origen de lista: customrecord_psg_ei_standards

Lógica POST

Al enviar el formulario, el Suitelet captura los valores, cifra la contraseña, busca un registro de configuración existente por subsidiaria y lo actualiza o crea uno nuevo con los datos recibidos, luego redirige a sí mismo

Búsquedas

Obtener lista de subsidiarias, departamentos, clases y ubicaciones activos (y subsidiarias de Argentina)

Filtros:

Retorna: internalid y nombre para poblar los campos SELECT

Obtener códigos de impuesto de ventas activos para Argentina

Filtros:

Retorna: internalid y nombre para los campos de tax code

Buscar registro de configuración inbound existente para la subsidiaria seleccionada

Filtros:

Retorna: internalid del registro

Operaciones

TIPO	TARGET	PROPÓSITO
load	customrecord_lmry_ar_setup_inbound	cargar registro existente y actualizar sus campos
create	customrecord_lmry_ar_setup_inbound	crear nuevo registro cuando no existe para la subsidiaria
setValue	registro de configuración	asignar valores de tax codes, departamento, clase, ubicación, ítems, credenciales AFIP y paquete
save	registro de configuración	persistir los cambios en NetSuite
sendRedirect	Suitelet (mismo script)	recargar la página después de guardar

Comportamiento del Suitelet

CREA REGISTROS

Sí

ACTUALIZA REGISTROS

Sí

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: Redirige al mismo Suitelet tras guardar para refrescar la vista

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construir y renderizar el formulario
N/search	Obtener listas de subsidiarias, departamentos, clases, ubicaciones y códigos de impuesto
N/runtime	Verificar la activación de features y obtener contexto de ejecución
N/record	Cargar, crear y guardar el registro customrecord_lmry_ar_setup_inbound
N/encode	Codificar la contraseña cifrada en Base64

Resumen Técnico

El Suitelet genera un formulario con serverWidget, llena los campos mediante búsquedas, procesa la sumisión creando o actualizando un registro customrecord_lmry_ar_setup_inbound y cifra la contraseña con XOR + Base64 antes de guardarla.

INTERNALMODULE · PROVEEDOR DE TRADUCCIONES PARA AR INBOUND

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: utils

Proveedor de traducciones para AR Inbound

Resumen Ejecutivo

Este módulo centraliza las etiquetas de texto para la funcionalidad AR Inbound, permitiendo que otros scripts obtengan traducciones en español o inglés de forma sencilla y sin efectos colaterales.

API pública

getTranslations

Devuelve el conjunto de etiquetas de texto según el idioma solicitado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código de idioma (es o en). Si no coincide, se usa inglés por defecto	

Retorna

Objeto con pares clave-valor que representan los textos traducidos

Detalles de implementación

El módulo contiene un diccionario estático con todas las etiquetas en español e inglés. La función pública `getTranslations` recibe un código de idioma, selecciona el bloque correspondiente y lo devuelve; si el idioma no está definido, devuelve el bloque inglés.

CLIENTSCRIPT · GESTIÓN DE PAGOS ENTRANTES EN CLIENTE

SuiteScript 2.1 · Módulos: 3

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Facilita la gestión de pagos entrantes en la interfaz del cliente, permitiendo marcar o desmarcar todas las líneas y asegurando que las validaciones se ejecuten antes de guardar.

Lifecycle Hooks

pageInit

Inicializa el manejador del cliente y delega la lógica de carga del formulario

Lógica: Crea una instancia del handler, la almacena y ejecuta su método pageInit pasando el contexto

Salida: Puede modificar campos o sublistas según la lógica del handler

fieldChanged

Reenvía el evento de cambio de campo al handler para su procesamiento

Lógica: Llama al método fieldChanged del handler pasando el contexto

Salida: El handler puede actualizar valores o mostrar mensajes

saveRecord

Ejecuta la validación final antes de guardar delegando al handler

Lógica: Invoca handler.saveRecord y devuelve su resultado (true permite guardar, false lo bloquea)

Salida: Booleano que determina si el guardado procede

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
apply	cualquier sublista especificada en toggleCheckBoxes	escritura	Establece el valor del checkbox apply en todas las líneas de la sublista según el parámetro isApplied

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

Sí

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: La mayor parte de la lógica, incluidas validaciones y mensajes, se delega al handler externo

Dependencias

NOMBRE	PROPÓSITO
../Handlers/ LMRY_CO_Inbound_CLNT_HNDL	Contiene la lógica del manejador del cliente para pagos entrantes
N/url	Provee utilidades para construir URLs (usado por el handler)
N/currentRecord	Permite obtener y manipular el registro actual en la función toggleCheckBoxes

Resumen Técnico

Implementa pageInit, fieldChanged y saveRecord delegando a un handler externo y ofrece toggleCheckBoxes para establecer el campo apply en todas las líneas de una sublista.

CLIENTSCRIPT · DETALLE INBOUND CO CLIENTE

SuiteScript 2.1 · Módulos: 6

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

El script controla la edición de facturas inbound, valida la consistencia con los purchase orders y evita errores de datos antes de guardar.

Lifecycle Hooks

pageInit

Inicializa el formulario, carga datos de configuración, desactiva campos de la sublista de ítems y establece el idioma del script.

Lógica: Obtiene valores de campos ocultos, desactiva los campos de PO en la sublista, determina el idioma del usuario, carga la configuración de la subsidiaria y, según el estado, desactiva los campos de la sublista de ítems.

Salida: Campos de la sublista marcados como disabled; variable global LANGUAGE establecida.

validateField

Gestiona la activación del campo 'custpage_bill_manually' según el valor del checkbox 'assign_flag'.

Lógica: Si el campo modificado es assign_flag, habilita o deshabilita custpage_bill_manually y lo reinicia a 0.

Validaciones

- **assign_flag:** cambio de valor

Acción si falla: habilita o deshabilita custpage_bill_manually

Salida: Estado disabled del campo custpage_bill_manually actualizado.

fieldChanged

Responde a cambios en varios campos: controla la selección única de PO, carga datos del PO seleccionado, recalcula montos y asigna el código de proveedor al ítem seleccionado.

Lógica: • Si el campo es po_checked, permite solo una selección y carga los ítems del PO. • Si el campo es purchase_order, completa datos del PO. • Si el campo es amount o tax_rate, recalcula impuestos y totales. • Si el campo es custpage_item_po, busca el vendor code del ítem y lo escribe en el registro.

Validaciones

- **po_checked:** checkbox marcado

Acción si falla: deshabilita los demás checkboxes y carga ítems del PO

- **purchase_order:** valor ingresado

Acción si falla: completa fecha, monto y tipo de documento del PO

- **[Markup('amount'), Markup('tax_rate')]:** cambio de valor

Acción si falla: recalcula tax_amount y total_amount

- **custpage_item_po:** selección de ítem

Acción si falla: asigna vendor_code al ítem

Salida: Actualización de campos de la sublista y de la cabecera según la lógica descrita.

validateLine

Impide la creación de líneas adicionales cuando el número de líneas supera la cantidad de ítems del XML.

Lógica: Si la sublista es `custpage_items_list` y el índice actual supera la longitud del arreglo `items` del XML, devuelve `false` para bloquear la línea.

Validaciones

- **línea de ítem:** excede número de ítems del XML

Acción si falla: bloquea la inserción de la línea

Salida: Retorno `false` cuando la condición se cumple.

validateInsert

Prohíbe la inserción manual de nuevas líneas en la sublista de ítems.

Lógica: Si la sublista es `custpage_items_list`, siempre devuelve `false`.

Validaciones

- **línea de ítem:** cualquier intento de inserción

Acción si falla: bloquea la inserción

Salida: Retorno `false`.

validateDelete

Permite la eliminación de líneas en la sublista de ítems sin restricciones adicionales.

Lógica: Simplemente devuelve `true`; mantiene la bandera interna `isNotRunning` para evitar recursividad.

Salida: Retorno `true`.

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
assign_flag	-	validación	Al marcarlo habilita el campo custpage_bill_manually; al desmarcarlo lo deshabilita.
custpage_bill_manually	-	lectura escritura	Se habilita o deshabilita según assign_flag y se reinicia a 0 en ambos casos.
po_checked	custpage_pos_list	validación	Permite seleccionar solo un PO; deshabilita los demás checkboxes.
purchase_order	custpage_pos_list	validación escritura	Al cambiarlo completa fecha, monto y tipo de documento del PO seleccionado.
amount	custpage_items_list	cálculo	Al modificarse recalcula tax_amount y total_amount.
tax_rate	custpage_items_list	cálculo	Al modificarse recalcula tax_amount y total_amount.
custpage_item_po	custpage_items_list	validación escritura	Al seleccionar un ítem busca y asigna el vendor_code correspondiente.
vendor_code	custpage_items_list	escritura	Se rellena automáticamente con el código del proveedor del ítem seleccionado.
description	custpage_items_list	lectura	Desactivado en modo solo lectura cuando el estado final es 3 o 4.
quantity	custpage_items_list	lectura	Desactivado en modo solo lectura cuando el estado final es 3 o 4.
tax_code	custpage_items_list	lectura	Desactivado en modo solo lectura cuando el estado final es 3 o 4.
tax_amount	custpage_items_list	lectura	Desactivado en modo solo lectura cuando el estado final es 3 o 4.
total_amount	custpage_items_list	lectura	Desactivado en modo solo lectura cuando el estado final es 3 o 4.
department	custpage_items_list	lectura	Desactivado en modo solo lectura cuando el estado final es 3 o 4.
class	custpage_items_list	lectura	Desactivado en modo solo lectura cuando el estado final es 3 o 4.
location	custpage_items_list	lectura	Desactivado en modo solo lectura cuando el estado final es 3 o 4.

Validaciones

UBICACIÓN	CAMPO	CRITERIO	ACCIÓN	BLOQUEA
validateField	assign_flag	cambio de valor	habilita o deshabilita custpage_bill_manually	No
fieldChanged	po_checked	solo un checkbox puede estar marcado	deshabilita los demás checkboxes y carga ítems del PO	No
fieldChanged	purchase_order	valor ingresado	completa datos del PO en la cabecera	No
fieldChanged	[Markup('amount'), Markup('tax_rate')]	cambio de valor	recalcula tax_amount y total_amount	No
fieldChanged	custpage_item_po	selección de ítem	asigna vendor_code al ítem	No
validateLine	línea de ítem	excede número de ítems del XML	bloquea la inserción de la línea	Sí
validateInsert	línea de ítem	cualquier intento de inserción	bloquea la inserción	Sí

Interacción con Usuario

alert: Ocurrió un error

Trigger: cualquier excepción capturada

alert: Debe completar la fecha

Trigger: intento de aprobar sin fecha cuando bill_manually es falso y statusFinal no es 3

alert: El monto de purchase order es cero o menor al monto de la factura emitida, por favor modifique su purchase order

Trigger: aprobación con PO cuyo monto es insuficiente

alert: Se generará sin orden de compra.

Trigger: aprobación sin PO seleccionado

Comportamiento del Script

VALIDA DATOS

Sí

BLOQUEA GUARDADO

Sí

OCULTA CAMPOS

Sí

MODIFICA VALORES

Sí

MUESTRA ALERTAS

Sí

Notas: Desactiva campos en modo solo lectura, calcula impuestos automáticamente y muestra alertas ante errores de validación.

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener la preferencia de idioma del usuario.
N/currentRecord	Acceder al registro actual del cliente.
N/record	Cargar registros de Purchase Order y otros tipos cuando es necesario.
N/search	Ejecutar búsquedas de ítems, PO y códigos de proveedor.
N/format	Formatear valores (no usado directamente en la lógica principal).
N/ui/dialog	Mostrar diálogos de confirmación o información al usuario.

Resumen Técnico

Implementa validaciones en los eventos de campo, línea y guardado, desactiva campos según el estado y recalcula impuestos en tiempo real usando búsquedas y lógica de negocio.

INTERNALMODULE · MANEJADOR CLIENTE PARA PROCESAMIENTO DE INBOUND

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador cliente para procesamiento de inbound

Dependencias: N/RUNTIME N/SEARCH N/URL N/RECORD N/HTTPS N/FORMAT
N/CURRENTRECORD N/QUERY N/CURRENCY N/UI/MESSAGE
../TRANSLATOR/LMRY_CO_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza la lógica de validación y procesamiento de documentos inbound en la interfaz cliente, facilitando la generación de logs y la visualización de detalles.'), 'technical_summary': Markup('Expone una clase con métodos de ciclo de vida (pageInit, fieldChanged, saveRecord) que interactúan con APIs de NetSuite para búsquedas, consultas, creación de registros y manejo de UI.')}

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener información del entorno, validar características y preferencias del usuario
N/search	Ejecutar búsquedas de registros y obtener datos de configuración
N/url	Construir URLs a Suitelets y scripts externos
N/record	Crear y guardar registros personalizados de log de inbound
N/https	No utilizado directamente, pero importado para posibles llamadas externas
N/format	Parsear y formatear fechas
N/currentRecord	Acceder al registro actual del cliente
N/query	Obtener información de moneda del proveedor mediante consultas
N/currency	No utilizado directamente, pero importado para posibles conversiones
N/ui/message	Mostrar mensajes informativos al usuario
../Translator/ LMRY_CO_Inbound_Translator	Proveer traducciones de textos según el idioma del usuario

API pública

Clnthandler

Clase que controla la lógica del cliente para la pantalla de inbound, incluye validaciones, apertura de ventanas y generación de logs

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto de configuración opcional (no utilizado internamente)	

Efectos colaterales

- Acceso a runtime, búsqueda de configuraciones, manipulación de campos del registro

Retorna

Instancia de Clnthandler

back

Navega de regreso a la suitelet principal según el estado del registro

Efectos colaterales

- Redirección de la ventana del navegador

Retorna

void

Funciones internas

NOMBRE	ROL	USADO POR
validate_open_window	Verifica si la ventana de detalle ya está abierta antes de crear una nueva	fieldChanged
createMprdIsRunningAlert	Muestra un mensaje informativo de proceso en ejecución	
validateMandatoryFields	Comprueba que los campos obligatorios del formulario tengan valores	saveRecord
validateDates	Valida que la fecha de fin no sea anterior a la fecha de inicio	saveRecord
validateRangeDates	Asegura que las fechas de inicio y fin pertenezcan al mismo mes y año	saveRecord
validateSublist	Confirma que al menos una línea del sublist está marcada para procesamiento	saveRecord
validateExecution	Detecta si ya existe una ejecución de Map/Reduce en curso y muestra alerta	pageInit , saveRecord
fillPeriod	Carga periodos contables activos en el campo de selección	
fillDepartments	Pobla la lista de departamentos según la subsidiaria y la disponibilidad	
fillClasses	Pobla la lista de clases según la subsidiaria y la disponibilidad	
fillLocations	Pobla la lista de ubicaciones según la subsidiaria y la disponibilidad	
fillCurrenciesByEntity	Obtiene y muestra las monedas asociadas al proveedor seleccionado	
createLogInbound	Genera un registro de log con los datos de las transacciones seleccionadas	saveRecord

NOMBRE	ROL	USADO POR
getSetup	Recupera la configuración de taxcode, departamento, clase, ubicación e ítem para una subsidiaria	pageInit
getText	Devuelve el texto traducido correspondiente a una clave	pageInit , fieldChanged , saveRecord , validateMandatoryFields , validateDates , validateRangeDates , validateSublist , validateExecution , createMprdIsRunningAlert
getTranslations	Obtiene el objeto de traducciones para el idioma del usuario	constructor

Generado automáticamente 2026-01-23 18:01 UTC

INTERNALMODULE · MANEJADOR DE POPUP INBOUND CO

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador de Popup Inbound CO

Dependencias: N/URL N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET N/HTTPS
N/FORMAT N/XML N/QUERY N/CONFIG ../TRANSLATOR/LMRY_CO_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo muestra un popup con la información de una factura inbound y permite a los usuarios aprobar, rechazar o crear la factura, facilitando la gestión de documentos de compra en Colombia.'), 'technical_summary': Markup('Expone la clase LibraryHandler que construye dinámicamente un formulario Suitelet, parsea XML inbound, consulta datos maestros y de transacciones, y actualiza el estado del proceso mediante búsquedas y redirecciones.')}

Dependencias

NOMBRE	PROPÓSITO
N/url	Construir URLs a registros y scripts
N/search	Ejecutar búsquedas guardadas y dinámicas
N/runtime	Obtener información del entorno y parámetros de script
N/redirect	Redirigir a otro Suitelet
N/ui/serverWidget	Crear y manipular formularios y sublistas en Suitelets
N/https	Realizar llamadas HTTP externas (no usado directamente en este módulo)
N/format	Formatear y parsear fechas y números
N/xml	Parsear el contenido XML del registro inbound
N/query	Obtener la lista de monedas del proveedor mediante Query
N/config	Leer preferencias contables, como UNRECEIVEDBILLS
../Translator/ LMRY_CO_Inbound_Translator	Proveer traducciones de textos según el idioma del usuario

API pública

LibraryHandler.constructor

Inicializa la instancia con parámetros de la solicitud, detecta características de la cuenta y carga datos XML y de configuración

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto que contiene params (datos del inbound) y method (GET o POST)	

Efectos colaterales

- Carga configuraciones de la cuenta
- Obtiene traducciones cuando el método es GET

Retorna

Instancia de LibraryHandler

createForm

Construye el formulario principal del Suitelet con campos de información primaria y botones de acción según el estado del registro

Efectos colaterales

- Registra depuración mediante log.debug

Retorna

Objeto serverWidget.Form

addVendor

Busca información del proveedor y agrega la opción al campo de selección de vendor

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
vendorField	Campo serverWidget.Field donde se añadirá la opción	

Efectos colaterales

- Realiza una búsqueda lookupFields sobre el registro vendor

createPOSublist

Agrega la pestaña y los campos relacionados a la orden de compra, incluyendo enlaces a PO y XML/PDF cuando corresponda

Efectos colaterales

- Resuelve URLs a registros y scripts

createSublist

Genera sublistas de PO y de ítems según el estado final del proceso

Efectos colaterales

- Crea sublistas LIST e INLINEEDITOR

fillItemSublist

Pobla la sublista de ítems con datos del XML, códigos de proveedor y cálculos de impuestos

Efectos colaterales

- Llama a getItemDetailsByVendorCodes para obtener ítems por código de proveedor

getItemDetailsByVendorCodes

Busca ítems en NetSuite que coincidan con los códigos de proveedor suministrados

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
entity	Internal id del proveedor	
itemNames	Arreglo de códigos de proveedor	

Efectos colaterales

- Ejecuta una búsqueda de tipo ITEM

Retorna

Objeto cuyo key es el código de proveedor y el value es el internal id del ítem

fillPoSublist

Rellena la sublista de órdenes de compra con los resultados obtenidos de getPurchaseOrder

Efectos colaterales

- Construye enlaces a registros de purchaseorder

getPurchaseOrder

Obtiene las órdenes de compra pendientes de facturación para el proveedor y subsidiaria indicados

Efectos colaterales

- Ejecuta una búsqueda paginada de purchaseorder

Retorna

Objeto con internal id como clave y datos de PO como valor

createAssignManually

Agrega la sección para asignar manualmente una factura cuando el estado final es 4

fillManualBill

Llena el campo de selección con facturas de proveedor que no fueron creadas por inbound

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
manualBill	Campo select donde se agregarán las opciones	

Efectos colaterales

- Ejecuta una búsqueda de vendorbill

setFormValues

Actualiza valores por defecto del formulario con datos de la solicitud y, si corresponde, datos de la subsidiaria

Efectos colaterales

- Realiza lookupFields sobre la subsidiaria

getEntities

Construye un mapa de proveedores colombianos (VAT + dígito) a su nombre comercial

Efectos colaterales

- Ejecuta una búsqueda paginada de vendor

Retorna

Objeto map donde la clave es el número de identificación y el valor es el nombre del proveedor

toLogSuitelet

Redirige al Suitelet de registro de pagos WHT

Efectos colaterales

- Llama a `redirect.toSuitelet`

getTranslations

Obtiene el diccionario de traducciones para el idioma indicado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código ISO de dos letras	

Retorna

Objeto con pares clave-valor de textos traducidos

getText

Devuelve el texto traducido para la clave solicitada

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
key	Clave del diccionario de traducciones	

Retorna

Cadena de texto traducido

getDataXML

Parsea el contenido XML del registro inbound y devuelve un objeto con información de la factura y sus líneas

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
inboundEdocId	Internal id del registro <code>customrecord_psg_ei_inbound_edoc</code>	

Efectos colaterales

- Realiza `lookupFields` para obtener el XML
- Usa `N/xml.Parser` para parsear el documento

Retorna

Objeto con `purchaseorder`, `itemreceipt`, `rateIVA` y arreglo de items

addOptions

Agrega opciones a un campo select a partir de una lista de resultados de búsqueda

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
field	Campo select del formulario	
searchList	Arreglo de objetos con id, text y opcional default	

getTaxCodes

Obtiene los códigos de impuesto de venta activos para la subsidiaria y país CO

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
subsidiary	Internal id de la subsidiaria	

Efectos colaterales

- Ejecuta una búsqueda paginada de salestaxitem

Retorna

Arreglo de objetos {id, text}

getDepartments

Obtiene la lista de departamentos activos para la subsidiaria

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
subsidiary	Internal id de la subsidiaria	

Efectos colaterales

- Ejecuta una búsqueda paginada de department

Retorna

Arreglo de objetos {id, text}

getClasses

Obtiene la lista de clasificaciones activas para la subsidiaria

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
subsidiary	Internal id de la subsidiaria	

Efectos colaterales

- Ejecuta una búsqueda paginada de classification

Retorna

Arreglo de objetos {id, text}

getLocations

Obtiene la lista de ubicaciones activas para la subsidiaria

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
subsidiary	Internal id de la subsidiaria	

Efectos colaterales

- Ejecuta una búsqueda paginada de location

Retorna

Arreglo de objetos {id, text}

getCurrencies

Recupera la moneda primaria y secundarias del proveedor mediante Query

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
vendor	Internal id del proveedor	

Efectos colaterales

- Ejecuta una consulta Query sobre vendor y su join currencylist

Retorna

Arreglo de objetos {id, text, default?}

setStatus

Determina el estado final del proceso según la presencia de purchaseorder en el XML

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
jsonXML	Objeto resultante de getDataXML	

Efectos colaterales

- Puede invocar searchPO

searchPO

Busca una orden de compra por número de referencia y monto, estableciendo el estado final y el id de PO encontrado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
folioPO	Número de referencia de la orden de compra	

Efectos colaterales

- Ejecuta una búsqueda de purchaseorder

setMessagePO

Construye el mensaje HTML que se muestra en la pestaña PO según el estado final

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
messagePO	Campo inlinehtml donde se establecerá el mensaje	

convertDate

Convierte una cadena en formato YYYY-MM-DD a objeto Date

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
date	Cadena con formato de fecha	

Retorna

Objeto Date o cadena vacía

getPeriods

Llena un campo select con periodos contables activos y selecciona el periodo actual

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
periodField	Campo select del formulario	

Efectos colaterales

- Ejecuta una búsqueda de accountingperiod

Modelos de datos / estructuras

DataXML

Representa la información extraída del XML inbound

CAMPO	TIPO	DESCRIPCIÓN
purchaseorder	Cadena con número de orden de compra (puede estar vacío)	
itemreceipt	Cadena con número de recibo de ítem (no usado en este módulo)	
rateIVA	Número que indica la tasa de IVA global	
items	Arreglo de objetos con nombre, cantidad, precio, descuento, tasa de IVA, importe de línea, importe de impuesto y opcionalmente código de proveedor	

PurchaseOrderResult

Objeto que agrupa los datos relevantes de una orden de compra encontrada

CAMPO	TIPO	DESCRIPCIÓN
entity	Nombre del proveedor	
status	Estado interno de la PO	
transactionname	Tipo de transacción	
approvalstatus	Estado de aprobación	
trandate	Fecha de la PO	
amount	Monto en moneda extranjera	
otherrefnum	Número de referencia externo	
tranid	Número interno de la PO	

INTERNALMODULE · MANEJADOR SUITELET INBOUND CO

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: class

Manejador Suitelet Inbound CO

Dependencias: N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET N/RECORD
N/FORMAT N/XML N/URL N/TASK ../TRANSLATOR/LMRY_CO_INBOUND_TRANSLATOR

Resumen Ejecutivo

Este módulo centraliza la visualización, filtrado y procesamiento de documentos inbound de Colombia, ofreciendo una interfaz Suitelet que permite consultar, ver PDFs/XML y lanzar conversiones en lote. Técnicamente expone la clase LibraryHandler que construye formularios, consulta registros mediante búsquedas, parsea XML, genera sublistas y orquesta tareas Map/Reduce y registros de log.

Dependencias

NOMBRE	PROPÓSITO
N/search	Ejecutar búsquedas de registros y obtener resultados.
N/runtime	Acceder a información del script, usuario y características de la cuenta.
N/redirect	Redirigir a otros Suitelets o scripts.
N/ui/serverWidget	Construir formularios y sublistas de la interfaz Suitelet.
N/record	Crear y guardar registros de log de inbound.
N/format	Parsear y formatear fechas.
N/xml	Parsear documentos XML de inbound.
N/url	Construir URLs a registros y scripts.
N/task	Ejecutar un Map/Reduce para procesar conversiones.
../Translator/ LMRY_CO_Inbound_Translator	Obtener traducciones de textos según el idioma del usuario.

API pública

LibraryHandler

Constructor que inicializa la instancia con parámetros de solicitud y determina características de la cuenta.

Parámetros: options: objeto que contiene params (objeto con parámetros del request) y method (cadena GET o POST).

Retorna

Instancia de LibraryHandler.

createFilterForm

Genera el formulario de filtro con campos de subsidiaria, fechas y contadores de documentos.

Parámetros: Ninguno.

Efectos colaterales: Crea campos UI y agrega mensajes de licencia cuando no hay subsidiarias.

Retorna

Objeto serverWidget.Form listo para mostrarse.

createApplySublist

Añade una pestaña y sublista al formulario para mostrar los resultados de inbound.

Parámetros: Ninguno.

Efectos colaterales: Define columnas y botones de la sublista.

Retorna

Objeto serverWidget.Sublist configurado.

loadSublist

Llena la sublista con los datos obtenidos del método getResponse y actualiza contadores de estado.

Parámetros: Ninguno.

Efectos colaterales: Realiza búsquedas, resuelve URLs y escribe valores en la sublista.

Retorna

Ninguno (modifica la sublista del formulario).

fillSubsidiaries

Puebla el campo de selección de subsidiarias con las opciones activas.

Parámetros: Ninguno.

Efectos colaterales: Modifica el campo UI del formulario.

Retorna

Ninguno.

getRedirectParams

Construye un objeto con los parámetros necesarios para redirigir al Suitelet de procesamiento.

Parámetros: Ninguno.

Retorna

Objeto con subsidiary, startdate, enddate y status.

setFormValues

Establece los valores por defecto del formulario a partir de los parámetros recibidos.

Parámetros: Ninguno.

Efectos colaterales: Actualiza valores de campos UI y realiza lookup de subsidiaria.

Retorna

Ninguno.

toLogSuitelet

Redirige al Suitelet de log de inbound.

Parámetros: Ninguno.

Efectos colaterales: Ejecuta redirect.toSuitelet.

Retorna

Ninguno (redirección).

executeMapReduce

Inicia la ejecución de un script Map/Reduce para procesar la conversión de documentos.

Parámetros: convertData: objeto con datos a convertir, subsidiary: id de la subsidiaria.

Efectos colaterales: Crea y envía un task de tipo MAP_REDUCE.

Retorna

Id del task Map/Reduce creado.

createLogInbound

Crea un registro customrecord_lmry_co_ei_inbound_po con la información del proceso.

Parámetros: convertData: datos convertidos, params: objeto con fechas, subsidiaria y usuario.

Efectos colaterales: Crea y guarda un registro en NetSuite.

Retorna

Id del registro de log creado.

getResponse

Obtiene y transforma los documentos inbound en un arreglo de resultados estructurados.

Parámetros: Ninguno.

Efectos colaterales: Realiza búsquedas, parsea XML y construye URLs a PDFs y XML.

Retorna

Objeto con results (array de resultados) y campos request/response vacíos.

getTranslations

Obtiene el diccionario de traducciones para el idioma especificado.

Parámetros: language: código de idioma (ej. 'es').

Retorna

Objeto con pares clave-valor de traducciones.

getText

Devuelve el texto traducido para la clave indicada.

Parámetros: key: cadena de la clave de traducción.

Retorna

Texto traducido.

convertDate

Convierte una fecha en formato DD-MM-YYYY a un string formateado según la zona GMT.

Parámetros: date: cadena con formato DD-MM-YYYY.

Retorna

Fecha formateada como string.

Funciones internas

NOMBRE	ROL	USADO POR
getInbounStandarData	helper que recupera los registros inbound, filtra por fechas y devuelve XML y metadatos asociados.	getResponse
getBillsByinboundRecord	helper que busca facturas vendorbill vinculadas a documentos inbound convertidos.	loadSublist
getLatestAuditTrailByState	helper que obtiene el último registro de auditoría para documentos en estado específico.	loadSublist

Modelos de datos / estructuras

InboundResult

Objeto que representa un documento inbound procesado.

CAMPO	TIPO	DESCRIPCIÓN
issuer_name	Nombre del emisor	
issuer_rut	RUT del emisor	
folio	Número de folio del documento	
total_amount	Valor total del documento	
document_type	Tipo de documento (ej. Factura)	
date	Fecha de emisión en formato legible	
see_pdf	URL al PDF del documento	
see_xml	Id del registro XML inbound	
status	Código de estado interno	
statustype	Estado legible (Pending, Converted, etc.)	

InboundStandardData

Estructura devuelta por getInbounStandarData con información agregada.

CAMPO	TIPO	DESCRIPCIÓN
inboundRecordsXml	String con la concatenación de los XML de los inbound filtrados	
vendorList	Arreglo de ids de proveedores asociados	
statusList	Arreglo de códigos de estado de cada inbound	
xmlList	Arreglo de ids de registros XML inbound	
pdfList	Arreglo de ids de PDFs adjuntos	

Efectos colaterales

- **Type:** UI; **Details:** Creación y actualización de formularios y sublistas Suitelet.
- **Type:** SEARCH; **Details:** Ejecución de búsquedas de subsidiarias, inbound records, facturas y auditorías.
- **Type:** RECORD; **Details:** Creación y guardado de registros de log de inbound.
- **Type:** TASK; **Details:** Creación y envío de un task Map/Reduce para procesamiento de conversiones.
- **Type:** REDIRECT; **Details:** Redirección a otros Suitelets de log o PDF.
- **Type:** LOG; **Details:** Uso de log.debug para trazas de ejecución.

Detalles de implementación

Al recibir una solicitud GET o POST, la clase construye un formulario de filtro con campos de subsidiaria y rango de fechas. Cuando el usuario envía el filtro, se ejecuta `getResponse` que busca los registros inbound, parsea sus XML, extrae datos relevantes y los devuelve como resultados. Con `loadSublist` se rellenan los valores en la sublista del formulario, incluyendo enlaces a PDFs, XML y facturas cuando corresponda. Dependiendo del estado del proceso, se habilitan botones para volver o avanzar y se pueden lanzar tareas Map/Reduce para convertir documentos, registrando la actividad en un registro custom. La clase también gestiona traducciones y ajustes de idioma.

Generado automáticamente 2026-01-23 18:02 UTC

INTERNALMODULE · MANEJADOR URET DE VENDOR BILL

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: userEvent

Manejador URET de Vendor Bill

Dependencias: N/LOG N/RUNTIME N/SEARCH N/RECORD N/XML

Resumen Ejecutivo

{'business_summary': Markup('Este script procesa los registros inbound de facturas de proveedores para Colombia y Perú, asignando automáticamente la plantilla de e-doc y manteniendo información adicional como la fecha del documento.'), 'technical_summary': Markup('En beforeSubmit valida y transforma el XML, selecciona la plantilla mediante búsquedas y actualiza el estado; afterSubmit extrae la fecha del XML y sincroniza un registro personalizado con esos datos.')}

Dependencias

NOMBRE	PROPÓSITO
N/log	Registro de logs y errores
N/runtime	Obtener contexto de ejecución
N/search	Realizar búsquedas y lookup de campos
N/record	Cargar, crear y guardar registros personalizados
N/xml	Parsear y validar documentos XML

API pública

beforeLoad

Se ejecuta antes de cargar el registro; actualmente no contiene lógica.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, type, form y request	

beforeSubmit

Valida y actualiza campos del registro inbound de Vendor Bill según país y tipo de documento, asigna plantilla y actualiza el estado del registro.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, oldRecord y type	

Efectos colaterales

- Modifica campos del registro inbound (template, content, status)
- Realiza búsquedas de vendor y plantillas
- Transforma contenido XML

afterSubmit

Después de guardar el registro, extrae la fecha del XML y crea o actualiza el registro customrecord_lmry_inbound_fields asociado.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, oldRecord y type	

Efectos colaterales

- Crea o actualiza registros personalizados de campos extra
- Registra errores en N/log

Funciones internas

NOMBRE	ROL	USADO POR
getXmlData	Extrae la fecha del contenido XML del registro inbound	afterSubmit
createOrUpdateInboundExtraFields	Crea o actualiza el registro customrecord_lmry_inbound_fields con la fecha extraída y el vendor relacionado	afterSubmit
COXMLContent	Obtiene el contenido del nodo Description dentro de Attachment para documentos de Colombia	beforeSubmit
updateXMLContent	Elimina atributos xmlns del XML para normalizar el contenido	beforeSubmit
getEdocTemplate	Busca la plantilla de e-doc adecuada y valida el XML contra su XSD	beforeSubmit

Modelos de datos / estructuras

InboundFields

Registro personalizado que almacena la fecha del documento y el vendor relacionado al inbound

CAMPO	TIPO	DESCRIPCIÓN
custrecord_lmry_inbound_edoc	internalid del registro inbound	
custrecord_lmry_inbound_date	fecha extraída del XML	
custrecord_lmry_inbound_related_vendor	internalid del vendor	

Efectos colaterales

- **Type:** recordModification; **Details:** Modifica campos del registro inbound (template, content, status) durante beforeSubmit
- **Type:** recordCreationOrUpdate; **Details:** Crea o actualiza registros personalizados customrecord_lmry_inbound_fields en afterSubmit
- **Type:** log; **Details:** Registra errores y mensajes mediante N/log

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
createOrUpdateInboundExtraFields	Captura cualquier excepción al crear o actualizar el registro de campos extra	Error registrado en N/log
getEdocTemplate	Captura errores de validación XML contra XSD	Error registrado en log.error

Detalles de implementación

En beforeSubmit, cuando el registro se crea o edita y no proviene de la interfaz de usuario, se obtienen datos del vendor, se determina el país (PE o CO) y se selecciona automáticamente la plantilla de e-doc correspondiente. El contenido XML se normaliza y, si corresponde, se transforma para Colombia. Finalmente se actualiza el estado del registro según la disponibilidad de vendor y plantilla. En caso de ejecución desde UI, solo se normaliza el XML. En afterSubmit se extrae la fecha del XML, se busca o crea un registro customrecord_lmry_inbound_fields y se guarda la información adicional.

Generado automáticamente 2026-01-23 18:01 UTC

SUITELET · VISUALIZADOR DE PDF Y ARCHIVOS INBOUND COLOMBIA

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a usuarios autorizados visualizar o descargar archivos inbound (PDF, texto) de Colombia directamente desde NetSuite.

Flujo de Ejecución

GET

Parámetros de Entrada

Lógica POST

Carga el archivo indicado, determina su tipo y codificación, y envía el contenido con el encabezado adecuado o redirige al URL del archivo

Operaciones

TIPO	TARGET	PROPÓSITO
load	Archivo del File Cabinet	Obtener el contenido y metadatos del archivo solicitado
redirect	URL del archivo	Redirigir al archivo cuando es texto con codificación ISO-8859-1
write	Respuesta HTTP	Entregar el contenido del archivo con el encabezado Content-Type correspondiente

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: Solo lectura del archivo, sin modificar registros

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construcción de formularios (no usado en este Suitelet)
N/search	No utilizado en este Suitelet
N/runtime	Obtener usuario actual y verificar la feature SUBSIDIARIES
N/record	No utilizado en este Suitelet
N/redirect	Redirigir al URL del archivo cuando corresponde
N/file	Cargar el archivo desde el File Cabinet

Resumen Técnico

Carga el archivo mediante N/file, verifica tipo y codificación, y envía el contenido con encabezado adecuado o redirige al URL usando N/redirect.

SUITELET · VISUALIZADOR DE XML INBOUND CO

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar de forma legible el XML asociado a un registro inbound de CO.

Flujo de Ejecución

GET

POST

No hay lógica para solicitudes POST

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_id_message	Message	INLINEHTML	Muestra mensaje de error en HTML dentro del formulario

Lógica POST

En modo GET, busca el contenido XML del registro inbound, lo parsea y lo formatea para devolverlo como texto plano; en caso de error muestra un formulario con mensaje de acceso no permitido.

Búsquedas

Obtener el XML almacenado en el registro inbound

Filtros: search.lookupFields en customrecord_psg_ei_inbound_edoc, columna custrecord_psg_ei_inbound_content

Operaciones

TIPO	TARGET	PROPÓSITO
parse	XML string a documento XML	Convertir el texto XML en un objeto XML para manipularlo
serialize	Documento XML a cadena con indentación	Generar una representación legible del XML
customFormat	Cadena XML	Aplicar sangrado y saltos de línea adicionales mediante la función formatXMLString
responseWrite	Respuesta HTTP	Enviar el XML formateado como texto plano al cliente
formCreate	Formulario de error	Mostrar mensaje de acceso no permitido cuando ocurre una excepción

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Solo muestra datos; no modifica registros ni envía notificaciones

Dependencias

NOMBRE	PROPÓSITO
N/search	Buscar y obtener el campo XML del registro inbound
N/encode	Importado pero no utilizado en el script actual
N/ui/serverWidget	Crear formulario de error y campo INLINEHTML
N/url	Importado pero no utilizado en el script actual
N/xml	Parsear y serializar documentos XML

Resumen Técnico

Obtiene el XML mediante `search.lookupFields`, lo parsea y lo formatea con `N/xml` y una función personalizada, devolviéndolo como texto plano; muestra un formulario de error si ocurre una excepción.

SUITELET · DETALLE DE REGISTRO INBOUND DE COLOMBIA

SuiteScript 2.x

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar los detalles de los documentos inbound de Colombia, incluyendo su estado y vínculo a la factura correspondiente.

Flujo de Ejecución

GET

POST

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
id_internalid	Contador	TEXT	solo lectura
id_tran_id	ID Interno	TEXT	solo lectura
id_name	Nombre Archivo	TEXT	solo lectura
id_vendor	Proveedor	TEXT	solo lectura
id_subsidiary	Subsidiaria	TEXT	solo lectura
id_doc_status	Estado de Documento EI	TEXT	solo lectura
id_tran_ur	Transacción	TEXT	solo lectura

Lógica POST

Recupera la lista de IDs de documentos inbound desde el registro padre, los consulta en bloques y muestra sus datos en la sublista, añadiendo enlaces a facturas cuando existen.

Búsquedas

Obtener los campos del registro customrecord_lmry_co_ei_inbound_po que contiene la lista de IDs y la información de facturas

Filtros: Búsqueda de registro por ID (logid) usando lookupFields

Recuperar los detalles de cada documento inbound (internalid, name, vendor, status, referencia) para los IDs obtenidos

Filtros: Búsqueda en customrecord_psg_ei_inbound_edoc con filtro internalid anyof (bloque de 900 IDs)

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Solo muestra información en una sublista estática; no modifica ni crea registros

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener parámetros de script y verificar si la característica de subsidiarias está activa
N/log	Registrar información de depuración y errores
N/search	Ejecutar búsquedas y lookup de registros
N/record	Importado pero no utilizado en este Suitelet
N/ui/serverWidget	Crear el formulario y la sublista estática
N/url	Construir enlaces a facturas de proveedor

Resumen Técnico

Construye un formulario con sublista estática, consulta registros mediante lookup y búsqueda paginada, y genera enlaces a facturas usando N/url.

Generado automáticamente 2026-01-23 18:11 UTC

CLIENTSCRIPT · CANCEL NAVIGATION SCRIPT FOR INBOUND LOG

SuiteScript 2.1

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Proporciona una función de cancelación que permite al usuario volver a la página anterior desde el formulario de registro de inbound log.

Lifecycle Hooks

saveRecord

Ejecutado al intentar guardar el registro; actualmente no contiene lógica y siempre permite el guardado

Lógica: No realiza ninguna validación ni modificación; retorna implícitamente undefined

Interacción con Usuario

inline_message: Navega a la página anterior al invocar la función cancelar

Trigger: Al ejecutar la función funcionCancel

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

No

MUESTRA ALERTAS

No

Notas: El script no implementa lógica de validación ni manipulación de campos; solo ofrece una función de cancelación que retrocede en el historial del navegador

Resumen Técnico

El Client Script expone una función que llama a `window.history.back()`; no afecta la validación ni la presentación del formulario.

SUITELET · LR - LOG CO INBOUND

SuiteScript 2.0

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar el historial de procesos inbound de Colombia, mostrando estado, número de transacciones y acceso a detalle.

Flujo de Ejecución

GET

POST

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_hidden	HIDDEN	TEXT	oculto (display type HIDDEN)
custpage_id_message	MESSAGE	INLINEHTML	muestra mensaje informativo
id_internalid	Internal ID	TEXT	solo lectura
id_subsidiary	Subsidiary	TEXT	solo lectura
id_datefrom	Date From	TEXT	solo lectura
id_dateto	Date To	TEXT	solo lectura
id_employee	Employee	TEXT	solo lectura
id_total	# Transactions	TEXT	solo lectura
id_transactions	Transactions	TEXTAREA	solo lectura (puede contener enlace a detalle)
id_state	State	TEXTAREA	solo lectura

Lógica POST

Ejecuta una búsqueda sobre el registro `customrecord_lmry_co_ei_inbound_po` para obtener los logs de procesos inbound y los muestra en la sublista. Según el estado del proceso (pending, error, finished) muestra texto o un enlace al detalle.

Búsquedas

Obtener los registros de log de inbound con estado definido

Filtros: Filtro: `custrecord_lmry_co_inb_state_po` no vacío; columnas: `internalid`, usuario, subsidiaria, transacciones, estado, fecha de creación, conteo, fecha desde, fecha hasta; orden descendente por `internalid`

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Muestra historial de procesos inbound, permite refrescar la sublista y acceder a detalle mediante enlace.

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener parámetros del script y verificar si la característica de subsidiarias está habilitada
N/log	Registrar información de depuración y errores
N/search	Ejecutar la búsqueda sobre customrecord_lmry_co_ei_inbound_po
N/record	Importado pero no utilizado directamente en este Suitelet
N/ui/ serverWidget	Crear el formulario, campos y sublista que se muestra al usuario
N/url	Construir la URL al Suitelet de detalle de transacciones

Resumen Técnico

Construye un formulario con sublista estática, ejecuta una búsqueda sobre customrecord_lmry_co_ei_inbound_po y muestra los resultados; genera enlaces a detalle mediante N/url.

SUITELET · DETALLE INBOUND CO SUITELET

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar el detalle de inbound CO mediante un formulario interactivo.

Flujo de Ejecución

GET

POST

No se procesa la solicitud POST

Flujo: El script no contiene lógica para el método POST; cualquier solicitud POST será ignorada.

Genera: Ninguna salida específica

Lógica POST

Utiliza un handler externo para construir dinámicamente el formulario y sus sub-listas, y gestiona errores mostrando un formulario de mensaje.

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Solo atiende solicitudes GET; POST no está implementado

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Crear y manipular formularios y campos de la UI
N/runtime	Obtener información del script en ejecución (id y deployment)
N/redirect	Importado pero no utilizado en este script
../Handlers/ LMRY_CO_Inbound_Detail_STLT_HNDL	Contiene la lógica para crear el formulario y las sub-listas

Resumen Técnico

Construye el formulario y sus sub-listas usando un handler externo y lo muestra en respuesta a solicitudes GET.

MAP/REDUCE · CONVERSIÓN DE DOCUMENTOS ELECTRÓNICOS ENTRANTES A FACTURAS DE PROVEEDOR

Pipeline de procesamiento por lotes (SuiteScript Map/Reduce).

Procesos masivos, búsquedas y operaciones sobre registros.

Metadatos

Tipo: MapReduce

SuiteScript: 2.0

Conversión de documentos electrónicos entrantes a facturas de proveedor

Resumen Ejecutivo

{'business_summary': Markup('Convierte documentos electrónicos entrantes en facturas de proveedor y actualiza el registro de control con el resultado.'), 'technical_summary': Markup('Lee los IDs de documentos inbound, procesa cada uno en la fase map mediante plantillas y plugins personalizados, transforma o crea facturas, y en summarize persiste los resultados en el registro de log.')}

getInputData

Obtiene los IDs de los registros customrecord_psg_ei_inbound_edoc asociados al registro de control inbound PO mediante una búsqueda y los devuelve como arreglo JSON.

Filtros

CAMPO	OPERADOR	VALOR
internalid	anyof	paramIdDocuments

Columnas

NOMBRE	ETIQUETA
custrecord_lmry_co_inb_transactions_po	

map

Parsea el JSON, busca campos del registro inbound (contenido XML y vendor), carga el registro inbound, actualiza o asigna códigos de proveedor en los ítems, invoca la conversión del documento mediante plantillas y plugins personalizados, y escribe como salida la clave id del registro inbound y el resultado de la conversión.

Campos utilizados

- custrecord_psg_ei_inbound_content
- custrecord_psg_ei_inbound_vendor
- tranid
- custrecord_psg_ei_inbound_template
- custrecord_psg_ei_inbound_status
- custrecord_psg_ei_inbound_parsed_content
- custrecord_psg_ei_inbound_refnum
- custrecord_psg_ei_inbound_po
- custrecord_psg_ei_attach_pdf

Operaciones en registros (map)

OPERACIÓN	RECORD	CAMPOS
load	customrecord_psg_ei_inbound_edoc	
load	vendor	
load	purchaseOrder	
submitFields	customrecord_psg_ei_inbound_edoc	custrecord_psg_ei_inbound_status
submitFields	customrecord_psg_ei_inbound_edoc	custrecord_psg_ei_inbound_parsed_content, custrecord_psg_ei_inbound_refnum, custrecord_psg_ei_inbound_po
submitFields	customrecord_psg_ei_inbound_edoc	custrecord_psg_ei_inbound_status
submitFields	customrecord_psg_ei_inbound_edoc	custrecord_psg_ei_inbound_status
transform		custbody_psg_ei_inbound_edocument
create	vendorBill	entity, custbody_psg_ei_inbound_edocument
save	vendorBill	
attach	file	
submitFields	customrecord_lmry_co_ei_inbound_po	custrecord_lmry_co_inb_bill_po, custrecord_lmry_co_inb_state_po
load	item	
save	item	

summarize

Registra en el log de ejecución los resultados agregados y actualiza el registro de control inbound PO con el JSON de resultados y el estado final.

Errores

- Captura y registra cualquier excepción lanzada durante la iteración de resultados

Acciones finales

- Actualiza el registro customrecord_lmry_co_ei_inbound_po con el JSON de resultados y cambia su estado a 2
- Guarda el registro actualizado

Búsquedas utilizadas

ID	FILTROS	COLUMNAS
Inbound PO Search	{'field': Markup('internalid'), 'operator': Markup('anyof'), 'value': Markup('paramIdDocuments')}	custrecord_lmry_co_inb_transactions_po
Setup Inbound Search	{'field': Markup('custrecord_lmry_co_setup_subsidiary'), 'operator': Markup('anyof'), 'value': Markup('subsidiary')}, {'field': Markup('isinactive'), 'operator': Markup('is'), 'value': Markup('F')}	custrecord_lmry_co_setup_item
Created From PO Search	{'field': Markup('tranid'), 'operator': Markup('is'), 'value': Markup('createdFrom')}, {'field': Markup('entity'), 'operator': Markup('anyof'), 'value': Markup('entityId')}	internalid, entity
Bills By Reference Number Search	{'field': Markup('entity'), 'operator': Markup('is'), 'value': Markup('vendorId')}, {'field': Markup('tranid'), 'operator': Markup('is'), 'value': Markup('refNum')}, {'field': Markup('mainline'), 'operator': Markup('is'), 'value': Markup('T')}	tranid
Item Details By Vendor Codes Search	{'group': Markup('OR'), 'conditions': [{ 'field': Markup('vendorcode'), 'operator': Markup('is'), 'value': Markup('each vendor code in list') }], { 'field': Markup('othervendor'), 'operator': Markup('is'), 'value': Markup('entity') }}	internalid, vendorcode
Item Record Type Search	{'field': Markup('internalid'), 'operator': Markup('is'), 'value': Markup('itemId')}	itemid

Operaciones con registros

OPERACIÓN	RECORD	CAMPOS
customrecord_psg_ei_inbound_edoc		
vendor		
purchaseOrder		
vendorBill		
customrecord_lmry_co_ei_inbound_po		
item		
customrecord_lmry_co_setup_inbound		
customrecord_psg_ei_template		

Side effects

- Actualiza el estado del registro inbound a CONVERTING, CONVERTED o FAILED
- Crea o transforma una factura de proveedor (vendor bill) a partir del PO o directamente
- Adjunta el archivo PDF del documento inbound a la factura creada
- Registra auditoría de conversión exitosa o fallida mediante auditManager
- Actualiza códigos de proveedor en los ítems cuando corresponde
- Guarda el JSON de resultados y cambia el estado del registro de control inbound PO

Generado automáticamente 2026-01-23 18:01 UTC

SUITELET · FORMULARIO Y PROCESAMIENTO INBOUND CO

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios filtrar, visualizar y procesar registros inbound de CO, ejecutando procesos de actualización masiva y mostrando los resultados en una interfaz amigable.

Flujo de Ejecución

GET

POST

Procesa la acción del formulario y ejecuta procesos de backend o redirección

Flujo: Al recibir una petición POST, el script lee `custpage_status`. Si el valor es `falsy`, redirige al mismo Suitelet con los parámetros de filtro generados por el handler. Si el valor es verdadero, extrae `custpage_log_id` y `custpage_subsiary`, invoca el handler para ejecutar un proceso Map/Reduce con esos datos y luego redirige al Suitelet de logs.

Genera: Redirección al mismo Suitelet (para volver a cargar filtros) o al Suitelet de logs después de ejecutar Map/Reduce

Parámetros de Entrada

Lógica POST

Construye dinámicamente el formulario y la sublista, gestiona redirecciones y ejecuta un proceso Map/Reduce para procesar logs inbound

Operaciones

TIPO	TARGET	PROPÓSITO
redirect	Suitelet (mismo script)	volver a cargar el formulario con los parámetros de filtro
mapReduce	Proceso de procesamiento de logs inbound	actualizar registros relacionados con el log seleccionado
redirect	Suitelet de logs	mostrar el resultado del procesamiento al usuario

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

Sí

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: El Suitelet depende de un handler externo para la lógica de negocio y utiliza Map/Reduce para actualizar datos

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Crear y manipular formularios y campos de la UI
N/runtime	Obtener información del script en ejecución (id y deployment)
N/redirect	Redirigir al usuario a otro Suitelet o a la misma página con parámetros
../Handlers/ LMRY_CO_Inbound_STLT_HNDL	Contiene la lógica de negocio para crear formularios, sublistas y ejecutar Map/Reduce

Resumen Técnico

Construye dinámicamente formularios y sublistas mediante N/ui/serverWidget, gestiona flujos GET/POST, redirige según la acción del usuario y delega la lógica de negocio a un handler que ejecuta un proceso Map/Reduce.

INTERNALMODULE · CAPTURA Y PROCESAMIENTO DE E-DOCUMENTOS ENTRANTES VÍA EMAIL

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: plugin

Captura y procesamiento de e-documentos entrantes vía email

Dependencias: SUITESCRIPT 1.0 API CUSTOMSCRIPT_UNZIP_STLT

Resumen Ejecutivo

{'business_summary': Markup('Este plugin captura e-documents enviados por correo, los valida y los registra en NetSuite, garantizando que proveedores y clientes sean identificados correctamente y que se notifique al remitente sobre el estado del proceso.'), 'technical_summary': Markup('Procesa un ZIP adjunto, extrae XML/PDF mediante un Suitelet, crea un registro customrecord_psg_ei_inbound_edoc, genera un PDF asociado y envía correos de notificación, manejando errores mediante nlapiCreateError y logs.')}

Dependencias

NOMBRE	PROPÓSITO
SuiteScript 1.0 API	Provee funciones globales de NetSuite como nlapiSearchRecord, nlapiCreateRecord, nlapiSendEmail, etc.
customscript_unzip_stlt	Suitelet que recibe un archivo ZIP, lo descomprime y devuelve la lista de archivos internos

API pública

process

Punto de entrada del plugin que recibe un mensaje de correo, extrae el ZIP adjunto, procesa los archivos XML y PDF, crea el registro Inbound E-Document y envía notificaciones de éxito o error

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
email	Objeto de correo de NetSuite que contiene remitente, asunto y archivos adjuntos	

Efectos colaterales

- Creación de registro customrecord_psg_ei_inbound_edoc
- Creación de archivo PDF en el File Cabinet
- Llamada al Suitelet para descomprimir ZIP
- Envío de correos de notificación a administradores y remitente
- Registro de logs de depuración y errores

Retorna

void

Funciones internas

NOMBRE	ROL	USADO POR
retrieveFile	Obtiene el archivo XML o PDF válido dentro de una lista de archivos	process
unzipZipWithSuitelet	Invoca al Suitelet que descomprime el ZIP y devuelve la lista de archivos extraídos	process
getActiveAdmins	Busca empleados activos con rol de Administrador	sendEmail , sendErrorToSender , sendSuccessToSender
sendEmail	Envía un correo de error a los administradores o al destinatario configurado	process
sendErrorToSender	Notifica al remitente del correo original sobre un error de validación	process
sendSuccessToSender	Confirma al remitente la recepción y procesamiento exitoso del e-documento	process
identifyVendor	Localiza al proveedor a partir del NIT y del correo remitente	process
vendorGenericoFunction	Obtiene el proveedor genérico configurado para Colombia	identifyVendor
identifyCustomer	Localiza al cliente a partir del dominio del remitente o de una lista de correos autorizados	process
updateXMLContent	Elimina atributos xmlns del XML para evitar problemas de validación	process
createPDFFile	Genera un archivo PDF en el File Cabinet asociado al registro Inbound	process
getInboundFolder	Obtiene o crea la carpeta donde se almacenan los archivos XML y PDF inbound	process , createPDFFile
getFolderIdByName	Busca el ID de una carpeta por nombre	getInboundFolder
getSubFolderId	Busca el ID de una subcarpeta bajo un padre especificado	getInboundFolder
createFolder	Crea una carpeta en el File Cabinet	getInboundFolder
getParentSubsidiaryId	Obtiene el ID de la subsidiaria matriz (sin padre)	sendEmail

Modelos de datos / estructuras

InboundFile

Representa un archivo adjunto extraído del ZIP

CAMPO	TIPO	DESCRIPCIÓN
name	nombre del archivo	
content	contenido del archivo en texto plano	

Efectos colaterales

- Creación y actualización de registros customrecord_psg_ei_inbound_edoc
- Creación de carpetas y archivos en el File Cabinet
- Llamada HTTP a un Suitelet interno (customscript_unzip_stlt)
- Envío de correos electrónicos a administradores y remitente
- Escritura de logs de depuración y errores en el registro de ejecución

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
retrieveFile	Lanza error si no se encuentra exactamente un XML o si hay más de un PDF	EMAIL_CAPTURE_ERROR_CODE
process (validación de adjuntos)	Lanza error si el correo no contiene exactamente un archivo ZIP	EMAIL_CAPTURE_ERROR_CODE
identifyVendor	Devuelve "multiple" cuando la búsqueda por NIT devuelve varios proveedores	
identifyCustomer	Lanza error si no se puede identificar un cliente único	EMAIL_CAPTURE_ERROR_CODE
process (creación de registro)	Lanza error si no se asigna ninguna plantilla al registro inbound	EMAIL_CAPTURE_ERROR_CODE

Detalles de implementación

El script recibe un correo, valida que contenga exactamente un archivo ZIP, lo guarda en el File Cabinet y llama a un Suitelet para descomprimirlo. De los archivos extraídos se seleccionan el XML y, opcionalmente, el PDF. El XML se limpia de namespaces y se almacena en un registro customrecord_psg_ei_inbound_edoc junto con metadatos (tipo de transacción, proveedor o cliente, plantilla asociada). Si existe PDF, se crea un archivo PDF enlazado al registro. Finalmente se envía una notificación de éxito al remitente o, en caso de error, se notifica a los administradores y al remitente con detalle del problema.

SUITELET · EXTRACCIÓN Y GUARDADO DE XML Y PDF DESDE UN ZIP

SuiteScript 2.x

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios extraer automáticamente documentos XML y PDF de un archivo ZIP almacenado y guardarlos como archivos independientes en NetSuite.

Flujo de Ejecución

GET

POST

Procesa un ZIP almacenado en la File Cabinet y crea archivos XML y PDF individuales

Flujo: Parsea el cuerpo JSON, valida la presencia de zipFileId, carga el archivo ZIP, lo descomprime con JSZip (indicando que el contenido está en base64), recorre cada entrada y, si la extensión es .xml o .pdf, lee su contenido (texto o base64), crea un archivo NetSuite con el tipo correspondiente, asigna la codificación BASE_64 para PDFs, lo guarda en la carpeta indicada o en la del ZIP y acumula los resultados; envía una respuesta JSON única con la lista de archivos creados o con un mensaje de error

Genera: Respuesta JSON que indica éxito y lista de archivos creados, o error con mensaje descriptivo

Parámetros de Entrada

Lógica POST

Extrae archivos XML y PDF de un ZIP cargado desde la File Cabinet y los guarda como nuevos archivos en NetSuite

Operaciones

TIPO	TARGET	PROPÓSITO
load	N/file	cargar el archivo ZIP mediante su ID
create	N/file	crear un nuevo archivo NetSuite para cada XML o PDF encontrado
save	N/file	persistir el archivo creado y obtener su ID

Comportamiento del Suitelet

CREA REGISTROS

Sí

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: No realiza redirecciones; la respuesta se envía como JSON

Dependencias

NOMBRE	PROPÓSITO
N/file	cargar el ZIP y crear/guardar los archivos resultantes
N/log	registrar información de depuración y errores
JSZip	descomprimir el contenido del ZIP en memoria

Resumen Técnico

El Suitelet recibe un JSON vía POST, usa JSZip para descomprimir el ZIP (base64) y crea archivos mediante N/file, devolviendo un reporte JSON del proceso.

PLUG-IN IMPLEMENTATION · INJECTS CUSTOM DATA SOURCE FOR INBOUND ELECTRONIC INVOICE PROCESSING

SuiteScript 2.0

Documentación técnica del plug-in.

Metadatos

Tipo: PlugInImplementation

SuiteScript: 2.0

Injects custom data source for inbound electronic invoice processing

Resumen Ejecutivo

Provides a custom data source during the generation of electronic invoices, extracting XML data, mapping subsidiary setup fields, formatting dates, and returning a render.DataSource object for template rendering.

Entry points

NOMBRE	DESCRIPCIÓN	INPUTS	OUTPUTS
inject	Creates a custom data source for the rendering engine by extracting information from an inbound electronic invoice XML, applying subsidiary configuration, and formatting values.	Object params containing inboundEDocRec (record), templateRec (record), entityRec (record).	Object with customDataSources array containing a DataSource of type OBJECT, alias "item_type", and the assembled data object.

Búsquedas

PROPÓSITO	FILTROS	RETORNO
Obtain the internal name of a currency based on its symbol code.	Search on record type "currency" where field "symbol" equals the provided currency code.	The first matching currency name.
Retrieve subsidiary-specific setup values (department, class, location, tax codes).	Search on custom record "customrecord_lmry_co_setup_inbound" where field "custrecord_lmry_co_setup_subsidiary" equals the inbound document subsidiary.	Department, class, location texts and tax code IDs for rates 0, 5, and 19.

Dependencias

NOMBRE	PROPÓSITO
N/render	Create DataSource objects for template rendering.
N/log	Log debug and error information.
N/format	Formatting utilities (imported but not directly used).
N/search	Execute saved searches for currency and subsidiary setup.
N/https	HTTPS utilities (imported but not directly used).
N/runtime	Access script parameters and current user information.
N/xml	Parse XML content of the inbound electronic invoice.
/SuiteBundles/Bundle 436209/com.netsuite.electronicinvoicing/src/app/audit/app_audit_manager	Log audit events for conversion failures.
N/error	Create and throw custom NetSuite error objects.

Side effects

- auditManager.logConversionFailed is called when processing fails
- Debug and error logs are written via N/log

Errores

UBICACIÓN	DESCRIPCIÓN	EXCEPCIÓN

PLUG-IN IMPLEMENTATION · INJECT CUSTOM DATA SOURCE DURING INBOUND PROCESSING

SuiteScript 2.0

Documentación técnica del plug-in.

Metadatos

Tipo: PlugInImplementation

SuiteScript: 2.0

Inject custom data source during inbound processing

Resumen Ejecutivo

Provides an inject entry point that adds a custom data source named 'custom' with an empty object payload to the inbound rendering process.

Entry points

NOMBRE	DESCRIPCIÓN	INPUTS	OUTPUTS
inject	Adds a custom data source to the inbound rendering context.	params object containing inboundEDocRec, templateRec, and entityRec records.	Object with a customDataSources array containing a DataSource definition (format OBJECT, alias 'custom', data empty object).

Dependencias

NOMBRE	PROPÓSITO
N/render	Provides DataSource constants for defining the custom data source format.

INTERNALMODULE · ROUTER DE USER EVENT POR TIPO DE TRANSACCIÓN

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: router

Router de User Event por tipo de transacción

Dependencias: N/LOG

Resumen Ejecutivo

Este módulo permite obtener dinámicamente el script de User Event adecuado según el tipo de transacción, facilitando la reutilización y mantenimiento del código. Expone una función que carga el handler correspondiente mediante require y registra errores usando N/log.

Dependencias

NOMBRE	PROPÓSITO
N/log	Registro de mensajes y errores en NetSuite

API pública

getUserEventRouter

Obtiene el handler de User Event correspondiente al tipo de transacción solicitado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
type	Tipo de transacción (por ejemplo, vendorbill)	

Efectos colaterales

- Carga dinámica del módulo handler mediante require
- Escribe en el log de errores en caso de excepción

Retorna

Objeto del handler cargado dinámicamente o cadena vacía si no se encuentra

Efectos colaterales

- Carga dinámica de módulos externos
- Registro de errores en el log de NetSuite

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
getUserEventRouter	Captura cualquier excepción que ocurra al resolver la ruta del handler y la registra en el log	

Detalles de implementación

La función getUserEventRouter define un mapa de tipos de transacción a rutas de módulos handler. Si el tipo solicitado está en el mapa, utiliza require para cargar el módulo de forma asíncrona y asigna el resultado a una variable que luego se devuelve. En caso de error, captura la excepción y la registra mediante N/log.

CLIENTSCRIPT · CARGA DE CONFIGURACIÓN INBOUND AL CAMBIAR SUBSIDIARIA

SuiteScript 2.1 · Módulos: 2

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Automatiza la carga de datos de configuración inbound al seleccionar una subsidiaria, reduciendo errores y tiempo de ingreso manual.

Lifecycle Hooks

fieldChanged

Al cambiar la subsidiaria, busca la configuración inbound asociada y rellena los campos de tax code, departamento, clasificación, ubicación, proveedor y artículo; si no existe configuración, limpia esos campos.

Lógica: Si el campo cambiado es la subsidiaria y tiene valor, ejecuta una búsqueda en `customrecord_lmry_co_setup_inbound` filtrando por la subsidiaria seleccionada. Si se encuentra un registro, extrae los valores de `taxcode`, `department`, `class`, `location`, `vendor` e `item` y los asigna a los campos del formulario. Si no hay resultados, establece en blanco los campos relacionados.

Salida: Actualiza los valores de los campos `custpage_taxcode0`, `custpage_taxcode5`, `custpage_taxcode19`, `custpage_department`, `custpage_class`, `custpage_location`, `custpage_vendor` y `custpage_item` en el registro actual

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
custpage_taxcode0	-	escritura	Se rellena con el valor del taxcode 0 obtenido de la configuración inbound o se limpia si no hay configuración
custpage_taxcode5	-	escritura	Se rellena con el valor del taxcode 5 obtenido de la configuración inbound o se limpia si no hay configuración
custpage_taxcode19	-	escritura	Se rellena con el valor del taxcode 19 obtenido de la configuración inbound o se limpia si no hay configuración
custpage_department	-	escritura	Se rellena con el department extraído del JSON almacenado en la configuración inbound o se limpia si no hay configuración
custpage_class	-	escritura	Se rellena con la clasificación extraída del JSON almacenado en la configuración inbound o se limpia si no hay configuración
custpage_location	-	escritura	Se rellena con la ubicación extraída del JSON almacenado en la configuración inbound o se limpia si no hay configuración
custpage_vendor	-	escritura	Se rellena con el vendor obtenido de la configuración inbound o se limpia si no hay configuración
custpage_item	-	escritura	Se rellena con el item obtenido de la configuración inbound o se limpia si no hay configuración

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: El script solo actúa al cambiar la subsidiaria; no muestra mensajes ni impide el guardado.

Dependencias

NOMBRE	PROPÓSITO
N/search	Ejecutar la búsqueda del registro customrecord_lmry_co_setup_inbound según la subsidiaria seleccionada
N/ currentRecord	Obtener y manipular el registro del formulario activo

Resumen Técnico

Al cambiar la subsidiaria, el script busca la configuración inbound asociada y rellena los campos de tax code, department, class, location, vendor e item; si no existe configuración, limpia esos campos.

SUITELET · LR - CO INBOUND SETUP

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios configurar los parámetros inbound (subsidiaria, impuestos, departamento, clasificación, ubicación, vendor e ítem) para Colombia mediante un formulario sencillo.

Flujo de Ejecución

GET

POST

Procesa los datos enviados del formulario y crea o actualiza el registro de configuración inbound

Flujo: Recibe los parámetros del formulario, busca si ya existe un registro `customrecord_lmry_co_setup_inbound` para la subsidiaria. Si existe, lo carga y actualiza los campos con los valores recibidos (incluyendo JSON para departamento, clasificación y ubicación). Si no existe, crea un nuevo registro con esos valores. Finalmente redirige al mismo Suitelet para recargar la página.

Genera: Redirección al Suitelet (recarga la página)

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_subsidiary	Subsidiaria	SELECT	Obligatorio, opciones pobladas con subsidiarias activas en Colombia
custpage_taxcode0	Tax Code 0	SELECT	Opcional, opciones pobladas con tax codes activos en Colombia
custpage_taxcode5	Tax Code 5	SELECT	Opcional, opciones pobladas con tax codes activos en Colombia
custpage_taxcode19	Tax Code 19	SELECT	Opcional, opciones pobladas con tax codes activos en Colombia
custpage_department	Department	SELECT	Opcional, se llena solo si la característica DEPARTMENTS está habilitada
custpage_class	Class	SELECT	Opcional, se llena solo si la característica CLASSES está habilitada
custpage_location	Location	SELECT	Opcional, se llena solo si la característica LOCATIONS está habilitada
custpage_vendor	Vendor Generic	SELECT	Opcional, fuente vendor
custpage_item	Item Generic	SELECT	Opcional, fuente item

Lógica POST

Procesa la solicitud POST, actualiza o crea el registro de configuración inbound con los valores del formulario

Búsquedas

Obtener subsidiarias activas en Colombia para poblar el campo Subsidiaria

Filtros:

Retorna: internalid y name

Obtener departamentos activos (si la característica está habilitada)

Filtros:

Retorna: internalid y name

Obtener clasificaciones activas (si la característica está habilitada)

Filtros:

Retorna: internalid y name

Obtener ubicaciones activas (si la característica está habilitada)

Filtros:

Retorna: internalid y name

Obtener tax codes activos para Colombia y llenar los tres campos de tax code

Filtros:

Retorna: internalid y name

Buscar registro de configuración inbound existente para la subsidiaria seleccionada

Filtros:

Retorna: internalid

Operaciones

TIPO	TARGET	PROPÓSITO
load	customrecord_lmry_co_setup_inbound	Cargar registro existente para actualizarlo
create	customrecord_lmry_co_setup_inbound	Crear nuevo registro cuando no existe
setValue	campo del registro customrecord_lmry_co_setup_inbound	Asignar valores de tax codes, department, class, location, vendor e item
save	customrecord_lmry_co_setup_inbound	Persistir los cambios en el registro
sendRedirect	Suitelet (mismo script)	Recargar la página después de guardar

Comportamiento del Suitelet

CREA REGISTROS

Sí

ACTUALIZA REGISTROS

Sí

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: Almacena información de department, class y location como JSON en campos personalizados

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construir el formulario y sus campos
N/search	Ejecutar búsquedas para poblar listas y localizar registro de configuración
N/runtime	Verificar si están habilitadas las características de departamentos, ubicaciones y clases
N/record	Cargar, crear y guardar registros customrecord_lmry_co_setup_inbound

Resumen Técnico

El Suitelet muestra un formulario construido con serverWidget, llena listas mediante búsquedas y, al enviarlo, crea o actualiza un registro customrecord_lmry_co_setup_inbound usando la API de record, finalizando con una redirección al mismo Suitelet.

INTERNALMODULE · TRADUCTOR DE ETIQUETAS PARA INBOUND CO

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: utils

Traductor de etiquetas para Inbound CO

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza las traducciones de la interfaz Inbound para Colombia, facilitando su reutilización en otros scripts.'), 'technical_summary': Markup('Expone la función getTranslations que recibe un código de idioma y devuelve un objeto con todas las etiquetas traducidas.')}

API pública

getTranslations

Devuelve el conjunto de etiquetas traducidas según el idioma solicitado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código de idioma (por ejemplo, 'es' o 'en')	

Retorna

Objeto con pares clave-valor donde cada clave es el identificador de la etiqueta y el valor es el texto traducido

Modelos de datos / estructuras

Translations

Mapa de identificadores de etiquetas a su texto traducido

CAMPO	TIPO	DESCRIPCIÓN
clave	string que identifica la etiqueta	
valor	texto traducido correspondiente	

Detalles de implementación

Al recibir un código de idioma, la función selecciona el objeto de etiquetas correspondiente (es o en) y lo devuelve; si el idioma no está definido, se devuelve el conjunto en inglés.

Generado automáticamente 2026-01-23 18:00 UTC

INTERNALMODULE · OBTENCIÓN DINÁMICA DEL MANEJADOR DE EVENTOS DE USUARIO POR PAÍS

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: factory

Obtención dinámica del manejador de eventos de usuario por país

Dependencias: N/LOG N/SUITEAPPINFO

Resumen Ejecutivo

Este módulo permite a otros scripts obtener dinámicamente el manejador de eventos de usuario adecuado según el país y el tipo de transacción, facilitando la integración con los SuiteApps de inbound de LatamReady.

Dependencias

NOMBRE	PROPÓSITO
N/log	Registro de mensajes y errores en el log de NetSuite
N/suiteAppInfo	Comprobar si un SuiteApp está instalado en la cuenta

API pública

getUserEventHandler

Devuelve el objeto con las funciones del script de eventos de usuario correspondiente al país y tipo de transacción

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
country	Código del país asociado a la subsidiaria	
type	Tipo de transacción que dispara el evento	

Efectos colaterales

- Registra un error en el log si ocurre una excepción

Retorna

Objeto con las funciones del manejador de eventos de usuario o null si no se encuentra

Efectos colaterales

- **Type:** log; **Details:** Escribe en el log de errores cuando ocurre una excepción en getUserEventHandler

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
getUserEventHandler	Captura cualquier excepción lanzada durante la carga del router o la verificación del SuiteApp	No propaga la excepción, solo la registra

Detalles de implementación

Comprueba si existe un router definido para el país, verifica la instalación del SuiteApp asociado, carga de forma asíncrona el módulo router mediante require y llama a getUserEventRouter(type) para obtener el manejador, devolviendo el resultado; en caso de error registra el problema.

USEREVENTSCRIPT · DELEGACIÓN DE MANEJO DE FACTURAS DE PROVEEDOR POR PAÍS

SuiteScript 2.1

Eventos del sistema y validaciones de datos.

Resumen Ejecutivo

RESUMEN DE NEGOCIO

Este User Event delega la validación y procesamiento de facturas de proveedor a manejadores específicos por país.

RESUMEN TÉCNICO

Obtiene el país del proveedor y llama al método correspondiente del módulo LMRY_Inbound_Uret_HNDL para beforeSubmit y afterSubmit.

Eventos Implementados

beforeSubmit

Obtiene el país del proveedor y delega la lógica de validación y procesamiento al manejador específico para facturas de proveedor

Lógica: Si el registro tiene un proveedor, determina el país (según OneWorld o configuración de compañía), busca el manejador correspondiente y ejecuta su método beforeSubmit; captura y registra cualquier error

afterSubmit

Obtiene el país del proveedor y delega la lógica posterior al guardado al manejador específico para facturas de proveedor

Lógica: Si el registro tiene un proveedor, determina el país mediante búsqueda, obtiene el manejador correspondiente y ejecuta su método afterSubmit; captura y registra cualquier error

Validaciones

UBICACIÓN	CAMPO	CRITERIO	FUENTE	BLOQUEA
beforeSubmit	custrecord_psg_ei_inbound_vendor	debe existir	campo del registro inbound	No
afterSubmit	custrecord_psg_ei_inbound_vendor	debe existir	campo del registro inbound	No

Búsquedas

Obtener el país del proveedor a partir de su subsidiaria principal

Filtros: internalid del vendor y subsidiaria principal marcada como true

Retorna: valor del campo country de la subsidiaria del vendor

Comportamiento del Script

MODIFICA REGISTRO

No

BLOQUEA GUARDADO

No

LANZA ERRORES

No

CREA REGISTROS

No

ENVÍA NOTIFICACIONES

No

Notas: Delegación a manejadores externos según país; no modifica directamente el registro

Resumen Técnico

Obtiene el país del proveedor y llama al método correspondiente del módulo LMRY_Inbound_Uret_HNDL para beforeSubmit y afterSubmit.

CLIENTSCRIPT · DETALLE INBOUND MX - POPUP DE ORDEN DE COMPRA

SuiteScript 2.1 · Módulos: 6

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

El popup permite validar y completar datos de órdenes de compra y líneas de ítems antes de aprobar una factura inbound, asegurando que la información sea coherente y completa.

Lifecycle Hooks

pageInit

Inicializa el popup, carga datos del registro y deshabilita los campos de la sublista de ítems cuando el XML contiene ítems.

Lógica: Lee valores de campos ocultos, determina el idioma del usuario, llama a setLanguage y, si el XML tiene ítems, ejecuta disableSublistFields. Captura cualquier error y muestra un mensaje genérico.

Salida: Campos de la sublista deshabilitados y variable global LANGUAGE establecida

validateField

Gestiona el comportamiento del campo asignación de bandera y habilita o deshabilita el campo de facturación manual.

Lógica: Si el campo es assign_flag, habilita custpage_bill_manually cuando el valor es verdadero y lo deshabilita cuando es falso, siempre reiniciando su valor a 0. Captura errores y muestra un mensaje genérico.

Salida: Campo custpage_bill_manually habilitado o deshabilitado

fieldChanged

Responde a cambios en varios campos de sublistas para mantener la consistencia y recalcular totales.

Lógica: • Cuando se marca po_checked, deshabilita los checkboxes de otras líneas. • Al cambiar purchase_order en la sublista de PO, completa fecha, monto y tipo de documento mediante completePoValues. • Al modificar amount o tax_rate en la sublista de ítems, recalcula tax_amount y total_amount mediante recalculate. Otros casos están comentados o sin acción.

Salida: Actualización de campos dependientes y recálculo de totales

validateLine

Valida que el número de líneas de ítems no exceda la cantidad declarada en el XML.

Lógica: Obtiene el índice de la línea actual y la compara con la longitud del arreglo items del XML. Si la línea supera el número de ítems, devuelve false para bloquear la validación.

Validaciones

- **línea de ítem:** índice mayor o igual al número de ítems del XML

Acción si falla: bloquear la línea retornando false

validateInsert

Impide la inserción de nuevas líneas en la sublista de ítems.

Lógica: Si la sublista es custpage_items_list, devuelve false para bloquear la inserción.

Validaciones

- **inserción de línea:** sublista custpage_items_list

Acción si falla: bloquear la inserción retornando false

validateDelete

Impide la eliminación de líneas en la sublista de ítems.

Lógica: Si la sublista es `custpage_items_list`, devuelve `false` para bloquear la eliminación.

Validaciones

- **eliminación de línea:** sublista `custpage_items_list`

Acción si falla: bloquear la eliminación retornando `false`

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
assign_flag	-	validación	Al marcarlo, habilita el campo custpage_bill_manually; al desmarcarlo, lo deshabilita.
custpage_bill_manually	-	lectura escritura	Se habilita o deshabilita según el valor de assign_flag; siempre se reinicia a 0.
po_checked	custpage_pos_list	validación	Permite seleccionar solo una línea; al marcarla deshabilita los checkboxes de las demás líneas.
purchase_order	custpage_pos_list	cálculo	Al cambiarlo, completa automáticamente fecha, monto y tipo de documento de la PO seleccionada.
tax_code	custpage_items_list	lectura	Actualmente no ejecuta lógica adicional al cambiarse.
amount	custpage_items_list	cálculo	Al modificarse, recalcula tax_amount y total_amount usando el valor de tax_rate.
tax_rate	custpage_items_list	cálculo	Al modificarse, recalcula tax_amount y total_amount usando el valor de amount.
description	custpage_items_list	lectura	Deshabilitado en pageInit para evitar edición manual.
quantity	custpage_items_list	lectura	Deshabilitado en pageInit.
tax_amount	custpage_items_list	lectura	Deshabilitado en pageInit; se calcula automáticamente.
total_amount	custpage_items_list	lectura	Deshabilitado en pageInit; se calcula automáticamente.
department	custpage_items_list	lectura	Deshabilitado en pageInit; se rellena con valor de configuración.
class	custpage_items_list	lectura	Deshabilitado en pageInit; se rellena con valor de configuración.
location	custpage_items_list	lectura	Deshabilitado en pageInit; se rellena con valor de configuración.
rate	custpage_items_list	lectura	Deshabilitado en pageInit.

Validaciones

UBICACIÓN	CAMPO	CRITERIO	ACCIÓN	BLOQUEA
validateLine	línea de ítem	índice de línea mayor o igual al número de ítems del XML	bloquear la línea retornando false	Sí
validateInsert	inserción de línea	sublista custpage_items_list	bloquear la inserción retornando false	Sí
validateDelete	eliminación de línea	sublista custpage_items_list	bloquear la eliminación retornando false	Sí
updateParent (aprobación)	fecha	aprobado, no facturación manual, fecha vacía y statusFinal distinto de 3	mostrar alerta de fecha requerida y bloquear la operación	Sí
updateParent (aprobación)	monto total	statusFinal 4, suma de totales de ítems distinta al monto total del documento o cero	mostrar alerta de cantidad requerida y bloquear la operación	Sí
updateParent (aprobación)	moneda	statusFinal 4 y campo moneda vacío	mostrar alerta de moneda requerida y bloquear la operación	Sí
updateParent (aprobación)	orden de compra	statusFinal 3, PO seleccionada con monto cero o menor al total de la factura	mostrar alerta de monto de PO insuficiente y bloquear la operación	Sí

Interacción con Usuario

alert: Ocurrió un error

Trigger: cualquier excepción capturada

alert: Debe completar la fecha

Trigger: aprobación sin fecha y sin facturación manual

alert: Debe completar el monto total indicado en el campo Monto Total

Trigger: aprobación cuando la suma de ítems no coincide con el total

alert: Debe completar el campo Currency

Trigger: aprobación cuando el campo moneda está vacío

alert: El monto de purchase order es cero o menor al monto de la factura emitida, por favor modifique su purchase order

Trigger: aprobación con PO cuyo monto es insuficiente

alert: Se generará sin orden de compra.

Trigger: aprobación cuando no se selecciona ninguna PO

Comportamiento del Script

VALIDA DATOS

Sí

BLOQUEA GUARDADO

No

OCULTA CAMPOS

Sí

MODIFICA VALORES

Sí

MUESTRA ALERTAS

Sí

Notas: Deshabilita campos de la sublista, completa datos de PO y de ítems, y muestra alertas de validación antes de aprobar.

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener la preferencia de idioma del usuario y verificar características de la cuenta
N/currentRecord	Acceder al registro que se está editando en el cliente
N/url	No utilizado directamente en la lógica actual, reservado para generación de URLs
N/search	Buscar órdenes de compra, ítems, configuraciones y códigos de impuesto
N/format	Formatear valores (no usado en la versión actual del script)
N/ui/dialog	Mostrar diálogos y alertas al usuario

Resumen Técnico

El Client Script controla la habilitación de campos, rellena automáticamente datos de PO, recalcula impuestos y bloquea inserciones o eliminaciones de líneas, mostrando alertas cuando se incumplen reglas de negocio.

CLIENTSCRIPT · MANEJO DE INBOUND DE PAGOS EN CLIENTE

SuiteScript 2.1 · Módulos: 3

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Facilita la interacción del usuario en el formulario de inbound de pagos, permitiendo marcar/desmarcar rápidamente todas las casillas de aplicación y delegando la validación y guardado a lógica especializada.

Lifecycle Hooks

pageInit

Inicializa el handler y delega la lógica de carga del formulario

Lógica: Crea una instancia del handler, la almacena y ejecuta su método pageInit pasando el contexto

Salida: Inicializa valores y configuraciones del registro según la lógica del handler

fieldChanged

Reenvía el evento de cambio de campo al handler

Lógica: Llama al método fieldChanged del handler con el contexto recibido

Salida: Ejecuta validaciones o actualizaciones definidas en el handler

saveRecord

Valida y procesa el guardado delegando al handler

Lógica: Invoca handler.saveRecord y devuelve su resultado (true permite guardar, false lo bloquea)

Salida: Bloquea o permite el guardado según la lógica del handler

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
apply	sublistId dinámico	escritura	Se marca o desmarca en todas las líneas del sublist mediante toggleCheckBoxes

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: Incluye función toggleCheckBoxes para marcar o desmarcar la casilla 'apply' en todas las líneas de una sublista

Dependencias

NOMBRE	PROPÓSITO
../Handlers/ LMRY_MX_Inbound_CLNT_HNDL	Contiene la lógica de negocio para la inicialización, cambios de campo y validación al guardar
N/url	Proporciona utilidades de URL que pueden ser usadas por el handler
N/currentRecord	Permite acceder al registro actual para manipular valores de sublistas

Resumen Técnico

Instancia un handler externo para gestionar pageInit, fieldChanged y saveRecord, y expone una utilidad toggleCheckBoxes que actualiza el campo 'apply' en cada línea de una sublista mediante la API currentRecord.

INTERNALMODULE · MANEJADOR DE POPUP INBOUND MX PARA DETALLE DE TRANSACCIONES

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: class

Manejador de Popup Inbound MX para detalle de transacciones

Dependencias: N/URL N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET N/HTTPS
N/FORMAT N/XML N/QUERY N/CONFIG N/LOG
../TRANSLATOR/LMRY_MX_INBOUND_TRANSLATOR

Resumen Ejecutivo

Este módulo centraliza la lógica de visualización y validación de documentos inbound para México, permitiendo a los usuarios revisar la información del XML, comparar con órdenes de compra existentes y crear o asociar facturas de proveedor. Técnicamente, expone la clase LibraryHandler que construye dinámicamente un Suitelet con formularios, sublistas y botones, consumiendo búsquedas, traducciones y configuraciones específicas de la subsidiaria.

Dependencias

NOMBRE	PROPÓSITO
N/url	Construir URLs a registros y scripts para enlaces en el formulario
N/search	Ejecutar búsquedas de registros (vendor, purchaseorder, item, etc.)
N/runtime	Obtener parámetros de script, preferencias de usuario y estado de features
N/redirect	Redirigir a otro Suitelet (log de pagos)
N/ui/serverWidget	Crear y manipular formularios, campos, sublistas y botones
N/https	No utilizado directamente, pero disponible para llamadas externas
N/format	Parsear y formatear fechas
N/xml	Parsear el contenido XML del registro inbound
N/query	Obtener la lista de monedas del vendor mediante Query API
N/config	Leer la preferencia de contabilidad UNRECEIVEDBILLS
N/log	Registrar información de depuración y errores
../Translator/ LMRY_MX_Inbound_Translator	Proveer traducciones de textos según el idioma del usuario

API pública

LibraryHandler.constructor

Inicializa la instancia con parámetros de solicitud, determina features y preferencias, carga el XML y la configuración de la subsidiaria, y prepara el traductor si el método es GET

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto que contiene params (parámetros de la petición) y method (GET o POST)	

Efectos colaterales

- Lectura de parámetros de script
- Lectura de preferencias de usuario
- Carga de configuración mediante búsqueda
- Parseo de XML mediante N/xml

Retorna

Instancia de LibraryHandler

LibraryHandler.createForm

Construye el formulario principal del popup, agrega campos de información primaria, retenciones, enlaces a XML/PDF, y botones de acción según el estado del registro

Efectos colaterales

- Creación de campos y sublistas en el formulario
- Llamadas a search.lookupFields para obtener datos del vendor
- Registro de depuración mediante log.debug

Retorna

Objeto serverWidget.Form

LibraryHandler.addVendor

Busca el nombre del vendor y lo asigna al campo SELECT correspondiente

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
vendorField	Campo SELECT del formulario donde se añadirá la opción del vendor	

Efectos colaterales

- Búsqueda de campos del registro vendor

LibraryHandler.createSublist

Crea la pestaña PO y la sublista de órdenes de compra, y si corresponde, la pestaña Items con su sublista editable

Efectos colaterales

- Creación de pestañas y sublistas en el formulario
- Llamadas a `getPurchaseOrder` y a búsquedas de códigos de retención

LibraryHandler.fillItemsSublist

Puebla la sublista de items con la información extraída del XML y, cuando corresponde, agrega líneas de retención

Efectos colaterales

- Seteo de valores en la sublista
- Búsqueda de códigos de ítem por vendorcode

LibraryHandler.getPurchaseOrder

Ejecuta una búsqueda paginada de órdenes de compra que cumplen con los filtros de subsidiaria, vendor y estado, devolviendo un mapa de información relevante

Efectos colaterales

- Búsqueda de purchaseorder

Retorna

Objeto con claves internalid y valores con datos de la PO

LibraryHandler.setStatus

Determina el estado final del registro a partir del número de orden de compra presente en el XML

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
jsonXML	Objeto resultante del parseo del XML	

LibraryHandler.searchPO

Busca una orden de compra por folio y monto, actualizando la propiedad statusFinal y guardando el internalId encontrado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
folioPO	Folio de la orden de compra extraído del XML	

Efectos colaterales

- Búsqueda de purchaseorder mediante filtros dinámicos

LibraryHandler.getDataXML

Obtiene el contenido XML del registro inbound, lo parsea y devuelve un objeto estructurado con items, retenciones y datos de impuestos

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
xmlData	InternalId del registro customrecord_psg_ei_inbound_edoc	

Efectos colaterales

- Lectura de campo custrecord_psg_ei_inbound_content
- Parseo de XML mediante N/xml

Retorna

Objeto con propiedades purchaseorder, itemreceipt, rateIVA, items y retenciones

LibraryHandler.getTranslations

Obtiene el diccionario de traducciones para el idioma especificado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código ISO de dos letras (ej. 'es')	

Retorna

Objeto con pares clave-texto traducido

LibraryHandler.toLogSuitelet

Redirige al Suitelet de log de pagos

Efectos colaterales

- Redirección mediante N/redirect

Modelos de datos / estructuras

InboundXMLData

Representa la información extraída del XML del documento inbound

CAMPO	TIPO	DESCRIPCIÓN
purchaseorder	string - folio de la orden de compra	
itemreceipt	string - número de recibo de ítem (no usado)	
rateIVA	número - tasa de IVA del documento	
items	array de objetos con name, quantity, price, discount, rateIvaItem, taxAmount y opcional vlrCodigo	
retenciones	array de objetos que contiene retenciones y el importe total	

SetupData

Configuración por subsidiaria obtenida de customrecord_lmry_mx_setup_inbound

CAMPO	TIPO	DESCRIPCIÓN
taxcode0	internalid del código de impuesto 0 %	
taxcode8	internalid del código de impuesto 8 %	
taxcode16	internalid del código de impuesto 16 %	
department	internalid del departamento por defecto	
class	internalid de la clasificación por defecto	
location	internalid de la ubicación por defecto	
item	internalid del ítem por defecto	
withholding	JSON string con tabla de retenciones	

PurchaseOrderInfo

Datos relevantes de una orden de compra obtenidos en la búsqueda

CAMPO	TIPO	DESCRIPCIÓN
entity	texto - nombre del vendor	
status	texto - estado interno de la PO	
transactionname	texto - tipo de transacción	
approvalstatus	texto - estado de aprobación	
trandate	fecha - fecha de la PO	
amount	número - monto en moneda extranjera	
otherrefnum	texto - número de referencia (folio)	
tranid	texto - número interno de la transacción	

Efectos colaterales

- Búsqueda de registros mediante N/search
- Lectura de campos mediante search.lookupFields
- Creación y manipulación de formularios y sublistas (N/ui/serverWidget)
- Resolución de URLs a registros y scripts (N/url)
- Redirección a otro Suitelet (N/redirect)
- Registro de mensajes de depuración y errores (N/log)
- Parseo de XML (N/xml)
- Consulta de monedas mediante N/query

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
createForm - parsing de tabla de retenciones	Error al parsear la tabla JSON de retenciones	capturado y registrado con log.error
fillItemsSublist - agregando líneas de retención	Error al procesar la información de retención y asignar ítems de retención	capturado y registrado con log.error
getDataXML - extracción de datos del XML	Cualquier excepción durante el parseo del XML o acceso a atributos	no se captura explícitamente, se propaga

Detalles de implementación

La clase LibraryHandler se utiliza en un Suitelet para mostrar el detalle de un documento inbound (CFDI). Al instanciarse, carga el XML, traduce los textos y recupera la configuración de la subsidiaria. createForm construye el formulario principal con campos de información del vendor, importe, folio y retenciones. Según el estado (statusFinal) se añaden botones de aprobar, rechazar o crear factura y se generan sublistas de órdenes de compra y de ítems. La sublista de ítems se llena con los conceptos del XML y, cuando existen retenciones, se añaden líneas de retención usando los códigos de ítem configurados. El módulo también permite buscar PO existentes, obtener listas de códigos de impuestos, departamentos, clases, ubicaciones y monedas, y redirigir a un Suitelet de log. Todo el proceso está orientado a validar y preparar la información para la creación o asociación de una factura de proveedor.

INTERNALMODULE · MANEJADOR CLIENTE PARA INBOUND DE MÉXICO

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: class

Manejador cliente para inbound de México

Dependencias: N/RUNTIME N/RECORD N/SEARCH N/URL N/HTTPS N/FORMAT
N/CURRENTRECORD N/QUERY N/CURRENCY N/UI/MESSAGE
../TRANSLATOR/LMRY_MX_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo gestiona la validación, selección y registro de documentos inbound para México, permitiendo que otros scripts inicien su procesamiento de forma segura.'), 'technical_summary': Markup('Expone la clase ClntHandler con métodos de UI, validaciones y creación de un registro de log, y una función back para la navegación al suitelet.')}

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Acceder a información de tiempo de ejecución y preferencias del usuario
N/record	Crear y manipular registros de NetSuite
N/search	Ejecutar búsquedas guardadas y consultas dinámicas
N/url	Construir URLs a Suitelets y scripts
N/https	Realizar llamadas HTTP externas
N/format	Parsear y formatear valores de tipo fecha
N/currentRecord	Obtener el registro actual en la UI del cliente
N/query	Ejecutar consultas tipo SQL sobre entidades NetSuite
N/currency	Operaciones relacionadas con monedas
N/ui/message	Mostrar mensajes informativos al usuario
../Translator/ LMRY_MX_Inbound_Translator	Proveer traducciones de textos según idioma

API pública

Clnthandler

Clase que encapsula la lógica de validación, carga de datos y generación de logs para el proceso inbound mexicano

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto de configuración opcional (no utilizado)	

Retorna

Instancia de la clase

Clnthandler.pageInit

Inicializa la pantalla, deshabilita campos según el estado y valida la ejecución concurrente

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
context	Objeto de contexto con la referencia al registro actual	

Efectos colaterales

- Modifica valores y propiedades de campos del registro
- Muestra alertas al usuario

Retorna

true si la inicialización se completa

Clnthandler.fieldChanged

Maneja cambios en la sublista, abre ventana de detalle y controla selección de documentos

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
context	Información del campo modificado	

Efectos colaterales

- Abre ventana emergente
- Actualiza valores de sublista
- Muestra alertas

Retorna

true al finalizar el manejo

ClntHandler.saveRecord

Valida datos antes de guardar y, si corresponde, crea un registro de log para el proceso MRPD

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
context	Contexto de guardado con referencia al registro	

Efectos colaterales

- Crea registro customrecord_lmry_ei_inbound_po
- Muestra alertas

Retorna

true si el registro puede guardarse, false en caso de error de validación

ClntHandler.validate_open_window

Verifica si ya existe una ventana de detalle abierta para la línea indicada

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
line	Índice de la línea de la sublista	

Retorna

true si se puede abrir una nueva ventana, false si ya está abierta

ClntHandler.createMprdIsRunningAlert

Muestra mensaje informativo indicando que el proceso de conversión está en ejecución

Efectos colaterales

- Muestra mensaje UI

ClntHandler.reloadMprdDone

Método placeholder sin implementación

ClntHandler.validateMandatoryFields

Comprueba que los campos obligatorios del formulario tengan valores

Efectos colaterales

- Muestra alertas en caso de falta de datos

Retorna

true si todos los campos obligatorios están completos

ClntHandler.validateDates

Valida que la fecha de fin no sea anterior a la fecha de inicio

Efectos colaterales

- Muestra alertas

Retorna

true si las fechas son consistentes

ClntHandler.validateRangeDates

Asegura que las fechas de inicio y fin pertenezcan al mismo mes y año

Retorna

true si están en el mismo período, false en caso contrario

ClntHandler.validateSublist

Verifica que al menos una línea de la sublista esté marcada para procesamiento

Efectos colaterales

- Muestra alertas

Retorna

true si hay líneas seleccionadas, false si la lista está vacía o sin selección

ClntHandler.validateExecution

Comprueba si existe una ejecución programada del script MRPD en curso y notifica al usuario

Efectos colaterales

- Actualiza campo `custpage_is_processing`
- Muestra mensaje UI

Retorna

false si hay una ejecución en proceso, true en caso contrario

Clnthandler.fillPeriod

Carga los periodos contables activos en el campo de selección del formulario

Efectos colaterales

- Modifica opciones del campo de periodo

Clnthandler.fillDepartments

Pobla la lista de departamentos según la subsidiaria y la disponibilidad

Efectos colaterales

- Modifica opciones del campo de departamento
- Realiza búsqueda asíncrona

Clnthandler.fillClasses

Pobla la lista de clases contables según la subsidiaria

Efectos colaterales

- Modifica opciones del campo de clase
- Realiza búsqueda asíncrona

Clnthandler.fillLocations

Pobla la lista de ubicaciones según la subsidiaria

Efectos colaterales

- Modifica opciones del campo de ubicación
- Realiza búsqueda asíncrona

Clnthandler.fillCurrenciesByEntity

Obtiene las monedas asociadas al proveedor seleccionado y las inserta en el campo de moneda

Efectos colaterales

- Modifica opciones del campo de moneda
- Ejecuta consulta query

Clnthandler.createLogInbound

Genera un registro de log con los datos de las líneas seleccionadas para su posterior procesamiento MRPD

Efectos colaterales

- Crea registro customrecord_lmry_ei_inbound_po
- Actualiza campo custpage_log_id
- Escribe en consola

Retorna

false (valor de retorno no utilizado)

ClntHandler.getSetup

Recupera la configuración de inbound para la subsidiaria indicada

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
subsidiary	Id interno de la subsidiaria	

Retorna

Objeto con los valores de configuración

ClntHandler.getText

Devuelve la cadena traducida correspondiente a la clave solicitada

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
key	Identificador del texto	

Retorna

Texto traducido

ClntHandler.getTranslations

Obtiene el conjunto de traducciones para el idioma especificado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código de idioma (dos letras)	

Retorna

Objeto con pares clave-valor de traducciones

back

Navega de regreso al suitelet de búsqueda con los parámetros actuales

Efectos colaterales

- Redirige la ventana del navegador
- Muestra mensaje en consola

Funciones internas

NOMBRE	ROL	USADO POR
handleError	Registra errores en la consola y, opcionalmente, los muestra al usuario	ClntHandler.pageInit , ClntHandler.fieldChanged , ClntHandler.saveRecord

Efectos colaterales

- **Type:** UI; **Details:** Muestra alertas, mensajes informativos y ventanas emergentes
- **Type:** Record; **Details:** Crea y actualiza registros customrecord_lmry_ei_inbound_po y campos del registro actual
- **Type:** Search; **Details:** Ejecuta búsquedas y consultas para poblar listas desplegables
- **Type:** Network; **Details:** Construye URLs y abre ventanas popup para detalle de documentos
- **Type:** Console; **Details:** Escribe información de depuración y errores

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
pageInit	Captura cualquier excepción durante la inicialización	Muestra alerta genérica al usuario
fieldChanged	Captura errores al procesar cambios de campo	Muestra alerta genérica
saveRecord	Captura errores al validar o crear el log	Muestra alerta y devuelve false

Detalles de implementación

El módulo expone la clase ClntHandler que controla la interacción del usuario con la pantalla de inbound mexicano. En pageInit se preparan los campos y se verifica que no haya procesos concurrentes. fieldChanged abre una ventana de detalle para la línea seleccionada y controla la selección única. saveRecord valida los datos ingresados; si el proceso está en estado de ejecución crea un registro de log con la información de las líneas marcadas. El método validateExecution evita ejecuciones simultáneas del script MRPD. Los métodos fill* cargan listas de selección (periodos, departamentos, clases, ubicaciones, monedas) según la configuración de la subsidiaria. createLogInbound construye un registro customrecord_lmry_ei_inbound_po con los datos de las transacciones a procesar. La función back devuelve al usuario al suitelet de búsqueda original.

INTERNALMODULE · MANEJADOR DE SUITELET INBOUND MÉXICO

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador de Suitelet Inbound México

Dependencias: N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET N/RECORD
N/FORMAT N/XML N/URL N/TASK ../TRANSLATOR/LMRY_MX_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza la lógica de visualización y procesamiento de documentos inbound para México, ofreciendo filtros por subsidiaria y fechas, y permitiendo la conversión masiva mediante Map/Reduce.'), 'technical_summary': Markup('Expone la clase LibraryHandler que construye formularios Suitelet, consulta registros inbound mediante búsquedas, genera sublistas con enlaces a PDF/XML y controla la ejecución de procesos asíncronos y de registro de auditoría.')}

Dependencias

NOMBRE	PROPÓSITO
N/search	Ejecutar búsquedas y paginación de registros
N/runtime	Obtener información del script, usuario y características activas
N/redirect	Redirigir a Suitelets o registros
N/ui/serverWidget	Construir formularios y sublistas en la UI del Suitelet
N/record	Crear y guardar registros de log de inbound
N/format	Formatear fechas
N/xml	Parsear y manipular XML de documentos inbound
N/url	Construir URLs a scripts y registros
N/task	Ejecutar Map/Reduce
../Translator/LMRY_MX_Inbound_Translator	Obtener traducciones según idioma del usuario

API pública

LibraryHandler.constructor

Inicializa la instancia con parámetros, método y verifica características del entorno

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto que contiene params y method	

Retorna

instancia de LibraryHandler

LibraryHandler.areThereSubsidiaries

Determina si existen subsidiarias activas y las almacena en this.subsidiaries

Retorna

boolean indicando si hay al menos una subsidiaria activa

LibraryHandler.getSubsidiaries

Obtiene la lista de subsidiarias mexicanas activas mediante una búsqueda

Retorna

array de objetos {value, text, active}

LibraryHandler.createFilterForm

Construye el formulario de filtro del Suitelet con campos de subsidiaria, fechas y contadores de documentos

Efectos colaterales

- crea campos UI
- agrega botones y mensajes HTML

Retorna

objeto serverWidget.Form

LibraryHandler.createApplySublist

Agrega una sublista de resultados al formulario con columnas para datos del documento inbound

Efectos colaterales

- crea columnas UI
- configura botones de selección

Retorna

objeto serverWidget.Sublist

LibraryHandler.loadSublist

Puebla la sublista con los resultados obtenidos de getResponse, calcula totales y genera enlaces a PDF/XML

Efectos colaterales

- setea valores en la sublista
- actualiza contadores en campos del formulario

LibraryHandler.fillSubsidiaries

Llena el campo SELECT de subsidiarias con las opciones activas

LibraryHandler.getRedirectParams

Construye un objeto con los parámetros necesarios para redirigir al Suitelet en modo de visualización

Retorna

objeto con subsidiary, startdate, enddate, uuidInExpense y status

LibraryHandler.setFormValues

Establece los valores por defecto del formulario a partir de los parámetros recibidos

Efectos colaterales

- actualiza valores por defecto de campos UI
- realiza lookup de datos de subsidiaria

LibraryHandler.toLogSuitelet

Redirige al Suitelet de log de inbound

Efectos colaterales

- redirección a otro Suitelet

LibraryHandler.executeMapReduce

Inicia la ejecución de un script Map/Reduce pasando los datos de conversión y la subsidiaria

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
convertData	Datos a procesar en el Map/Reduce	
subsidiary	Id de la subsidiaria	

Efectos colaterales

- crea y envía tarea Map/Reduce

Retorna

id del job Map/Reduce

LibraryHandler.createLogInbound

Crea un registro customrecord_lmry_ei_inbound_po con información del proceso de inbound

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
convertData	Datos convertidos a registrar	
params	Objeto con fechas, subsidiaria, usuario y conteo	

Efectos colaterales

- crea y guarda un registro custom

Retorna

internalId del registro creado

LibraryHandler.getTranslations

Obtiene el diccionario de traducciones para el idioma indicado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código ISO del idioma (ej. 'es')	

Retorna

objeto con pares clave-valor de traducción

LibraryHandler.getText

Devuelve la cadena traducida para la clave indicada

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
key	Clave del diccionario de traducción	

Retorna

texto traducido

LibraryHandler.getResponse

Construye la lista de resultados a partir de los registros inbound y sus archivos XML/PDF

Retorna

objeto {results: array, request: string, response: string}

LibraryHandler.getInboundStandardData

Recupera los XML, vendors, estados y PDFs de los registros inbound dentro del rango de fechas

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
startDate	Fecha inicial del filtro	
endDate	Fecha final del filtro	
subsidiary	Id de la subsidiaria	

Efectos colaterales

- ejecuta búsquedas paginadas

Retorna

objeto con inboundStandardXml, vendorList, statusList, documentIds, pdfList

LibraryHandler.getBillsByInboundRecord

Obtiene los vendor bills asociados a documentos inbound convertidos

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
data	Array de resultados de getResponse	

Efectos colaterales

- búsqueda de vendor bills

Retorna

objeto map con internalid y tranid por inboundId

LibraryHandler.getLatestAuditTrailByState

Recupera el último registro de auditoría para cada inbound en el estado indicado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
data	Array de resultados de getResponse	
state	Estado a filtrar (ej. 'Failed')	

Efectos colaterales

- búsqueda de customrecord_psg_ei_audit_trail

Retorna

objeto con details por inboundId

LibraryHandler.getDateFromXml

Extrae la fecha del atributo Fecha del XML del documento

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
xmlData	String XML del documento	

Retorna

fecha en formato YYYY-MM-DD o null

LibraryHandler.getUuidInExpense

Verifica si los UUID de los documentos inbound están registrados en algún expense report

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
uuidList	Array de UUID a buscar	
subsidiary	Id de la subsidiaria	

Efectos colaterales

- búsqueda de expensereport

Retorna

objeto map UUID → internalId de expense report

LibraryHandler.getSetup

Obtiene la configuración de validación de expense report para la subsidiaria

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
subsidiary	Id de la subsidiaria	

Efectos colaterales

- búsqueda de customrecord_lmry_mx_setup_inbound

Retorna

objeto con flag validateExpenseReport

LibraryHandler.getInboundRecordsIds

Devuelve los IDs de los registros inbound que cumplen con el rango de fechas y subsidiaria

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
startDate	Fecha inicial	
endDate	Fecha final	
subsidiary	Id de la subsidiaria	

Efectos colaterales

- búsqueda paginada de customrecord_lmry_inbound_fields

Retorna

array de internalId de registros inbound

LibraryHandler.convertDate

Convierte una fecha en formato YYYY-MM-DD a la representación de NetSuite

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
date	String fecha en formato YYYY-MM-DD	

Retorna

fecha formateada como string

Modelos de datos / estructuras

InboundResult

Objeto que representa un documento inbound procesado

CAMPO	TIPO	DESCRIPCIÓN
issuer_name	nombre del emisor (vendedor)	
issuer_rut	RFC del emisor	
folio	folio del CFDI	
total_amount	importe total del documento	
document_type	tipo de documento (ej. Factura)	
date	fecha del documento (YYYY-MM-DD)	
see_pdf	URL al PDF generado	
see_xml	ID del registro inbound (XML)	
status	código de estado interno	
statustype	texto descriptivo del estado (Pending, Converted, etc.)	
uuid	UUID del timbre fiscal	

INTERNALMODULE · MANEJO DE VENDOR BILL INBOUND URET

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: userEvent

Manejo de Vendor Bill inbound URET

Dependencias: N/LOG N/RUNTIME N/SEARCH N/RECORD

Resumen Ejecutivo

Este módulo gestiona la lógica de procesamiento de facturas de proveedor inbound, extrayendo información del XML y manteniendo campos auxiliares para su uso en otros procesos. Se ejecuta como User Event afterSubmit, utiliza búsquedas y operaciones de registro para crear/actualizar un registro de campos extra y, cuando corresponde, actualiza el estado del registro inbound.

Dependencias

NOMBRE	PROPÓSITO
N/log	Registro de mensajes de depuración y errores
N/runtime	Acceso a información del entorno de ejecución
N/search	Ejecución de búsquedas guardadas y dinámicas
N/record	Creación, carga y actualización de registros de NetSuite

API pública

beforeLoad

Se ejecuta antes de cargar el registro; actualmente no contiene lógica

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, type, form y request	

Retorna

void

beforeSubmit

Se ejecuta antes de enviar el registro; actualmente no contiene lógica

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, oldRecord y type	

Retorna

void

afterSubmit

Procesa el registro inbound después de guardarse, extrae la fecha del XML, crea o actualiza campos extra y, si el proveedor es genérico, actualiza el estado del registro

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, oldRecord y type	

Efectos colaterales

- record.submitFields
- search.create
- record.load
- record.create
- record.save
- log.error

Retorna

void

Funciones internas

NOMBRE	ROL	USADO POR
getSetupData	Busca la configuración inbound para determinar si el proveedor está marcado como genérico	afterSubmit
getXmlData	Extrae la fecha del atributo Fecha del contenido XML del registro inbound	afterSubmit , createOrUpdateInboundExtraFields
createOrUpdateInboundExtraFields	Crea o actualiza el registro customrecord_lmry_inbound_fields con la fecha extraída y el proveedor relacionado	afterSubmit

Modelos de datos / estructuras

XmlData

Objeto que contiene la fecha extraída del XML

CAMPO	TIPO	DESCRIPCIÓN
xmlDate	Objeto Date con la fecha encontrada o valor vacío si no se encontró	

Efectos colaterales

- **Type:** record; **Details:** submitFields para actualizar el estado del registro inbound
- **Type:** search; **Details:** Búsqueda de configuración inbound y de registros de campos extra
- **Type:** record; **Details:** load o create y save del registro customrecord_lmry_inbound_fields
- **Type:** log; **Details:** Registro de errores en caso de excepciones

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
getSetupData	Si vendorId no está definido se registra un error y la función devuelve false	
createOrUpdateInboundExtraFields	Captura cualquier excepción y la registra mediante nLog.error	

Detalles de implementación

En `afterSubmit` se obtiene el registro `inbound`, se extrae la fecha del XML mediante `getXmlData`, se llama a `createOrUpdateInboundExtraFields` para crear o actualizar el registro de campos extra, y si el evento es de creación y el proveedor está configurado como genérico, se actualiza el campo de estado a 18 mediante `record.submitFields`.

Generado automáticamente 2026-01-23 17:50 UTC

SUITELET · INBOUND MÉXICO - VISUALIZADOR DE PDF

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a usuarios autorizados visualizar archivos PDF o texto almacenados en NetSuite mediante un Suitelet.

Flujo de Ejecución

GET

Parámetros de Entrada

Lógica POST

Carga el archivo indicado, determina su tipo y codificación y lo entrega al cliente mediante redirección o escritura directa

Operaciones

TIPO	TARGET	PROPÓSITO
load	file	cargar el contenido del archivo a partir del ID proporcionado
redirect	url	redirigir al URL del archivo cuando es texto con codificación ISO-8859-1
write	response	escribir el contenido del archivo en la respuesta HTTP con el encabezado Content-Type apropiado

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: Entrega PDFs o textos según tipo y codificación; solo usuarios autorizados pueden acceder

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	incluido por convención aunque no se utiliza en este Suitelet
N/search	no se utiliza directamente en la lógica actual
N/runtime	obtener información del usuario y verificar la característica SUBSIDIARIES
N/record	no se utiliza directamente en la lógica actual
N/redirect	realizar redirección a la URL del archivo cuando corresponde
N/file	cargar el archivo desde el File Cabinet

Resumen Técnico

Carga el archivo por ID, verifica su tipo y codificación y lo entrega como PDF o texto, usando redirección cuando es texto con codificación ISO-8859-1.

MAP/REDUCE · INBOUND SAT DOCUMENTS PROCESSING

Pipeline de procesamiento por lotes (SuiteScript Map/Reduce).

Procesos masivos, búsquedas y operaciones sobre registros.

Metadatos

Tipo: MapReduce

SuiteScript: 2.1

Inbound SAT Documents Processing

Resumen Ejecutivo

{'business_summary': Markup('Importa documentos fiscales SAT recibidos de un servicio externo, crea registros inbound y adjunta PDFs, evitando duplicados.'), 'technical_summary': Markup('Obtiene documentos vía HTTP POST, filtra por folios existentes, asigna vendor o datos genéricos, valida XML, crea registros customrecord_psg_ei_inbound_edoc y archivos PDF, y registra auditoría.')}

getInputData

Llama al servicio externo mediante HTTP POST con el processID, recibe un JSON con filesArray, elimina documentos cuyo folio ya existe y agrega datos de configuración.

map

Parsea el JSON, busca el vendor por RFC del emisor, crea un registro inbound mediante createInboundEdoc, registra logs y maneja errores.

Campos utilizados

- Folio Fiscal
- RFC Emisor
- genericVendor
- genericPackage
- subsiSetup
- edocType
- datos.xml
- datos.pdf

Operaciones en registros (map)

OPERACIÓN	RECORD	CAMPOS
create	customrecord_psg_ei_inbound_edoc	name, custrecord_psg_ei_inbound_vendor, custrecord_psg_ei_inbound_status, custrecord_psg_ei_inbound_source, custrecord_psg_ei_inbound_tran_type, custrecord_psg_ei_inbound_content, custrecord_psg_ei_attach_pdf
submitFields	vendor	custentity_psg_ei_entity_edoc_standard
create	folder	name, parent
create	file	PDF file (name, contents, folder)

summarize

Registra auditoría con total de registros procesados y número de errores.

Búsquedas utilizadas

ID	FILTROS	COLUMNAS
buscarVendorPorRFC	[Markup('vatregnumber'), Markup('is'), Markup('RFC')]	internalid
getTemplate	[Markup('custrecord_psg_ei_template_edoc_standard'), Markup('anyof'), Markup('edocPackage')], [Markup('custrecord_psg_ei_template_trans_type'), Markup('is'), Markup('tranType')], [Markup('custrecord_psg_ei_template_subsiary'), Markup('is'), Markup('subsiSetup')], [Markup('custrecord_edoc_template_xsd'), Markup('noneof'), [Markup('@NONE@')]]	custrecord_edoc_template_xsd, internalid
existingDocuments	OR combinaciones de name contains cada folio	name
getSetupData	[Markup('custrecord_lmry_mx_setup_sat_user'), Markup('is'), Markup('receptorRfc')]	custrecord_lmry_mx_setup_vendor, custrecord_lmry_mx_setup_package, custrecord_lmry_mx_setup_subsiary
findFolderByName	[Markup('name'), Markup('is'), Markup('folderName')]	internalid
getSubFolderId	[Markup('name'), Markup('is'), Markup('folderName')], AND, [Markup('parent'), Markup('anyof'), Markup('parentId')]	internalid
lookupVendorPackage		custentity_psg_ei_entity_edoc_standard

Operaciones con registros

OPERACIÓN	RECORD	CAMPOS
create	customrecord_psg_ei_inbound_edoc	
submitFields	vendor	
create	folder	
create	file	

Side effects

- Creación de registros inbound para cada documento SAT
- Actualización del campo custentity_psg_ei_entity_edoc_standard en vendor cuando falta
- Creación de carpetas y archivos PDF en el File Cabinet
- Llamada HTTP POST al servicio externo
- Generación de logs de auditoría y errores

RESTLET • RESTLET QUE RECIBE UN PROCESSID Y LANZA UN MAP/REDUCE PARA INICIAR LA DESCARGA DE DATOS SAT.

SuiteScript 2.1

Documentación técnica del servicio RESTlet.

Metadatos

Tipo: RESTlet

SuiteScript: 2.1

RESTlet que recibe un processID y lanza un Map/Reduce para iniciar la descarga de datos SAT.

Resumen Ejecutivo

El RESTlet expone únicamente el método POST; recibe un JSON con processID, lanza un script Map/Reduce (customscript_lmry_mx_descarga_sat_mprd) pasando ese ID como parámetro, y devuelve el estado de la ejecución con el taskId generado.

Endpoints

GET

Definición inválida o no estructurada.

POST

Recibe un cuerpo JSON con la propiedad processID, crea y envía un task de tipo Map/Reduce que ejecuta el script de descarga SAT.

Body

NOMBRE	DESCRIPCIÓN	TIPO
processID		

Lógica

Construye un objeto con processID, lo serializa a JSON, crea un task Map/Reduce asignando scriptId y deploymentId, pasa el JSON como parámetro custscript_lmry_sat_processid, envía el task y devuelve un objeto con estado, mensaje, taskId y el JSON enviado.

Salida

Objeto JSON con status, message, taskId y jobRecord (cadena JSON del proceso).

PUT

Definición inválida o no estructurada.

DELETE

Definición inválida o no estructurada.

Errores

UBICACIÓN	DESCRIPCIÓN	EXCEPCIÓN
post	Error capturado en bloque try/catch y relanzado.	error object capturado (e)

Side effects globales

- log.debug al iniciar el Map/Reduce con el ID de task
- log.error en caso de excepción
- creación y envío de un task Map/Reduce mediante N/task

SCHEDULED SCRIPT · DESCARGA DE DOCUMENTOS SAT MEDIANTE SERVICIO EXTERNO

SuiteScript 2.1

Procesos periódicos y lotes programados.

Metadatos

Tipo: ScheduledScript

SuiteScript: 2.1

Descarga de documentos SAT mediante servicio externo

Resumen Ejecutivo

{'business_summary': Markup('Descarga automáticamente los documentos del SAT para la subsidiaria configurada mediante un servicio externo.'), 'technical_summary': Markup('El script obtiene la fecha, RFC y contraseña de configuración, envía un POST JSON al endpoint y gestiona la respuesta, con manejo de errores y registro de logs.')}

Entrada

Obtiene el parámetro custscript_lmry_mx_sat_subsidiary_rfc y busca credenciales SAT en customrecord_lmry_mx_setup_inbound

Parámetros

PARÁMETRO	DESCRIPCIÓN
custscript_lmry_mx_sat_subsidiary_rfc	

Filtros

CAMPO	OPERADOR	VALOR
custrecord_lmry_mx_setup_sat_user	is	subsiParam

Columnas

NOMBRE	ETIQUETA
custrecord_lmry_mx_setup_sat_user	
custrecord_lmry_mx_setup_sat_pass	

Lógica

Obtiene el parámetro de subsidiaria, calcula la fecha actual, recupera RFC y password mediante búsqueda, arma el cuerpo JSON, envía petición POST al endpoint y registra la respuesta; captura y registra cualquier excepción.

Condiciones

- Si response.code es 200 registra debug, de lo contrario registra error

Búsquedas

TIPO	FILTROS	COLUMNAS
search	{'field': Markup('custrecord_lmry_mx_setup_sat_user'), 'operator': Markup('is'), 'value': Markup('subsiParam')}	custrecord_lmry_mx_setup_sat_user, custrecord_lmry_mx_setup_sat_pass
search	{'field': Markup('custentity_mx_rfc'), 'operator': Markup('is'), 'value': Markup('rfc')}	internalid

Governance

Errores

ESCENARIO	MANEJO

Side effects

- Registro de logs de depuración y errores
- Llamada HTTP POST al endpoint externo para iniciar descarga de documentos SAT

Generado automáticamente 2026-01-23 17:48 UTC

SUITELET · VISUALIZADOR DE XML INBOUND

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a usuarios visualizar de forma legible el XML almacenado en registros inbound de México.

Flujo de Ejecución

GET

POST

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_id_message	Message	INLINEHTML	Muestra mensaje de error en HTML cuando la solicitud falla

Lógica POST

Procesa la solicitud GET, recupera el XML del registro inbound, lo formatea y lo devuelve; en caso de error genera un formulario con mensaje informativo

Búsquedas

Obtener el contenido XML almacenado en el registro inbound

Filtros: Tipo de registro customrecord_psg_ei_inbound_edoc, id inboundRecordId, columna custrecord_psg_ei_inbound_content

Operaciones

TIPO	TARGET	PROPÓSITO
lookupFields	customrecord_psg_ei_inbound_edoc	Recuperar el campo custrecord_psg_ei_inbound_content
parse	N/xml.Parser	Convertir la cadena XML en un documento XML
format	N/xml.Parser	Convertir el documento XML a cadena con indentación
customFunction	formatXMLString	Aplicar formato adicional al XML para mejor legibilidad
writeResponse	context.response	Enviar el XML formateado como texto plano al cliente
drawFormError	serverWidget	Mostrar formulario con mensaje de error cuando ocurre una excepción

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Devuelve XML formateado; en caso de error muestra un formulario con mensaje de acceso no permitido

Dependencias

NOMBRE	PROPÓSITO
N/search	Realizar lookupFields para obtener el contenido XML del registro inbound
N/encode	Módulo incluido pero no utilizado en la lógica actual
N/ui/serverWidget	Crear formulario de error con campo INLINEHTML
N/url	Módulo incluido pero no utilizado en la lógica actual
N/xml	Parsear y serializar documentos XML

Resumen Técnico

Recupera el contenido mediante lookupFields, lo parsea con N/xml, lo formatea y lo envía como texto plano; en caso de error muestra un formulario con mensaje informativo.

SUITELET · DETALLE INBOUND MX SUITELET

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite visualizar el detalle de inbound MX mediante un formulario con sublistas de POS y artículos.

Flujo de Ejecución

GET

Lógica POST

La lógica de creación y llenado del formulario se delega al handler; en caso de error se genera un formulario estático con mensaje

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Muestra formulario o mensaje de error

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Crear formularios, campos y sublistas
N/runtime	Obtener ID del script y despliegue actual
N/redirect	Importado pero no utilizado en este archivo
../Handlers/ LMRY_MX_InboundDetail_STLT_HNDL	Contiene la lógica para crear el formulario y sus sublistas

Resumen Técnico

Construye el formulario usando una clase handler, asigna script cliente y muestra la página; en caso de error muestra un mensaje estático.

SUITELET · DETALLE DE REGISTRO INBOUND DE DOCUMENTOS ELECTRÓNICOS

SuiteScript 2.x

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Este Suitelet muestra de forma detallada los registros inbound de documentos electrónicos, facilitando la auditoría y el seguimiento de facturas asociadas.

Flujo de Ejecución

GET

POST

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
id_internalid	Count	TEXT	solo lectura, muestra el número de fila
id_tran_id	Internal ID	TEXT	solo lectura, muestra el ID interno del registro inbound
id_name	Name File	TEXT	solo lectura, muestra el nombre del archivo
id_vendor	Vendor	TEXT	solo lectura, muestra el nombre del proveedor
id_subsidiary	Subsidiary	SELECT	inline, muestra la subsidiaria cuando la característica está habilitada
id_doc_status	E-Document Status	TEXT	solo lectura, muestra el estado del documento
id_tran_ur	Transaction	TEXT	solo lectura, muestra un enlace a la factura cuando está disponible
audit_trial	Audit Trial	TEXT	solo lectura, muestra los detalles del último registro de auditoría

Lógica POST

En modo GET, el Suitelet recupera los IDs de inbound desde el registro de log, los procesa en bloques, busca los registros de detalle y auditoría, y llena la sublista con la información correspondiente.

Búsquedas

Obtener datos del registro de log inbound

Filtros:

Retorna: Subsidiaria, JSON de transacciones y JSON de facturas

Buscar registros de detalle inbound (customrecord_psg_ei_inbound_edoc)

Filtros:

Retorna: internalid, name, referencia, estado, proveedor

Obtener el último registro de auditoría por inbound edoc (customrecord_psg_ei_audit_trail)

Filtros:

Retorna: detalle de auditoría del registro más reciente

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Maneja grandes volúmenes de IDs mediante segmentación de 900 registros; incluye enlaces a facturas y muestra auditoría.

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener parámetros del script y verificar la característica de subsidiarias
N/log	Registrar información de depuración y errores
N/search	Ejecutar búsquedas y lookupFields sobre registros personalizados
N/record	No se utiliza directamente para crear o actualizar registros en este Suitelet
N/ui/serverWidget	Construir el formulario y la sublista estática
N/url	Generar enlaces a facturas de proveedor

Resumen Técnico

Construye un formulario con sublista estática usando serverWidget, recupera datos mediante lookupFields y búsquedas paginadas, y muestra enlaces a facturas mediante url.resolveRecord.

Generado automáticamente 2026-01-23 17:49 UTC

CLIENTSCRIPT · FUNCIÓN DE CANCELACIÓN Y GUARDADO SIN LÓGICA

SuiteScript 2.1

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Proporciona una función de cancelación que devuelve al usuario a la página anterior sin aplicar lógica adicional en el formulario.

Lifecycle Hooks

pageInit

Se ejecuta al cargar la página pero no realiza ninguna acción

Lógica: No contiene lógica alguna

fieldChanged

Se invoca cuando un campo cambia, pero no ejecuta código alguno

Lógica: No contiene lógica alguna

validateField

Se llama para validar un campo, pero no devuelve valor ni realiza comprobaciones

Lógica: No contiene lógica alguna y no retorna un booleano

saveRecord

Se ejecuta al intentar guardar el registro; no contiene lógica ni devuelve valor

Lógica: No contiene lógica alguna y no retorna un booleano

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

No

MUESTRA ALERTAS

No

Notas: Exporta una función de cancelación que retrocede en el historial del navegador; los métodos de ciclo de vida están vacíos y no realizan validaciones ni modificaciones.

Resumen Técnico

Implementa métodos de ciclo de vida vacíos y una función de cancelación que utiliza `window.history.back`; no valida datos ni bloquea el guardado.

SUITELET · REGISTRO DE INBOUND MÉXICO

SuiteScript 2.0

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Este Suitelet permite a los usuarios visualizar el historial de procesos inbound en México, mostrando estado, fechas y detalle de cada ejecución.

Flujo de Ejecución

GET

POST

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_hidden	HIDDEN	TEXT	valor predeterminado "Hidden" y tipo de visualización HIDDEN
custpage_id_message	MESSAGE	INLINEHTML	mensaje informativo con texto de importancia y guía de uso
id_internalid	Internal ID	TEXT	campo de texto que muestra el ID interno del registro de log
id_subsidiary	Subsidiary	SELECT	lista desplegable de subsidiarias, mostrada como INLINE
id_datefrom	Date From	TEXT	campo de texto que muestra la fecha de inicio del rango procesado
id_dateto	Date To	TEXT	campo de texto que muestra la fecha de fin del rango procesado
id_employee	Employee	TEXT	campo de texto que muestra el usuario que ejecutó el proceso
id_total	# Transactions	TEXT	campo de texto que muestra la cantidad de transacciones procesadas
id_transactions	Transactions	TEXTAREA	campo de texto multilínea que muestra detalle o enlace a detalle según el estado
id_state	State	TEXTAREA	campo de texto multilínea que muestra el estado del proceso (Pending, Error, Complete)

Lógica POST

Construye el formulario, ejecuta una búsqueda de logs de inbound y rellena la sublista con la información obtenida, generando enlaces a detalle cuando el proceso está completo.

Búsquedas

Obtener los registros de log de inbound que tengan estado definido

Filtros: Filtro: custrecord_lmry_inbound_state_po no vacío; columnas: internalid, usuario, subsidiaria, transacciones, estado, fecha de creación, conteo, fecha desde, fecha hasta

Operaciones

TIPO	TARGET	PROPÓSITO
search	customrecord_lmry_ei_inbound_po	Recuperar datos para poblar la sublista
url.resolveScript	customscript_lmry_mx_inb_log_det_stlt	Construir la URL del detalle del log cuando el estado es completado

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Solo muestra información; no modifica registros ni envía correos

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener parámetros del script y verificar si la característica de subsidiarias está activa
N/log	Registrar información de depuración y errores
N/search	Ejecutar la búsqueda de registros de log de inbound
N/record	No se utiliza directamente en este Suitelet, pero está cargado por consistencia
N/ui/serverWidget	Crear el formulario, campos y sublista que se muestra al usuario
N/url	Construir la URL al script de detalle del log

Resumen Técnico

Genera una página con un formulario estático usando serverWidget, consulta los logs mediante una búsqueda y muestra los resultados en una sublista, incluyendo enlaces a detalle mediante url.resolveScript.

MAP/REDUCE · LMRY_MX_INBOUND_MPRD

Pipeline de procesamiento por lotes (SuiteScript Map/Reduce).

Procesos masivos, búsquedas y operaciones sobre registros.

Metadatos

Tipo: MapReduce

SuiteScript: 2.0

LMRY_MX_Inbound_MPRD

Resumen Ejecutivo

{'business_summary': Markup('Convierte documentos electrónicos entrantes en facturas de proveedor, enlazándolas con órdenes de compra cuando corresponde y registra el proceso para auditoría.'), 'technical_summary': Markup('Lee IDs de documentos inbound, procesa cada uno en la fase map usando un plugin y plantillas de render, crea o transforma facturas, actualiza varios registros personalizados y genera un resumen JSON en la fase summarize.')}

getInputData

Devuelve un arreglo con los IDs de los documentos inbound (customrecord_psg_ei_inbound_edoc) asociados al registro customrecord_lmry_ei_inbound_po indicado en el parámetro custscript_lmry_inb_log_id.

Filtros

CAMPO	OPERADOR	VALOR
	anyof	

Columnas

NOMBRE	ETIQUETA
custrecord_lmry_inbound_transactions_po	

map

Para cada registro inbound se obtienen datos del vendor y del contenido XML, se asigna o actualiza el código de vendor en los ítems, se invoca el plugin de conversión, se parsea el documento con la plantilla correspondiente y se crea o transforma una factura (Vendor Bill) a partir del contenido. Se actualiza el estado del registro inbound y se adjunta el PDF generado a la factura. Finalmente se escribe en la salida la clave (ID del inbound) y el resultado de la conversión (objeto con billId y flag success).

Campos utilizados

- custrecord_psg_ei_inbound_content
- custrecord_psg_ei_inbound_vendor
- custrecord_psg_ei_inbound_template
- custrecord_psg_ei_inbound_status
- custrecord_psg_ei_inbound_parsed_content
- custrecord_psg_ei_inbound_refnum
- custrecord_psg_ei_inbound_po
- custrecord_psg_ei_inbound_attach_pdf
- custrecord_lmry_inbound_state
- custrecord_lmry_inbound_bill_po
- custrecord_lmry_inbound_state_po

Operaciones en registros (map)

OPERACIÓN	RECORD	CAMPOS
load	customrecord_psg_ei_inbound_edoc	
submitFields	customrecord_psg_ei_inbound_edoc	custrecord_psg_ei_inbound_status
load	customrecord_psg_ei_inbound_edoc	
load	vendor	
transform	purchaseOrder	
create	vendorbill	
setValue	vendorbill	entity, custbody_psg_ei_inbound_edocument
save	vendorbill	
attach	file	
load	customrecord_lmry_mx_inb_bill	
create	customrecord_lmry_mx_inb_bill	
setValue	customrecord_lmry_mx_inb_bill	custrecord_lmry_mx_inb_edoc_id, custrecord_lmry_mx_inb_bill_id
save	customrecord_lmry_mx_inb_bill	

reduce

summarize

Acciones finales

- Construye un objeto JSON con los resultados de todas las conversiones.
- Carga el registro customrecord_lmry_ei_inbound_po indicado en el parámetro custscript_lmry_inb_log_id.
- Actualiza los campos custrecord_lmry_inbound_bill_po (JSON con resultados) y custrecord_lmry_inbound_state_po (valor '2').
- Guarda el registro.

Búsquedas utilizadas

ID	FILTROS	COLUMNAS
search_getInboundID	{'name': Markup('internalid'), 'operator': Markup('anyof'), 'values': Markup('paramIdDocuments')}	custrecord_lmry_inbound_tran
search_getSetup	{'name': Markup('custrecord_lmry_mx_setup_subsidiary'), 'operator': Markup('equalto'), 'values': Markup('subsidiaryParam')}, {'name': Markup('isinactive'), 'operator': Markup('is'), 'values': Markup('F')}	custrecord_lmry_mx_setup_ite
search_getCreatedFromTransaction	{'name': Markup('tranid'), 'operator': Markup('is'), 'values': Markup('createdFromNumber')}	internalid, entity
search_getBillsUsingReferenceNumber	{'name': Markup('entity'), 'operator': Markup('is'), 'values': Markup('vendorId')}, {'name': Markup('tranid'), 'operator': Markup('is'), 'values': Markup('referenceNumber')}, {'name': Markup('mainline'), 'operator': Markup('is'), 'values': Markup('T')}	tranid
search_getItemDetailsByVendorCodes	{'name': Markup('vendorcode'), 'operator': Markup('anyof'), 'values': Markup('arrayVendorCodes')}, {'name': Markup('othervendor'), 'operator': Markup('is'), 'values': Markup('entityId')}	vendorcode, othervendor
search_getItemRecordType	{'name': Markup('internalid'), 'operator': Markup('is'), 'values': Markup('itemId')}	itemid
search_createOrUpdateConvertedRecord	{'name': Markup('custrecord_lmry_mx_inb_edoc_id'), 'operator': Markup('anyof'), 'values': Markup('inboundRecordId')}	internalid

Operaciones con registros

OPERACIÓN	RECORD	CAMPOS
customrecord_psg_ei_inbound_edoc		
customrecord_lmry_ei_inbound_po		
vendorbill		
customrecord_lmry_mx_inb_bill		
item		
purchaseorder		

Side effects

- Creación o actualización de facturas de proveedor (Vendor Bill).
- Adjunto de archivo PDF al Vendor Bill.
- Actualización del estado del registro inbound (customrecord_psg_ei_inbound_edoc).
- Creación o actualización del registro customrecord_lmry_mx_inb_bill que enlaza el inbound con la factura.
- Actualización del registro customrecord_lmry_ei_inbound_po con el JSON de resultados y cambio de estado.
- Registro de auditoría mediante auditManager.logConverted y auditManager.logConversionFailed.

Generado automáticamente 2026-01-23 17:49 UTC

SUITELET · MX INBOUND SUITELET

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios filtrar y aplicar acciones sobre registros inbound, y ejecutar procesos de integración mediante Map/Reduce.

Flujo de Ejecución

GET

POST

Procesa la acción del formulario y redirige o ejecuta un Map/Reduce

Flujo: Lee el parámetro `custpage_status`; si está vacío redirige al mismo Suitelet con los parámetros de filtro generados por el handler; si está presente toma `custpage_log_id` y `custpage_subsidiary`, ejecuta el proceso Map/Reduce correspondiente y luego redirige a la suitelet de registro de logs

Genera: Redirección a otro Suitelet o ejecución de Map/Reduce seguida de redirección a la suitelet de logs

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
<code>custpage_id_message</code>	Message	INLINEHTML	campo de solo lectura que muestra un mensaje de error cuando ocurre una excepción

Lógica POST

El Suitelet delega la mayor parte de la lógica a la librería LibraryHandler; en GET construye y muestra formularios y sublistas, en POST controla redirecciones y lanza procesos Map/Reduce

Operaciones

TIPO	TARGET	PROPÓSITO
redirect	Suitelet (mismo script)	Volver a cargar el formulario con los parámetros de filtro después de una acción POST sin status
executeMapReduce	Map/Reduce script definido en LibraryHandler	Procesar los registros de log especificados para la subsidiaria seleccionada
redirect	Suitelet de logs	Mostrar el resultado del proceso Map/Reduce al usuario

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: El Suitelet solo muestra formularios y delega la creación/actualización de registros a procesos externos (Map/Reduce).

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construir y renderizar formularios y campos UI
N/runtime	Obtener información del script en ejecución (ID y deployment)
N/redirect	Redirigir al usuario a otro Suitelet o a la misma suitelet con parámetros
../Handlers/ LMRY_MX_Inbound_STLT_HNDL	Contiene la lógica de negocio para crear formularios, cargar sublistas y ejecutar procesos Map/Reduce

Resumen Técnico

Construye formularios con N/ui/serverWidget, gestiona la lógica mediante una librería externa y controla la navegación con N/redirect, delegando el procesamiento intensivo a un script Map/Reduce.

Generado automáticamente 2026-01-23 17:49 UTC

INTERNALMODULE · PLUGIN CAPTURA DE E-DOCUMENTOS ENTRANTES

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: service

Plugin captura de e-documentos entrantes

Dependencias: SUITESCRIPT 1.0 NLAPI API

Resumen Ejecutivo

{'business_summary': Markup('Este plugin automatiza la captura y registro de e-documents recibidos por email, reduciendo la intervención manual y notificando rápidamente cualquier error a los administradores.'), 'technical_summary': Markup('Procesa el email, valida adjuntos XML/PDF, identifica proveedor o cliente, crea un registro personalizado, adjunta el PDF y gestiona errores mediante logs y notificaciones por email.')}

Dependencias

NOMBRE	PROPÓSITO
nlapiGetContext	Obtener preferencias de contexto, como el idioma
nlapiCreateRecord	Crear registros personalizados de e-documentos entrantes
nlapiSubmitRecord	Guardar el registro creado en NetSuite
nlapiSearchRecord	Buscar empleados, proveedores, clientes, carpetas y plantillas
nlapiLookupField	Obtener valores de campos sin cargar el registro completo
nlapiCreateFile	Crear archivos PDF en el File Cabinet
nlapiSubmitFile	Guardar el archivo PDF creado
nlapiSendEmail	Notificar errores a administradores o destinatario configurado
nlapiLogExecution	Registrar información de depuración y errores
nlapiResolveURL	Construir enlace al registro de e-documento creado
nlapiSubmitField	Actualizar campo de adjunto PDF en el registro
nlapiCreateError	Generar errores personalizados para el flujo de captura

API pública

process

Punto de entrada del plugin; procesa el email, valida adjuntos, identifica entidad, crea registro de e-documento, adjunta PDF y maneja errores

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
email	Objeto de email provisto por NetSuite	

Efectos colaterales

- crea registro customrecord_psg_ei_inbound_edoc
- crea archivo PDF en el File Cabinet
- envía email de notificación en caso de error

Funciones internas

NOMBRE	ROL	USADO POR
retrieveFile	helper	process
getActiveAdmins	helper	sendEmail
sendEmail	helper	process
identifyVendor	validator	process
identifyCustomer	validator	process
updateXMLContent	transformer	process
createPDFFile	helper	process
getPDFFolder	helper	createPDFFile
getFolderIdByName	helper	getPDFFolder
createPDFFolder	helper	getPDFFolder
getParentSubsidiaryId	helper	process

Modelos de datos / estructuras

InboundEdocRecord

Registro personalizado que almacena el e-documento entrante y sus metadatos

CAMPO	TIPO	DESCRIPCIÓN
custrecord_psg_ei_inbound_source	origen del e-documento (email)	
custrecord_psg_ei_inbound_content	XML del e-documento sin namespace	
custrecord_psg_ei_inbound_transtype	Tipo de transacción (Bill, BillCredit, SalesOrder)	
custrecord_psg_ei_inbound_vendor	Id interno del proveedor asociado	
custrecord_psg_ei_inbound_customer	Id interno del cliente asociado	
custrecord_psg_ei_inbound_template	Id de la plantilla E-Document seleccionada	
custrecord_psg_ei_attach_pdf	Id del archivo PDF adjunto	

Efectos colaterales

- **Type:** record_creation; **Details:** Creación y guardado de registros customrecord_psg_ei_inbound_edoc y carpetas de archivo
- **Type:** file_creation; **Details:** Generación y almacenamiento de archivos PDF en el File Cabinet
- **Type:** email_sending; **Details:** Envío de notificaciones de error a administradores o destinatario configurado
- **Type:** logging; **Details:** Uso de nlapiLogExecution para depuración y registro de errores
- **Type:** search; **Details:** Ejecución de búsquedas de empleados, proveedores, clientes, carpetas y plantillas
- **Type:** field_lookup; **Details:** Consultas de campos mediante nlapiLookupField

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
retrieveFile	Validación de número de archivos XML o PDF adjuntos	EMAIL_CAPTURE_ERROR_CODE cuando la cantidad no cumple los requisitos
identifyVendor	Búsqueda de proveedor por RFC; lanza error si no se encuentra ninguno	EMAIL_CAPTURE_ERROR_CODE cuando no hay proveedor para el RFC
identifyCustomer	Búsqueda de cliente por dominio o email; lanza error si no se identifica un único cliente	EMAIL_CAPTURE_ERROR_CODE cuando no se puede identificar al cliente
process	Bloque try/catch captura cualquier excepción lanzada durante el flujo	Se reenvía el error mediante sendEmail después de registrar

Detalles de implementación

El script recibe un email, extrae los archivos adjuntos y valida que exista un único XML y, opcionalmente, un único PDF. El XML se limpia de namespaces y se guarda en un registro customrecord_psg_ei_inbound_edoc junto con el tipo de transacción inferido del asunto. Según el tipo, se identifica al proveedor (por RFC) o al cliente (por dominio o registro de envío) y se enlaza al registro. Se busca la plantilla E-Document correspondiente al estándar del proveedor/cliente y se asigna. Si hay PDF, se crea un archivo en la carpeta "Inbound PDF E-Documents" y se enlaza al registro. Finalmente se envía el registro; en caso de cualquier error se captura, se registra y se envía un email de notificación a los administradores o al destinatario configurado.

PLUG-IN IMPLEMENTATION • INJECTS CUSTOM DATA SOURCE FOR MEXICAN ELECTRONIC INVOICE RENDERING

SuiteScript 2.0

Documentación técnica del plug-in.

Metadatos

Tipo: PlugInImplementation

SuiteScript: 2.0

Injects custom data source for Mexican electronic invoice rendering

Resumen Ejecutivo

Provides a custom data source during the rendering of inbound Mexican electronic invoices, extracting XML data, validating SAT, and preparing tax and withholding information for the template.

Deployment / Enablement

Se habilita al asociar este archivo a un Plug-in Type definido en NetSuite y activarlo en la configuración del tipo de plug-in correspondiente.

Entry points

NOMBRE	DESCRIPCIÓN	INPUTS	OUTPUTS
inject	Genera y devuelve un data source personalizado con la información extraída y transformada del documento XML de la factura electrónica.	Objeto params que contiene inboundEDocRec (registro de documento entrante), templateRec (registro de plantilla) y entityRec (registro de entidad/vendedor).	Objeto con la propiedad customDataSources que incluye un DataSource de tipo OBJECT, alias "item_type" y los datos de inyección preparados.

Búsquedas

PROPÓSITO	FILTROS	RETORNO
Obtener el método de pago mexicano correspondiente al código de forma de pago.	custrecord_lmry_clave = formaPago y custrecord_lmry_country_pm = 157 (México).	Nombre del método de pago.
Recuperar la configuración de la subsidiaria para la validación de inbound invoices.	custrecord_lmry_mx_setup_subsidiary = id de la subsidiaria.	Departamento, clase, ubicación, validación SAT, códigos de impuesto y tabla de retenciones.
Buscar expense reports que ya tengan registrado el UUID del documento.	custcol_lmry_inbound_folio no vacío y coincide con el UUID; opcionalmente filtra por subsidiaria.	InternalId del expense report.
Obtener el número de orden de compra que coincide con el total y el vendor del documento inbound.	mainline = T, totalamount = totalValue, vendor.internalid = inboundEDocVendorID.	TranId de la orden de compra.
Determinar la configuración de tax point (accrual o payment) para la subsidiaria.	custrecord_4601_wts_nexus = id del nexus mexicano.	Valor del campo custrecord_4601_wts_purctaxpoint (onaccrual / onpayment).
Obtener la información del código de impuesto de retención (WHT) a partir de la tabla de retenciones.	internalid = withholdingTaxCode (valor de la tabla de retenciones).	Tasa y tipo de impuesto de retención.
Obtener el ítem de compra asociado al tipo de retención.	internalid = witaxtypeId (tipo de retención obtenido previamente).	custrecord_4601_wtt_purcitem (id del ítem).
Obtener el nexus mexicano para determinar la configuración fiscal.	country = MX.	InternalId del nexus.
Listar las monedas filtradas (MXN y USD) para incluirlas en el data source.	symbol contiene MXN o USD.	Mapa de símbolo a nombre de moneda.

Dependencias

NOMBRE	PROPÓSITO
N/render	Crear el DataSource que será devuelto al motor de renderizado.
N/log	Registrar información de depuración y errores.
N/record	Acceder a tipos de registro (p.ej., PURCHASE_ORDER).
N/search	Ejecutar búsquedas de registros y lookup de campos.
N/https	Invocar el servicio SOAP del SAT para validar el CFDI.
N/runtime	Obtener parámetros del script y datos del usuario actual.
N/xml	Parsear el XML del CFDI y extraer atributos.
/SuiteBundles/Bundle 436209/com.netsuite.electronicinvoicing/src/app/audit/app_audit_manager	Registrar auditoría de conversiones fallidas.
N/error	Crear y lanzar errores personalizados.

Side effects

- Registra auditoría de fallas mediante auditManager.
- Lanza errores que abortan la generación del data source.
- Escribe logs de depuración y errores.

Errores

UBICACIÓN	DESCRIPCIÓN	EXCEPCIÓN

PLUG-IN IMPLEMENTATION · INECTA UNA FUENTE DE DATOS PERSONALIZADA EN EL PROCESO INBOUND

SuiteScript 2.0

Documentación técnica del plug-in.

Metadatos

Tipo: PlugInImplementation

SuiteScript: 2.0

Inyecta una fuente de datos personalizada en el proceso inbound

Resumen Ejecutivo

El plug-in agrega una fuente de datos personalizada al proceso de entrada, permitiendo que plantillas de renderizado consuman datos definidos por el desarrollador.

Deployment / Enablement

Se habilita al asignar este plug-in a la configuración de tipo de plug-in correspondiente en NetSuite.

Entry points

NOMBRE	DESCRIPCIÓN	INPUTS	OUTPUTS
inject	Inyecta una fuente de datos personalizada en el proceso inbound.	Objeto params que contiene inboundEDocRec, templateRec y entityRec.	Objeto con la propiedad customDataSources que incluye alias, formato y datos del DataSource.

Dependencias

NOMBRE	PROPÓSITO
N/render	Proporciona la constante DataSource y permite definir la estructura de la fuente de datos.

INTERNALMODULE · ROUTER DE USER EVENT POR TIPO DE TRANSACCIÓN

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: utils

Router de User Event por tipo de transacción

Dependencias: N/LOG

Resumen Ejecutivo

{'business_summary': Markup('Este módulo permite a otros scripts obtener dinámicamente el handler de User Event correspondiente al tipo de documento, facilitando la reutilización y el mantenimiento del código.'), 'technical_summary': Markup('Expone la función getUserEventRouter que mapea tipos de transacción a rutas de módulos y los carga mediante require, devolviendo el módulo o una cadena vacía.')}

Dependencias

NOMBRE	PROPÓSITO
N/log	Registro de mensajes de depuración y errores en SuiteScript

API pública

getUserEventRouter

Obtiene el manejador de User Event correspondiente al tipo de transacción solicitado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
type	Tipo de transacción (por ejemplo vendorbill)	

Efectos colaterales

- Registra error en log si ocurre una excepción
- Carga dinámicamente el módulo mediante require

Retorna

Módulo de manejador cargado o cadena vacía si no existe

Efectos colaterales

- **Type:** log; **Details:** Registra mensaje de error cuando se captura una excepción en getUserEventRouter
- **Type:** module load; **Details:** Carga dinámicamente el handler de User Event mediante la función require

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
getUserEventRouter	Captura cualquier excepción durante la búsqueda y carga del handler	No lanza, solo registra el error en el log

Detalles de implementación

Recibe el tipo de transacción, busca la ruta del handler en un mapa interno y, si la encuentra, utiliza require para cargar el módulo de forma asíncrona; asigna el módulo cargado a la variable result y lo devuelve, devolviendo una cadena vacía cuando no hay coincidencia o en caso de error.

CLIENTSCRIPT · CARGA DE CONFIGURACIÓN INBOUND SEGÚN SUBSIDIARIA

SuiteScript 2.1 · Módulos: 3

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Permite que los usuarios carguen automáticamente la configuración inbound de México al seleccionar una subsidiaria, evitando ingreso manual y errores.

Lifecycle Hooks

pageInit

Inicializa el formulario cargando valores de configuración inbound para la subsidiaria predeterminada cuando la característica de subsidiarias está desactivada.

Lógica: Comprueba si la característica de subsidiarias está inactiva; si es así, fija la subsidiaria en 1, ejecuta una búsqueda del registro `customrecord_lmry_mx_setup_inbound`, extrae los campos configurados y los asigna a los campos del formulario, parsea JSON para departamento, clasificación y ubicación, determina si se valida gasto, oculta la contraseña con asteriscos, y rellena la sublista de retenciones mediante `poblarSublistaWithholding`; si no hay resultados, limpia los campos.

Salida: Establece valores en `custpage_subsidiary`, `custpage_taxcode0`, `custpage_taxcode8`, `custpage_taxcode16`, `custpage_department`, `custpage_class`, `custpage_location`, `custpage_item`, `custpage_vendor`, `custpage_expense_validate`, `custpage_user`, `custpage_password`, `custpage_withholdingdata` y la sublista `custpage_sublist_withholding`

fieldChanged

Actualiza los campos del formulario cuando el usuario cambia la subsidiaria, cargando la configuración correspondiente.

Lógica: Si el campo modificado es `custpage_subsidiary`, obtiene la subsidiaria seleccionada, ejecuta una búsqueda del registro `customrecord_lmry_mx_setup_inbound` filtrado por esa subsidiaria, asigna los valores encontrados a los campos del formulario (incluyendo paquete), parsea JSON para departamento, clasificación y ubicación, determina la validación de gasto, oculta la contraseña con asteriscos y rellena la sublista de retenciones; si no hay resultados, limpia los campos relacionados.

Salida: Actualiza los mismos campos que `pageInit`, añadiendo `custpage_package`

saveRecord

Antes de guardar, recopila los datos de la sublista de retenciones y los almacena como JSON en un campo oculto.

Lógica: Itera sobre todas las líneas de la sublista `custpage_sublist_withholding`, recoge nombre, tasa y código de impuesto si existen, construye un arreglo de objetos, serializa el arreglo a JSON y lo guarda en `custpage_withholdingdata`; siempre devuelve `true` para permitir el guardado.

Salida: Actualiza el campo `custpage_withholdingdata` con la representación JSON de la sublista

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
custpage_subsiary	-	lectura escritura	Se establece en 1 cuando la característica de subsidiarias está desactivada; al cambiarlo dispara la carga de configuración correspondiente
custpage_taxcode0	-	escritura	Recibe el código de impuesto 0% de la configuración inbound
custpage_taxcode8	-	escritura	Recibe el código de impuesto 8% de la configuración inbound
custpage_taxcode16	-	escritura	Recibe el código de impuesto 16% de la configuración inbound
custpage_department	-	escritura	Se llena con el valor department extraído del JSON almacenado en la configuración
custpage_class	-	escritura	Se llena con la clasificación extraída del JSON almacenado en la configuración
custpage_location	-	escritura	Se llena con la ubicación extraída del JSON almacenado en la configuración
custpage_item	-	escritura	Recibe el ítem predeterminado de la configuración inbound
custpage_vendor	-	escritura	Recibe el proveedor predeterminado de la configuración inbound
custpage_package	-	escritura	Recibe el paquete predeterminado de la configuración inbound (solo en cambio de subsidiaria)
custpage_expense_validate	-	escritura	Bandera booleana que indica si la validación de gastos está habilitada
custpage_user	-	escritura	Usuario SAT obtenido de la configuración

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
custpage_password	-	escritura	Contraseña SAT mostrada enmascarada con asteriscos
custpage_withholdingdata	-	escritura	Almacena la tabla de retenciones como cadena JSON
custpage_sublist_namewithholding	custpage_sublist_withholding	lectura escritura	Nombre de la retención en cada línea de la sublista
custpage_sublist_ratewithholding	custpage_sublist_withholding	lectura escritura	Tasa de la retención en cada línea de la sublista
custpage_sublist_whttaxcode	custpage_sublist_withholding	lectura escritura	Código de impuesto de retención en cada línea de la sublista

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: Carga y sincroniza configuración inbound; oculta la contraseña con asteriscos; gestiona la sublista de retenciones

Dependencias

NOMBRE	PROPÓSITO
N/search	Ejecuta búsquedas para obtener la configuración inbound según la subsidiaria seleccionada
N/currentRecord	Accede al registro actual del formulario para leer y establecer valores de campos y sublistas
N/runtime	Verifica si la característica de subsidiarias está habilitada en la cuenta

Resumen Técnico

En pageInit y fieldChanged busca el registro customrecord_lmry_mx_setup_inbound, asigna los valores a los campos del formulario y rellena la sublista de retenciones; al guardar, serializa la sublista en JSON.

Generado automáticamente 2026-01-23 17:48 UTC

SUITELET · CONFIGURACIÓN INBOUND MX

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios configurar y mantener la integración inbound de México, definiendo impuestos, subsidiarias, departamentos y datos de retención en un único formulario.

Flujo de Ejecución

GET

POST

Procesa los datos enviados, crea o actualiza el registro de configuración y redirige

Flujo: Al recibir una solicitud POST, el Suitelet extrae los parámetros del formulario, construye objetos JSON para departamento, clasificación y ubicación, y busca un registro `customrecord_lmry_mx_setup_inbound` asociado a la subsidiaria seleccionada. Si existe, lo carga y actualiza sus campos; si no, crea uno nuevo con los valores recibidos. Después guarda el registro y redirige al mismo Suitelet para recargar la página.

Genera: Redirección al Suitelet después de guardar la configuración

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_subsidiary	Subsidiaria	SELECT	obligatorio, opciones dinámicas según subsidiarias mexicanas
custpage_taxcode0	Tax Code 0	SELECT	opción vacía incluida, opciones de impuestos MX
custpage_taxcode8	Tax Code 8	SELECT	opción vacía incluida, opciones de impuestos MX
custpage_taxcode16	Tax Code 16	SELECT	opción vacía incluida, opciones de impuestos MX
custpage_department	Department	SELECT	opción vacía incluida, visible solo si la característica DEPARTMENTS está activa
custpage_class	Class	SELECT	opción vacía incluida, visible solo si la característica CLASSES está activa
custpage_location	Location	SELECT	opción vacía incluida, visible solo si la característica LOCATIONS está activa
custpage_item	Item Generic	SELECT	fuentes de lista 'item'
custpage_vendor	Vendor Generic	SELECT	fuentes de lista 'vendor'
custpage_package	Inbound Package	SELECT	fuentes de lista 'customrecord_psg_ei_standards'
custpage_expense_validate	Validate expense report	CHECKBOX	opcional
custpage_user	User	TEXT	opcional, dentro del grupo SAT Connection
custpage_password	Password	TEXT	opcional, dentro del grupo SAT Connection
custpage_withholdingdata	Withholding Data	LONGTEXT	oculto, almacena JSON de la sublista de retenciones

Lógica POST

Al recibir POST, el Suitelet valida la subsidiaria, construye objetos JSON para campos opcionales, busca un registro de configuración existente y lo actualiza o crea uno nuevo con los valores del formulario, luego redirige al mismo Suitelet.

Búsquedas

Obtener el registro de configuración inbound asociado a la subsidiaria seleccionada

Filtros:

Retorna: internalid del registro (si existe)

Operaciones

TIPO	TARGET	PROPÓSITO
load	customrecord_lmry_mx_setup_inbound	Cargar registro existente para actualizar
create	customrecord_lmry_mx_setup_inbound	Crear nuevo registro cuando no existe
setValue	varios campos del registro	Asignar valores del formulario, incluyendo JSON para department, class y location
save	customrecord_lmry_mx_setup_inbound	Persistir los cambios en NetSuite
redirect	Suitelet customscript_lmry_mx_inbsetup_stlt	Recargar la página después de guardar

Comportamiento del Suitelet

CREA REGISTROS

Sí

ACTUALIZA REGISTROS

Sí

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: Campo withholding data se mantiene oculto; la contraseña solo se guarda si no está enmascarada

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construir y renderizar el formulario y sublistas
N/search	Ejecutar búsquedas para poblar listas desplegables y localizar registros existentes
N/runtime	Verificar la activación de características (departamentos, ubicaciones, clases, subsidiarias)
N/record	Cargar, crear y guardar registros customrecord_lmry_mx_setup_inbound

Resumen Técnico

Genera dinámicamente un formulario Suitelet con datos de listas mediante búsquedas, procesa la información enviada y crea o actualiza un registro customrecord, finalizando con una redirección al mismo Suitelet.

INTERNALMODULE · TRADUCTOR DE ETIQUETAS PARA INBOUND MX

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: utils

Traductor de etiquetas para Inbound MX

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza todas las etiquetas de la interfaz Inbound MX, permitiendo que otros scripts obtengan los textos traducidos según el idioma del usuario.'), 'technical_summary': Markup('Expone la función getTranslations que recibe un código de idioma y devuelve un objeto con las claves de etiqueta y sus valores en español o inglés.')}

API pública

getTranslations

Devuelve el conjunto de etiquetas traducidas según el código de idioma solicitado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código de idioma, por ejemplo 'es' o 'en'	

Retorna

Objeto cuyas claves son identificadores de etiqueta y cuyos valores son los textos traducidos

Modelos de datos / estructuras

Translations

Mapa de claves de etiqueta a su texto traducido

Campos: clave: string, valor: string

Detalles de implementación

El módulo mantiene un diccionario interno con las etiquetas en español e inglés y expone la función `getTranslations` que, al recibir un código de idioma, devuelve el objeto correspondiente; si el idioma no está definido, se devuelve el conjunto en inglés por defecto.

Generado automáticamente 2026-01-23 17:49 UTC

CLIENTSCRIPT · GESTIÓN DE DETALLE DE PO/BILL EN POPUP

SuiteScript 2.1 · Módulos: 7

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

El script controla la captura y validación de datos de órdenes de compra y facturas en un popup, asegurando que la información sea coherente antes de enviarla al registro principal.

Lifecycle Hooks

pageInit

Inicializa el formulario, carga datos de la orden de compra o factura y establece el idioma.

Lógica: Lee valores de campos ocultos, deshabilita campos de PO en la sublista de ítems, determina el idioma del usuario y, si el estado es 3 o 4, prepara la información del XML para su visualización.

Salida: Deshabilita campos de PO en la sublista de ítems y establece la variable de idioma.

fieldChanged

Responde a cambios en campos y sublistas, controla la selección única de PO y Bill, y actualiza valores dependientes.

Lógica: Gestiona toggles de asignación, habilita/deshabilita checkboxes de PO y Bill, carga ítems del PO seleccionado, actualiza opciones del selector de ítems, recalcula montos cuando cambian amount o tax_rate y completa valores de PO o Bill al cambiar sus campos.

Validaciones

- **assign_flag:** cambio de valor

Acción si falla: habilita o deshabilita el campo custpage_bill_manually

- **po_checked:** checkbox marcado

Acción si falla: deshabilita los demás checkboxes, carga ítems del PO y muestra campos de PO

- **bill_checked:** checkbox marcado

Acción si falla: deshabilita los demás checkboxes de Bill

- **purchase_order:** cambio de PO en sublista custpage_pos_list

Acción si falla: completa valores de la PO en la línea

- **bill_internalid:** cambio de Bill en sublista custpage_pos_list

Acción si falla: completa valores del Bill en la línea

- **[Markup('amount'), Markup('tax_rate')]:** cambio de monto o tasa de impuesto en sublista custpage_items_list

Acción si falla: recalcula tax_amount y total_amount

Salida: Actualiza valores de campos, habilita/deshabilita checkboxes y actualiza opciones de selección.

validateField

Valida el campo assign_flag y habilita o deshabilita el campo de facturación manual.

Lógica: Si el campo es assign_flag, habilita o deshabilita custpage_bill_manually según el valor.

Validaciones

- **assign_flag:** cambio de valor

Acción si falla: actualiza estado de custpage_bill_manually

Salida: Campo custpage_bill_manually habilitado o deshabilitado.

validateLine

Impide la creación de líneas de ítems que superen la cantidad de ítems del XML.

Lógica: Compara el índice de la línea actual con la longitud del arreglo items del XML; si supera, devuelve false.

Validaciones

- **línea de custpage_items_list:** índice mayor que número de ítems del XML

Acción si falla: bloquea la línea retornando false

validateInsert

Bloquea la inserción de nuevas líneas en la sublista de ítems.

Lógica: Si la sublista es custpage_items_list, siempre devuelve false.

Validaciones

- **inserción en custpage_items_list:** cualquier intento de inserción

Acción si falla: bloquea la inserción retornando false

validateDelete

Bloquea la eliminación de líneas en la sublista de ítems.

Lógica: Si la sublista es custpage_items_list, siempre devuelve false.

Validaciones

- **eliminación en custpage_items_list:** cualquier intento de borrado

Acción si falla: bloquea la eliminación retornando false

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
assign_flag	-	validación	Al marcarlo habilita el campo custpage_bill_manually; al desmarcarlo lo deshabilita.
custpage_bill_manually	-	escritura	Se habilita o deshabilita según assign_flag; su valor se reinicia a 0 en ambos casos.
po_checked	custpage_pos_list	validación	Permite seleccionar solo una PO a la vez; al marcarla carga ítems del PO y habilita campos de PO.
bill_checked	custpage_pos_list	validación	Permite seleccionar solo un Bill a la vez; al marcarla deshabilita los demás checkboxes.
purchase_order	custpage_pos_list	cálculo	Al cambiarlo completa fecha, monto y tipo de documento de la PO en la línea.
bill_internalid	custpage_pos_list	cálculo	Al cambiarlo completa fecha, monto y tipo de documento del Bill en la línea.
tax_code	custpage_items_list	cálculo	Al cambiarlo (código de impuesto) se podría actualizar la tasa de impuesto; lógica actual está comentada.
amount	custpage_items_list	cálculo	Al cambiarlo se recalcula tax_amount y total_amount.
tax_rate	custpage_items_list	cálculo	Al cambiarlo se recalcula tax_amount y total_amount.
custpage_item	custpage_items_list	lectura escritura	Se deshabilita cuando se selecciona una PO; su valor se limpia al habilitar PO.
custpage_item_po	custpage_items_list	lectura escritura	Se habilita cuando se marca una PO; muestra los ítems del PO seleccionado.
vendor_code	custpage_items_list	escritura	Se rellena con el código de proveedor del XML o con 'GENÉRICO' si no existe.
department	custpage_items_list	escritura	Se establece con el valor de configuración del subsidiary.
class	custpage_items_list	escritura	Se establece con el valor de configuración del subsidiary.
location	custpage_items_list	escritura	Se establece con el valor de configuración del subsidiary.

Validaciones

UBICACIÓN	CAMPO	CRITERIO	ACCIÓN	BLOQUEA
validateLine	custpage_items_list	índice de línea mayor que número de ítems del XML	bloquea la línea y muestra mensaje de error	Sí
validateInsert	custpage_items_list	cualquier intento de inserción	bloquea la inserción	Sí
validateDelete	custpage_items_list	cualquier intento de borrado	bloquea la eliminación	Sí
updateParent	date	fecha vacía cuando se aprueba y no es facturación manual	muestra alerta y cancela la operación	Sí
updateParent	total_amount vs sum items	suma de ítems distinta al total de la factura	muestra alerta de cantidad y cancela la operación	Sí
updateParent	currency	moneda vacía al aprobar	muestra alerta de moneda y cancela la operación	Sí
updateParent	purchase_order amount	monto de PO cero o menor al total de la factura	muestra alerta amount_error y cancela la operación	Sí
updateParent	bill_credit amount	monto de Nota de Crédito mayor al monto del Bill seleccionado	muestra alerta bill_credit_amount_error y cancela la operación	Sí
updateParent	selección de Bill	ningún Bill seleccionado al aprobar una Nota de Crédito	muestra alerta y cancela la operación	Sí
updateParent	selección de PO	ningún PO seleccionado al aprobar una Factura	muestra alerta informativa (se generará sin orden de compra)	No

Interacción con Usuario

alert: Debe completar el campo Periodo Contable

Trigger: cuando falta el periodo contable

alert: Debe completar el monto total indicado en el campo Monto Total

Trigger: cuando la suma de ítems no coincide con el total

alert: Debe completar el campo Currency

Trigger: cuando la moneda está vacía al aprobar

alert: Debe completar la fecha

Trigger: cuando la fecha está vacía al aprobar

alert: Ocurrió un error

Trigger: cualquier excepción capturada

alert: El monto de purchase order es cero o menor al monto de la factura emitida, por favor modifique su purchase order

Trigger: cuando el monto de PO es insuficiente

alert: El monto de la Nota de Crédito no puede ser mayor al Bill seleccionado.

Trigger: cuando el monto de la Nota de Crédito supera al Bill

alert: El numero de lineas del XML es diferente al numero de lineas del purchase order.

Trigger: cuando la validación de número de líneas está activada y difiere

alert: Debe seleccionar al menos una factura para aplicar la Nota de Crédito.

Trigger: cuando se aprueba una Nota de Crédito sin seleccionar Bill

alert: Se generará sin orden de compra.

Trigger: cuando se aprueba una Factura sin PO seleccionado

alert: Se generará el Vendor Credit sin aplicar a ningún Bill.

Trigger: cuando se aprueba una Nota de Crédito sin Bill

Comportamiento del Script

VALIDA DATOS

Sí

BLOQUEA GUARDADO

No

OCULTA CAMPOS

Sí

MODIFICA VALORES

Sí

MUESTRA ALERTAS

Sí

Notas: Deshabilita campos según selección, controla la inserción y eliminación de líneas, y muestra alertas de validación.

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener la preferencia de idioma del usuario y verificar características de la cuenta.
N/currentRecord	Acceder al registro actual del formulario.
N/url	No utilizado directamente en la lógica actual, reservado para generación de URLs.
N/search	Buscar ítems, PO, Bills y configuraciones de subsidiary.
N/record	Cargar registros de Purchase Order y Bill para obtener sus líneas.
N/format	No utilizado directamente en la lógica actual, reservado para formateo de datos.
N/ui/dialog	No utilizado directamente; potencialmente para mostrar diálogos personalizados.

Resumen Técnico

Implementa validaciones de campos, controla la selección única de PO y Bill, recalcula montos y bloquea inserciones o eliminaciones de líneas mediante los métodos de ciclo de vida de Client Script.

Generado automáticamente 2026-01-23 18:00 UTC

CLIENTSCRIPT · GESTIÓN DE PAGOS INBOUND EN FORMULARIO CLIENTE

SuiteScript 2.1 · Módulos: 3

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

Facilita la gestión de pagos inbound, permite aplicar o desmarcar rápidamente todas las líneas y delega validaciones al handler especializado.

Lifecycle Hooks

pageInit

Inicializa el handler del script y delega la lógica de carga del formulario

Lógica: Crea una instancia del handler, la almacena y ejecuta su método pageInit pasando el contexto

Salida: Puede modificar campos o UI según la lógica del handler

fieldChanged

Reenvía el evento de cambio de campo al handler para su procesamiento

Lógica: Llama al método fieldChanged del handler pasando el contexto

Salida: El handler puede actualizar valores o mostrar mensajes

saveRecord

Ejecuta la validación y lógica de guardado delegada al handler

Lógica: Invoca el método saveRecord del handler y devuelve su resultado (true permite guardar, false lo bloquea)

Salida: Resultado booleano que determina si el registro se guarda

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: Incluye función toggleCheckBoxes que marca o desmarca el campo apply en todas las líneas de una sublista

Dependencias

NOMBRE	PROPÓSITO
../Handlers/ LMRY_PE_Inbound_CLNT_HNDL	Contiene la lógica de negocio para pageInit, fieldChanged y saveRecord
N/url	Proporciona utilidades de URL que pueden ser usadas por el handler
N/currentRecord	Permite acceder al registro actual para manipular sublistas en toggleCheckBoxes

Resumen Técnico

Instancia un handler y delega los eventos pageInit, fieldChanged y saveRecord; incluye una utilidad para marcar/desmarcar checkboxes en sublistas mediante N/currentRecord.

INTERNALMODULE · MANEJADOR DE POPUP INBOUND PE

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador de Popup Inbound PE

Dependencias: N/URL N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET N/HTTPS
N/FORMAT N/XML N/QUERY N/ENCODE N/CONFIG
../TRANSLATOR/LMRY_PE_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo genera el popup de detalle para documentos inbound en Perú, permitiendo al usuario revisar la información del XML, buscar órdenes de compra o facturas relacionadas y crear los ítems correspondientes con sus impuestos y detracciones.'),
'technical_summary': Markup('Expone la clase LibraryHandler que, mediante búsquedas, consultas a registros y traducciones, construye un formulario Suitelet con sublistas dinámicas, procesa el XML inbound y gestiona la lógica de aprobación y creación de asientos.')}

Dependencias

NOMBRE	PROPÓSITO
N/url	Construir URLs a scripts y registros
N/search	Ejecutar búsquedas y lookup de campos
N/runtime	Obtener información del script, parámetros y características activas
N/redirect	Redirigir a otro Suitelet
N/ui/serverWidget	Crear y manipular formularios y sublistas en Suitelet
N/https	Realizar llamadas HTTP externas (no usado directamente en este módulo, pero cargado por compatibilidad)
N/format	Parsear y formatear fechas
N/xml	Parsear el contenido XML del documento inbound
N/query	Obtener listas de monedas del proveedor mediante Query API
N/encode	Codificar/decodificar datos (cargado por compatibilidad)
N/config	Leer preferencias de contabilidad, como UNRECEIVEDBILLS
../Translator/ LMRY_PE_Inbound_Translator	Proveer traducciones de textos según idioma del usuario

API pública

LibraryHandler

Clase principal que construye el formulario de detalle inbound, procesa el XML, busca PO, facturas y items, y gestiona la lógica de aprobación.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto con propiedades params (parámetros del Suitelet) y method (GET o POST).	

Efectos colaterales

- Crea formularios y sublistas en la UI
- Ejecuta búsquedas y lecturas de registros
- Registra logs en el script

Retorna

Instancia de LibraryHandler

LibraryHandler.createForm

Construye el formulario principal con campos de información primaria y botones de acción.

Efectos colaterales

- Añade campos ocultos y visibles al formulario
- Registra logs de depuración

Retorna

Objeto serverWidget.Form

LibraryHandler.addVendor

Busca datos del proveedor y agrega la opción al campo vendedor del formulario.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
vendorField	Campo SELECT del formulario donde se añadirá la opción del proveedor.	

Efectos colaterales

- Realiza lookup de campos del registro vendedor

LibraryHandler.createPOSublist

Crea la sublista de PO o Bill según el tipo de documento y muestra enlaces a XML/PDF.

Efectos colaterales

- Añade campos y sublistas al formulario

LibraryHandler.createSublist

Genera las pestañas y sublistas de PO/Bill y de Items, y las llena con datos obtenidos de búsquedas.

Efectos colaterales

- Crea sublistas de tipo LIST e INLINEEDITOR

LibraryHandler.getSetup

Obtiene la configuración de impuestos, departamentos, clases, ubicaciones, ítem y tabla de detracción para la subsidiaria indicada.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
subsidiary	Id interno de la subsidiaria	

Retorna

Objeto con propiedades taxcode0, taxcode10, taxcode18, department, class, location, item, detraction

LibraryHandler.fillItemSublist

Puebla la sublista de items con la información extraída del XML y la configuración de impuestos.

Efectos colaterales

- Calcula montos, impuestos y posible ítem de detracción

LibraryHandler.fillPoSublist

Llena la sublista de órdenes de compra pendientes con datos de búsqueda.

LibraryHandler.fillBillSublist

Llena la sublista de facturas del proveedor y marca la que corresponde al documento inbound.

LibraryHandler.getPurchaseOrder

Busca órdenes de compra no facturadas del proveedor, aplicando filtros de subsidiaria y estado.

Retorna

Objeto map con id de PO como clave y datos de la PO como valor

LibraryHandler.getVendorBills

Obtiene facturas del proveedor dentro de periodos abiertos y que tengan el flag inbound.

Retorna

Objeto map con id de factura como clave y datos de la factura como valor

LibraryHandler.getDataXML

Parsea el registro customrecord_psg_ei_inbound_edoc y devuelve un objeto con la información relevante del documento (items, pagos, tipo, etc.).

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
xmlData	Id interno del registro que contiene el XML	

Retorna

Objeto con propiedades purchaseorder, itemreceipt, rateIVA, items, paymentMeansID, paymentPercent, documentType, referencedInvoiceId

LibraryHandler.getTranslations

Obtiene el diccionario de traducciones para el idioma del usuario.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código ISO del idioma (ej. 'es')	

Retorna

Objeto con pares clave-texto

LibraryHandler.getText

Devuelve el texto traducido para la clave indicada.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
key	Clave del diccionario de traducciones	

Retorna

Texto traducido

LibraryHandler.convertDate

Convierte una cadena en formato YYYY-MM-DD a objeto Date.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
date	Cadena de fecha	

Retorna

Objeto Date o cadena vacía

LibraryHandler.getPeriods

Llena un campo SELECT con los periodos contables abiertos y selecciona el actual.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
periodField	Campo SELECT del formulario	

Modelos de datos / estructuras

SetupData

Configuración por subsidiaria utilizada para impuestos y listas de selección.

CAMPO	TIPO	DESCRIPCIÓN
taxcode0	Id interno del código de impuesto 0 %	
taxcode10	Id interno del código de impuesto 10 %	
taxcode18	Id interno del código de impuesto 18 %	
department	Id interno del departamento por defecto	
class	Id interno de la clasificación por defecto	
location	Id interno de la ubicación por defecto	
item	Id interno del ítem por defecto	
detraction	JSON con tabla de conceptos de detracción	

XmlData

Representación estructurada del contenido XML del documento inbound.

CAMPO	TIPO	DESCRIPCIÓN
purchaseorder	Texto del número de orden de compra (si aplica)	
itemreceipt	Texto del número de recibo de ítem (si aplica)	
rateIVA	Tasa de IVA general del documento	
items	Arreglo de objetos ItemData	
paymentMeansID	Código del medio de pago	
paymentPercent	Porcentaje de detracción	
documentType	Tipo de documento (Factura o Nota de crédito)	
referencedInvoiceId	Id de factura referenciada en caso de nota de crédito	

ItemData

Detalle de cada línea del documento XML.

CAMPO	TIPO	DESCRIPCIÓN
name	Descripción del ítem	
quantity	Cantidad facturada	
price	Precio unitario	
discount	Descuento aplicado	
rateIvaItem	Tasa de IVA del ítem (0,10,18)	
amount	Importe total de la línea	
vlrCodigo	Código del proveedor para el ítem (opcional)	

Generado automáticamente 2026-01-23 18:09 UTC

INTERNALMODULE · MANEJADOR CLIENTE PARA INBOUND DE PERÚ

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador cliente para inbound de Perú

Dependencias: N/RUNTIME N/RECORD N/SEARCH N/URL N/HTTPS N/FORMAT
N/CURRENTRECORD N/QUERY N/CURRENCY N/UI/MESSAGE
../TRANSLATOR/LMRY_PE_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza la lógica de validación y registro de documentos inbound en la interfaz cliente, facilitando su uso por otros scripts de NetSuite.'),
'technical_summary': Markup('Expone una clase con manejadores de eventos (pageInit, fieldChanged, saveRecord) y una función de navegación; utiliza búsquedas, consultas y creación de registros para procesar la información.')}

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener información del entorno, validar características y preferencias del usuario
N/record	Crear y guardar registros personalizados de log
N/search	Ejecutar búsquedas de periodos, departamentos, clases y ubicaciones
N/url	Construir URLs a suitelets para abrir pop-ups
N/https	No utilizado directamente, pero disponible para llamadas externas
N/format	Parsear y formatear fechas
N/currentRecord	Acceder al registro actual del cliente
N/query	Obtener la lista de monedas asociadas a un vendor
N/currency	No utilizado directamente, pero expuesto para posibles conversiones
N/ui/message	No utilizado directamente, pero disponible para mostrar mensajes UI
../Translator/ LMRY_PE_Inbound_Translator	Proveer traducciones de textos según el idioma del usuario

API pública

ClntHandler

Clase que contiene los manejadores de eventos del cliente y la lógica de validación

back

Navega a la suitelet principal según el estado del registro

Efectos colaterales

- Redirección de la ventana del navegador
- Uso de `url.resolveScript`

Funciones internas

NOMBRE	ROL	USADO POR
validate_open_window	Verifica que la ventana de detalle para la línea no esté ya abierta	fieldChanged
validateMandatoryFields	Comprueba que los campos obligatorios del formulario tengan valor	saveRecord
validateDates	Valida que la fecha de fin no sea anterior a la fecha de inicio	saveRecord
validateRangeDates	Asegura que ambas fechas pertenezcan al mismo mes y año	saveRecord
validateSublist	Confirma que al menos una línea de la sublista esté seleccionada	saveRecord
fillPeriod	Carga los periodos contables activos en el campo de selección	
fillDepartments	Pobla la lista de departamentos según la subsidiaria y la configuración de NetSuite	
fillClasses	Pobla la lista de clases según la subsidiaria	
fillLocations	Pobla la lista de ubicaciones según la subsidiaria	
fillCurrenciesByEntity	Obtiene y muestra las monedas asociadas al vendor seleccionado	
createLogInbound	Genera un registro de log con la información de los documentos inbound procesados	saveRecord
getSetup	Recupera la configuración de inbound para la subsidiaria indicada	
getText	Devuelve la cadena traducida para la clave solicitada	pageInit , fieldChanged , saveRecord , validateMandatoryFields , validateDates , validateRangeDates , validateSublist , createLogInbound , back

NOMBRE	ROL	USADO POR
getTranslations	Obtiene el objeto de traducciones del módulo Translator	constructor

Generado automáticamente 2026-01-23 17:57 UTC

INTERNALMODULE · MANEJADOR DE SUITELET PARA INBOUND PE

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: class

Manejador de Suitelet para Inbound PE

Dependencias: N/CONFIG N/SEARCH N/RUNTIME N/REDIRECT N/UI/SERVERWIDGET
N/RECORD N/FORMAT N/XML N/URL N/TASK
../TRANSLATOR/LMRY_PE_INBOUND_TRANSLATOR

Resumen Ejecutivo

{'business_summary': Markup('Este módulo centraliza la lógica de visualización y procesamiento de documentos inbound para Perú, ofreciendo filtros por subsidiaria y fechas, y permitiendo la conversión masiva mediante Map/Reduce.'), 'technical_summary': Markup('Construye dinámicamente un formulario Suitelet, consulta registros inbound, parsea XML, muestra resultados en una sublista y orquesta tareas de Map/Reduce y logs mediante la API de NetSuite.')}

Dependencias

NOMBRE	PROPÓSITO
N/config	Acceder a la configuración de la compañía y cargar información de la empresa.
N/search	Ejecutar búsquedas guardadas y paginadas sobre registros de NetSuite.
N/runtime	Obtener información del entorno de ejecución, como parámetros de script y características activas.
N/redirect	Redirigir al usuario a otros Suitelets o páginas.
N/ui/serverWidget	Construir formularios y sublistas que se presentan en la UI del Suitelet.
N/record	Crear y guardar registros personalizados de log.
N/format	Parsear y formatear fechas.
N/xml	Parsear contenido XML y ejecutar consultas XPath.
N/url	Construir URLs a scripts y registros.
N/task	Ejecutar tareas de Map/Reduce.
../Translator/ LMRY_PE_Inbound_Translator	Obtener traducciones de texto según el idioma del usuario.

API pública

LibraryHandler.constructor

Inicializa la instancia con parámetros, método y verifica características del entorno.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
options	Objeto que contiene params y method.	

Retorna

Instancia de LibraryHandler.

LibraryHandler.createFilterForm

Construye el formulario de filtros y botones según el estado y la configuración de subsidiarias.

Efectos colaterales

- Crea campos y grupos en el formulario.

Retorna

Objeto serverWidget.Form configurado.

LibraryHandler.createApplySublist

Agrega una pestaña y sublista al formulario para mostrar resultados de documentos inbound.

Efectos colaterales

- Define columnas de la sublista.

Retorna

Objeto serverWidget.Sublist configurado.

LibraryHandler.loadSublist

Llena la sublista con datos obtenidos del método getResponse y genera enlaces a PDF, XML y registros relacionados.

Efectos colaterales

- Actualiza valores de la sublista.
- Genera URLs dinámicas.

LibraryHandler.fillSubsidiaries

Puebla el campo de subsidiarias con las opciones activas.

LibraryHandler.getRedirectParams

Construye un objeto con los parámetros necesarios para redirigir al Suitelet de procesamiento.

Retorna

Objeto con subsidiary, startdate, enddate y status.

LibraryHandler.setFormValues

Establece los valores por defecto del formulario a partir de los parámetros recibidos y la configuración de la compañía.

LibraryHandler.toLogSuitelet

Redirige al Suitelet de log de inbound.

Efectos colaterales

- Ejecuta una redirección.

LibraryHandler.executeMapReduce

Inicia una tarea Map/Reduce pasando los datos de conversión y la subsidiaria como parámetros.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
convertData	Datos a procesar en el Map/Reduce.	
subsidiary	Identificador de la subsidiaria.	

Efectos colaterales

- Crea y envía una tarea de Map/Reduce.

Retorna

Id de la tarea Map/Reduce.

LibraryHandler.createLogInbound

Crea un registro personalizado de log de inbound con la información de la ejecución.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
convertData	Datos convertidos que se guardan en el registro.	
params	Objeto con fechas, subsidiaria, usuario y conteo de registros.	

Efectos colaterales

- Crea y guarda un registro personalizado.

Retorna

Id del registro creado.

LibraryHandler.getTranslations

Obtiene el diccionario de traducciones para el idioma especificado.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código de idioma (dos letras).	

Retorna

Objeto con pares clave-valor de traducciones.

LibraryHandler.getText

Devuelve el texto traducido para la clave indicada.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
key	Clave del diccionario de traducciones.	

Retorna

Texto traducido.

LibraryHandler.getResponse

Recupera y procesa los registros inbound dentro del rango de fechas, devolviendo una lista estructurada de documentos.

Efectos colaterales

- Ejecuta búsquedas y parsea XML.

Retorna

Objeto con la propiedad results que contiene un arreglo de documentos.

LibraryHandler.getStatusType

Mapea el código de estado interno a una etiqueta legible.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
status	Código numérico del estado.	

Retorna

Etiqueta de estado (Pending, Converted, etc.).

LibraryHandler.extraerFolioDelXML

Extrae el número de folio del contenido XML mediante expresión regular.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
xmlContent	Cadena XML completa.	

Retorna

Folio como cadena o vacío.

LibraryHandler.getInboundStandardData

Obtiene los registros inbound que cumplen con el rango de fechas y los agrupa en estructuras auxiliares.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
startDate	Fecha de inicio del filtro.	
endDate	Fecha de fin del filtro.	
subsidiary	Identificador de la subsidiaria (opcional).	

Efectos colaterales

- Ejecuta una búsqueda paginada.

Retorna

Objeto con XML concatenado y listas de vendor, status, ids y folios.

LibraryHandler.getBillsByinboundRecord

Busca facturas y notas de crédito vinculadas a documentos inbound convertidos.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
data	Arreglo de resultados del método getResponse.	

Efectos colaterales

- Ejecuta búsquedas de vendorbill y vendorcredit.

Retorna

Mapa de inboundId a datos de factura o crédito.

LibraryHandler.getLatestAuditTrialByState

Obtiene el registro de auditoría más reciente para los documentos en un estado específico.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
data	Arreglo de resultados del método getResponse.	
state	Estado a filtrar (Failed, Incomplete, etc.).	

Efectos colaterales

- Ejecuta una búsqueda sobre el registro de auditoría.

Retorna

Mapa de inboundId a detalle de auditoría.

LibraryHandler.getDateFromXml

Extrae la fecha de emisión del XML mediante expresión regular.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
xmlData	Contenido XML del documento.	

Retorna

Fecha en formato YYYY-MM-DD o null.

LibraryHandler.convertDate

Convierte una fecha en formato YYYY-MM-DD a la representación de NetSuite.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
date	Cadena de fecha en formato YYYY-MM-DD.	

Retorna

Fecha formateada según la configuración de NetSuite.

Funciones internas

NOMBRE	ROL	USADO POR
areThereSubsidiaries	Determina si la cuenta tiene subsidiarias activas y prepara la lista para el formulario.	constructor , createFilterForm
getSubsidiaries	Ejecuta una búsqueda de subsidiarias activas en Perú con taxid.	areThereSubsidiaries

Modelos de datos / estructuras

InboundDocument

Representa un documento inbound procesado para la sublista.

CAMPO	TIPO	DESCRIPCIÓN
issuer_name	Nombre del proveedor	
issuer_rut	RUT del proveedor	
folio	Número de folio del documento	
total_amount	Monto total	
document_type	Tipo legible (Factura o Nota de credito)	
date	Fecha de emisión formateada	
see_pdf	Id interno del PDF adjunto	
see_xml	Id interno del XML	
status	Código de estado interno	
statustype	Etiqueta de estado legible	

InboundStandardData

Estructura que agrupa los XML concatenados y listas auxiliares obtenidas de la búsqueda.

CAMPO	TIPO	DESCRIPCIÓN
inboundRecordsXml	Cadena XML completa de todos los documentos	
vendorList	Lista de proveedores	
statusList	Lista de códigos de estado	
xmlList	Lista de internalId de los registros inbound	
pdfList	Lista de internalId de los PDFs adjuntos	
foliosList	Lista de folios extraídos del XML	

INTERNALMODULE · MANEJO DE FACTURAS DE PROVEEDOR INBOUND URET

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: functions

Manejo de facturas de proveedor inbound URET

Dependencias: N/LOG N/RUNTIME N/SEARCH N/RECORD N/XML N/FILE

Resumen Ejecutivo

Este módulo procesa facturas de proveedor inbound URET, genera archivos XML en el File Cabinet y registra información adicional (fecha, proveedor y referencia al archivo) en un registro personalizado para su uso en integraciones posteriores.

Dependencias

NOMBRE	PROPÓSITO
N/log	Registro de mensajes de depuración y errores
N/runtime	Acceso a información del entorno de ejecución
N/search	Búsqueda y consulta de carpetas y registros personalizados
N/record	Creación, carga y actualización de registros NetSuite
N/xml	Parseo y extracción de datos del contenido XML del inbound
N/file	Creación y guardado de archivos XML en el File Cabinet

API pública

beforeLoad

Hook ejecutado antes de cargar el registro; no realiza ninguna acción en este script

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, type y form	

beforeSubmit

Hook ejecutado antes de enviar el registro; actualmente vacío

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, oldRecord y type	

afterSubmit

Procesa el contenido XML del registro inbound, guarda el archivo XML y actualiza los campos extra asociados

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
scriptContext	Objeto que contiene newRecord, oldRecord y type	

Efectos colaterales

- Creación de carpetas en el File Cabinet
- Guardado de archivo XML
- Creación o actualización de registro customrecord_lmry_inbound_fields

Funciones internas

NOMBRE	ROL	USADO POR
getInboundFolder	Obtiene o crea la subcarpeta donde se almacenan los XML inbound	afterSubmit
getFolderIdByName	Busca el internalId de una carpeta por nombre	getInboundFolder
getSubFolderId	Busca el internalId de una subcarpeta bajo un padre especificado	getInboundFolder
createFolder	Crea una carpeta en el File Cabinet y devuelve su internalId	getInboundFolder
getXmlData	Parsea la cadena XML y devuelve un objeto con los campos relevantes	afterSubmit
extractXMLData	Extrae valores específicos (RUC, serie, número, fecha, monto, moneda) del documento XML	getXmlData
createOrUpdateInboundExtraFields	Crea o actualiza el registro customrecord_lmry_inbound_fields con la información extra del inbound	afterSubmit

Modelos de datos / estructuras

InboundXmlData

Objeto que representa los datos extraídos del XML inbound

CAMPO	TIPO	DESCRIPCIÓN
numRuc	RUC del proveedor	
numeroSerie	Serie del documento	
numero	Número del documento	
fechaEmision	Fecha de emisión como objeto Date	
monto	Importe total del documento	
moneda	Código de moneda	

Efectos colaterales

- Creación de carpetas en el File Cabinet si no existen
- Guardado de archivos XML en la carpeta 'XML and PDF Inbound'
- Creación o actualización de registros personalizados de tipo customrecord_lmry_inbound_fields
- Registro de mensajes de depuración y errores en el log de NetSuite

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
createOrUpdateInboundExtraFields	Captura y registra cualquier excepción que ocurra al crear o actualizar el registro de campos inbound	Error registrado mediante nLog.error

Detalles de implementación

Después de que el registro inbound se guarda, afterSubmit extrae el XML del campo `custrecord_psg_ei_inbound_content`, lo parsea para obtener datos clave, genera un nombre de archivo y lo guarda en una carpeta específica del File Cabinet. Luego, crea o actualiza un registro `customrecord_lmry_inbound_fields` vinculando la fecha, el proveedor y el archivo XML guardado.

Generado automáticamente 2026-01-23 17:57 UTC

SUITELET · INBOUND MÉXICO - VISUALIZADOR DE PDF

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a usuarios autorizados visualizar archivos PDF o texto almacenados en NetSuite mediante un enlace seguro.

Flujo de Ejecución

GET

Parámetros de Entrada

Lógica POST

Procesa la solicitud GET cargando el archivo indicado y devolviéndolo según su tipo y codificación

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: Redirige a la URL del archivo cuando el tipo es FREEMARKER, XMLDOC o PLAINTEXT y la codificación es ISO-8859-1; de lo contrario escribe el contenido directamente

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Proporciona la API para crear formularios (no utilizado en este Suitelet)
N/search	Disponible para búsquedas, aunque no se emplea en la lógica actual
N/runtime	Obtiene información del usuario y verifica la característica SUBSIDIARIES
N/record	Incluido por convención, no se usa directamente
N/redirect	Realiza redirecciones a URLs externas cuando corresponde
N/file	Carga el archivo a partir del idcontenido y accede a su contenido y metadatos

Resumen Técnico

Carga el archivo por ID, verifica tipo y codificación y lo entrega como PDF o texto, redirigiendo a la URL del archivo cuando la codificación es ISO-8859-1.

SUITELET · VISUALIZADOR DE XML INBOUND FORMATEADO

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a usuarios y sistemas visualizar de forma legible el XML inbound asociado a un registro en NetSuite.

Flujo de Ejecución

GET

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_id_message	Message	INLINEHTML	Muestra mensaje estático de acceso no permitido

Lógica POST

Recupera el XML almacenado en un registro inbound, lo formatea para legibilidad y lo devuelve al cliente; en caso de excepción muestra un formulario de error

Búsquedas

Obtener el contenido XML del registro inbound

Filtros: Tipo customrecord_psg_ei_inbound_edoc, id igual a inboundRecordId, columna custrecord_psg_ei_inbound_content

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Devuelve XML formateado como texto plano; muestra formulario de error si falla la búsqueda o el parseado

Dependencias

NOMBRE	PROPÓSITO
N/search	Realizar lookup del campo XML en el registro inbound
N/encode	Disponible pero no utilizado directamente en la lógica actual
N/ui/serverWidget	Construir formulario de error con campo INLINEHTML
N/url	Disponible para generación de URLs, no usado en este script
N/xml	Parsear y serializar el documento XML para formatearlo

Resumen Técnico

Recupera el XML mediante search.lookupFields, lo parsea y lo formatea con N/xml, devolviéndolo como texto plano; en caso de error muestra un formulario con mensaje estático.

INTERNALMODULE · UTILIDAD PARA CREAR CAMPO CKEDITOR Y PLANTILLA DE EJEMPLO

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: utils

Utilidad para crear campo CKEditor y plantilla de ejemplo

Dependencias: N/RUNTIME

Resumen Ejecutivo

Business: Facilita la incorporación de editores de texto enriquecido en formularios y provee una plantilla de mensaje lista para usar en comunicaciones. Technical: Exporta ckEditorField que genera dinámicamente HTML y scripts según la versión de NetSuite, y getExample que devuelve un HTML estático de ejemplo.

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener la versión de NetSuite en tiempo de ejecución

API pública

ckEditorField

Crea y configura un campo de texto enriquecido CKEditor en un formulario Suitelet, adaptando la versión de NetSuite.

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
form	Objeto SuiteScript Form donde se añadirá el campo	
id	Identificador único del campo	
content	HTML opcional que se inserta como contenedor del editor	

Efectos colaterales

- Añade un campo al formulario con configuración de CKEditor
- Inyecta scripts y estilos en el valor por defecto del campo

Retorna

Objeto Field añadido al formulario

getExample

Devuelve una cadena HTML con una plantilla de mensaje de ejemplo para correos electrónicos.

Retorna

Cadena HTML con la plantilla de ejemplo

Efectos colaterales

- **Type:** form modification; **Details:** ckEditorField añade un nuevo campo richtext al formulario proporcionado
- **Type:** script injection; **Details:** ckEditorField inserta scripts y estilos de CKEditor en el valor por defecto del campo

Detalles de implementación

El módulo exporta dos utilidades. ckEditorField recibe un formulario, un id y contenido opcional, construye la configuración del campo richtext, determina la versión de NetSuite mediante runtime.version y genera el HTML necesario para CKEditor5 (versión 2023 o superior) o CKEditor clásico (versiones anteriores). Configura ancho, alto y valor por defecto del campo antes de devolverlo. getExample simplemente devuelve una cadena HTML estática que representa una plantilla de correo electrónico con estilos y enlaces de redes sociales.

SUITELET · DETALLE DE INBOUND PE

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar el detalle de Inbound PE con sublistas de información relevante.

Flujo de Ejecución

GET

Lógica POST

La lógica principal se delega al handler que crea el formulario y las sublistas

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Muestra un formulario de error si la generación del formulario falla

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Crear formularios y campos UI
N/runtime	Obtener información del script en ejecución (ID y deployment)
N/redirect	Importado para posibles redirecciones (no usado en este archivo)
../Handlers/ LMRY_PE_InboundDetail_STLT_HNDL	Contiene la lógica de creación del formulario y sublistas

Resumen Técnico

Construye el formulario y sublistas mediante una clase handler y muestra la página, gestionando errores con un formulario de mensaje.

SUITELET · LR - PE INBOUND LOG DETAILS

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar de forma detallada los documentos inbound y su estado, facilitando la revisión de facturas y notas de crédito asociadas.

Flujo de Ejecución

GET

POST

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
id_internalid	Count	TEXT	valor calculado, solo lectura
id_tran_id	Internal ID	TEXT	solo lectura
id_name	Name File	TEXT	solo lectura
id_vendor	Vendor	TEXT	solo lectura
id_subsidiary	Subsidiary	SELECT	display inline, solo lectura
id_doc_status	E-Document Status	TEXT	solo lectura
id_tran_ur	Transaction	TEXT	contiene enlace HTML a factura o nota de crédito
audit_trial	Audit Trial	TEXT	solo lectura

Lógica POST

Recupera los IDs de inbound desde el registro de log, busca los documentos inbound y su estado, obtiene el último registro de auditoría con estado 'failed' y completa la sublista con información del documento, proveedor, subsidiaria, estado y enlace a la factura o nota de crédito

Búsquedas

Obtener datos del registro de log inbound

Filtros: lookupFields en customrecord_lmry_pe_ei_inbound_po por ID logid

Retorna: Campos custrecord_lmry_pe_inbound_subsi_po, custrecord_lmry_pe_inbound_trans_po y custrecord_lmry_pe_inbound_bill_po

Buscar los registros inbound edoc asociados

Filtros: search en customrecord_psg_ei_inbound_edoc con filtro internalid anyof lista de IDs inbound

Retorna: internalid, name, referencia, estado, vendor y tipo de transacción

Obtener el último audit trail por inbound con estado 'failed'

Filtros: search en customrecord_psg_ei_audit_trail filtrando custrecord_psg_ei_audit_inbound_edoc anyof lista de IDs inbound, ordenado por created descendente

Retorna: custrecord_psg_ei_audit_inbound_edoc y custrecord_psg_ei_audit_details

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Genera una página con una sublista estática; incluye enlaces a facturas o notas de crédito

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener parámetros del script y verificar si la característica Subsidiaries está activa
N/log	Registrar información de depuración y errores
N/search	Ejecutar búsquedas y lookupFields sobre registros personalizados
N/record	Importado pero no utilizado directamente en este script
N/ui/ serverWidget	Crear el formulario y la sublista que se muestra al usuario
N/url	Construir URLs a facturas (vendorbill) o notas de crédito (vendorcredit) para los enlaces de la sublista

Resumen Técnico

Construye un formulario con una sublista estática, consulta registros mediante búsquedas y lookup, y muestra enlaces a los documentos usando N/url.

CLIENTSCRIPT · VALIDACIÓN Y CREACIÓN DE REGISTRO INBOUND EN FORMULARIO

SuiteScript 2.1 · Módulos: 5

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

El script impide el guardado del formulario si faltan datos obligatorios o si no se selecciona ninguna transacción, y crea automáticamente un registro inbound con la información proporcionada.

Lifecycle Hooks

pageInit

Inicializa el idioma del script según el parámetro del script y verifica si la característica de subsidiarias está activa.

Lógica: Obtiene el parámetro LANGUAGE, lo recorta a dos caracteres y llama a setLanguage para asegurar un idioma soportado.

fieldChanged

Se ejecuta al cambiar cualquier campo, pero no realiza ninguna acción adicional.

Lógica: Retorna true para permitir la continuación del flujo.

validateField

Valida individualmente un campo cuando su valor cambia, pero siempre permite la edición.

Lógica: Retorna true sin aplicar validaciones específicas.

saveRecord

Al guardar el registro verifica campos obligatorios, crea un registro inbound con los documentos seleccionados y actualiza el estado del formulario.

Lógica: Obtiene valores de estado, subsidiaria, fechas y lista de transacciones. Si el estado está vacío valida que subsidiaria, fecha desde y fecha hasta estén completos y muestra alertas en caso contrario. Si el estado tiene valor, recorre la sublista para obtener los documentos marcados; si ninguno está seleccionado muestra alerta. Crea un registro customrecord_lmry_pe_ei_inbound_po, asigna subsidiaria, fechas, lista de transacciones, contador, usuario y estado, lo guarda y escribe el id resultante en custpage_state.

Validaciones

- **custpage_lmry_subsidiari:** debe contener un valor cuando la característica de subsidiarias está activa

Acción si falla: muestra alerta y bloquea guardado

- **custpage_date:** no puede estar vacío ni ser cero

Acción si falla: muestra alerta y bloquea guardado

- **custpage_date_2:** no puede estar vacío ni ser cero

Acción si falla: muestra alerta y bloquea guardado

- **custpage_sublista.id_apply:** al menos una fila debe estar marcada

Acción si falla: muestra alerta y bloquea guardado

Salida: actualiza custpage_state con el id del registro inbound creado y genera un registro inbound en la base de datos

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
custpage_state	-	escritura	se actualiza con el id del registro inbound creado al guardar
custpage_lmry_subsidiari	-	lectura validación	se valida como obligatorio cuando la característica de subsidiarias está activa
custpage_date	-	lectura validación	debe contener una fecha válida antes de permitir el guardado
custpage_date_2	-	lectura validación	debe contener una fecha válida antes de permitir el guardado
id_apply	custpage_sublista	validación	indica si la fila está seleccionada para el proceso; al menos una debe estar marcada
id_tran	custpage_sublista	lectura	valor del documento que se agrega a la lista de transacciones del registro inbound

Validaciones

UBICACIÓN	CAMPO	CRITERIO	ACCIÓN	BLOQUEA
saveRecord	custpage_lmry_subsidiari	valor obligatorio cuando la característica de subsidiarias está activa	muestra alerta y devuelve false	Sí
saveRecord	custpage_date	no puede estar vacío ni ser cero	muestra alerta y devuelve false	Sí
saveRecord	custpage_date_2	no puede estar vacío ni ser cero	muestra alerta y devuelve false	Sí
saveRecord	custpage_sublista.id_apply	al menos una fila debe estar marcada	muestra alerta y devuelve false	Sí

Interacción con Usuario

alert: EL CAMPO SUBSIDIARIA ES OBLIGATORIO

Trigger: saveRecord cuando subsidiaria está vacía

alert: EL CAMPO FECHA DESDE ES OBLIGATORIO

Trigger: saveRecord cuando fecha desde está vacía

alert: EL CAMPO FECHA HASTA ES OBLIGATORIO

Trigger: saveRecord cuando fecha hasta está vacía

alert: NO HA SELECCIONADO NINGUNA TRANSACCIÓN

Trigger: saveRecord cuando ninguna fila está marcada

alert: NO HAY NINGUNA TRANSACCIÓN POR PROCESAR

Trigger: saveRecord cuando la sublista está vacía

alert: OCURRIÓ UN ERROR, NO SE PUEDE CONTINUAR CON EL PROCESO

Trigger: saveRecord en caso de excepción

Comportamiento del Script

VALIDA DATOS

Sí

BLOQUEA GUARDADO

Sí

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

Sí

Notas: Crea un registro inbound dinámico y actualiza el campo de estado del formulario

Dependencias

NOMBRE	PROPÓSITO
N/search	no se utiliza directamente en el código actual
N/currentRecord	acceso al registro del formulario en tiempo de ejecución
N/record	creación y guardado del registro customrecord_lmry_pe_ei_inbound_po
N/url	comentado; se usaba para generar enlaces de redirección
N/runtime	obtener parámetros del script, usuario actual y verificar la característica de subsidiarias

Resumen Técnico

En saveRecord se validan campos críticos, se recopilan los documentos seleccionados, se crea un registro custom dinámico y se actualiza el campo de estado; cualquier error muestra una alerta y bloquea el guardado.

SUITELET · REGISTRO DE INBOUND PE

SuiteScript 2.0

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios visualizar el estado de los procesos inbound de PE y acceder al detalle de cada transacción.

Flujo de Ejecución

GET

POST

No se procesa ninguna solicitud POST

Flujo: N/A

Genera: N/A

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_hidden	HIDDEN	TEXT	valor predeterminado "Hidden" y tipo de visualización oculto
custpage_id_message	MESSAGE	INLINEHTML	Muestra mensaje informativo con estilo y texto traducido
id_internalid	Internal ID	TEXT	Valor leído de la búsqueda, visible
id_subsidiary	Subsidiary	SELECT	Fuente "subsidiary", visualización en línea, solo si la característica de subsidiarias está activa
id_datefrom	Date From	TEXT	Valor de fecha de inicio del proceso
id_dateto	Date To	TEXT	Valor de fecha de fin del proceso
id_employee	Employee	TEXT	Nombre del usuario que ejecutó el proceso
id_total	# Transactions	TEXT	Número de transacciones procesadas con texto descriptivo
id_transactions	Transactions	TEXTAREA	Muestra "Detail" o enlace al detalle según el estado
id_state	State	TEXTAREA	Indica el estado del proceso (Pending, Error, Complete)

Lógica POST

Ejecuta una búsqueda de registros de log inbound, recorre los resultados y llena la sublista con la información del proceso, generando enlaces al detalle cuando el proceso está completo

Búsquedas

Obtener los logs de procesos inbound con estado definido

Filtros:

Retorna: Internal ID, usuario, subsidiaria, transacciones, estado, fecha de creación, conteo, fechas de rango

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

No

ELIMINA REGISTROS

No

REDIRIGE

No

ENVÍA NOTIFICACIONES

No

Notas: Muestra una vista de los procesos inbound y permite abrir el detalle mediante enlaces

Dependencias

NOMBRE	PROPÓSITO
N/runtime	Obtener parámetros del script y verificar si la característica de subsidiarias está activa
N/log	Registrar información de depuración y errores
N/search	Ejecutar la búsqueda de registros de log inbound
N/record	Importado pero no utilizado en la lógica actual
N/ui/serverWidget	Construir el formulario, campos y sublista
N/url	Resolver la URL del script de detalle para generar enlaces

Resumen Técnico

Construye un formulario con una sublista estática mediante serverWidget y rellena los datos mediante una búsqueda sobre el registro customrecord_lmry_pe_ei_inbound_po.

Generado automáticamente 2026-01-23 17:57 UTC

MAP/REDUCE · LMRY_PE_INBOUND_MPRD

Pipeline de procesamiento por lotes (SuiteScript Map/Reduce).

Procesos masivos, búsquedas y operaciones sobre registros.

Metadatos

Tipo: MapReduce

SuiteScript: 2.1

LMRY_PE_Inbound_MPRD

Resumen Ejecutivo

{'business_summary': Markup('Convierte documentos electrónicos inbound de Perú en facturas o notas de crédito de proveedor, actualiza el registro de control y notifica al proveedor cuando corresponde.'), 'technical_summary': Markup('Lee los datos inbound, los parsea con una plantilla y un plugin de validación, crea o transforma transacciones (Vendor Bill / Vendor Credit), actualiza registros custom, adjunta PDFs, envía correos y registra auditoría.')}

getInputData

Obtiene los IDs de documentos inbound desde el registro customrecord_lmry_pe_ei_inbound_po usando el parámetro custscript_lmry_pe_inb_log_id y los devuelve como un arreglo de objetos JSON.

Filtros

CAMPO	OPERADOR	VALOR
	anyof	paramIdDocuments

Columnas

NOMBRE	ETIQUETA
custrecord_lmry_pe_inbound_trans_po	

map

Para cada registro inbound se carga la plantilla y el registro de origen, se asigna o actualiza el vendor code, se determina si es nota de crédito o factura, se invoca la función de conversión correspondiente (convertInboundEDoc o convertInboundBillCredit), se actualiza el estado del registro inbound, se envía email al vendor cuando corresponde y se escribe el resultado (clave = id del registro inbound, valor = objeto con éxito, id de factura o crédito y bandera de método de pago).

Campos utilizados

- custrecord_psg_ei_inbound_content
- custrecord_psg_ei_inbound_vendor
- custrecord_psg_ei_inbound_template
- custrecord_psg_ei_inbound_status
- custrecord_psg_ei_inbound_parsed_content
- custrecord_psg_ei_inbound_refnum
- custrecord_psg_ei_inbound_po
- custentity_lmry_pe_receive_mail
- email

Operaciones en registros (map)

OPERACIÓN	RECORD	CAMPOS
load	customrecord_psg_ei_inbound_edoc	
submitFields	customrecord_psg_ei_inbound_edoc	custrecord_psg_ei_inbound_status
transform	purchaseorder -> vendorbill	
create	vendorbill	
create	vendorcredit	
save	vendorbill vendorcredit	
attach	file -> vendorbill	
load	vendor	
load	customrecord_lmry_pe_ei_inbound_po	
emailSend	email	

summarize

Acciones finales

- **Type:** load; **Recordtype:** customrecord_lmry_pe_ei_inbound_po; **Fields:** {}
- **Type:** setValue; **Recordtype:** customrecord_lmry_pe_ei_inbound_po; **Fields:** {'custrecord_lmry_pe_inbound_bill_po': Markup('JSON string con resultados'), 'custrecord_lmry_pe_inbound_state_po': Markup('2')}
- **Type:** save; **Recordtype:** customrecord_lmry_pe_ei_inbound_po; **Fields:** {}

Búsquedas utilizadas

ID	FILTROS	COLUMNAS
getInboundID	[Markup('internalid'), Markup('anyof'), Markup('paramIdDocuments')]	custrecord_lmry_pe_inbound_trans_po
getSetup	[Markup('custrecord_lmry_pe_setup_subsidiary'), Markup('equalto'), Markup('subsidiary')], AND, [Markup('isinactive'), Markup('is'), Markup('F')]	custrecord_lmry_pe_setup_item, custrecord_lmry_pe_message, custrecord_lmry_pe_message_sender, custrecord_lmry_pe_message_subject
getCreatedFromTransaction	[Markup('trandid'), Markup('is'), Markup('createdFrom')]	internalid, entity
getBillsUsingReferenceNumber	[Markup('entity'), Markup('is'), Markup('vendorId')], AND, [Markup('trandid'), Markup('is'), Markup('refNum')], AND, [Markup('mainline'), Markup('is'), Markup('T')]	trandid
searchItemsByVendorCode	[Markup('vendorcode'), Markup('is'), Markup('vendorCodeValues (OR)')], AND, [Markup('othervendor'), Markup('is'), Markup('entity')]	othervendor, vendorcode
convertInboundBillCredit - bill search	[Markup('entity'), Markup('is'), Markup('entityId')], AND, [Markup('trandid'), Markup('contains'), Markup('referencedInvoiceId')], AND, [Markup('mainline'), Markup('is'), Markup('T')]	internalid, trandid, fxamount
convertInboundBillCredit - bill lookup		trandid
convertInboundBillCredit - bill amount lookup		fxamount

Operaciones con registros

OPERACIÓN	RECORD	CAMPOS
customrecord_psg_ei_inbound_edoc		
purchaseorder		
vendorbill		
vendorcredit		
vendor		
customrecord_lmry_pe_ei_inbound_po		
customrecord_lmry_mx_inb_bill		
customrecord_psg_ei_template		
file		
email		

Side effects

- Creación o transformación de Vendor Bill o Vendor Credit en NetSuite.
- Actualización del estado del registro inbound (customrecord_psg_ei_inbound_edoc) a CONVERTING, CONVERTED o FAILED.
- Adjunto del PDF del documento inbound al Vendor Bill creado.
- Envío de correo electrónico al vendor cuando está configurado para recibir notificaciones.
- Registro de auditoría mediante auditManager.logConverted y auditManager.logConversionFailed.
- Actualización del registro de control customrecord_lmry_pe_ei_inbound_po con los resultados de la conversión.

Generado automáticamente 2026-01-23 17:56 UTC

SUITELET · FORMULARIO INBOUND PE SUITELET

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios filtrar y aplicar acciones sobre registros inbound, y ejecutar procesos de integración mediante Map/Reduce.

Flujo de Ejecución

GET

POST

Procesa la acción del formulario y redirige o ejecuta el proceso de Map/Reduce

Flujo: Obtiene `custpage_status`; si es `falsy` redirige al mismo Suitelet con los parámetros de filtro generados por el handler. Si tiene valor, extrae `custpage_log_id` y `custpage_subsidiary`, ejecuta el Map/Reduce correspondiente y luego redirige al Suitelet de logs.

Genera: Redirección a la misma Suitelet o ejecución de Map/Reduce y redirección al Suitelet de logs

Parámetros de Entrada

Lógica POST

El Suitelet delega la creación y manejo del formulario a la librería `LibraryHandler`; en POST decide entre redirigir al mismo Suitelet con parámetros de filtro o ejecutar un proceso de Map/Reduce y luego redirigir al Suitelet de logs.

Operaciones

TIPO	TARGET	PROPÓSITO
redirect	Suitelet (mismo script)	Volver a cargar el formulario con los filtros seleccionados
executeMapReduce	Map/Reduce script definido en <code>LibraryHandler</code>	Procesar el log seleccionado para la subsidiaria indicada
redirect	Suitelet de logs	Mostrar el resultado del procesamiento al usuario

Comportamiento del Suitelet

CREA REGISTROS

No

ACTUALIZA REGISTROS

Sí

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: El script actualiza datos mediante el proceso de Map/Reduce y realiza redirecciones según la lógica del formulario

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construir y renderizar formularios y campos
N/runtime	Obtener información del script y del entorno de ejecución
N/redirect	Redirigir al usuario a otro Suitelet o a la misma con parámetros
../Handlers/ LMRY_PE_Inbound_STLT_HNDL	Contiene la lógica de creación de formularios, sublistas y ejecución de Map/Reduce

Resumen Técnico

Construye dinámicamente formularios con serverWidget, gestiona la lógica mediante una librería externa y controla el flujo mediante redirecciones y ejecución de Map/Reduce.

PLUG-IN IMPLEMENTATION · INYECCIÓN DE DATOS PERSONALIZADOS PARA GENERACIÓN DE DOCUMENTOS ELECTRÓNICOS

SuiteScript 2.0

Documentación técnica del plug-in.

Metadatos

Tipo: PlugInImplementation

SuiteScript: 2.0

Inyección de datos personalizados para generación de documentos electrónicos

Resumen Ejecutivo

Este Plug-in provee una fuente de datos personalizada que extrae información de un XML de factura electrónica, calcula conceptos de detracción, asigna códigos de impuesto y devuelve los datos estructurados para el motor de renderizado de NetSuite.

Deployment / Enablement

Se habilita al asignar este Plug-in como implementación de DataSource en la configuración del proceso de generación de facturas electrónicas.

Entry points

NOMBRE	DESCRIPCIÓN	INPUTS	OUTPUTS
inject	Genera la fuente de datos personalizada que será consumida por el motor de renderizado; extrae valores del XML, determina moneda, fechas, conceptos de detracción, códigos de impuesto y referencias a documentos origen.	Objeto params con inboundEDocRec (registro de documento entrante), templateRec (registro de plantilla) y entityRec (registro de entidad proveedora).	Objeto que contiene un array customDataSources con un DataSource de tipo OBJECT (alias "item_type") que incluye todos los campos calculados y una cadena con el método de pago.

Búsquedas

PROPÓSITO	FILTROS	RETORNO
Obtener el nombre interno de la moneda a partir del símbolo.	symbol = código de moneda del XML.	Nombre de la moneda (campo name).
Determinar la configuración de punto de impuesto (accrual o payment) según el nexus de Perú.	custrecord_4601_wts_nexus = id del nexus PE.	Valor del campo custrecord_4601_wts_purctaxpoint.
Obtener la tasa y tipo de impuesto de detracción a partir del código WHT.	ID del registro customrecord_4601_witaxcode.	Campos custrecord_4601_wtc_rate y custrecord_4601_wtc_witaxtype.
Obtener el ítem de compra asociado al tipo de detracción.	ID del registro customrecord_4601_witaxtype.	Campo custrecord_4601_wtt_purcitem.
Cargar la configuración de la subsidiaria para la generación de documentos inbound.	custrecord_lmry_pe_setup_subsidiary = id de la subsidiaria, isinactive = F	department, class, location (texto), taxcode0, taxcode10, taxcode18, tabla de detracción
Obtener el nexus interno de Perú para usar en la búsqueda de configuración tributaria.	country = PE	internalid del nexus.
Buscar la factura de proveedor (Vendor Bill) que corresponde al número de documento de referencia en una nota de crédito.	type = VendBill, tranid = número de referencia, subsidiary = id de la subsidiaria (opcional)	internalid de la Vendor Bill.

Dependencias

NOMBRE	PROPÓSITO
N/render	Crear y devolver el DataSource para el motor de renderizado.
N/log	Registrar información de depuración y errores.
N/format	Formato de datos (no usado directamente en la lógica principal).
N/search	Ejecutar búsquedas y lookups de configuración tributaria y de subsidiarias.
N/https	Comunicación con los servicios REST de SUNAT (consultaSunat, obtenerToken).
N/runtime	Obtener información del script, usuario actual y parámetros de ejecución.
N/xml	Parsear y navegar el XML de la factura electrónica.
N/error	Crear y lanzar errores personalizados.

Side effects

- Registra mensajes de depuración y errores en el log de NetSuite.
- Lanza errores personalizados que pueden detener la generación del documento.
- Realiza llamadas HTTP a la API de SUNAT (aunque no se invocan en el flujo principal).

Errores

UBICACIÓN	DESCRIPCIÓN	EXCEPCIÓN

Generado automáticamente 2026-01-23 17:54 UTC

PLUG-IN IMPLEMENTATION · INJECTS CUSTOM DATA SOURCE INTO INBOUND EDOC PROCESSING

SuiteScript 2.1

Documentación técnica del plug-in.

Metadatos

Tipo: PlugInImplementation

SuiteScript: 2.1

Injects custom data source into inbound eDoc processing

Resumen Ejecutivo

Provides an inject entry point that adds a custom data source to the inbound eDoc record during processing.

Deployment / Enablement

Activated when the inbound eDoc workflow calls the plugin's inject entry point.

Entry points

NOMBRE	DESCRIPCIÓN	INPUTS	OUTPUTS
inject	Adds a custom data source to the inbound eDoc processing context.	params object containing inboundEDocRec, templateRec, and entityRec records.	Object with a customDataSources array that includes a DataSource of type OBJECT, alias "custom", and an empty data object.

Dependencias

NOMBRE	PROPÓSITO
N/render	Create and return a custom data source for rendering.

INTERNALMODULE · CAPTURA Y PROCESAMIENTO DE E-DOCUMENTS ENTRANTES POR EMAIL

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Módulo: plugin

Captura y procesamiento de e-documents entrantes por email

Dependencias: NLAPIGETCONTEXT NLAPICREATEERROR NLAPISEARCHRECORD
NLAPILOGEXECUTION NLAPISENDEMAIL NLAPICREATERECORD NLAPISUBMITRECORD
NLAPICREATEFILE NLAPISUBMITFILE NLAPILOOKUPFIELD NLAPIRESOLVEURL
NLAPISUBMITFIELD NLOBJECTSEARCHFILTER NLOBJECTSEARCHCOLUMN

Resumen Ejecutivo

{'business_summary': Markup('Este plugin automatiza la captura de facturas y notas de crédito enviadas por correo, creando registros estructurados que facilitan su posterior procesamiento contable.'), 'technical_summary': Markup('Procesa el email, valida y extrae XML/PDF, crea un registro custom, asocia entidad (proveedor o cliente) mediante búsquedas, asigna una plantilla E-Document y gestiona errores enviando notificaciones a administradores.')}

Dependencias

NOMBRE	PROPÓSITO
nlapiGetContext	Obtener la configuración del contexto de ejecución, como el idioma
nlapiCreateError	Generar errores personalizados para detener el flujo y ser capturados
nlapiSearchRecord	Ejecutar búsquedas sobre empleados, proveedores, clientes, carpetas y plantillas
nlapiLogExecution	Registrar información de depuración y errores en el log de NetSuite
nlapiSendEmail	Enviar notificaciones por correo a administradores o destinatario configurado
nlapiCreateRecord	Crear el registro customrecord_psg_ei_inbound_edoc
nlapiSubmitRecord	Guardar el registro inbound y opcionalmente omitir validaciones obligatorias
nlapiCreateFile	Crear un archivo PDF en el File Cabinet
nlapiSubmitFile	Persistir el archivo PDF creado
nlapiLookupField	Obtener valores de campos de registros sin cargar el registro completo
nlapiResolveURL	Construir la URL de edición del registro inbound para incluirla en los correos de error
nlapiSubmitField	Actualizar el campo de adjunto PDF del registro inbound
nlobjSearchFilter	Definir criterios de filtrado para las búsquedas
nlobjSearchColumn	Definir columnas a devolver en las búsquedas

API pública

process

Punto de entrada del plugin; procesa el email entrante, extrae XML y PDF, crea el registro inbound, asocia proveedor o cliente, selecciona plantilla y maneja errores

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
email	Objeto de email provisto por NetSuite	

Efectos colaterales

- creación de registro customrecord_psg_ei_inbound_edoc
- creación de archivo PDF en el File Cabinet
- envío de correo de notificación en caso de error

Funciones internas

NOMBRE	ROL	USADO POR
retrieveFile	extrae el archivo XML o PDF válido de los adjuntos	process
getActiveAdmins	busca empleados activos con rol de administrador	sendEmail
sendEmail	envía correo de error a administradores o destinatario configurado	process
identifyVendor	localiza el proveedor mediante NIT y verifica que sea único	process
identifyCustomer	localiza el cliente mediante dominio y correo electrónico y verifica unicidad	process
updateXMLContent	elimina atributos xmlns del XML para evitar problemas de parsing	process
createPDFFile	crea y guarda el archivo PDF asociado al registro inbound	process
getInboundFolder	obtiene o crea la carpeta donde se almacenan los PDFs	createPDFFile
getFolderIdByName	busca el ID de una carpeta por nombre	getInboundFolder
getSubFolderId	busca el ID de una subcarpeta bajo una carpeta padre	getInboundFolder
createFolder	crea una nueva carpeta en el File Cabinet	getInboundFolder
getParentSubsidiaryId	obtiene el ID de la subsidiaria madre para determinar destinatario de notificaciones	sendEmail

Modelos de datos / estructuras

InboundEdocRecord

Registro customrecord_psg_ei_inbound_edoc que almacena la información del e-documento recibido

CAMPO	TIPO	DESCRIPCIÓN
custrecord_psg_ei_inbound_source	Origen del e-documento (valor constante 2 para email)	
custrecord_psg_ei_inbound_content	Contenido XML del documento	
custrecord_psg_ei_inbound_transtype	Tipo de transacción (Bill, BillCredit o SalesOrder)	
custrecord_psg_ei_inbound_vendor	ID interno del proveedor asociado (cuando corresponde)	
custrecord_psg_ei_inbound_customer	ID interno del cliente asociado (cuando corresponde)	
custrecord_psg_ei_inbound_template	ID de la plantilla E-Document seleccionada	
custrecord_psg_ei_attach_pdf	ID del archivo PDF adjunto	

INTERNALMODULE · ROUTER DE USER EVENT POR TIPO DE TRANSACCIÓN

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: factory

Router de User Event por tipo de transacción

Dependencias: N/LOG

Resumen Ejecutivo

{'business_summary': Markup('Este módulo permite a otros scripts obtener dinámicamente el handler de User Event adecuado según el tipo de documento, facilitando la reutilización y el mantenimiento del código.'), 'technical_summary': Markup('Expone la función getUserEventRouter que mapea tipos de transacción a rutas de módulos y los carga mediante require, devolviendo el handler o una cadena vacía.')}

Dependencias

NOMBRE	PROPÓSITO
N/log	Registro de mensajes y errores en el log de NetSuite

API pública

getUserEventRouter

Obtiene el handler de User Event correspondiente al tipo de transacción solicitado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
type	Tipo de transacción (por ejemplo vendorbill)	

Efectos colaterales

- Carga dinámicamente el módulo handler mediante require
- Registra errores en el log si ocurre una excepción

Retorna

Objeto del handler cargado o cadena vacía si no existe

Efectos colaterales

- Carga dinámica de módulos externos
- Escritura de mensajes de error en el log

Manejo de errores

UBICACIÓN	DESCRIPCIÓN	THROW
getUserEventRouter	Captura cualquier excepción durante la búsqueda y carga del handler	No lanza, solo registra el error en el log

Detalles de implementación

Recibe un tipo de transacción, busca la ruta del handler en un mapa interno, utiliza require para cargar el módulo de forma asíncrona y devuelve el handler o una cadena vacía cuando no hay coincidencia.

CLIENTSCRIPT · CARGA Y POBLADO DE CONFIGURACIÓN INBOUND POR SUBSIDIARIA

SuiteScript 2.1 · Módulos: 3

Eventos de cliente y validaciones en UI.

Resumen Ejecutivo

El script pre-llena automáticamente los campos de configuración inbound según la subsidiaria seleccionada, facilitando la captura de datos de impuestos, usuarios SAT y mensajes de email.

Lifecycle Hooks

pageInit

Inicializa el formulario, determina la versión de NetSuite para usar CKEditor o RTE, y carga la configuración inbound predeterminada cuando la característica de subsidiarias está desactivada.

Lógica: Comprueba la versión de NetSuite para seleccionar la librería de editor. Si la característica de subsidiarias está desactivada, fija la subsidiaria en 1 y ejecuta una búsqueda del registro `customrecord_lmry_pe_setup_inbound` filtrado por esa subsidiaria. Si existen resultados, asigna los valores de códigos de impuesto, departamento, clasificación, ubicación, ítem, usuario SAT, contraseña (ocultada con asteriscos), datos de email y tabla de detracción a los campos del formulario y rellena la sublista de detracción. Si no hay resultados, limpia los campos y vacía la sublista.

Salida: Actualiza múltiples campos del formulario y la sublista de detracción

fieldChanged

Reacciona al cambio del campo de subsidiaria, busca la configuración correspondiente y actualiza los campos del formulario.

Lógica: Si el campo cambiado es `custpage_subsidiary`, realiza una búsqueda del registro `customrecord_lmry_pe_setup_inbound` filtrado por la subsidiaria seleccionada. Cuando se encuentran resultados, asigna los mismos campos que en `pageInit`. Si no hay resultados, limpia los campos y la sublista.

Salida: Actualiza los mismos campos que `pageInit` según la subsidiaria seleccionada

saveRecord

Antes de guardar, captura el contenido del editor de email y serializa los datos de la sublista de detracción en JSON.

Lógica: Obtiene el contenido del editor (CKEditor o RTE) y lo guarda en `custfield_email`. Recorre la sublista `custpage_sublist_detraction`, construye un arreglo de objetos con concepto, código y código de retención, lo serializa a JSON y lo guarda en `custpage_detractiondata`.

Salida: Actualiza `custfield_email` y `custpage_detractiondata` con los datos actuales del formulario

Campos Controlados

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
custpage_subsidary	-	lectura escritura	Se establece en 1 cuando la característica de subsidiarias está desactivada; al cambiar, dispara la carga de configuración
custpage_taxcode0	-	escritura	Se rellena con el código de impuesto 0 obtenido del registro de configuración
custpage_taxcode10	-	escritura	Se rellena con el código de impuesto 10 obtenido del registro de configuración
custpage_taxcode19	-	escritura	Se rellena con el código de impuesto 18 (mapeado como 19) obtenido del registro de configuración
custpage_department	-	escritura	Se rellena con el valor de departamento extraído del JSON almacenado en el registro de configuración
custpage_class	-	escritura	Se rellena con la clasificación extraída del JSON almacenado en el registro de configuración
custpage_location	-	escritura	Se rellena con la ubicación extraída del JSON almacenado en el registro de configuración
custpage_item	-	escritura	Se rellena con el ítem configurado
custpage_user	-	escritura	Se rellena con el usuario SAT configurado
custpage_password	-	escritura	Se muestra la contraseña SAT enmascarada con asteriscos
custfield_email_sender	-	escritura	Se rellena con el remitente del mensaje de email configurado
custfield_email_subject	-	escritura	Se rellena con el asunto del mensaje de email configurado
custfield_email	-	escritura	Contiene el cuerpo del mensaje de email; se carga en el editor RTE o CKEditor

CAMPO	SUBLISTA	ROL	COMPORTAMIENTO
custfield_email_body	-	lectura escritura	Editor de texto enriquecido que muestra el contenido de custfield_email
custpage_detractiiondata	-	escritura	Almacena la tabla de detracción en formato JSON
custpage_sublist_detractiion	custpage_sublist_detractiion	lectura escritura	Sublista que muestra conceptos, códigos y códigos de retención de detracción; se rellena dinámicamente
custpage_sublist_conceptdetra	custpage_sublist_detractiion	escritura	Campo de concepto dentro de la sublista de detracción
custpage_sublist_codedetra	custpage_sublist_detractiion	escritura	Campo de código dentro de la sublista de detracción
custpage_sublist_whttaxcode	custpage_sublist_detractiion	escritura	Campo de código de retención dentro de la sublista de detracción

Comportamiento del Script

VALIDA DATOS

No

BLOQUEA GUARDADO

No

OCULTA CAMPOS

No

MODIFICA VALORES

Sí

MUESTRA ALERTAS

No

Notas: El script solo carga y guarda datos; no realiza validaciones de negocio ni muestra mensajes al usuario

Dependencias

NOMBRE	PROPÓSITO
N/search	Ejecutar búsquedas sobre el registro customrecord_lmry_pe_setup_inbound
N/runtime	Verificar la versión de NetSuite y la disponibilidad de la característica de subsidiarias
../Libraries/ LMRY_PE_Message_LBRY	Proveer un ejemplo de mensaje de email para cargar en el editor

Resumen Técnico

En pageInit y fieldChanged se busca el registro de configuración y se asignan los valores a los campos del formulario; al guardar, el contenido del editor y la sublista de detracción se serializan en JSON.

SUITELET · CONFIGURACIÓN INBOUND PE

SuiteScript 2.1

Procesos de UI y rutas Suitelet.

Resumen Ejecutivo

Permite a los usuarios definir la configuración inbound por subsidiaria, incluyendo códigos de impuesto, departamentos, ubicación y mensaje de notificación.

Flujo de Ejecución

GET

POST

Procesa los datos enviados y guarda la configuración

Flujo: Al recibir una solicitud POST, el Suitelet lee los parámetros del formulario, verifica si ya existe un registro de configuración para la subsidiaria seleccionada, actualiza ese registro o crea uno nuevo con los valores recibidos, guarda los cambios y redirige al mismo Suitelet para recargar la página.

Genera: Redirección al Suitelet

Parámetros de Entrada

Campos del Formulario

CAMPO	ETIQUETA	TIPO	COMPORTAMIENTO
custpage_subsiary	Subsidiary	SELECT	Obligatorio, opciones cargadas por búsqueda de subsidiarias de Perú
custpage_taxcode0	Tax Code 0	SELECT	Opciones cargadas por búsqueda de códigos de impuesto de Perú
custpage_taxcode10	Tax Code 10	SELECT	Opciones cargadas por búsqueda de códigos de impuesto de Perú
custpage_taxcode16	Tax Code 18	SELECT	Opciones cargadas por búsqueda de códigos de impuesto de Perú
custpage_department	Department	SELECT	Opciones cargadas si la característica Departments está activa
custpage_class	Class	SELECT	Opciones cargadas si la característica Classes está activa
custpage_location	Location	SELECT	Opciones cargadas si la característica Locations está activa
custpage_item	Item Generic	SELECT	Fuente de lista de ítems estándar
custpage_detractiiondata	Detraction Data	LONGTEXT	Oculto, almacena datos de la sublista en JSON
custfield_email_sender	Sender	SELECT	Opciones cargadas con empleados activos
custfield_email_subject	Subject	TEXT	Campo de texto para asunto del correo
custfield_email_body	Message Body	RICH TEXT (CKEditor)	Editor enriquecido para cuerpo del mensaje
custfield_email	Message Template	LONGTEXT	Oculto, almacena la plantilla del mensaje

Lógica POST

Crea o actualiza el registro customrecord_lmry_pe_setup_inbound con los valores enviados desde el formulario

Búsquedas

Buscar registro de configuración existente para la subsidiaria seleccionada

Filtros:

Retorna: internalid del registro encontrado

Operaciones

TIPO	TARGET	PROPÓSITO
load	customrecord_lmry_pe_setup_inbound	Obtener registro existente y actualizar sus campos
create	customrecord_lmry_pe_setup_inbound	Crear nuevo registro cuando no existe uno para la subsidiaria
setValue	registro de configuración	Asignar códigos de impuesto, departamentos, clase, ubicación, ítem, datos de detracción y mensaje
save	registro de configuración	Persistir los cambios en NetSuite
redirect	Suitelet	Recargar la página después de guardar

Comportamiento del Suitelet

CREA REGISTROS

Sí

ACTUALIZA REGISTROS

Sí

ELIMINA REGISTROS

No

REDIRIGE

Sí

ENVÍA NOTIFICACIONES

No

Notas: Utiliza CKEditor para el cuerpo del mensaje y almacena datos de la sublista en un campo LONGTEXT

Dependencias

NOMBRE	PROPÓSITO
N/ui/serverWidget	Construir y renderizar el formulario y sus campos
N/search	Ejecutar búsquedas para poblar listas de selección y localizar registros existentes
N/runtime	Verificar si ciertas características (departamentos, ubicaciones, clases, subsidiarias) están activas
N/record	Cargar, crear y guardar registros customrecord_lmry_pe_setup_inbound
../Libraries/LMRY_PE_Message_LBRY	Proveer campo CKEditor para el cuerpo del mensaje

Resumen Técnico

Muestra un formulario dinámico, procesa la entrada y crea o actualiza un registro customrecord_lmry_pe_setup_inbound, usando búsquedas y redirección al final.

INTERNALMODULE · TRADUCTOR DE ETIQUETAS INBOUND PE

Módulo compartido / servicios internos

Referencia funcional del módulo compartido.

Metadatos

Tipo: InternalModule

Lenguaje: JavaScript

Módulo: utils

Traductor de etiquetas Inbound PE

Resumen Ejecutivo

Este módulo centraliza las traducciones de todas las etiquetas usadas en la funcionalidad Inbound de Perú, permitiendo que otros scripts obtengan los textos en español o inglés. Expone la función `getTranslations` que recibe el código de idioma y devuelve el objeto de etiquetas traducidas, sin dependencias externas ni efectos colaterales.

API pública

`getTranslations`

Devuelve un objeto con todas las etiquetas traducidas al idioma solicitado

Parámetros

NOMBRE	DESCRIPCIÓN	TIPO
language	Código del idioma (es o en)	

Retorna

Objeto cuyas claves son los identificadores de etiqueta y los valores son los textos traducidos

Modelos de datos / estructuras

Translations

Mapa de etiquetas traducidas por idioma

Campos: Cada clave corresponde a un identificador de etiqueta y su valor es una cadena de texto en el idioma seleccionado

Detalles de implementación

La función getTranslations selecciona el conjunto de etiquetas correspondiente al idioma indicado y lo devuelve; si el idioma no está soportado, devuelve el conjunto en inglés.

Generado automáticamente 2026-01-23 17:55 UTC

AUDIT REPORT · PROYECTOS INBOUND.ZIP

107 files · 2026-01-23 18:55 UTC

File review summary with hotspots, overviews, and fix plans.

Audit Metrics

Audits: 106

High hotspots: 258

PROYECTOS INBOUND/Inbound_CL/Client/ LMRY_CL_Inbound_CLNT.js

Script type: ClientScript

Audit score (0-10): 8

Final score (0-100): 80

API version: 2.1

El script está declarado correctamente como SuiteScript 2.1, usa sintaxis moderna y está bien modularizado. Presenta pequeñas áreas de mejora como manejo de nulos y eliminación de logs de depuración.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	medium	Las funciones fieldChanged y saveRecord asumen que 'handler' está inicializado; si pageInit falla o no se ejecuta, se producirá un error de referencia.	Agregar validación: if (!handler) { handler = new lbryPayment.Clnthandler(); } o lanzar error controlado.	22, 26	-
debug_artifact	low	Presencia de console.log(handler) en código de producción.	Eliminar o envolver en condición de entorno de desarrollo.	18	-
governance_risk	low	Iteración sobre todas las líneas del sublist sin límite; en sublists muy extensas puede generar latencia en el cliente.	Considerar paginación o limitar la operación a rangos visibles, o usar batch updates si NetSuite lo permite.	34, 37	-

Fix Plan

- Implementar manejo de errores y validaciones de 'handler' antes de usarlo.
- Eliminar logs de depuración antes de mover a producción.
- Optimizar toggleCheckBoxes para grandes sublists mediante técnicas de batch o limitación de rango.

PROYECTOS INBOUND/Inbound_CL/Client/ LMRY_CL_Inbound_Detail_CLNT.js

Script type: ClientScript

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.1

Script declarado como SuiteScript 2.1 pero mezcla APIs de SuiteScript 1.0 (nlapi*), lo que genera errores críticos de compatibilidad en tiempo de ejecución. Además, la función updateParent concentra demasiada lógica (validaciones, construcción de payload, interacción UI) violando SRP y generando un entry point inflado. Se detectan prácticas de depuración (console.log) y uso de magic strings que dificultan el mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Uso de APIs de SuiteScript 1.0 (nlapiSetLineItemValue, nlapiGetLineItemCount, nlapiGetLineItemValue) dentro de un script declarado como 2.1. Estas llamadas no existen en el runtime GraalVM y provocarán errores de runtime.	Reemplazar todas las llamadas nlapi* por sus equivalentes 2.x (record, currentRecord, sublist APIs) o migrar el script a 1.0 si se requiere esas funciones.	216, 268, 286, 357, 360, 363, 366, 369, 372, 374	-
fat_entry_point	high	La función updateParent contiene más de 150 líneas, mezcla validaciones de negocio, construcción de objeto de datos, lógica de UI y llamadas a nlapi*.	Dividir la lógica en módulos: validación de datos, construcción del payload, y comunicación con la ventana padre. Cada módulo debe ser una función pura con una única responsabilidad.	229, 386	-
code_smell	medium	Variables globales mutables (p.ej., isNotRunning, itemValues, setupData) son usadas para controlar flujo y estado entre callbacks.	Encapsular estado dentro de objetos o usar closures; evitar mutaciones globales.	16 - 24	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
debug_artifact	low	Presencia de numerosos console.log en código de producción.	Eliminar o envolver los logs bajo una bandera de depuración.	61, 120, 124, 126, 133, 138, 146, 150, 155, 162, 165, 170, 176, 182, 186, 190, 194, 198, 202, 206, 210, 214, 218, 222, 226, 230, 236, 240, 244, 248, 252, 256, 260, 264, 268, 272, 276, 280, 284, 288, 292, 296, 300, 304, 308, 312, 316, 320, 324, 328, 332, 336, 340, 344, 348, 352, 356, 360, 364, 368, 372, 376, 380, 384	-
magic_string	low	Uso de strings literales para IDs de campos y sublistas en todo el código.	Definir constantes descriptivas para IDs de campos/sublistas y reutilizarlas.	3 - 6, 65, 71 - 1111	-

Fix Plan

- Reemplazar todas las llamadas nlapi* por APIs 2.x (record, currentRecord).
- Dividir updateParent en funciones: validación, construcción de payload, comunicación con ventana padre.
- Eliminar console.log y usar un logger controlado por entorno.
- Externalizar IDs de campos y sublistas a constantes.
- Implementar pruebas unitarias para cada módulo después del refactor.

PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_CLNT_HNDL.js

Script type: UserModule.js

Audit score (0-10): 5

Final score (0-100): 55

API version: 2.1

El script funciona bajo SuiteScript 2.1 pero presenta graves violaciones de SRP, métodos inflados (fieldChanged > 200 líneas) y varios errores lógicos que comprometen mantenibilidad y seguridad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	Método fieldChanged supera los 200 líneas, contiene múltiples bucles anidados, lógica de UI y de negocio mezclada, y uso de variables globales (window).	Dividir la lógica en módulos de servicio (ej. PopupService, UrlBuilder, ValidationService) y delegar desde fieldChanged.	103, 254	-
anti_pattern	high	La clase agrupa responsabilidades de UI, validación, búsqueda, creación de registros y manejo de traducciones, violando el Principio de Responsabilidad Única (SRP).	Refactorizar separando la lógica en clases o módulos independientes (Validator, RecordLogger, SetupProvider, UIHandler).	36, 892	-
logic_error	medium	Se formatea el campo subsidiary como fecha usando format.Type.DATE, lo que genera valores incorrectos en la URL.	Obtener el valor del subsidiary directamente sin formatearlo como fecha.	906, 909	-
security_issue	medium	Los parámetros de la URL se construyen a partir de valores del registro sin sanitización ni validación de tipo.	Validar y sanitizar todos los valores antes de incluirlos en params; usar encodeURIComponent si corresponde.	916, 920	-
obsolete_style	low	Uso de 'var' en un script 2.1 donde 'let' y 'const' son preferidos.	Reemplazar 'var' por 'let' o 'const' según corresponda.	850, 866	-

Fix Plan

- Refactorizar fieldChanged en varios servicios pequeños.
- Crear una capa de validación independiente para separar lógica de negocio de UI.
- Implementar manejo de errores centralizado en lugar de console.log/alert dispersos.
- Reemplazar 'var' por 'let/const' y aplicar linting para consistencia de estilo.
- Añadir pruebas unitarias para los métodos de validación y creación de logs.

PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_Detail_STLT_HNDL.js

Script type: UserModule.js

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

El script está escrito en SuiteScript 2.1 y aprovecha funcionalidades modernas (AMD, query, pagination). Sin embargo, la clase LibraryHandler concentra demasiada lógica (UI, búsquedas, reglas de negocio) violando SRP, contiene errores lógicos y prácticas de logging inadecuadas que aumentan la complejidad y el riesgo de mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
logic_error	high	Condición sin paréntesis: `if (this.statusFinal == 4 && !bill this.statusFinal == 3)` produce evaluación inesperada por precedencia de operadores.	Agrupar la condición: `if ((this.statusFinal == 4 && !bill) this.statusFinal == 3)`.	550	-
logging_practice	medium	Uso de `log.error` para imprimir información de depuración.	Reemplazar por `log.debug` o `log.audit` según corresponda.	88 - 90	-
anti_pattern	medium	La clase maneja UI, lógica de negocio, acceso a datos y utilidades de conversión, violando el Principio de Responsabilidad Única (SRP).	Dividir la clase en módulos: un Service para lógica de negocio, un Repository para búsquedas, y un UIBuilder para la generación de formularios/ sub-listas.	40, 81, 122, 148, 170, 210, 244, 255, 370, 543, 714, 735, 762, 787, 808, 887, 914, 939, 989, 998, 1026, 1455, 1475	-
deprecated_api	low	Uso de `getRange(start, end)` sin objeto de opciones; la firma recomendada es `getRange({ start, end })`.	Actualizar a la forma moderna: `search.run().getRange({ start: 0, end: 1000 })`.	752, 1388, 1467	-
variable_declaration	low	Uso de `var` en lugar de `let` / `const` para `taxCodesObj`.	Reemplazar por `const taxCodesObj = search.create(...);`.	1443	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
hardcoded_strings	low	IDs de script y deployment están hardcodeados en `url.resolveScript`.	Obtener los IDs mediante parámetros de script o constantes configurables.	298 - 300	-

Fix Plan

- Refactorizar LibraryHandler en varios módulos (Service, Repository, UIBuilder).
- Implementar inyección de dependencias para facilitar pruebas unitarias.
- Reemplazar logs de error por niveles adecuados.
- Centralizar IDs de scripts y despliegues en parámetros de script.
- Actualizar todas las llamadas a `getRange` a la firma moderna.

PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_STLT_HNDL.js

Script type: UserModule.js

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa AMD correctamente, pero presenta graves problemas de arquitectura, seguridad y lógica. Funciones críticas son excesivamente largas, violan SRP y contienen código duplicado. Existe un bug crítico al usar splice() en la construcción de filtros, lo que genera búsquedas incorrectas. La generación del SOAP envelope concatena datos sin escape, exponiendo a inyección. Además, se usan hash MD5 débiles y falta validación de parámetros de entrada.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	Función con más de 200 líneas que combina lógica de UI, validación de licencias y generación de enlaces, violando el principio de responsabilidad única.	Dividir en sub-módulos: generación de formulario, carga de licencias y construcción de enlaces. Cada sub-función debe ser ≤ 60 líneas.	144, 422	-
spaghetti_code	high	Método extenso que mezcla iteración de datos, generación de URLs, actualización de sublist y cálculo de contadores.	Separar la lógica de transformación de datos, generación de URLs y actualización de la sublista en funciones independientes.	603, 788	-
bug_logic	critical	Uso de <code>filtersDetail = filtersDetail.splice(1);`</code> elimina todos los filtros excepto el primero, devolviendo un array con el elemento removido. El filtro resultante es incorrecto y la búsqueda falla o devuelve resultados inesperados.	Reemplazar por <code>filtersDetail = filtersDetail.slice(1);`</code> o construir el array sin el primer elemento de forma segura.	1199	-
bug_logic	critical	Mismo error de <code>splice(1)`</code> que en <code>getBills</code> , generando filtros incorrectos para créditos.	Aplicar la corrección indicada en <code>getBills</code> .	1267	-
security_issue	high	Construcción del SOAP envelope mediante interpolación de variables (<code>`\$ {...}`</code>) sin escape XML, lo que permite inyección de contenido malicioso si algún parámetro proviene del usuario.	Escapar todos los valores con <code>`xml.escape`</code> o usar un builder de XML seguro antes de concatenar.	1079, 1092	-
security_issue	medium	Uso de algoritmo MD5 para generar token de autenticación, considerado criptográficamente débil.	Reemplazar MD5 por SHA-256 o algoritmo recomendado por el proveedor del servicio.	1075, 1076	-
security_issue	medium	No se valida ni sanitiza los parámetros de entrada (<code>`params.subsidiary`</code> , <code>`params.startdate`</code> , <code>`params.enddate`</code> , <code>`params.filter`</code>).	Implementar validación estricta (tipo, rango, formato) antes de utilizarlos.	58, 59, 62, 63	-
bad_naming	low	Variables <code>`anysubsidiary`</code> , <code>`licenses`</code> y <code>`subsidiaries`</code> usan nombres genéricos y a veces contradictorios.	Renombrar a <code>`hasActiveSubsidiary`</code> , <code>`licenseList`</code> , <code>`availableSubsidiaries`</code> .	76, 78, 79	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	low	IDs de scripts y despliegues están hardcodeados en varios lugares.	Mover a Script Parameters o a un archivo de configuración.	966, 967, 970, 971	-
style	low	Uso extensivo de `let` donde `const` sería más apropiado.	Reemplazar por `const` cuando la variable no se reasigna.	76, 78, 79, 84 - 90	-

Fix Plan

- Refactorizar createFilterForm, loadSublist y getResponse en módulos de servicio separados.
- Corregir el uso de splice() por slice() en la construcción de filtros.
- Sanitizar todos los valores interpolados en el SOAP envelope.
- Reemplazar MD5 por SHA-256 para la generación de tokens.
- Implementar validación y saneamiento de todos los parámetros de entrada.
- Utilizar const en lugar de let cuando la variable no cambia.
- Externalizar IDs de scripts y despliegues a parámetros de script.
- Agregar paginación (runPaged) en búsquedas que pueden superar 1000 registros.

PROYECTOS INBOUND/Inbound_CL/LMRY_CL_Inbound_XML_STLT.js

Script type: Suitelet

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

Suitelet escrito en SuiteScript 2.1 con arquitectura monolítica, falta de validación de entrada y riesgos de seguridad (SSRF). Uso inconsistente de var/const, manejo de codificación incorrecto y ausencia de controles de gobernanza. Necesita refactorización urgente.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
security_issue	critical	Se permite la descarga de cualquier URL provista por el parámetro 'xml' sin validación, lo que abre la posibilidad de Server-Side Request Forgery (SSRF).	Validar estrictamente el parámetro 'xml' contra una lista blanca de dominios permitidos y usar https.request con timeout y validación de certificado.	23 - 25	-
security_issue	high	No se verifica la existencia ni la validez del parámetro 'xml' antes de usarlo.	Comprobar que context.request.parameters.xml esté definido y cumpla con un patrón de URL permitido; devolver error controlado si no.	18	-
spaghetti_code	high	El manejador onRequest supera los 50 líneas y combina múltiples responsabilidades (validación, descarga, decodificación, parsing, formateo, sustitución y respuesta).	Dividir la lógica en módulos/ servicios independientes (p.ej., XmlFetcher, XmlEncoder, XmlFormatter, ResponseWriter) e inyectarlos mediante define.	15 - 18, 23 - 25, 27 - 29, 31, 32, 34 - 44, 46 - 53, 55 - 60	-
bad_naming	medium	Función con nombre en español y lógica de sustitución de caracteres codificados manualmente.	Renombrar a `normalizeXmlEncoding` y documentar su propósito; considerar usar una librería de decodificación en lugar de reemplazos manuales.	64 - 81	-
obsolete_style	low	Uso de `var` en lugar de `const` o `let` en un entorno ES2019+.	Reemplazar `var` por `const` cuando la variable no se reasigna.	43, 47	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	low	Se utiliza <code>`encode.convert`</code> con <code>`inputEncoding: BASE_64`</code> sobre un cuerpo que ya es texto, lo que puede producir resultados incorrectos.	Determinar la codificación real del cuerpo y usar <code>`encode.convert`</code> con los parámetros adecuados o confiar en la detección automática de NetSuite.	37 - 41	-
content_type_mismatch	low	Se envía XML con header <code>`Content-Type: text/plain; charset=utf-8`</code> en lugar de <code>`application/xml`</code> .	Cambiar a <code>`application/xml; charset=utf-8`</code> .	55	-

Fix Plan

- Refactorizar `onRequest` en servicios independientes para cumplir SRP.
- Implementar whitelist de dominios y validar la URL antes de `https.get`.
- Reemplazar la lógica manual de sustitución de caracteres por una decodificación adecuada.
- Usar `const/let` en todo el código y eliminar `var`.
- Ajustar el header `Content-Type` a `application/xml`.
- Agregar control de uso de gobernanza con `runtime.getCurrentScript().getRemainingUsage()` en caso de ampliación futura.

PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_Detail_STLT.js

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script declara SuiteScript 2.1 y usa arquitectura AMD correctamente, pero contiene errores críticos que impedirán su ejecución y vulnerabilidades de seguridad que requieren corrección inmediata.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se invoca log.error sin haber importado el módulo N/log, provocando ReferenceError en tiempo de ejecución.	Agregar "N/log" al arreglo de dependencias y al parámetro de la función, p.ej. function(serverWidget, runtime, redirect, librarySuitelet, lbryMail, log).	47	-
security_issue	high	Se envía el mensaje de error completo por email, exponiendo potencialmente información sensible del entorno.	Enviar solo un mensaje genérico y registrar el error completo en un log interno seguro.	51	-
bad_naming	medium	Variable 'params' declarada pero nunca utilizada.	Eliminar la declaración o utilizarla para validar los parámetros de entrada.	29	-
security_issue	medium	El Suitelet solo maneja peticiones GET; las peticiones POST son ignoradas sin respuesta ni validación.	Implementar lógica explícita para POST o devolver un mensaje de método no soportado.	53	-
hardcoded_path	low	Ruta del client script está hardcodeada como './LMRY_CL_Inbound_Detail_CLNT.js'.	Externalizar la ruta a un Script Parameter o constante configurada.	16	-
obsolete_style	low	Ruta del bundle contiene espacio ('Bundle 37714'), lo que puede generar problemas de resolución en algunos entornos.	Renombrar el bundle sin espacios o usar la API de SuiteBundler para referenciarlo.	13	-

Fix Plan

- Agregar la dependencia N/log y refactorizar el manejo de errores.
- Implementar validación y sanitización de los parámetros de entrada.
- Crear una rama de manejo para peticiones POST o devolver un mensaje de método no permitido.
- Externalizar rutas y constantes a Script Parameters para mayor flexibilidad.
- Revisar y limpiar variables no utilizadas.

PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_Log_Detail_STLT.js

Script type: Suitelet

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

El script está escrito en SuiteScript 2.1 y sigue la arquitectura AMD, pero presenta varios problemas de gobernanza, seguridad y mantenimiento que requieren refactorización.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
governance_risk	high	Se realizan operaciones de sublist dentro de un bucle sin verificar el consumo de gobernanza. Con gran número de registros puede superar el límite de uso y lanzar SSS_USAGE_LIMIT_EXCEEDED.	Agregar verificación de remaining usage dentro del bucle y dividir la carga en páginas o usar runPaged con límites menores.	137 - 171	-
security_issue	high	Se inserta HTML directamente en los valores de sublist usando datos del registro (po_name, bill_name, vendorcredit_name) sin sanitizar.	Escapar o sanitizar los valores antes de incluirlos en el HTML, o usar campos de tipo URL en lugar de HTML.	146 - 149, 151 - 154, 156 - 159	-
governance_risk	medium	Iteración sobre resultados de búsqueda sin control de uso de gobernanza y sin manejo de paginación más allá de 1000 registros.	Utilizar runPaged con verificación de remaining usage y procesar cada página de forma incremental.	206 - 247	-
anti_pattern	medium	Uso de search.run().getRange(0,1000) sin paginación y sin control de gobernanza.	Cambiar a runPaged y procesar por páginas, añadiendo checks de remaining usage.	279 - 291	-
bad_naming	low	Parámetro llamado 'Language' con mayúscula inicial, rompe convención camelCase.	Renombrar a 'language' y aplicar camelCase en todo el código.	255 - 263	-
obsolete_style	low	No se valida que la preferencia LANGUAGE exista antes de aplicar substring, lo que puede lanzar error si es undefined.	Agregar verificación de existencia y fallback a 'en' o 'es' antes de usar substring.	46	-

Fix Plan

- Implementar checks de remaining usage y dividir la carga en lotes.
- Sanitizar valores antes de generar HTML para evitar XSS.
- Refactorizar funciones largas (loadSublist, getRecords) en módulos de servicio.
- Externalizar valores estáticos (códigos de país, IDs) a constantes configurables.
- Cambiar getDocuments a runPaged y aplicar paginación.
- Añadir validación de preferencias de usuario y manejo de valores nulos.

PROYECTOS INBOUND/Inbound_CL/Module/LMRY_CL_Inbound_Log_STLT.js

Script type: Suitelet

Audit score (0-10): 7

Final score (0-100): 70

API version: 2.1

El script está escrito en SuiteScript 2.1 con buenas prácticas de modularidad y uso de paginación en búsquedas, pero presenta fallos críticos de dependencias no declaradas y algunas áreas de mejora en robustez y mantenibilidad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION
missing_dependency	critical	Se utilizan los objetos log.error y log.debug sin haber importado el módulo N/log en la definición de dependencias.	Agregar "N/log" a la lista de módulos de parámetro correspondiente (e.g., search, redirect, runtime, libran
null_reference	high	Se asume que runtime.getCurrentUser().getPreference('LANGUAGE') siempre devuelve una cadena; si es null, substring lanzará error.	Validar que el valor exista antes proporcionar un fallback (e.g., o runtime.getCurrentUser().getPr 'en_US');
hardcoded_id	low	Los IDs de script y deployment están escritos como literals.	Externalizar los IDs a Script Par al inicio del módulo.
mutable_variable	low	Se declara con let variables que nunca se reasignan; podrían ser const.	Reemplazar let por const donde
potential_xss	low	Se inserta HTML en un campo de tipo TEXT para crear un enlace.	Escapar o validar el valor antes campo de tipo INLINEHTML si

Fix Plan

- Agregar la dependencia N/log para evitar errores de referencia.
- Implementar validaciones de null para preferencias de usuario.
- Convertir IDs de script y deployment en parámetros configurables.
- Reemplazar let por const en variables inmutables.
- Sanitizar valores antes de generar HTML en sublist.

PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_MPRD.js

Script type: Map/Reduce

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script usa SuiteScript 2.1 pero presenta graves problemas de arquitectura, gobernanza y seguridad. Duplicación de módulos, búsquedas y cargas de registros dentro de bucles, falta de retorno en caso de error y construcción insegura de SOAP hacen que el código sea poco mantenible y propenso a fallos en producción.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	high	Módulo "N/format" importado dos veces y asignado al mismo parámetro "format", provocando sombra de variable y redundancia.	Eliminar la importación duplicada y mantener un único parámetro con nombre descriptivo.	9, 14, 19	-
spaghetti_code	high	En el bloque catch no se devuelve ningún valor, lo que hace que la función retorne undefined y falle el Map/Reduce.	Agregar `return []` al final del catch o re-throw después de registrar el error.	23, 52, 55, 69	-
governance_risk	high	Se ejecutan búsquedas (`getSetup`) y llamadas a `getDocumentType` por cada registro procesado.	Cachear resultados por subsidiaria y documento; ejecutar búsquedas fuera del bucle.	85, 126, 131, 133	-
performance	medium	Se verifica `runtime.isFeatureInEffect` dentro del bucle de ítems.	Evaluar las características una sola vez antes del bucle y reutilizar los valores.	431 - 433	-
governance_risk	high	Carga de registro de ítem (`record.load`) dentro de un bucle por cada ítem y búsqueda de sublist para cada carga.	Reestructurar para cargar cada ítem una sola vez, usar `search.lookupFields` o caché de datos.	598, 602, 607	-
security_issue	high	Construcción del SOAP envelope mediante concatenación de strings sin escape de XML.	Escapar correctamente los valores o usar una librería de construcción XML segura.	523, 540	-
obsolete_style	low	Uso de `var` en lugar de `let` / `const`.	Reemplazar `var` por `const` o `let` según corresponda.	27	-
anti_pattern	medium	Variables `action`, `statusFinal`, `actionEnternet` declaradas pero no siempre utilizadas, generando código innecesario.	Eliminar variables no usadas o consolidar lógica.	85 - 87	-
bad_naming	low	Códigos de estado ("1", "2", "3", "4") hardcoded.	Definir constantes descriptivas para los estados.	39, 40, 42	-

Fix Plan

- Cachear resultados de `getSetup` y `getDocumentType` por subsidiaria/documento.

- Eliminar importaciones duplicadas y renombrar parámetros para evitar sombra.
- Mover verificaciones de características (`runtime.isFeatureInEffect`) fuera de bucles.
- Reemplazar concatenación de XML por un builder seguro o escapado de valores.
- Agregar retorno explícito en los bloques catch de `getInputData`.
- Crear constantes para códigos de estado y valores mágicos.
- Separar la lógica de negocio del manejo de errores y de la capa de integración (SOAP).

PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_STLT.js

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.1

El script declara SuiteScript 2.1 pero falla en tiempo de ejecución por no importar el módulo N/log, genera un entry point excesivamente grande y carece de manejo de casos inesperados y validación de entradas.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se utiliza el objeto 'log' sin haber importado el módulo N/log, provocando ReferenceError en tiempo de ejecución.	Agregar 'N/log' al arreglo de dependencias y al parámetro de la función define.	31, 32, 67, 89	-
spaghetti_code	high	El entry point onRequest supera las 50 líneas y contiene lógica de GET, POST, manejo de formularios, redirecciones y envío de correos, violando SRP.	Dividir la lógica en servicios o clases separadas (e.g., GetHandler, PostHandler) y delegar desde onRequest.	26 - 28, 30 - 36, 38 - 41, 43 - 48, 50 - 61, 63 - 95	-
anti_pattern	high	No existe caso por defecto para valores de 'status' fuera de 0,1,2; el objeto 'form' quedaría undefined y provocaría error al asignar clientScriptModulePath.	Agregar un bloque else que genere un formulario de error o redirección, o usar switch con default.	44 - 48, 50 - 61	-
security_issue	medium	Los parámetros de la request se convierten a Number sin validar que sean numéricos; valores no numéricos producirán NaN y lógica inesperada.	Validar que los parámetros existan y sean numéricos antes de usarlos; manejar NaN con mensajes claros.	30, 75	-
governance_risk	warning	No se verifica el consumo de gobernanza (runtime.getCurrentScript().getRemainingUsage()) en bucles ni en operaciones potencialmente costosas.	Incorporar chequeos de uso restante y dividir procesos en etapas de Map/Reduce si la carga lo justifica.	26 - 28, 33 - 36, 38 - 41, 43 - 48, 50 - 61, 63 - 95	-

Fix Plan

- Agregar importación de N/log.
- Refactorizar onRequest en clases/servicios separados para GET y POST.
- Implementar manejo de caso por defecto para 'status' y validar parámetros.

- Incluir chequeos de gobernanza y, de ser necesario, migrar lógica pesada a Map/Reduce.
- Reemplazar múltiples if por switch o tabla de configuración para mejorar legibilidad.

PROYECTOS INBOUND/Inbound_CL/Transaltor/ LMRY_CL_Inbound_Translator.js

Script type: UserModule.js

Audit score (0-10): 7

Final score (0-100): 70

API version: 2.1

Módulo simple y bien estructurado con uso correcto de SuiteScript 2.1 y ES6. Sin embargo, contiene claves duplicadas que provocan sobrescritura de traducciones, lo que genera errores de mantenimiento y posibles fallos de UI.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
duplicate_key	high	Clave 'case4' definida dos veces con valores diferentes; la segunda sobrescribe la primera, perdiendo una traducción.	Eliminar la clave duplicada o renombrar una de ellas (p.ej., case4_alt) y actualizar las referencias.	24, 25	-
duplicate_key	medium	Clave 'pending' definida dos veces con el mismo valor; la segunda sobrescribe la primera sin aportar valor.	Mantener una única definición de 'pending' y remover la duplicada.	43, 106	-
duplicate_key	medium	Clave 'pending' definida dos veces en el bloque inglés; la segunda sobrescribe la primera.	Consolidar la clave 'pending' en una única definición.	152, 216	-

Fix Plan

- Externalizar las traducciones a archivos JSON o a un registro de NetSuite para facilitar la gestión y evitar duplicados.
- Implementar una validación automática que detecte claves duplicadas durante el build.
- Crear una función genérica para cargar traducciones por idioma, permitiendo extensibilidad sin tocar el código fuente.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Client/
LMRY_AR_InboundDetail_CLNT.js**

Script type: ClientScript

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

Script declarado como SuiteScript 2.1 pero mezcla APIs legacy 1.0 (nlapi*), mantiene estado global mutable y contiene funciones gigantes con lógica de UI y de dominio mezclada. Violaciones críticas de compatibilidad, seguridad y gobernanza.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Uso de funciones legacy nlapiSetLineItemValue y nlapiGetFieldValue dentro de un script 2.1. Estas APIs no están disponibles en el runtime GraalVM y provocarán errores de runtime.	Reemplazar todas las llamadas nlapi* por sus equivalentes 2.x (e.g., record.submitFields, currentRecord, N/record).	255 - 278	-
compatibility_break	critical	Uso extensivo de nlapiGetLineItemCount, nlapiGetLineItemValue y nlapiGetFieldValue dentro de bucles para validar totales.	Utilizar record.getLineCount y record.getSublistValue con 2.x, o mejor, cargar los datos mediante search.lookupFields fuera del bucle.	291 - 336	-
spaghetti_code	high	Función de más de 200 líneas que mezcla validaciones, cálculos, UI, llamadas a ventana padre y lógica de negocio.	Dividir en servicios: validación, cálculo de totales, construcción de payload, y manejo de ventana. Cada módulo ≤ 60 líneas.	289 - 527	-
global_state	high	Variables mutables (purchaseOrderList, itemList, taxCodes, isNotRunning, itemValues, setupData, statusFinal, LANGUAGE) declaradas en ámbito global del módulo.	Encapsular dentro de funciones o usar closures; inicializar en pageInit y evitar mutar globalmente.	16 - 24	-
security_issue	medium	Se inserta HTML sin sanitizar en campos de NetSuite mediante setValue, lo que puede generar XSS si los valores provienen de fuentes externas.	Escapar o sanitizar los valores antes de incluirlos en HTML, o usar APIs de NetSuite que acepten texto plano.	712 - 715	-
governance_risk	medium	Dentro del bucle se llama a nlapiGetLineItemValue en cada iteración, consumiendo gobernanza de forma lineal y sin control de uso restante.	Obtener todos los valores mediante search.lookupFields o cargar la sublista completa una sola vez.	319 - 327	-
anti_pattern	low	Se llama a replaceAll en cada cálculo aunque el valor ya fue limpiado previamente; redundancia de operaciones.	Limpiar una sola vez y reutilizar la variable.	626, 630 - 635	-

Fix Plan

- Migrar todas las llamadas `nlapi*` a APIs 2.x (`N/record`, `N/search`).
- Eliminar variables globales y pasar contexto explícito.
- Dividir `updateParent` en módulos: validación, cálculo, payload, comunicación con ventana padre.
- Implementar manejo de errores centralizado y registro de logs en lugar de `console.log/alert`.
- Sanitizar cualquier contenido HTML antes de insertarlo en campos.

PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/ com.latamready.lmryarinbound/Client/LMRY_AR_Inbound_CLNT.js

Script type: ClientScript

Audit score (0-10): 8

Final score (0-100): 80

API version: 2.1

El script está declarado correctamente como SuiteScript 2.1 y utiliza sintaxis moderna. La arquitectura básica es adecuada, pero existen pequeños problemas de robustez y buenas prácticas que deben corregirse.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	medium	La variable 'handler' se inicializa en pageInit, pero si fieldChanged o saveRecord se disparan antes de pageInit, 'handler' será null y provocará una excepción.	Agregar verificación de existencia antes de usar handler, o inicializarlo de forma segura en el módulo.	22, 26	-
debug_artifact	low	Se mantiene un console.log(handler) en producción.	Eliminar o envolver el log en una condición de entorno de desarrollo.	18	-
input_validation	low	Los parámetros 'isApplied' y 'sublistId' no son validados antes de usarse.	Validar que sublistId exista en el registro y que isApplied sea booleano antes de iterar.	34, 35	-
error_handling	low	No se captura ninguna excepción proveniente del handler.	Envolver las llamadas al handler en try/catch y manejar los errores de forma amigable.	15, 22, 26	-

Fix Plan

- Implementar validación de parámetros y manejo de errores en todas las funciones públicas.
- Eliminar logs de depuración antes de pasar a producción.
- Añadir JSDoc a las funciones para mejorar la mantenibilidad y la generación automática de documentación.
- Considerar extraer la lógica de toggleCheckBoxes a un módulo de utilidad reutilizable.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Handlers/
LMRY_AR_InboundDetail_STLT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script declara SuiteScript 2.1 y usa sintaxis moderna, pero contiene errores críticos de ejecución (log no importado, expresión de filtros mal construida en getItemDetailsByVendorCodes), funciones extremadamente largas y alta complejidad, falta de validación de parámetros y varios anti-patrones que comprometen mantenibilidad y gobernanza.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se utiliza log.debug sin haber importado el módulo N/log, provocando ReferenceError en tiempo de ejecución.	Agregar "N/log" al arreglo de dependencias y al parámetro de la función define.	81 - 84	-
compatibility_break	critical	Construcción incorrecta del filtro vendorCodeFilter: se inserta un array dentro de otro array, resultando en una expresión de filtro inválida para search.create.	Construir el filtro como una expresión plana (ej. [['vendorcode', 'anyof', ...], 'OR', ...]) o usar search.createFilter para cada código y combinar con 'OR'.	862, 866 - 868, 870, 872, 888, 889	-
spaghetti_code	high	El método createForm contiene más de 200 líneas de lógica directa, mezclando creación de campos, lógica de negocio y generación de HTML.	Dividir la lógica en sub-métodos (ej. buildPrimaryFields, addActionButtons, populateHiddenFields) y delegar la generación de HTML a una clase de utilidades.	75 - 277	-
governance_risk	high	Se realizan múltiples llamadas a setSublistValue dentro de un bucle que puede procesar cientos de ítems, consumiendo gran cantidad de gobernanza y riesgo de SSS_USAGE_LIMIT_EXCEEDED.	Utilizar sublist.setSublistValue en bloques o emplear sublist.addLine con objetos, y validar uso de remainingUsage en bucles críticos.	690 - 861	-
bad_naming	medium	Uso de nombres de variables en mayúsculas (FEAT_SUBS, FEAT_DPT, etc.) y mezcla de snake_case y camelCase, lo que rompe la convención de estilo.	Adoptar camelCase para variables y constantes en mayúsculas solo para valores inmutables.	38, 40 - 73	-
obsolete_style	low	Se utilizan declaraciones var en lugar de let/const, lo que puede generar hoisting inesperado.	Reemplazar var por const o let según corresponda.	862, 864, 866 - 868, 870, 872, 874, 875, 877 - 902	-

Fix Plan

- Importar N/log y eliminar referencias a log no definidas.
- Corregir la construcción del filtro en getItemDetailsByVendorCodes.
- Refactorizar createForm y fillItemsSublist en módulos más pequeños.
- Aplicar validación y sanitización de parámetros recibidos en Suitelet.
- Reemplazar var por let/const y unificar estilo de nombres.
- Implementar control de remainingUsage en bucles críticos.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Handlers/
LMRY_AR_Inbound_CLNT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa AMD correctamente, pero concentra demasiadas responsabilidades en una única clase, viola SRP y varios principios SOLID, contiene código obsoleto (var), valores mágicos, riesgos de gobernanza y posibles vulnerabilidades de inyección en la generación de URLs.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	high	La clase ClnHandler agrupa lógica de UI, validaciones, búsquedas, creación de registros y manejo de pop-ups, violando el Principio de Responsabilidad Única (SRP).	Dividir la clase en módulos especializados: UIHandler, ValidationService, DataService, LogService y Exportar cada uno como dependencia independiente.	36, 59, 104, 255, 307, 320, 465, 517, 566, 616, 666, 736, 798	-
governance_risk	medium	Ejecuta una búsqueda que devuelve hasta 1000 periodos en una sola llamada sin paginación ni control de uso de gobernanza.	Implementar paginación con runPaged() o limitar la búsqueda a los periodos necesarios; monitorear el consumo de gobernanza.	473, 492, 493, 514	-
security_issue	high	Se construye una URL con parámetros provenientes directamente del registro (xml, rut, vendor, amount, etc.) sin sanitización.	Validar y sanitizar todos los valores antes de incluirlos en params; usar encodeURIComponent o la API de parámetros seguros de NetSuite.	218 - 234	-
bad_naming	medium	Uso de la palabra reservada 'var' en un script 2.1 y nombres de variables genéricos (departmentObj) que no describen su propósito.	Reemplazar 'var' por 'const' o 'let' y renombrar variables a nombres descriptivos (e.g., setupSearch).	800 - 802	-
obsolete_style	low	Uso de declaración var en lugar de let/const en SuiteScript 2.1.	Cambiar a const para objetos inmutables.	801	-
anti_pattern	medium	Se asume que startDate y endDate siempre están definidos; si alguna es null, la llamada a getMonth()/getYear() lanzará excepción.	Agregar validaciones de null antes de comparar fechas.	389 - 392	-
anti_pattern	low	Uso de window["details;" + line] para rastrear pop-ups, creando variables globales y posible fuga de memoria.	Encapsular el estado en una estructura interna del handler o usar un Map.	308, 311, 312	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	low	Manejo de errores solo con console.log; el usuario no recibe feedback y el flujo continúa sin indicar fallo.	Mostrar mensaje de error al usuario mediante N/ui/message y registrar el error en un registro de auditoría.	538, 545 - 548, 555, 558, 562, 587, 594, 595, 604, 607, 610, 637, 644 - 646, 654, 657, 660	-
magic_number	low	IDs de script y despliegue están hardcodeados.	Mover a Script Parameters o constantes configurables en el registro del script.	34, 35	-

Fix Plan

- Refactorizar Clnthandler en varios servicios especializados.
- Implementar paginación en fillPeriod y en cualquier búsqueda que pueda devolver gran número de resultados.
- Sanitizar y validar todos los parámetros antes de construir URLs.
- Reemplazar var por let/const y eliminar variables globales en window.
- Agregar manejo de errores visible para el usuario en operaciones asíncronas.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Handlers/
LMRY_AR_Inbound_STLT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara SuiteScript 2.1 y usa AMD correctamente, pero mezcla demasiadas responsabilidades en una única clase, contiene funciones gigantes (más de 200 líneas), hard-codea IDs y strings, genera HTML sin sanitizar (riesgo XSS), maneja fechas con offsets manuales y tiene lógica confusa en areThereSubsidiaries. Además, varios bucles realizan búsquedas sin paginación adecuada y violan principios SOLID.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
security_issue	critical	Se insertan valores provenientes de registros (issuer_name, issuer_rut, etc.) directamente en campos de sublista como HTML sin escape, lo que permite XSS si algún dato contiene código malicioso.	Sanitizar todos los valores antes de asignarlos a sublistValue o usar encodeURIComponent / N/format.escape.	705, 710, 715, 724, 730, 735	-
spaghetti_code	high	Función supera ampliamente el límite de 50 líneas, contiene lógica de UI, validaciones, generación de URLs y manejo de estado mezclados.	Dividir en métodos privados: construirCampos, construirBotones, aplicarValoresPredeterminados, generarEnlaces.	123, 172, 310, 380, 424, 447	-
bad_naming	high	Nombre de método contiene errores tipográficos (Standar) y no sigue convención camelCase.	Renombrar a getInboundStandardData y actualizar todas sus llamadas.	1099	-
anti_pattern	medium	Variable anysubsidiary se inicializa siempre como true y nunca se actualiza, por lo que la función siempre devuelve true sin validar realmente la existencia de subsidiarias.	Eliminar anysubsidiary o calcular su valor basado en la lista obtenida.	70, 73, 92	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
governance_risk	medium	Dentro del bucle principal se realizan llamadas a url.resolveRecord y a búsquedas de auditoría que pueden consumir gobernanza rápidamente en grandes volúmenes.	Pre-calcular URLs fuera del bucle o usar caché de resultados; limitar la cantidad de registros procesados por petición.	684, 715, 724, 730	-
magic_number	medium	Se asume formato de fecha 'YYYY-MM-DD' y se reordena manualmente sin validar la entrada.	Utilizar N/format.parse y validar antes de convertir.	1289, 1290	-
obsolete_style	low	Se resta 4 horas al Date para ajustar zona horaria en lugar de usar la API de NetSuite para manejo de zonas.	Usar N/format con timezone o N/runtime.getCurrentUser().getPreference('TIMEZONE').	994	-
anti_pattern	low	Se crea una búsqueda con filtro 'anyof' usando una lista de IDs sin validar su longitud; si la lista supera 1000 elementos la búsqueda fallará.	Dividir la lista en lotes de 1000 o usar search.createFilter con operator 'anyof' y paginación.	1198, 1200	-

Fix Plan

- Refactorizar LibraryHandler en varios módulos: UIBuilder, DataFetcher, Translator, BusinessLogic.
- Sanitizar todos los valores que se insertan en HTML o sublist para prevenir XSS.
- Aplicar paginación y batch processing en búsquedas que usan listas largas (>1000 IDs).
- Reemplazar cálculos manuales de zona horaria por N/format con timezone.
- Renombrar métodos con nombres claros y consistentes.
- Eliminar código muerto y comentarios innecesarios.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Handlers/
LMRY_AR_VendorBillInbound_URET_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

El script declara SuiteScript 2.1 y usa AMD correctamente, pero carece de anotaciones críticas (@NScriptType), mezcla responsabilidades en afterSubmit y no controla el consumo de gobernanza. Hay varios problemas de estilo y nomenclatura que generan deuda técnica moderada.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION
compatibility_break	critical	Falta la anotación @NScriptType UserEvent, obligatoria para que NetSuite reconozca el script como User Event.	Agregar @NScriptType UserEvent justo antes de @NModuleScope.
bad_naming	low	El valor de @Name (LMRY_PE_VendorBill_URET_HNDL.js) no coincide con el nombre real del archivo (LMRY_AR_VendorBillInbound_URET_HNDL.js).	Actualizar @Name para que refleje el nombre exacto del archivo.
spaghetti_code	high	afterSubmit combina la extracción de datos XML y la lógica de creación/actualización de registros, violando SRP.	Dividir la lógica en módulos separados: parsear XML y otro para gestionar los registros inbound.
governance_risk	medium	No se verifica el consumo de gobernanza (remaining usage) antes de cargar o crear registros.	Consultar runtime.getCurrentScript().getRemainingUsage() y abortar o reprogramar si el límite está próximo.
obsolete_style	low	Conversión redundante de Date: se crea un Date y luego se vuelve a envolver en otro Date.	Eliminar la segunda llamada a new Date() asignando directamente el objeto Date a la variable.
bad_naming	low	Uso de strings literales para IDs de campos y tipos de registro en varios lugares.	Definir constantes descriptivas para los campos y tipos de registro.
obsolete_style	low	Variables declaradas con let que nunca se reasignan (inboundRecord, xmlData).	Cambiar a const para mayor claridad.
anti_pattern	low	Funciones vacías sin lógica ni comentario explicativo.	Eliminar funciones no usadas o agregar comentario // No operation si se mantiene por requerimiento.

Fix Plan

- Agregar @NScriptType UserEvent para que el script sea reconocido.
- Refactorizar afterSubmit en dos módulos: parseo XML y gestión de registros inbound.
- Implementar verificación de remaining usage antes de operaciones costosas.
- Reemplazar strings de IDs por constantes centralizadas.
- Optimizar la conversión de fechas eliminando la doble instanciación.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/LMRY_AR_InboundAFIP_MPRD.js**

Script type: Map/Reduce

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

Script declara SuiteScript 2.1 y usa arquitectura AMD, pero contiene fallos críticos de seguridad, fugas de variables globales, falta de validaciones y riesgos de gobernanza que impiden su ejecución fiable.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	critical	Variable 'templateByDocument' se asigna sin declaración (let/const), creando una fuga al ámbito global.	Declarar la variable con const o let dentro del bloque donde se usa.	168	-
security_issue	high	Se realiza una petición HTTP a una URL sin TLS (http://) y a una IP fija.	Cambiar a HTTPS y mover la URL a un parámetro de script o a un almacén seguro.	22 - 26	-
compatibility_break	high	No se verifica que getSetupData devuelva un objeto válido antes de acceder a sus propiedades.	Validar que setupData no sea null y manejar el caso con error descriptivo.	52, 54 - 58	-
governance_risk	high	Ejecuta un search.run().each sin límite de resultados, potencialmente consumiendo toda la gobernanza en un bucle.	Limitar la búsqueda con getRange o paginar los resultados; usar lookupFields si solo se necesita el nombre.	352 - 359	-
anti_pattern	medium	El método map devuelve siempre el número 1, sin escribir datos al contexto.	Eliminar el return o usar context.write si se necesita pasar información a la siguiente fase.	73	-
bad_naming	medium	Se usan valores numéricos mágicos (11, 18, 3, 17) para estados y tipos.	Definir constantes descriptivas con nombres claros y usarlas en lugar de literales.	156, 165, 178, 182	-
obsolete_style	low	Módulo 'N/https' se importa pero nunca se utiliza.	Eliminar la dependencia no usada.	6	-
bad_naming	low	Uso de 'var' para declarar objDocumentoCFDI.	Reemplazar por 'const' o 'let'.	63	-
obsolete_style	low	Bloque de código que nunca se ejecuta porque pdfFile está inicializado a cadena vacía y nunca se asigna.	Eliminar el bloque o implementar la lógica real de manejo de PDF.	195 - 205	-
security_issue	high	En el catch se registra el error pero no se propaga, dejando la función sin retorno claro.	Rethrow el error o devolver un estado que indique fallo y manejarlo en summarize.	224, 225	-

Fix Plan

- Migrar la llamada externa a HTTPS y parametrizar la URL.

- Implementar constantes para códigos de estado y tipos de documento.
- Reescribir existingDocuments con búsqueda paginada o lookupFields.
- Eliminar dependencias y código muerto (N/https, bloque PDF).
- Agregar control de remainingUsage en bucles críticos.
- Refactorizar createInboundEdoc en sub-módulos (XML, vendor lookup, record build).
- Manejar adecuadamente errores críticos propagándolos al summarize.

PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/ com.latamready.lmryarinbound/LMRY_AR_InboundAFIP_RSLT.js

Script type: RESTlet

Audit score (0-10): 8

Final score (0-100): 80

API version: 2.1

Script Restlet en SuiteScript 2.1 que lanza un Map/Reduce. Cumple con la arquitectura AMD y usa características ES6+. Presenta buenas prácticas generales, pero carece de validación de entrada y usa var en lugar de const/let. No hay problemas críticos de compatibilidad ni de gobernanza.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
security_issue	high	No se valida el contenido de requestBody.processID antes de usarlo, lo que permite inyección de datos o errores inesperados.	Validar que requestBody.processID exista, sea del tipo esperado (string/number) y cumpla con el formato requerido antes de crear processObj.	11, 12	-
bad_naming	medium	El mensaje 'Process Beggin' contiene un error tipográfico y no sigue un estándar de mensajes.	Corregir a 'Process Begin' o usar mensajes estandarizados.	25	-
obsolete_style	low	Uso de var en lugar de const/let para variables que no cambian (jsonBody, mrTask, taskId).	Reemplazar var por const cuando la variable no se reasigna.	12, 14 - 16, 21	-
missing_import	low	Se utiliza el objeto global log sin importarlo explícitamente mediante N/log.	Agregar 'N/log' al arreglo de dependencias y usar log como módulo importado.	global scope	-

Fix Plan

- Implementar validación robusta de requestBody.processID.
- Refactorizar para usar const/let y eliminar var.
- Importar explícitamente N/log para mayor claridad.
- Separar la construcción de la tarea en un módulo de servicio reutilizable.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/LMRY_AR_InboundAFIP_SCHDL.js**

Script type: Scheduled

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script presenta errores críticos de ejecución por orden de declaración, vulnerabilidades de seguridad, ausencia de control de gobernanza y código muerto. Necesita una re-arquitectura completa.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	
compatibility_break	critical	Se invoca getAFIPCredentials antes de su declaración (const arrow function). Genera ReferenceError en tiempo de ejecución.	Mover la declaración de getAFIPCredentials antes de su uso o convertirla en function declaration.	34	.
compatibility_break	critical	getAFIPCredentials utiliza simpleDecrypt que aún no está definido (const arrow function). Produce ReferenceError.	Definir simpleDecrypt antes de getAFIPCredentials o usar function declarations.	104, 105	.
security_issue	high	Endpoints hardcodeados con protocolo HTTP y direcciones IP públicas.	Mover URLs a parámetros de script o a un registro de configuración y usar HTTPS.	15, 17	.
security_issue	high	Implementación de cifrado XOR con clave estática 'inbound', vulnerabilidad crítica.	Reemplazar por N/crypto con algoritmo AES y almacenar la clave de forma segura (ej. SuiteVault).	109 - 123	.
governance_risk	high	Llamada a http.post sin control de remaining usage.	Consultar runtime.getCurrentScript().getRemainingUsage() antes y después de la llamada, y dividir en lotes si es necesario.	40 - 44	.
governance_risk	high	Búsqueda de vendor dentro de una función que podría ser llamada en bucle, sin control de gobernanza.	Cachear resultados o usar search.lookupFields cuando solo se necesita un campo.	64 - 78	.
governance_risk	high	Búsqueda de configuración sin control de gobernanza y sin manejo de resultados nulos más allá de lanzar excepción.	Validar uso de remaining usage y manejar ausencia de registro de forma controlada.	91 - 99	.
obsolete_style	low	Módulos N/record, N/file y N/task son importados pero nunca utilizados.	Eliminar importaciones no usadas.	5 - 13	.
dead_code	low	Función definida pero nunca invocada en el script.	Eliminar la función o integrarla si se necesita en el futuro.	62 - 86	.
logging_issue	low	Se registra el error con log.debug en lugar de log.error.	Cambiar a log.error para una mayor visibilidad.	46	.

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES
validation_missing	medium	No se valida que el parámetro custscript_subsidary_cuit exista o tenga formato válido.	Agregar validación y manejo de valores nulos antes de usarlo.	31

Fix Plan

- Reordenar declaraciones de funciones para evitar ReferenceError.
- Externalizar URLs a parámetros de script o registros de configuración y usar HTTPS.
- Reemplazar simpleDecrypt por N/crypto con algoritmo robusto.
- Implementar control de remaining usage antes y después de llamadas a http y search.
- Eliminar imports y funciones no utilizadas.
- Agregar validación de parámetros de script y manejo de errores consistente.
- Dividir la lógica del entry point en módulos de servicio (ej. CredentialService, HttpService).

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/LMRY_AR_InboundPdf_STLT.js**

Script type: Suitelet

Audit score (0-10): 3

Final score (0-100): 32

API version: 2.1

Script declarado como SuiteScript 2.1 pero contiene múltiples violaciones críticas de compatibilidad, errores de sintaxis y malas prácticas que comprometen la ejecución y el mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES
compatibility_break	critical	Define incluye "N/redirect" dos veces y la función runSuitelet recibe ocho parámetros pero solo siete módulos son declarados, provocando desalineación de dependencias.	Eliminar la duplicación de "N/redirect" y ajustar la lista de dependencias y parámetros para que coincidan 1:1.	19, 21
bad_naming	critical	Variables inconsistentes: se declara 'ActiveSubsi' pero se usa 'isSubsi' sin inicializar; typo en 'tipoDocuemnto' que afecta la lógica de tipo de archivo.	Unificar nombres de variables (p.ej., isSubsi) y corregir el typo a 'tipoDocumento'.	36, 43, 46, 66
security_issue	critical	Se lanza una cadena con throw en lugar de un objeto Error y se usa log.error para información de depuración, exponiendo potencialmente datos internos.	Lanzar new Error('Access denied') y usar log.debug para trazas no críticas.	48, 49
obsolete_style	high	Declaración de variables globales (UI, SEARCH, etc.) antes del define y uso de var en lugar de const/let.	Eliminar esas declaraciones y usar const dentro de la función runSuitelet.	15 - 17
anti_pattern	high	Construcción manual de URL con concatenación sin validación, potencial de open redirect.	Validar que 'Host' provenga del dominio esperado o usar redirect.toSuitelet con parámetros seguros.	71, 78, 79
governance_risk	medium	Carga de archivo dentro del flujo de solicitud sin control de uso de gobernanza; si el archivo es grande puede agotar el límite.	Verificar uso restante con runtime.getCurrentScript().getRemainingUsage() y limitar tamaño de archivo o usar streaming.	58 - 60
bad_practice	medium	Uso incorrecto de context.response.setHeader con objeto; la API espera setHeader(name, value).	Reemplazar por context.response.setHeader('Content-Type', 'application/pdf');	87 - 90
unused_dependency	low	Módulo "N/redirect" está listado dos veces y la variable REDIRECT se asigna pero solo se usa en una rama.	Eliminar la duplicación y revisar si REDIRECT es necesario en todo el script.	19

Fix Plan

- Refactorizar la función `execute` separando autorización, carga de archivo y generación de respuesta en módulos independientes.
- Actualizar a `const/let` y eliminar variables globales.
- Corregir la lista de dependencias del `define` y alinear parámetros.
- Implementar manejo de errores con objetos `Error` y logs apropiados.
- Validar y sanitizar la URL antes de redirigir para evitar open redirect.

PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/ com.latamready.lmryarinbound/LMRY_AR_InboundXML_STLT.js

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script declara SuiteScript 2.1 y usa AMD correctamente, pero falla en tiempo de ejecución por no importar el módulo N/log, lo que genera ReferenceError crítico. Además carece de validación de parámetros, contiene imports sin usar y mezcla estilos de variable (var) con ES6, lo que indica deuda técnica alta.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	El módulo N/log no está importado pero se usa (log.debug, log.error), provocando ReferenceError en tiempo de ejecución.	Agregar "N/log" al arreglo de dependencias de define() y al parámetro de la función.	15, 35	-
security_issue	high	El parámetro inboundRecordId se toma directamente de la query sin validación ni sanitización.	Validar que inboundRecordId exista, sea numérico y pertenezca al tipo de registro esperado antes de usar lookupFields.	18, 19	-
bad_naming	medium	Uso de var y nombres genéricos (formatted, reg, pad) en lugar de const/let y nombres descriptivos.	Reescribir la función usando const/let y nombres más claros; evitar var.	68, 96	-
obsolete_style	low	Módulos N/url y N/encode son importados pero nunca utilizados.	Eliminar los imports no usados.	8 - 13	-
anti_pattern	low	Construcción manual de HTML mediante concatenación de strings.	Utilizar serverWidget para generar contenido estático o sanitizar cualquier dato dinámico.	49, 59	-

Fix Plan

- Agregar validación y sanitización de inboundRecordId.
- Refactorizar onRequest en servicios: InboundService, XmlFormatter, ResponseWriter.
- Reemplazar var por const/let y mejorar nombres de variables.
- Eliminar imports innecesarios y añadir N/log.
- Considerar usar serverWidget para generar la página de error en lugar de HTML manual.

PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/ com.latamready.lmryarinbound/Module/ LMRY_AR_InboundDetail_STLT.js

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script declara SuiteScript 2.1 y usa arquitectura modular, pero contiene errores críticos que impedirán su ejecución en NetSuite.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES
missing_dependency	critical	Se invoca log.error sin haber importado el módulo N/log, lo que genera ReferenceError en tiempo de ejecución.	Agregar "N/log" al arreglo de dependencias de define y pasar el parámetro log a la función.	47
invalid_enum	critical	Se usa serverWidget.FieldLayoutType.OUTSIDEBELOW, que no existe; el valor correcto es serverWidget.FieldLayoutType.OUTSIDE_BELOW.	Reemplazar OUTSIDEBELOW por OUTSIDE_BELOW.	59
unused_import	high	El módulo "N/redirect" se importa pero nunca se utiliza.	Eliminar la dependencia si no se necesita.	12
incomplete_http_handling	medium	El Suitelet solo procesa peticiones GET; las peticiones POST quedan sin respuesta.	Implementar lógica para manejar POST o devolver un mensaje claro de método no soportado.	24, 34
hardcoded_html	low	Se construye HTML mediante concatenación de strings sin sanitizar.	Utilizar plantillas seguras o escapado de contenido antes de insertarlo.	62, 63

Fix Plan

- Agregar validación y sanitización de parámetros de entrada.
- Implementar manejo de POST o devolver respuesta 405.
- Refactorizar la generación de HTML usando plantillas o `serverWidget.createField` con valores estáticos seguros.
- Eliminar dependencias no usadas y documentar claramente los módulos requeridos.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Module/
LMRY_AR_InboundLogDetail_STLT.js**

Script type: Suitelet

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.

El script funciona pero presenta graves problemas de arquitectura, gobernanza y buenas prácticas. Versión declarada ambigua, punto de entrada inflado, uso excesivo de var y lógica de UI mezclada con búsquedas, riesgo de consumo de gobernanza y posible vulnerabilidad XSS.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	high	JSDoc declara @NApiVersion 2.x, práctica prohibida. Ambigüedad entre 2.0 y 2.1 puede generar errores de runtime.	Cambiar a @NApiVersion 2.0 o 2.1 explícitamente según el código real.	10	-
spaghetti_code	high	Punto de entrada supera ampliamente las 50 líneas, contiene lógica de UI, búsquedas, procesamiento y generación de HTML.	Dividir en módulos: servicio de búsqueda, generador de sublist, utilidades de idioma.	98, 335	-
governance_risk	high	Se ejecuta search.create y run().getRange dentro de un bucle while que puede iterar múltiples veces, consumiendo rápidamente unidades de gobernanza.	Reemplazar por search.runPaged con pageSize adecuado o consolidar IDs en una sola búsqueda usando filtros anyof con todos los IDs.	204, 228	-
security_issue	medium	Se inserta HTML sin sanitizar en sublistValue, potencial XSS si los campos provienen de fuentes externas.	Escapar o validar los valores antes de concatenar, usar encodeURIComponent o APIs de NetSuite para generar enlaces seguros.	292, 301	-
bad_naming	low	Variables LMRY_script y Language declaradas sin const/let y en ámbito global del módulo.	Usar const para valores inmutables y let para variables de ámbito limitado.	32, 33	-
anti_pattern	low	Declaración de var i en dos bucles anidados provoca colisión de variable.	Utilizar let para variables de bucle.	188, 230	-

Fix Plan

- Refactorizar onRequest en varios módulos: Service (búsquedas), UI Builder (formulario y sublist), Utils (idioma, sanitización).
- Cambiar @NApiVersion a 2.0 o 2.1 explícito y migrar a const/let/arrow functions si se usa 2.1.
- Reemplazar bucle while + slice por search.runPaged para obtener todos los registros en una sola llamada paginada.
- Implementar sanitización de valores antes de insertarlos como HTML en la sublista.
- Eliminar variables globales innecesarias y usar const/let para mejorar claridad de alcance.

PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/ com.latamready.lmryarinbound/Module/ LMRY_AR_InboundLog_CLNT.js

Script type: ClientScript

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

El script declara SuiteScript 2.1 y usa arquitectura AMD correcta, pero contiene funciones vacías y código comentado que no aporta valor, además de un nombre de función con typo. No hay problemas de compatibilidad ni de gobernanza, pero la deuda técnica es notable.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
dead_code	medium	Funciones pageInit, fieldChanged, validateField y saveRecord están vacías y no son exportadas (excepto saveRecord que está vacío).	Eliminar funciones no usadas o implementar la lógica requerida y exportarlas si son necesarias.	29, 44, 62, 75	-
bad_naming	low	El nombre de la función contiene un typo y mezcla de idiomas (funcionCancel).	Renombrar a functionCancel siguiendo camelCase y actualizar la exportación.	79	-
commented_export	low	Los métodos pageInit, fieldChanged y validateField están comentados en el objeto de exportación.	Descomentar y exportar solo los métodos que realmente se necesiten, o eliminar su definición si no se usarán.	90 - 92	-
inconsistent_header	low	El encabezado menciona un nombre de archivo diferente al real y una versión 2.0 mientras el script declara 2.1.	Actualizar el encabezado para reflejar el nombre y versión actuales del script.	4, 7	-

Fix Plan

- Eliminar código muerto y funciones vacías.

- Renombrar `funcionCancel` a `functionCancel` y aplicar convención `camelCase`.
- Implementar lógica real en `pageInit`, `fieldChanged`, `validateField` o removerlas si no son necesarias.
- Actualizar el encabezado del archivo para mantener consistencia documental.

PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/ com.latamready.lmryarinbound/Module/ LMRY_AR_InboundLog_STLT.js

Script type: Suitelet

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.0

El script funciona pero viola gravemente principios de arquitectura y buenas prácticas de SuiteScript 2.0. Presenta código espagueti en el entry point, uso incorrecto de tipos de campo, falta de validación de parámetros y riesgos de gobernanza.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	El entry point onRequest contiene más de 250 líneas de lógica directa (UI, búsqueda, mapeo de resultados).	Dividir la lógica en módulos de servicio (ej. UIBuilder, LogFetcher, ResultMapper) y delegar desde onRequest.	170, 464	-
anti_pattern	high	Se define el tipo de campo como cadena literal 'textarea' en lugar de usar serverWidget.FieldType.TEXTAREA.	Reemplazar por serverWidget.FieldType.TEXTAREA y validar que el campo se crea correctamente.	278	-
compatibility_break	medium	Se asume que el parámetro de script 'LANGUAGE' siempre está presente; si es null, .substring provocará error.	Validar que Language sea una cadena antes de aplicar substring, con fallback a 'en'.	37, 40, 466, 470	-
governance_risk	high	Uso de getRange(0,1000) dentro de un Suitelet puede consumir gran parte de la gobernanza y bloquear la ejecución si hay muchos registros.	Utilizar runPaged() con paginación o limitar la búsqueda mediante filtros más específicos.	304	-
bad_naming	low	Los IDs de script y deployment están hardcoded como strings.	Definir constantes o parámetros de script para los IDs.	433, 436	-
obsolete_style	low	Objeto jsonStatus declarado pero nunca utilizado.	Eliminar o integrar su uso si es necesario.	42, 46	-

Fix Plan

- Refactorizar onRequest en funciones/módulos separados (UI, Data Access, Mapping).
- Reemplazar tipos de campo por enums de serverWidget.
- Implementar validación robusta de parámetros de script.
- Cambiar la búsqueda a runPaged() con manejo de páginas.
- Externalizar IDs de scripts a constantes o parámetros.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Module/LMRY_AR_Inbound_MPRD.js**

Script type: Map/Reduce

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.0

El script está declarado como SuiteScript 2.0 y respeta la sintaxis de esa versión, pero presenta graves problemas de arquitectura, mantenimiento y gobernanza. Variables globales mutables, funciones monolíticas, anidación profunda y uso indiscriminado de búsquedas/loads dentro de bucles generan deuda técnica alta y riesgo de sobrepasar límites de uso.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	high	Variables globales mutables (objContext, hasSubsidiariesFeature, hasMultiVendor, actualUser, transRecord) son compartidas entre getInputData, map, summarize y funciones auxiliares.	Encapsular cada variable dentro del contexto de la función que la necesita o pasarla como parámetro; eliminar transRecord global y usar variables locales.	23, 25, 27 - 29	-
fat_entry_point	high	Función de más de 100 líneas que combina carga de registros, actualización de estado, carga de plug-in, parsing, validaciones y logging.	Dividir en sub-funciones: carga y actualización de estado, obtención de plug-in, parsing del documento, creación de transacción y auditoría.	178, 281	-
anti_pattern	high	Función con más de 180 líneas, anidación de bucles (hasta 4 niveles) y lógica compleja para manejo de ítems y líneas transformadas.	Refactorizar en módulos: extracción de códigos de proveedor, búsqueda de ítems, sincronización de líneas transformadas, manejo de errores.	436, 623	-
governance_risk	medium	Uso de getRange(1, 10) omite el primer resultado (índice 0) y puede devolver null inesperado.	Cambiar a getRange(0, 10) y validar existencia del resultado antes de usar.	395, 409	-
governance_risk	medium	Ejecuta search.run().getRange(0, 1000) sin paginación; en entornos con gran volumen puede superar el límite de uso.	Utilizar runPaged() con batch de 1000 o limitar la búsqueda con filtros más restrictivos.	415, 427	-
magic_numbers	medium	Estados de conversión definidos como números mágicos (14,15,16).	Definir un enum o constantes descriptivas (e.g., STATUS.CONVERTING = 16).	33 - 35	-
security_issue	low	No se valida que la clave de traducción exista; si falla se registra el stack completo en log.error.	Sanitizar la clave y registrar solo mensaje de error sin stack.	983, 999	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	low	Se asume que el plug-in tipo 'customscript_lmry_ar_inb_plgntype' está correctamente registrado; no se verifica la anotación @NScriptType plugintype en el script del tipo.	Agregar validación previa o documentación que garantice la presencia de @NScriptType plugintype.	212, 214	-

Fix Plan

- Refactorizar convertInboundEDoc y setItems en servicios independientes.
- Eliminar variables globales y pasar contexto explícitamente.
- Implementar paginación con runPaged() para búsquedas potencialmente grandes.
- Reemplazar números mágicos por constantes descriptivas.
- Añadir validación de parámetros de script (fechas, IDs) y sanitizar entradas.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Module/LMRY_AR_Inbound_STLT.js**

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script declara SuiteScript 2.1 y usa sintaxis moderna, pero presenta fallos críticos de compatibilidad y seguridad que impiden su ejecución correcta.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION
compatibility_break	critical	Falta la importación del módulo N/log, pero se utilizan log.debug y log.error en todo el script.	Agregar "N/log" al arreglo de dependencias, definir y pasar el objeto log al callback.
compatibility_break	critical	Se usa la constante serverWidget.FieldLayoutType.OUTSIDE_BELOW, que no existe; el valor correcto es OUTSIDE_BELOW.	Reemplazar por serverWidget.FieldLayoutType.OUTSIDE_BELOW.
governance_risk	high	Variable 'form' puede quedar undefined si el parámetro status no es 0, 1 o 2, y luego se accede a form.clientScriptModulePath.	Inicializar form con un formulario por defecto y validar que siempre se asigne antes de usarlo.
spaghetti_code	medium	Código repetitivo para los casos status 1 y 2; lógica duplicada que aumenta la superficie de mantenimiento.	Unificar la lógica común en una función y usar un switch o estructura de mapeo.
security_issue	high	Los parámetros de la request se usan directamente sin validación ni sanitización.	Validar y sanitizar todos los parámetros de entrada antes de utilizarlos.

Fix Plan

- Agregar la dependencia N/log y corregir la constante de layout.
- Refactorizar la lógica de manejo de status para eliminar duplicación.
- Implementar validación robusta de parámetros de entrada.
- Garantizar que siempre se cree un formulario antes de asignar clientScriptModulePath.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Plugin/
LMRY_AR_Inbound_Email_Capture.js**

Script type: UserModule.js

Audit score (0-10): 2

Final score (0-100): 15

Script escrito en SuiteScript 1.0 con graves violaciones de compatibilidad, errores críticos de runtime y alto acoplamiento. Necesita reescritura completa a 2.x o refactor profundo.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	fromAddress se asigna como string (email.getFrom()) y luego se invoca fromAddress.getEmail(), lo que genera TypeError en tiempo de ejecución.	Obtener la dirección de email directamente como string y pasarla sin llamar a getEmail().	354, 421	-
compatibility_break	critical	Se utiliza filterExpression (array de arrays) en nlapiSearchRecord, que solo es compatible con SuiteScript 2.x. En 1.0 se requieren nlobjSearchFilter objetos.	Reemplazar filterExpression por array de nlobjSearchFilter o migrar a SuiteScript 2.x.	196, 235	-
spaghetti_code	critical	Función principal supera los 150 líneas, contiene lógica de validación, transformación XML, creación de registros, búsqueda de plantillas y envío de email.	Dividir en módulos de servicio: EmailParser, VendorResolver, RecordBuilder, NotificationService.	353, 528	-
bad_naming	high	Variable innerXML se asigna sin declaración (var/let/const), creando una variable global implícita.	Declarar con const/let.	374	-
obscure_logic	high	En caso de excepción, la función devuelve la cadena 'false' y el llamador compara contra "false". Uso de string para indicar error es confuso.	Lanzar excepción o devolver null y manejar explícitamente.	270, 271	-
security_issue	high	Se registra el contenido completo del XML en nlapiLogExecution con nivel DEBUG, exponiendo potencialmente datos sensibles.	Eliminar o sanitizar logs que contengan datos de documentos.	372, 373	-
obsolete_style	medium	No se verifica si searchResults es null antes de iterar, lo que provocará error si no hay resultados.	Agregar verificación null/undefined antes del bucle.	104, 110	-
magic_number	medium	IDs y nombres de tipos están hardcodeados (ej. ADMIN_ROLE = 3).	Convertir a Script Parameters o a registros de configuración.	13 - 15, 24 - 28	-
undefined_variable	low	Se referencia edocStandardId en mensaje de error sin haberlo definido.	Usar la variable correcta (edocPackage) o definir edocStandardId.	444	-

Fix Plan

- Migrar a SuiteScript 2.1 para aprovechar async/await y módulos AMD.
- Refactorizar process() en servicios independientes y aplicar inyección de dependencias.
- Reemplazar búsquedas con filterExpression por nlobjSearchFilter o usar SuiteQL para mayor eficiencia.
- Externalizar valores de configuración a Script Parameters o registros de preferencias.
- Implementar manejo de errores consistente mediante excepciones y logging sanitizado.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Plugin/
LMRY_AR_Inbound_PLGNIMP.js**

Script type: PlugInImplementation

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.0

El script declara SuiteScript 2.0 pero contiene errores críticos de compatibilidad y buenas prácticas que impiden su ejecución fiable. La mayor falla es el uso incorrecto de record.getValue en la línea 49, lo que genera una excepción en tiempo de ejecución. Además, hay sombras de variables, falta de validación de datos de entrada y una función inject que mezcla varias responsabilidades.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION
compatibility_break	critical	Llamada a inboundEDocVendorRec.getValue('subsidiary') con parámetro string. En SuiteScript 2.0 la firma es getValue({ fieldId: 'subsidiary' }). Esto genera un error de runtime y rompe la compatibilidad.	Reemplazar por inboundEDocVendorRec.getValue({ fieldId: 'subsidiary' }) y validar que el valor no sea null antes de usarlo.
bad_naming	high	El @Name indica LMRY_AR_InboundCustomData_PLGN.js mientras que el archivo se llama LMRY_AR_Inbound_PLGNIMP.js. Inconsistencia que dificulta la trazabilidad.	Alinear el nombre del script con el valor del @Name o eliminar el tag si no es necesario.
shadow_variable	medium	El catch captura la excepción en una variable llamada 'error', que oculta el módulo N/error importado previamente.	Renombrar la variable del catch a 'e' o 'ex' para evitar la sombra.
missing_validation	medium	Se asume que inboundEDocRec.getValue({ fieldId: 'custrecord_psg_ei_inbound_content' }) devuelve XML válido. Si es null o mal formado, xml.Parser.fromString lanzará excepción no controlada.	Validar que xmlContent exista y sea una cadena no vacía antes de parsearla; envolver en try/catch específico con mensaje descriptivo.
anti_pattern	low	La función inject maneja múltiples responsabilidades (extracción XML, búsqueda de configuración, obtención de monedas) y supera los 50 líneas de lógica directa.	Dividir la lógica en servicios especializados y delegar desde inject.

Fix Plan

- Corregir la llamada a getValue en la línea 49 para cumplir con la API de SuiteScript 2.0.
- Renombrar la variable del catch en getCurrencies para evitar sombra de módulo.
- Añadir validaciones de entrada (XML, parámetros) y mensajes de error claros.
- Refactorizar inject en sub-módulos (XMLExtractor, ConfigProvider, CurrencyProvider).
- Alinear los nombres de archivo y los tags @Name para mejorar la trazabilidad.

PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/ com.latamready.lmryarinbound/Plugin/ LMRY_AR_Inbound_PLGNTYPE.js

Script type: PlugInImplementation

Audit score (0-10): 8

Final score (0-100): 85

API version: 2.0

Plug-in implementado para SuiteScript 2.0 con la firma requerida (inject). Cumple con la arquitectura de plug-in y no presenta errores críticos de compatibilidad ni de gobernanza. El código es sencillo pero carece de validaciones de entrada y contiene código comentado que indica posible deuda técnica.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
obsolete_style	low	Código comentado que declara variables no usadas (inboundEDocRec, templateRec, entityRec).	Eliminar o descomentar y utilizar esas variables si son necesarias; de lo contrario, borrar el código comentado.	20 - 22	-
missing_validation	medium	No se valida la existencia ni la estructura de los parámetros recibidos (params.inboundEDocRec, params.templateRec, params.entityRec).	Agregar validaciones explícitas y lanzar errores claros cuando falten datos obligatorios.	19	-
bad_naming	low	Objeto customObj se declara sin prefijo que indique su propósito; el nombre es genérico.	Renombrar a algo más descriptivo, por ejemplo, customDataObject.	25	-

Fix Plan

- Eliminar código comentado y limpiar variables no usadas.
- Implementar validaciones de entrada y manejo de errores robusto.
- Renombrar variables genéricas para mejorar la claridad del dominio.
- Considerar usar const/let en lugar de var para alinearse con mejores prácticas de ES5/ES6.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Router/LMRY_AR_Inbound_ROUT.js**

Script type: UserModule.js

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara SuiteScript 2.1 y usa AMD correctamente, pero contiene errores críticos de lógica y compatibilidad al cargar módulos de forma dinámica y devolver resultados antes de que el callback se ejecute. Además, carece de validación de entrada y vulnera principios de inyección de módulos.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Uso de require() con ruta calculada dinámicamente. SuiteScript 2.x requiere que los módulos se declaren estáticamente en la lista de dependencias; el loader no soporta carga dinámica y provocará errores en tiempo de ejecución.	Declarar todas las dependencias de forma estática en el array de define o usar N/module para cargar módulos de forma controlada. Si se necesita selección dinámica, crear un mapa de módulos pre-cargados y devolver la referencia sin volver a require.	23 - 26	-
spaghetti_code	critical	El valor devuelto (result) se asigna dentro del callback asíncrono de require, pero la función retorna inmediatamente, devolviendo siempre una cadena vacía.	Convertir la función a async/await (compatible con 2.1) y devolver una Promise que se resuelva con el módulo, o bien cargar todos los módulos estáticamente y devolver la referencia sin asincronía.	28	-
security_issue	high	El parámetro 'type' proviene de una fuente externa y se concatena directamente en la ruta del módulo, lo que permite potencialmente cargar scripts arbitrarios si el valor es manipulado.	Validar estrictamente el valor de 'type' contra una lista blanca de claves permitidas antes de usarlo para resolver la ruta del módulo.	15, 22	-
bad_naming	medium	Se inicializa con una cadena vacía (") que se usa como señal de ausencia de módulo. Es confuso y propenso a errores de tipo.	Inicializar con null o undefined y documentar claramente el valor de retorno cuando no exista una coincidencia.	21	-

Fix Plan

- Reescribir getUserEventRouter como función async que devuelve una Promise.
- Declarar todas las dependencias de módulos en la lista de define y eliminar require dinámico.
- Implementar validación de entrada contra un enum o mapa estático.
- Cambiar la inicialización de 'result' a null y documentar su uso.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Setup/
LMRY_AR_InboundSetup_CLNT.js**

Script type: ClientScript

Audit score (0-10): 5

Final score (0-100): 55

API version: 2.1

El script declara correctamente SuiteScript 2.1 y usa AMD, pero presenta puntos críticos de calidad: funciones de entrada muy extensas, lógica duplicada, uso de var en lugar de const/let, ausencia de manejo de errores en JSON.parse y uso de strings mágicos. La gobernanza está bien controlada, pero la arquitectura es frágil y necesita refactor.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	La función pageInit supera las 50 líneas y contiene lógica de negocio, mapeo de campos y limpieza de valores, violando el principio SRP.	Extraer la lógica de búsqueda y mapeo a un módulo de servicio separado y mantener pageInit solo como orquestador.	10, 11, 15, 21, 45, 50, 76, 81, 86, 105, 124, 126, 133	-
spaghetti_code	high	La función fieldChanged también supera las 50 líneas y replica la lógica de pageInit.	Reutilizar la lógica mediante una función común importada desde un módulo de utilidad.	137, 141, 146, 170, 175, 178, 204, 209, 214, 231, 250, 251, 258, 267	-
anti_pattern	high	Código duplicado para crear la búsqueda y mapear resultados en ambas funciones.	Crear una función genérica `loadSetup(subsidiaryId)` que devuelva el objeto mapeado y reutilizarla.	21, 146	-
obsolete_style	medium	Uso de `var` en un script declarado como SuiteScript 2.1, donde `const`/`let` son la práctica recomendada.	Reemplazar `var` por `const` o `let` según corresponda.	10, 138	-
bad_naming	medium	Uso de strings mágicos para IDs de campos y columnas en la búsqueda.	Definir constantes descriptivas para cada fieldId y columnId en un archivo de constantes.	21, 146, 170	-
security_issue	medium	`JSON.parse` se ejecuta sin manejo de excepciones; datos corruptos pueden lanzar errores en tiempo de ejecución.	Envolver `JSON.parse` en try/catch y fallback a valores seguros.	76, 81, 86, 204, 209, 214	-

Fix Plan

- Crear un módulo `SetupService.js` que encapsule la búsqueda y el mapeo de resultados.
- Reemplazar `var` por `const`/`let` y aplicar destructuring para mayor legibilidad.
- Definir un archivo de constantes con todos los fieldId y columnId.
- Agregar manejo de errores para `JSON.parse` y validar datos antes de asignarlos.
- Implementar pruebas unitarias para la lógica de mapeo independiente del cliente.

**PROYECTOS INBOUND/sdf-argentina-inbound/
com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/
com.latamready.lmryarinbound/Setup/
LMRY_AR_InboundSetup_STLT.js**

Script type: Suitelet

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El Suitelet está escrito en SuiteScript 2.1 y sigue la arquitectura AMD, pero presenta graves problemas de diseño y seguridad que lo hacen difícil de mantener y potencialmente inseguro.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
global_state	critical	Variables FEAT_DPT, FEAT_LOC, FEAT_CLASS y FEAT_SUBSI se declaran en el nivel del módulo y se modifican en cada petición, creando estado compartido entre ejecuciones concurrentes.	Mover estas variables al interior de onRequest y pasarlas como parámetros a las funciones auxiliares.	9 - 13	-
spaghetti_code	high	El entry point supera ampliamente las 50 líneas, combina lógica de UI, validación, búsqueda, carga/creación de registros y redirección en un solo bloque.	Dividir onRequest en servicios: UIBuilder, InputValidator, SetupService (carga/creación) y RedirectHandler.	13, 298	-
logic_error	high	La condición `if (FEAT_SUBSI recordType != 'subsidiary')` usa FEAT_SUBSI que nunca se inicializa en la fase GET, provocando que la lista de subsidiarias quede incompleta o incorrecta.	Inicializar FEAT_SUBSI al inicio de onRequest (runtime.isFeatureInEffect) y corregir la lógica a `if (!FEAT_SUBSI && recordType === 'subsidiary')` o usar una condición adecuada.	308, 313	-
error_handling	medium	No se captura ninguna excepción alrededor de record.save(), lo que puede provocar errores no controlados y respuestas HTTP 500.	Envolver los bloques de carga/creación y save() en try/catch y registrar errores mediante log.audit o log.error.	242, 265, 267, 288	-
security_issue	medium	Los parámetros recibidos del cliente se usan directamente sin validación ni sanitización (p.ej., subsidiaryId, tax codes).	Validar tipos, longitud y existencia de los IDs contra listas permitidas antes de usarlos.	196, 215	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
governance_risk	low	Se ejecutan búsquedas separadas por cada campo SELECT, lo que incrementa el consumo de gobernanza en cada petición.	Cachear resultados de búsqueda por tipo de registro dentro de la misma petición o usar una única búsqueda con columnas múltiples.	78, 86, 304, 327	-
style	low	Uso extensivo de `var` en lugar de `const`/`let`, lo que reduce la claridad del alcance y permite reasignaciones accidentales.	Reemplazar `var` por `const` cuando el valor no cambia y por `let` cuando sea mutable.	15, 31, 45, 89, 103, 117, 130, 138, 151, 158, 165, 172, 180, 188	-

Fix Plan

- Refactorizar onRequest en módulos de servicio separados.
- Eliminar variables globales y pasar contexto explícito.
- Implementar caché de resultados de búsqueda para reducir consumo de gobernanza.
- Agregar validación y sanitización de todos los parámetros de entrada.
- Incluir manejo de excepciones y logging estructurado.

PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.lmryarinbound/src/FileCabinet/SuiteApps/ com.latamready.lmryarinbound/Translator/ LMRY_AR_Inbound_Translator.js

Script type: UserModule.js

Audit score (0-10): 8

Final score (0-100): 80

API version: 2.1

Módulo de traducciones bien estructurado y compatible con SuiteScript 2.1, pero contiene redundancias y recrea un gran objeto en cada llamada, lo que impacta ligeramente en rendimiento y mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
duplicate_key	low	La clave 'pending' está definida dos veces en el bloque de español, lo que genera redundancia y posible confusión.	Eliminar la clave duplicada y conservar una única definición.	41, 111	-
performance_inefficiency	medium	El objeto 'labels' se crea dentro de la función en cada invocación, consumiendo memoria y CPU innecesariamente.	Mover la definición de 'labels' fuera de la función (const a nivel de módulo) y devolver una referencia o una copia según necesidad.	9, 242	-

Fix Plan

- Externalizar el objeto de traducciones a un archivo JSON estático y cargarlo una sola vez.
- Implementar validación del parámetro 'language' para evitar valores inesperados.
- Agregar pruebas unitarias que verifiquen la ausencia de claves duplicadas.

PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/SuiteApps/ com.latamready.coinbound/Client/LMRY_CO_Inbound_CLNT.js

Script type: ClientScript

Audit score (0-10): 8

Final score (0-100): 80

API version: 2.1

Script cliente en SuiteScript 2.1 bien estructurado, usa AMD, arrow functions y separación de responsabilidades. Presenta pequeñas áreas de mejora como trazas de consola en producción y manejo de errores.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
debug_artifact	low	Presencia de console.log(handler) que expone información interna en entornos productivos.	Eliminar o envolver en condición de entorno de desarrollo.	18	-
error_handling	medium	Llamadas directas a handler sin captura de excepciones.	Envolver en try/catch y devolver valores seguros (p.ej., false en saveRecord).	23, 27	-
global_state	warning	Variable mutable 'handler' a nivel de módulo.	Inicializar dentro de cada función o usar closure para evitar estado compartido.	13	-
input_validation	low	No se valida que 'sublistId' exista ni que 'isApplied' sea booleano.	Agregar validaciones de tipo y existencia antes de operar.	34, 35	-

Fix Plan

- Eliminar trazas de consola o usar un logger configurado por entorno.
- Implementar manejo de errores centralizado en el handler.
- Convertir la variable 'handler' a constante dentro de pageInit y pasarla como parámetro a otras funciones.
- Agregar validaciones de parámetros en toggleCheckBoxes para mayor robustez.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Client/LMRY_CO_Inbound_Detail_CLNT.js**

Script type: ClientScript

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.1

El script declara SuiteScript 2.1 pero mezcla llamadas legacy nlapi, tiene entry points inflados, variables globales mutables y lógica de negocio acoplada a UI, lo que genera incompatibilidades críticas y deuda técnica alta.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Uso de APIs legacy nlapiSetLineItemValue y nlapiGetFieldValue dentro de un script 2.1. Estas funciones no están disponibles en el runtime GraalVM y provocarán errores de runtime.	Reescribir la lógica usando la API 2.x (record.submitFields, currentRecord, etc.) y eliminar cualquier referencia a nlapi.	305 - 333	-
compatibility_break	critical	Presencia de llamadas nlapiGetFieldValue, nlapiGetLineItemCount dentro de un módulo 2.1, lo que rompe la compatibilidad del runtime.	Sustituir nlapi* por sus equivalentes 2.x (currentRecord.getValue, sublist.getLineCount, etc.).	340, 344 - 347, 350 - 352, 354 - 357	-
fat_entry_point	high	El entry point pageInit supera las 50 líneas y contiene lógica de carga de datos, manipulación de UI y validaciones mezcladas.	Dividir la función en módulos: carga de datos (getSetup, getPurchaseOrder), inicialización UI (disableFields), y manejo de errores.	50 - 99	-
global_state	high	Variables mutables globales (purchaseOrderList, itemList, taxCodes, isNotRunning, etc.) comparten estado entre llamadas y pueden producir efectos colaterales inesperados en entornos multi-usuario.	Encapsular el estado dentro de objetos o pasar como parámetros; evitar mutar globals.	16 - 25	-
governance_risk	medium	Se crea y ejecuta una búsqueda (search.create().run().getRange) cada vez que el campo custpage_item_po cambia, lo que puede consumir rápidamente la gobernanza si el usuario edita múltiples líneas.	Cachear resultados por vendorId y po_itemSelected o usar lookupFields para obtener solo el vendorcode.	206 - 220	-
hardcoded_values	medium	Uso de valores mágicos como '3', '4', 'Approved', 'Rejected' y comparaciones de strings sin enumeraciones ni constantes.	Definir constantes o enumeraciones descriptivas en un módulo de configuración.	358 - 361, 378 - 412	-
anti_pattern	low	Uso de var y bucles for tradicionales; se podría simplificar con map/filter y const.	Reescribir con const/let y métodos de array (map, forEach).	1192 - 1232	-

Fix Plan

- Migrar todas las llamadas nlapi* a la API 2.x.
- Eliminar variables globales y pasar estado mediante objetos de contexto.
- Dividir pageInit y updateParent en servicios de dominio y de UI.
- Implementar caché para búsquedas de vendorcode y purchase order.
- Reemplazar bucles for por métodos funcionales y usar const/let en lugar de var.
- Definir constantes para códigos de estado y textos de UI.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Handlers/
LMRY_CO_Inbound_CLNT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa AMD correctamente, pero presenta graves problemas de arquitectura y calidad: entry points inflados (>50 líneas), uso indiscriminado de var/alert, violaciones de la API de NetSuite y lógica de UI acoplada. La gobernanza es aceptable, pero la mantenibilidad es muy baja.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Uso incorrecto de record.setValue: se pasa la firma antigua (fieldId, value) en lugar de un objeto con fieldId y value.	Reemplazar por recordObj.setValue({ fieldId: 'custpage_is_processing', value: results[0].getValue('taskId') });	410	-
spaghetti_code	critical	El método pageInit contiene más de 50 líneas de lógica directa, mezcla de UI, búsquedas y manipulación de sublistas.	Extraer la lógica en servicios separados (e.g., InboundSetupService, SublistValidator) y mantener pageInit como orquestador.	58 - 100	-
spaghetti_code	high	fieldChanged supera el límite de 50 líneas y contiene lógica de UI, construcción de URLs, manejo de pop-ups y bucles anidados.	Dividir en métodos privados (e.g., _openDetailPopup, _collectIdPOs) y usar promesas/async donde corresponda.	102 - 219	-
spaghetti_code	high	saveRecord contiene lógica de validación, llamadas a métodos de dominio y creación de logs, superando los 50 líneas recomendados.	Externalizar validaciones a un validador dedicado y delegar la creación de logs a un servicio.	221 - 265	-
bad_naming	medium	Uso de `var` en un entorno ES6+; además nombres como `xml`, `pdf` son genéricos y pueden colisionar.	Reemplazar `var` por `let` o `const` y usar nombres descriptivos.	119, 125, 126, 130, 131, 136, 137, 141, 142, 146, 147, 151, 152, 157, 158, 161	-
security_issue	medium	Se muestra información de error al usuario mediante `alert`, lo que puede revelar detalles internos.	Registrar errores en un log interno y mostrar mensajes genéricos al usuario.	95, 216, 302, 328, 363, 378	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	low	Gestión de pop-ups mediante propiedades globales en `window`.	Encapsular en un módulo de UI o usar un patrón de singleton.	267 - 277	-
undefined_reference	low	Llamada a `setWindowChanged` que no está definido en el script ni importado.	Definir la función o eliminar la llamada si no es necesaria.	827	-

Fix Plan

- Refactorizar pageInit, fieldChanged y saveRecord en módulos de servicio separados.
- Reemplazar `var` por `let/const` y aplicar naming consistente.
- Implementar manejo centralizado de errores con registro interno y mensajes de usuario genéricos.
- Corregir la llamada a record.setValue en validateExecution.
- Eliminar dependencias a `window` y encapsular lógica de pop-ups.
- Añadir verificación de `runtime.getCurrentScript().getRemainingUsage()` en bucles críticos.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Handlers/
LMRY_CO_Inbound_Detail_STLT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script usa SuiteScript 2.1 correctamente, pero su arquitectura está sobrecargada: una única clase maneja UI, lógica de negocio y acceso a datos, con métodos gigantes y alta complejidad. Se detectan varios anti-patrones (fat entry points, magic strings, uso inconsistente de var/let, posible XSS en nombres de funciones) y riesgos de gobernanza por búsquedas dentro de bucles. Necesita una refactorización profunda.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	El método createForm supera ampliamente el límite de 50 líneas, mezclando creación de UI, lógica de negocio y decisiones de flujo.	Dividir en sub-módulos: un builder de UI, un service de datos y un controller que orquesta la lógica.	76, 247	-
bad_naming	high	Se inyectan valores de this.params.line directamente en strings de JavaScript para client-side, lo que permite inyección si el parámetro no está sanitizado.	Sanitizar los valores y pasar solo identificadores numéricos; usar funciones de callback predefinidas en lugar de strings.	211, 222, 231	-
anti_pattern	medium	Log.debug concatena strings de números con .toFixed, provocando concatenación en lugar de suma.	Usar parseFloat(netAmount) + parseFloat(item.taxLineAmount) antes de .toFixed.	666	-
magic_string	medium	Los estados se manejan con strings '1','2','3','4' sin enumeración ni constantes.	Definir un enum o constantes descriptivas (e.g., STATUS.APPROVED = '1').	86, 1334, 1338, 1340	-
obsolete_style	medium	Uso de var en varios métodos en lugar de let/const.	Reemplazar var por const o let según corresponda.	562, 688, 690, 695, 698, 704, 706, 718, 720	-
governance_risk	high	Construye filtros OR dinámicos que pueden crecer mucho y ejecutar una búsqueda de hasta 1000 resultados por cada llamada.	Limitar la cantidad de códigos de proveedor, usar lookupFields cuando sea posible o cachear resultados.	688, 716	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
security_issue	medium	Construye HTML con valores de traducción sin escape.	Escapar o sanitizar los textos antes de insertarlos en HTML.	1410, 1412, 1418	-
dead_code	low	Variable total calculada pero nunca utilizada.	Eliminar la variable o usarla si es necesaria.	598	-

Fix Plan

- Refactorizar LibraryHandler en varios módulos: UIBuilder, DataService, BusinessLogic.
- Definir constantes o enums para los códigos de estado.
- Reemplazar var por let/const y aplicar linting para evitar estilos obsoletos.
- Sanitizar todos los inputs/outputs que se usan en HTML o en funciones client-side.
- Cachear resultados de búsquedas estáticas (tax codes, departments, etc.) para reducir consumo de gobernanza.
- Implementar pruebas unitarias para cada módulo separado.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Handlers/
LMRY_CO_Inbound_STLT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 5

Final score (0-100): 55

API version: 2.1

El script está escrito para SuiteScript 2.1 y usa AMD correctamente, pero presenta graves problemas de arquitectura, mantenimiento y seguridad. Entradas de formulario y sublistas están infladas en un solo archivo, hay lógica defectuosa, uso de datos sin sanitizar y varios anti-patrones que comprometen la gobernanza y la extensibilidad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	Método createFilterForm contiene más de 200 líneas, mezcla lógica de UI, validaciones, generación de HTML y decisiones de negocio, violando el principio de Responsabilidad Única.	Dividir en sub-módulos: generación de campos, construcción de HTML, lógica de permisos y manejo de estados.	121, 395	-
bad_logic	medium	Variable anysubsidiary se inicializa en true y nunca se actualiza; el método siempre devuelve true aunque no haya subsidiarias activas.	Determinar correctamente la existencia de subsidiarias y devolver false cuando la lista esté vacía.	70, 90	-
security_issue	high	Se construyen enlaces HTML concatenando valores externos (see_xml, see_pdf) sin sanitizar, lo que permite XSS si esos campos contienen contenido malicioso.	Escapar o validar todos los valores antes de insertarlos en HTML; usar N/ui/serverWidget para generar enlaces seguros.	658, 666, 730, 736, 740, 751	-
governance_risk	warning	Llamadas a url.resolveRecord se ejecutan dentro de bucles que pueden procesar cientos de filas, consumiendo gobernanza innecesariamente.	Cachear URLs cuando sea posible o generar los enlaces en una fase posterior fuera del bucle.	658, 666, 740, 751	-
magic_numbers	low	Códigos de estado ('11','15', etc.) están hardcodeados en un switch, dificultando su mantenimiento y comprensión.	Definir constantes o enumeraciones descriptivas para los códigos de estado.	974, 992	-
obsolete_style	low	IDs de scripts y despliegues están escritos como strings literales en varios lugares.	Mover a constantes o parámetros de script.	854, 860, 862, 875	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
naming_issue	low	Nombre del método contiene errores tipográficos (Inboun, Standar).	Renombrar a <code>getInboundStandardData</code> .	1017	-
error_handling_missing	medium	Se parsea XML sin bloque try/catch; errores de parsing pueden lanzar excepciones no capturadas.	Envolver la lógica de parsing en try/catch y registrar errores de forma controlada.	949, 952	-
unused_variable	low	Constante <code>LMRY_script</code> declarada pero nunca utilizada.	Eliminar o usar la constante según sea necesario.	30	-
redundant_variable	low	Variable <code>anysubsidiary</code> siempre true y no aporta valor.	Eliminar variable y devolver directamente el resultado de la lógica.	71, 73	-

Fix Plan

- Refactorizar `createFilterForm` y `createApplySublist` en módulos de servicio separados.
- Implementar sanitización de datos antes de generar HTML.
- Centralizar IDs de scripts y constantes de estado.
- Agregar manejo de errores robusto alrededor de parsing XML y llamadas externas.
- Revisar lógica de `areThereSubsidiaries` para devolver valor correcto.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Handlers/
LMRY_CO_VendorBillInbound_URET_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.1

El script declara SuiteScript 2.1 pero contiene errores críticos de referencia (variables y módulos no definidos), uso inconsistente de logs y una función beforeSubmit inflada que viola SRP y presenta alta complejidad de anidamiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Variables constantes STATUS_FIELD, INCOMPLETE y FOR_CONVERSION no están definidas ni importadas, provocando errores de referencia en tiempo de ejecución.	Definir estas constantes al inicio del módulo o importarlas desde un archivo de configuración.	82 - 95	-
bad_naming	critical	Se utiliza `log.error` en lugar del logger importado `nLog`, lo que genera una referencia a una variable inexistente.	Reemplazar `log.error` por `nLog.error`.	228, 229	-
spaghetti_code	high	La función beforeSubmit supera ampliamente las 50 líneas, contiene múltiples niveles de anidamiento y combina lógica de validación, transformación XML y actualización de campos.	Dividir la lógica en servicios separados: validación de vendor, generación de template, transformación XML y actualización de estado.	30 - 98	-
anti_pattern	high	Más de tres niveles de anidamiento (`if` dentro de `if` dentro de `if`) generan código difícil de leer.	Utilizar guard clauses y funciones auxiliares para aplanar la estructura.	33, 46 - 80	-
governance_risk	medium	Se ejecuta una búsqueda y carga/creación de registro dentro de afterSubmit, lo que puede consumir gobernanza si el script se dispara en masa.	Optimizar usando `search.lookupFields` o `search.runPaged` y limitar la carga de registros a los necesarios.	135, 140, 141, 145 - 147, 150 - 153, 155	-
obsolete_style	low	Uso extensivo de `var` en lugar de `let` / `const` pese a estar en SuiteScript 2.1.	Reemplazar `var` por `const` o `let` según corresponda.	32, 35, 36, 38 - 45, 49 - 80, 82 - 95	-

Fix Plan

- Definir y centralizar constantes de estado (STATUS_FIELD, INCOMPLETE, FOR_CONVERSION).
- Corregir uso de logger (`nLog.error`).

- Dividir beforeSubmit en funciones de dominio: validarVendor, obtenerTemplate, procesarXML, actualizarEstado.
- Reemplazar `var` por `let`/`const` y aplicar destructuring donde sea posible.
- Añadir validaciones de existencia para campos de vendorSearch antes de acceder a índices.
- Implementar guard clauses para reducir niveles de anidamiento.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/LMRY_CO_Inbound_Pdf_STLT.js**

Script type: Suitelet

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara SuiteScript 2.1 pero contiene errores críticos de arquitectura (módulos duplicados, parámetros duplicados), dependencias no declaradas y prácticas de código inseguras que comprometen mantenibilidad y estabilidad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	high	Módulo 'N/redirect' está incluido dos veces en la lista de dependencias del define, generando conflicto de parámetros.	Eliminar la duplicación y dejar una única referencia a 'N/redirect' en el array de dependencias.	18	-
bad_naming	high	Parámetro 'redirect' está declarado dos veces en la firma de la función, sobrescribiendo el primer valor.	Eliminar el parámetro duplicado y ajustar la lista de parámetros para que coincida con el array de dependencias.	20	-
compatibility_break	high	Se invoca LMRY_EI_ENCRYPT.B62.decode sin que la librería esté importada ni declarada.	Importar el módulo correspondiente mediante define o eliminar la dependencia si no es necesaria.	53	-
anti_pattern	medium	Variable 'isSubsi' se asigna sin declaración previa (implícita global) y nunca se utiliza.	Declarar la variable con const/let y removerla si no se usa.	42	-
bad_practice	medium	Se lanzan strings directamente con throw en lugar de objetos Error.	Utilizar throw new Error('mensaje') o clases de error personalizadas.	48, 62	-
security_issue	medium	Construcción de URL mediante concatenación del host recibido del request sin validación.	Validar el host contra una lista blanca o usar la API de redirección de NetSuite que evita este riesgo.	77	-
obsolete_style	low	Declaración de variables globales con var en un módulo 2.1; debería usarse const/let.	Reemplazar var por const o let y limitar su alcance al módulo.	14, 15	-
anti_pattern	low	Validación insuficiente del parámetro idcontenido; se asume que la decodificación siempre es exitosa.	Agregar validación robusta y manejo de errores para valores no esperados.	52, 55	-

Fix Plan

- Refactorizar la función execute separando validación, carga de archivo y respuesta en módulos independientes.

- Corregir la lista de dependencias del define eliminando duplicados y alineando los parámetros de la función.
- Importar y declarar explícitamente LMRY_EI_ENCRYPT o reemplazar su uso por una utilidad interna.
- Utilizar const/let en lugar de var y evitar variables globales.
- Implementar manejo de errores con objetos Error y registrar logs estructurados.
- Validar y sanitizar el header host antes de usarlo en redirecciones.
- Considerar el uso de response.writeFile para PDFs en lugar de write con contenido binario.

PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/SuiteApps/ com.latamready.coinbound/LMRY_CO_Inbound_XML_STLT.js

Script type: Suitelet

Audit score (0-10): 7

Final score (0-100): 70

API version: 2.1

Script escrito en SuiteScript 2.1 con buena modularidad y separación de responsabilidades, pero presenta fallos críticos de dependencia (log no importado) y falta de validación de parámetros de entrada.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Módulo N/log no está importado pero se utiliza log.debug y log.error, provocando ReferenceError en tiempo de ejecución.	Agregar "N/log" al arreglo de dependencias del define y al parámetro correspondiente.	8	-
bad_naming	high	No se valida que inboundRecordId exista ni sea numérico antes de usarlo en search.lookupFields.	Validar inboundRecordId y devolver un mensaje de error amigable si es inválido.	18 - 22	-
obsolete_style	low	Módulo N/encode está importado pero nunca se utiliza.	Eliminar "N/encode" del define o usarlo si es necesario.	8	-
anti_pattern	low	Uso extensivo de var y concatenación manual para formatear XML; código legible pero poco mantenible.	Reescribir usando const/let, template literals y funciones auxiliares; considerar librerías de formateo XML si están disponibles.	67, 68, 70 - 72, 86 - 88, 90 - 92	-

Fix Plan

- Agregar validación de inboundRecordId y manejo de casos nulos.
- Incluir N/log en las dependencias y eliminar imports no usados.
- Refactorizar formatXMLString usando ES6+ y reducir complejidad.

- Implementar pruebas unitarias para `onRequest` y `formatXMLString`.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Library/jszip.min.js**

Script type: UserModule.js

Audit score (0-10): 2

Final score (0-100): 15

jszip.min.js es una biblioteca de terceros minificada que no declara @NApiVersion ni sigue la arquitectura AMD propia de SuiteScript 2.x. Detecta entornos Node/AMD y define un global JSZip, lo que genera incompatibilidad en NetSuite donde los módulos deben exportarse mediante define([...], ...) y no deben crear globals. Además, hace uso de require('lie') y Buffer, que no existen en el runtime de NetSuite, provocando errores de carga. Su tamaño y lógica compleja consumen gobernanza y dificultan el mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	El script detecta entornos Node/AMD y define un global JSZip, lo que rompe la arquitectura de módulos de SuiteScript 2.x que requiere define([...], ...) sin crear globals.	Encapsular la biblioteca como un módulo SuiteScript 2.x (exportar JSZip mediante return JSZip) o usar una versión compatible con AMD sin crear globals.	1, 13	-
obsoletestyle	critical	Uso de require('lie') y detección de Buffer; APIs inexistentes en el runtime de NetSuite.	Eliminar dependencias a Node (lie, Buffer) o reemplazarlas por implementaciones compatibles con SuiteScript.	todo el archivo	-
governance_risk	high	Operaciones intensivas (deflate/inflate) consumen gran cantidad de gobernanza, especialmente en bucles de procesamiento de archivos ZIP.	Limitar el uso de JSZip a scripts de tipo Scheduled o Map/Reduce con control de usage, o usar alternativas más ligeras.	funciones de compresión/deflación	-
spaghetti_code	medium	Código minificado y sin separación de responsabilidades, dificultando pruebas y mantenimiento.	Mantener la biblioteca como dependencia externa (npm) y no incluir el código minificado directamente; usar wrapper que exponga solo la API necesaria.	todo el archivo	-
security_issue	low	No hay validación de entradas al cargar datos ZIP, lo que podría permitir zip bombs o archivos maliciosos.	Implementar validaciones de tamaño y número de entradas antes de procesar ZIP.	función de carga de datos	-

Fix Plan

- Reescribir la integración como módulo SuiteScript 2.x que exporte JSZip sin crear globals.
- Eliminar dependencias a Node (lie, Buffer) o sustituirlas por implementaciones compatibles.
- Agregar validaciones de tamaño y número de archivos antes de descomprimir.
- Mover la lógica de compresión a scripts Scheduled o Map/Reduce con control de usage.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Module/LMRY_CO_InboundLog_CLNT.js**

Script type: ClientScript

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara la versión 2.1 correctamente y usa la arquitectura AMD, pero contiene funciones vacías y no exportadas, nombres inconsistentes y falta de documentación en la lógica principal, lo que indica deuda técnica alta y riesgo de mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
dead_code	high	Funciones pageInit, fieldChanged, validateField y saveRecord están declaradas pero vacías o no exportadas, generando código muerto.	Eliminar funciones no usadas o implementarlas plenamente y exportarlas según sea necesario.	27, 42, 59, 71	-
bad_naming	medium	El nombre 'funcionCancel' mezcla español e inglés y no sigue la convención camelCase en inglés.	Renombrar a 'cancel' o 'handleCancel' siguiendo la convención de nombres del proyecto.	74	-
missing_documentation	low	La función funcionCancel carece de JSDoc que describa parámetros, retorno y comportamiento.	Agregar bloque JSDoc descriptivo a la función.	74	-
console_logging	low	Uso de console.log en producción puede exponer información de depuración al cliente.	Reemplazar con un logger configurado o eliminar en entornos productivos.	78	-
anti_pattern	medium	Exporta solo funcionCancel y saveRecord, dejando comentadas otras funciones, lo que indica un patrón de código incompleto.	Definir claramente la API del módulo y remover código comentado o completarlo.	82, 86, 87	-

Fix Plan

- Eliminar funciones vacías o implementarlas según los requerimientos del cliente.
- Renombrar y documentar correctamente la función de cancelación.
- Eliminar o reemplazar console.log por un logger controlado.
- Definir una API de módulo clara y remover código comentado.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Module/LMRY_CO_InboundLog_STLT.js**

Script type: Suitelet

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.0

Script compatible con SuiteScript 2.0, sin uso de sintaxis ES6 prohibida y sin problemas graves de gobernanza. Sin embargo, el punto de entrada onRequest está inflado (>250 líneas) y contiene lógica de presentación, negocio y acceso a datos mezclados, lo que viola SRP y genera riesgo de mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES
fat_entry_point	high	El método onRequest supera ampliamente las 50 líneas y combina lógica de UI, acceso a datos y procesamiento de negocio.	Refactorizar extrayendo la generación del formulario, la carga de datos y la construcción de la tabla a módulos/servicios separados.	169, 428
magic_numbers	medium	Los valores de estado '0', '1', '2' están hardcodeados.	Definir constantes descriptivas (e.g., STATUS_PENDING = '0') y utilizarlas en el switch.	366, 382, 397
hardcoded_script_ids	medium	Los IDs de script y deployment están escritos literalmente.	Mover los IDs a parámetros de script o a un archivo de configuración central.	404 - 406
potential_null_reference	low	Si el parámetro 'LANGUAGE' no está definido, Language será null y .substring() lanzará excepción.	Validar que Language exista antes de aplicar substring, con fallback a 'en'.	36, 39, 430, 432
inconsistent_field_type	low	Se usa string literal 'textarea' en lugar del enum serverWidget.FieldType.TEXTAREA.	Reemplazar con serverWidget.FieldType.TEXTAREA.	273, 274
var_usage	low	Uso extensivo de var en lugar de let/const en SuiteScript 2.0.	Migrar a let/const donde corresponda para mejorar claridad y seguridad.	31, 32, 36, 41, 47, 169, 300, 304

Fix Plan

- Dividir onRequest en funciones: buildForm(), loadData(), populateSublist().
- Crear un módulo de constantes para estados y IDs de script.
- Implementar validación de parámetros de entrada (LANGUAGE) con valores por defecto seguros.
- Reemplazar var por let/const y usar enums de NetSuite para tipos de campo.
- Mover la generación de HTML a un client script o a un template separado.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Module/LMRY_CO_Inbound_Detail_STLT.js**

Script type: Suitelet

Audit score (0-10): 6

Final score (0-100): 68

API version: 2.1

Suitelet declara SuiteScript 2.1 y usa sintaxis moderna, pero carece de importación del módulo N/log, lo que provocará errores en tiempo de ejecución. No valida parámetros de entrada y no maneja solicitudes POST, lo que representa riesgos de seguridad y funcionales. El código es razonablemente modular, pero la falta de control de gobernanza y validaciones lo sitúan en un nivel medio.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LIN
compatibility_break	critical	Se invoca log.error sin haber importado el módulo N/log, lo que genera ReferenceError en tiempo de ejecución.	Agregar "N/log" a la lista de dependencias del define y pasar el parámetro log a la función.	46
security_issue	high	Los parámetros de la solicitud se pasan directamente al handler sin validación ni sanitización.	Validar y sanitizar todos los parámetros de entrada antes de utilizarlos o pasarlos al handler.	28,
anti_pattern	medium	Solo se maneja el método GET; las solicitudes POST son ignoradas sin respuesta ni registro.	Agregar manejo explícito para POST (y otros métodos) o devolver un error 405.	34,
governance_risk	low	No se verifica el uso de gobernanza dentro del handler, lo que podría causar SSS_USAGE_LIMIT_EXCEEDED en operaciones intensivas.	Dentro del handler, monitorear runtime.getCurrentScript().getRemainingUsage() y optimizar búsquedas/record loads.	Ha (no visi

Fix Plan

- Agregar la dependencia "N/log" y actualizar la firma del define.
- Implementar validación y sanitización de parámetros de entrada.
- Manejar explícitamente métodos HTTP no soportados (POST, PUT, DELETE).

- Revisar el handler para asegurar uso eficiente de gobernanza y considerar N/query para consultas intensivas.
- Documentar y aplicar pruebas unitarias para cubrir casos de error y validación de parámetros.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Module/LMRY_CO_Inbound_MPRD.js**

Script type: Map/Reduce

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.0

El script declara SuiteScript 2.0 y respeta la sintaxis ES5, pero contiene errores críticos de alcance y orden de declaración que provocarán excepciones en tiempo de ejecución, además de funciones monolíticas y uso intensivo de record/load dentro de bucles que comprometen la gobernanza. Necesita una refactorización profunda.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	En el bloque catch se hace referencia a la variable paramIdDocuments que está declarada dentro del try, por lo que está fuera de alcance y genera ReferenceError.	Declarar paramIdDocuments fuera del try o pasarla como argumento al catch; validar su existencia antes de usarla.	68 - 83	-
bad_naming	critical	logParams incluye la propiedad entity que hace referencia a la variable entityId antes de que sea declarada (entityId se define en la línea 190).	Definir entityId antes de construir logParams o mover la creación de logParams después de la línea 190.	183 - 190	-
fat_entry_point	high	La función convertInboundEDoc supera ampliamente el límite de 50 líneas, combinando lógica de validación, búsqueda, transformación, registro y logging.	Dividir la función en sub-módulos: validación, carga de datos, ejecución del plugin, generación de contenido, creación de transacción y logging.	180 - 302	-
governance_risk	high	Dentro de un bucle se carga cada registro de ítem (record.load) y potencialmente se guarda (itemRecord.save). Esto consume gran cantidad de unidades de gobernanza y puede superar los límites en lotes grandes.	Cachear los registros de ítem cuando sea posible, o usar búsquedas masivas (search.lookupFields) para obtener los datos necesarios sin cargar cada registro.	590 - 660	-
anti_pattern	medium	Múltiples bucles anidados (hasta 4 niveles) generan complejidad $O(n^2)$ y dificultan la lectura.	Refactorizar usando mapas (hash tables) para evitar bucles anidados y reducir la complejidad a $O(n)$.	417 - 524	-
obsolete_style	low	Variable templateId se declara dos veces consecutivas; la segunda sobrescribe la primera sin necesidad.	Eliminar la segunda asignación o reutilizar la variable ya obtenida.	194 - 198	-
anti_pattern	low	Uso de la variable global transRecord compartida entre varias funciones (convertInboundEDoc, setItems, addToItemSublist, etc.).	Pasar transRecord como argumento a las funciones que lo necesiten o encapsularlo dentro de un objeto de contexto.	29, 180, 306, 417, 590	-

Fix Plan

- Refactorizar convertInboundEDoc en módulos más pequeños.
- Eliminar variables globales y pasar contexto explícito.
- Cachear resultados de búsquedas y evitar cargar registros dentro de bucles.
- Corregir errores de alcance (paramIdDocuments, entityId).
- Simplificar bucles anidados usando estructuras de datos hash.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Module/LMRY_CO_Inbound_STLT.js**

Script type: Suitelet

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

El script está declarado correctamente como SuiteScript 2.1 y usa buenas prácticas de modularidad, pero el punto de entrada onRequest es demasiado extenso, contiene lógica de enrutamiento y construcción de UI mezclada, usa números mágicos y carece de validación robusta de parámetros, lo que genera riesgos de runtime y mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Variable 'form' puede quedar sin inicializar si el parámetro status no es 0, 1 o 2, provocando error al asignar clientScriptModulePath.	Agregar una rama default que cree un formulario de error o validar que 'form' esté definido antes de usarlo.	41, 61	-
spaghetti_code	high	El método onRequest supera las 50 líneas y contiene lógica de enrutamiento, construcción de UI y llamadas a procesos de Map/Reduce mezcladas.	Dividir la función en sub-módulos: un router, un builder de formularios y un controlador de acciones POST.	24, 90	-
bad_naming	high	Uso de números mágicos (status === 1, status === 2) sin constantes descriptivas.	Definir constantes descriptivas (e.g., STATUS_FILTER = 0, STATUS_LOAD_ALL = 1, STATUS_LOAD_STATES = 2).	48, 54	-
security_issue	medium	Los parámetros de entrada (custpage_status, custpage_log_id, custpage_subsidary) se convierten a Number sin validar y se usan directamente en redirecciones y llamadas a Map/Reduce.	Validar que los parámetros existan, sean numéricos y pertenezcan a los dominios esperados antes de procesarlos.	71, 80, 82	-
anti_pattern	low	Construcción de HTML mediante concatenación de strings dentro del Suitelet.	Utilizar plantillas o al menos escapar cualquier contenido dinámico; en este caso el HTML es estático, pero mantener la práctica.	101, 110	-

Fix Plan

- Refactorizar onRequest en un router que delegue a módulos específicos (formBuilder, postHandler).
- Definir constantes descriptivas para los valores de status y evitar números mágicos.
- Implementar validación robusta de parámetros de entrada (tipo, rango, existencia).
- Agregar manejo de caso default para status no reconocido y asegurar que 'form' siempre esté inicializado.
- Considerar usar plantillas HTML o serverWidget.addField con defaultValue seguro en lugar de concatenar strings.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Plugin/
LMRY_CO_Inbound_Email_Capture.js**

Script type: UserModule.js

Audit score (0-10): 2

Final score (0-100): 15

El script está escrito en SuiteScript 1.0 y contiene varios errores críticos que impiden su ejecución correcta, además de violaciones graves de SOLID y prácticas de Clean Code. Necesita una refactorización completa.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se trata a la variable fromAddress como objeto y se llama a fromAddress.getEmail(), pero email.getFrom() devuelve una cadena. Esto provocará una excepción en tiempo de ejecución.	Obtener el email como cadena y pasarla directamente a identifyVendor/identifyCustomer sin llamar a getEmail().	466, 589, 618	-
undefined_variable	critical	Se hace referencia a la variable edocStandardId en el log de error, pero nunca está declarada ni asignada.	Reemplazar edocStandardId por la variable correcta (por ejemplo, edocPackage) o declarar la variable antes de usarla.	610	-
spaghetti_code	high	El entry point supera ampliamente las 50 líneas y combina múltiples responsabilidades: validación de adjuntos, descompresión, parsing XML, búsqueda de vendor/customer, creación de registros, envío de correos, etc.	Dividir la lógica en servicios especializados (e.g., ZipService, XmlParser, VendorResolver, RecordBuilder, NotificationService) y delegar desde process.	465, 698	-
magic_number	medium	Se usan IDs numéricos duros (ADMIN_ROLE = 3, VENDOR_BILL_TYPE = 17, etc.) sin documentación ni parametrización.	Mover estos valores a Script Parameters o a un registro de configuración y leerlos en tiempo de ejecución.	14 - 16, 25	-
hardcoded_script_id	medium	Los IDs del Suitelet (customscript_unzip_stlt, customdeploy_unzip_stlt) están codificados.	Obtener los IDs mediante Script Parameters o buscar el registro por nombre.	81, 84	-
global_state	low	innerXML se asigna sin declarar con var/let/const, creando una variable global implícita.	Declarar con var/let dentro del bloque.	538, 543	-
redundant_search	low	Se llama a getActiveAdmins() en cada envío de correo, aunque el resultado no cambia durante la ejecución.	Cachear el resultado al inicio del script o pasar como parámetro.	159, 168	-

Fix Plan

- Refactorizar process() en varios módulos de servicio.
- Implementar caché para resultados de búsquedas estáticas (admins, carpeta inbound).
- Migrar a SuiteScript 2.1 para aprovechar async/await y módulos AMD.

- Reemplazar búsquedas por SuiteQL donde haya joins complejos.
- Externalizar IDs y configuraciones a Script Parameters.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Plugin/
LMRY_CO_Inbound_Email_Capture_STLT.js**

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.

El script declara @NApiVersion 2.x (interpretado como 2.0) pero utiliza Promise nativo y async/await implícitos, lo que rompe la compatibilidad con Nashorn. Además, el entry point supera ampliamente los 50 líneas, contiene lógica de negocio, manejo de archivos y control de errores, generando deuda técnica, riesgos de gobernanza y vulnerabilidades de seguridad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	JSDoc declara @NapiVersion 2.x, lo que el auditor interpreta como 2.0. SuiteScript 2.0 no soporta sintaxis ES6 ni Promise nativo.	Cambiar a @NapiVersion 2.1 o reescribir el código usando el módulo N/promise de SuiteScript 2.0.	16 - 20	-
anti_pattern	critical	Uso intensivo de promesas nativas (then/catch) y callbacks anidados dentro de un bucle, creando una cadena de callbacks profunda y difícil de mantener.	Reescribir usando el módulo N/promise o, mejor aún, migrar a SuiteScript 2.1 y emplear async/await con manejo de errores centralizado.	65 - 175	-
governance_risk	high	Se crean y guardan archivos (file.create / nsFile.save) dentro de un bucle asíncrono sin control de uso de gobernanza.	Pre-cargar los metadatos, usar batch o limitar la concurrencia; verificar remainingUsage en cada iteración y abortar de forma controlada.	80, 121, 138	-
spaghetti_code	high	El entry point supera los 50 líneas y combina validación de entrada, carga de archivos, lógica de negocio y respuesta HTTP.	Dividir en módulos: validador de request, servicio de descompresión, servicio de persistencia de archivos y controlador de respuesta.	22, 201	-
security_issue	high	No se valida que el usuario tenga permiso para acceder al fileId o folderId proporcionados; cualquier usuario con acceso al Suitelet podría crear archivos en cualquier carpeta.	Verificar permisos del usuario actual contra los IDs recibidos o restringir a carpetas predefinidas mediante script parameters.	25, 47 - 53	-
obsolete_style	medium	Los polyfills ejecutan la función inmediatamente, lo que altera el comportamiento esperado de setTimeout/setImmediate y puede ocultar errores de temporización.	Eliminar los polyfills o usar la API N/task para operaciones asíncronas en SuiteScript.	4 - 8, 10 - 14	-
obsolete_style	low	Uso extensivo de `var` en lugar de `let`/`const`, lo que aumenta el riesgo de hoisting y colisiones de ámbito.	Migrar a `const`/`let` y habilitar ESLint con reglas de estilo.	33, 34, 36 - 45, 47, 48, 56 - 201	-

Fix Plan

- Migrar a SuiteScript 2.1 y aprovechar async/await y clases ES6.
- Separar la lógica en módulos: validador, descompresor, creador de archivos y controlador de respuesta.
- Implementar control de gobernanza con `runtime.getCurrentScript().getRemainingUsage()`.
- Reemplazar los polyfills por la API `N/task` o `N/promise`.
- Añadir validación de permisos sobre `zipFileId` y `folderId` antes de operar.

PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Plugin/LMRY_CO_Inbound_PLGNIMP.js

Script type: PlugInImplementation

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.0

El script declara correctamente SuiteScript 2.0 y cumple con la arquitectura de plug-in, pero presenta varios problemas de calidad: lógica defectuosa en la validación de códigos de impuesto, redeclaraciones de variables, funciones sin uso y una violación clara del Principio de Responsabilidad Única en la función inject. No hay riesgos críticos de gobernanza, pero la mantenibilidad es limitada.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION
logic_error	high	La condición <code>`if(inboundSubsiSetupFields[inboundSubsiSetupFields])`</code> es incorrecta y nunca se cumple, por lo que el error de "TAX CODES NOT FOUND" nunca se lanza aunque falte el código de impuesto.	Reemplazar la condición por <code>inboundSubsiSetupFields[taxCode]</code> o validar explícitamente la existencia del código antes de asignarlo.
bad_naming variable_redeclaration	medium	Se redeclara la variable <code>`extractedJSON`</code> dos veces dentro del mismo ámbito, lo que genera confusión y puede ocultar errores.	Eliminar la declaración inicial y usar una sola asignación: <code>`vExtractedJSON = extractXMLData(xmlDocument)`</code> .
dead_code	low	La función <code>`dateFormat`</code> nunca es invocada en el script.	Eliminar la función o reutilizarla en el lugar de <code>`dateConvert`</code> si se requiere un único punto de formato de fecha.
spaghetti_code anti_pattern	medium	La función <code>`inject`</code> concentra múltiples responsabilidades: extracción de XML, búsquedas de configuración, conversión de fechas, construcción del objeto de inyección y manejo de errores.	Dividir <code>`inject`</code> en módulos especializados: parser XML, configuración, formateador de fechas y builder de data source.
validation_missing	medium	Se asume que los nodos XML existen y se accede directamente a <code>`[0].textContent`</code> sin comprobar su presencia.	Agregar validaciones de existencia de los nodos antes de acceder a los nodos y lanzar errores descriptivos si faltan datos o no son los esperados.

Fix Plan

- Refactorizar ``inject`` en servicios independientes para mejorar la mantenibilidad.
- Corregir la lógica de validación de códigos de impuesto y eliminar la condición errónea.
- Eliminar variables y funciones redundantes (redeclaraciones, ``dateFormat``).
- Añadir validaciones robustas al parser XML para evitar excepciones por datos faltantes.
- Implementar caché de resultados de búsqueda de configuración si el plug-in se usa repetidamente.

PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/SuiteApps/ com.latamready.coinbound/Plugin/LMRY_CO_Inbound_PLGNTYPE.js

Script type: PlugInImplementation

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.0

El script declara correctamente SuiteScript 2.0 y la arquitectura AMD, pero carece de validación de parámetros y potencialmente no implementa todos los métodos requeridos por el tipo de plug-in, lo que genera riesgo de incompatibilidad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
plugin_error	high	El método inject no valida ni utiliza los parámetros recibidos (inboundEDocRec, templateRec, entityRec). Además, la implementación podría no cumplir con el contrato completo del Plug-in Type si se esperan otros métodos.	Agregar validación de los parámetros, usar los objetos recibidos para poblar customObj y confirmar que todos los métodos obligatorios del Plug-in Type están implementados.	19, 21, 23	-
bad_naming	medium	Se declara la variable customObj con var, lo que permite re-asignación y hoisting innecesario.	Cambiar a const (o let si se requiere reasignación) para reforzar la inmutabilidad.	21	-
obsolete_style	low	El módulo solo importa N/render; si no se usan otras dependencias, está bien, pero se podría usar destructuring para mayor claridad.	Mantener la importación tal cual o usar const { DataSource } = N/render si se prefiere.	6	-

Fix Plan

- Implementar validación y uso real de los parámetros inboundEDocRec, templateRec y entityRec.
- Revisar la definición del Plug-in Type y asegurar que todos los métodos obligatorios están presentes.
- Cambiar var por const para mejorar la inmutabilidad.
- Agregar manejo de errores y logging para facilitar el diagnóstico.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Router/LMRY_CO_Inbound_ROUT.js**

Script type: UserModule.js

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara SuiteScript 2.1 y usa sintaxis moderna, pero contiene un error crítico de asincronía al cargar dinámicamente módulos con `require` y devolver el resultado de forma síncrona, lo que provoca que nunca se entregue el handler esperado. Además, mezcla responsabilidades y carece de validación de entrada y manejo de constantes.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se utiliza <code>`require([...], callback)`</code> que es asíncrono, pero la función devuelve <code>`result`</code> inmediatamente, por lo que siempre retorna una cadena vacía o <code>undefined</code> . El handler nunca se entrega al llamador.	Usar la versión síncrona de <code>`require`</code> (<code>`const UserEvent_HNDL = require(userEventRouter[type]);`</code>) o convertir la función en <code>async</code> y devolver una <code>Promise</code> que se resuelva cuando el módulo se cargue.	23 - 25, 28	-
bad_naming	high	El nombre sugiere que devuelve un router, pero en realidad devuelve el handler del User Event. La ambigüedad dificulta la comprensión.	Renombrar a <code>`loadUserEventHandler`</code> o <code>`getUserEventHandler`</code> para reflejar su propósito real.	15	-
magic_string	medium	Las rutas de los scripts están hardcodeadas como strings literales.	Definir constantes o parámetros de script para las rutas y referenciarlas desde allí.	18, 19	-
missing_input_validation	low	No se valida que <code>`type`</code> sea una cadena no vacía ni que pertenezca a los valores esperados.	Agregar validación explícita del parámetro y lanzar error descriptivo si es inválido.	15, 22	-

Fix Plan

- Refactorizar la carga del handler a una función síncrona o basada en `Promise`.
- Externalizar las rutas de scripts a constantes o parámetros de script.
- Añadir validación de entrada y manejo de errores más robusto.
- Renombrar la función para que su nombre refleje su comportamiento real.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Setup/LMRY_CO_InboundSetup_CLNT.js**

Script type: ClientScript

Audit score (0-10): 6

Final score (0-100): 68

API version: 2.1

El script está declarado correctamente como SuiteScript 2.1 y sigue la arquitectura AMD, pero presenta varios problemas de calidad: función fieldChanged demasiado extensa, falta de manejo de errores, importaciones innecesarias y uso de var en lugar de const/let. No hay riesgos críticos de compatibilidad, pero la mantenibilidad y robustez son medias.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
obsolete_style	low	Módulo 'N/currentRecord' es importado pero nunca utilizado.	Eliminar la importación o usarla si es necesaria.	8	-
bad_naming	medium	Se llama a JSON.parse sobre valores que pueden no ser JSON válido, lo que lanzará una excepción y romperá el script.	Envolver cada JSON.parse en try/catch y registrar el error con N/log; validar que el string sea JSON antes de parsear.	60, 66, 72	-
anti_pattern	low	Uso extensivo de 'var' en lugar de 'const' o 'let', lo que reduce la claridad del alcance y permite reasignaciones accidentales.	Reemplazar 'var' por 'const' cuando la variable no se reasigna y por 'let' cuando sí sea necesario.	11, 12, 14, 15, 17, 19, 20, 36, 41, 45, 50, 55, 60, 66, 72, 78, 84	-
governance_risk	medium	Se ejecuta una búsqueda completa (search.create + run().getRange) dentro del evento fieldChanged sin manejo de errores ni control de uso de recursos.	Agregar manejo de errores con try/catch, validar que la búsqueda devuelva resultados antes de procesar y considerar usar search.lookupFields si solo se necesita leer pocos campos.	19, 36 - 39	-
hard_coded_values	low	Los IDs de campos están escritos literalmente en el código.	Definir constantes o un objeto de mapeo para los IDs de campos y utilizarlos en todo el script.	45, 46, 50, 51, 55, 56, 60, 61, 66, 67, 72, 73, 78, 79, 84, 85, 89 - 92	-

Fix Plan

- Refactorizar fieldChanged en módulos más pequeños: un módulo para la búsqueda, otro para el mapeo de resultados y otro para la actualización de campos.
- Reemplazar var por const/let y eliminar importaciones no usadas.
- Implementar manejo de errores y logging con N/log para mejorar la depuración.
- Utilizar search.lookupFields o N/query (SuiteQL) si se migra a 2.1 para obtener los campos de forma más eficiente.

- Crear un objeto de constantes con los IDs de campos y usarlos en todo el script para facilitar futuros cambios.

**PROYECTOS INBOUND/sdf-colombia-inbound/
com.latamready.coinbound/src/FileCabinet/SuiteApps/
com.latamready.coinbound/Setup/LMRY_CO_InboundSetup_STLT.js**

Script type: Suitelet

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script funciona pero presenta graves problemas de arquitectura, mantenimiento y seguridad. Usa variables globales mutables, un entry point excesivamente grande y carece de validación de entrada.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
state_leakage	high	Variables FEAT_DPT, FEAT_LOC y FEAT_CLASS se declaran a nivel de módulo y se modifican en cada petición, provocando estado compartido entre ejecuciones concurrentes.	Mover la declaración de estas variables al interior de onRequest o calcularlas directamente en cada rama (GET). Eliminar cualquier estado mutable a nivel de módulo.	9 - 12	-
spaghetti_code	high	El método onRequest contiene más de 180 líneas combinando lógica de presentación (GET) y procesamiento de datos (POST).	Dividir la función en sub-módulos: una para construir el formulario (GET), otra para procesar la carga (POST) y utilidades comunes. Exportar esas funciones y mantener onRequest como simple delegador.	13 - 202	-
duplicate_logic	medium	Se repite el mismo bloque de setValue para actualizar o crear el registro custom.	Crear una función helper que reciba el record y los valores del formulario, y realice todos los setValue en un solo lugar.	167 - 174, 183 - 190	-
security_issue	high	Los parámetros recibidos del cliente se usan directamente sin validación ni sanitización.	Validar que los IDs sean numéricos y que los textos no contengan caracteres peligrosos. Utilizar N/validation o expresiones regulares y sanitizar antes de guardarlos.	127 - 138	-
security_issue	medium	Se almacena texto del usuario (departmentText, classificationText, locationText) dentro de JSON sin sanitizar.	Escapar o sanitizar los textos antes de incluirlos en JSON, o almacenar solo IDs y buscar nombres cuando sea necesario.	142 - 144	-
obsolete_style	low	Uso extensivo de var en lugar de const/let, lo que reduce la claridad del alcance y permite reasignaciones accidentales.	Reemplazar var por const o let según corresponda.	15, 24 - 28, 30 - 34, 37 - 51, 54 - 202	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
bad_naming	low	Se utilizan literales de cadena para IDs de campos, tipos de registro y filtros, dificultando cambios futuros.	Definir constantes descriptivas en la parte superior del módulo y reutilizarlas.	19 - 21, 34, 46, 50, 74, 78, 82, 86, 90, 94, 98, 102, 108, 109, 112 - 138	-
error_handling	medium	No se captura ninguna excepción durante la ejecución de búsquedas o carga/creación de registros.	Envolver la lógica crítica en try/catch, registrar errores mediante log.audit/error y devolver mensajes amigables al usuario.	146 - 152, 154 - 157, 162 - 175, 178 - 192	-

Fix Plan

- Refactorizar onRequest en módulos separados: buildForm(), processSubmission(), setRecordValues().
- Eliminar variables globales mutables y calcular FEAT_* dentro de la rama GET.
- Implementar validación y sanitización de todos los parámetros de entrada.
- Crear constantes para IDs de campos y tipos de registro.
- Agregar manejo de errores con try/catch y logging estructurado.
- Reemplazar var por const/let para mejorar la claridad del alcance.

PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/SuiteApps/ com.latamready.coinbound/Translator/ LMRY_CO_Inbound_Translator.js

Script type: UserModule.js

Audit score (0-10): 8

Final score (0-100): 85

API version: 2.1

Módulo de traducciones bien estructurado, usa SuiteScript 2.1 y AMD correctamente. No hay riesgos de gobernanza ni de seguridad. Presenta pequeñas oportunidades de mejora como duplicados de claves y recreación innecesaria del objeto de etiquetas.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
duplicate_key	low	La clave 'pending' está definida dos veces en el bloque de idioma español, la segunda sobrescribe la primera.	Eliminar la clave duplicada o consolidar su valor en una única definición.	41, 111	-
duplicate_key	low	La clave 'pending' está definida dos veces en el bloque de idioma inglés.	Mantener una única definición de 'pending' en el objeto inglés.	151, 221	-
performance	low	El objeto 'labels' se crea en cada invocación de getTranslations, consumiendo memoria y CPU innecesariamente.	Declarar 'labels' fuera de la función (como constante del módulo) y reutilizarlo.	10	-
validation	low	No se valida que el parámetro 'language' sea una cadena no vacía antes de acceder al objeto.	Agregar validación de tipo y valor, lanzar error descriptivo si el parámetro es inválido.	10	-

Fix Plan

- Externalizar las traducciones a archivos JSON o a un registro de tipo 'Custom Record' para facilitar la gestión multilingüe.
- Eliminar claves duplicadas y consolidar valores.
- Mover el objeto 'labels' fuera de la función para evitar recreación en cada llamada.
- Implementar validación de entrada y manejo de errores claros.

PROYECTOS INBOUND/sdf-latam-inbound/ com.latamready.lmrylataminbound/src/FileCabinet/SuiteApps/ com.latamready.lmrylataminbound/Handlers/ LMRY_Inbound_Uret_HNDL.js

Script type: UserModule.js

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script declara SuiteScript 2.1 y usa AMD correctamente, pero contiene un error crítico de lógica asíncrona que devuelve resultados vacíos, variable sin uso y falta de validación de entrada.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
logic_error	critical	Se utiliza require([...], callback) de forma asíncrona pero la función devuelve 'result' inmediatamente, por lo que siempre retorna una cadena vacía.	Reemplazar la carga asíncrona por require('module') síncrono o cargar todos los routers al iniciar el módulo y devolver la referencia directamente.	46 - 48, 52	-
unused_variable	high	Variable declarada pero nunca utilizada.	Eliminar la variable o usarla en logs si es necesaria.	13	-
missing_error_return	medium	En caso de excepción la función no devuelve nada explícitamente.	Retornar null o lanzar la excepción después de loggear.	53, 54	-
hardcoded_ids	low	Los IDs de SuiteApp y rutas están hardcodeados.	Mover a constantes o a parámetros de script.	17, 22, 26, 30	-

Fix Plan

- Refactorizar getUserEventHandler para usar carga síncrona de módulos.
- Eliminar variables y código muerto.
- Agregar validación de entrada (country, type) y manejo de errores consistente.
- Externalizar los IDs de SuiteApp a constantes o parámetros de script.

PROYECTOS INBOUND/sdf-latam-inbound/ com.latamready.lmrylataminbound/src/FileCabinet/SuiteApps/ com.latamready.lmrylataminbound/Urets/LMRY_Inbound_URET.js

Script type: UserEvent

Audit score (0-10): 8

Final score (0-100): 80

API version: 2.1

El script está bien estructurado, usa SuiteScript 2.1, AMD y separación de responsabilidades. Presenta pequeñas oportunidades de mejora en gestión de gobernanza y código muerto.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
dead_code	low	La función beforeLoad está declarada pero nunca exportada ni utilizada; el código está comentado en el retorno.	Eliminar la función o exportarla si se necesita en el futuro.	20, 22	-
governance_risk	medium	Se ejecuta un search completo por cada registro en beforeSubmit y afterSubmit, lo que consume gobernanza innecesariamente.	Reemplazar por N/lookupFields o cachear resultados cuando sea posible.	88, 102	-
redundancy	low	Carga de config.CompanyInformation en cada ejecución cuando no es OneWorld.	Cachear la configuración a nivel de módulo o usar runtime.getCurrentScript().getParameter si es constante.	45, 46	-
logic_consistency	medium	afterSubmit siempre llama a getCountry sin verificar si la cuenta es OneWorld, a diferencia de beforeSubmit que lo hace.	Repetir la lógica de detección de OneWorld o extraerla a una función reutilizable.	75	-

Fix Plan

- Eliminar código muerto (beforeLoad) o habilitarlo si es necesario.
- Reemplazar search en getCountry por lookupFields o cachear resultados.
- Unificar la detección de OneWorld en una función reutilizable para beforeSubmit y afterSubmit.
- Cachear la información de compañía cuando no sea OneWorld para reducir llamadas a config.load.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Client/
LMRY_MX_InboundDetail_CLNT.js**

Script type: ClientScript

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

Script declarado como SuiteScript 2.1 pero mezcla APIs de 1.0 (nlapi*), mantiene estado global mutable y contiene código muerto e inalcanzable. Violaciones críticas de compatibilidad y arquitectura que impiden un mantenimiento seguro.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Uso de APIs legacy nlapi* (nlapiSetLineItemValue, nlapiGetFieldValue, nlapiGetLineItemCount) dentro de un script declarado como SuiteScript 2.1.	Reemplazar todas las llamadas nlapi* por sus equivalentes 2.x (p.ej., currentRecord, record, or N/ui/message) y eliminar dependencias legacy.	262, 268, 298, 300 - 303, 312, 313, 320	-
global_state	high	Variables mutables (purchaseOrderList, itemList, taxCodes, isNotRunning, etc.) se comparten entre todas las invocaciones del cliente, provocando efectos colaterales y race conditions en sesiones concurrentes.	Encapsular el estado dentro de objetos o usar closures por instancia; evitar variables globales en módulos client.	16 - 24	-
dead_code	high	Las funciones están definidas pero nunca exportadas en el objeto retornado, por lo que nunca pueden ser invocadas desde NetSuite.	Eliminar funciones no usadas o exportarlas si son requeridas por otros scripts.	259, 294, 295, 456, 457	-
unreachable_code	medium	El `return false;` se ejecuta antes del código que manipula `itemValues`, dejando ese bloque inalcanzable.	Reordenar la lógica: mover el `return false;` al final del bloque o eliminarlo si no es necesario.	242, 244 - 248, 250, 251	-
anti_pattern	medium	Función `fieldChanged` contiene más de 70 líneas y múltiples bucles anidados, manejando varios sublist IDs y lógica de negocio.	Dividir en funciones específicas (ej. handlePoChecked, handlePurchaseOrderChange, handleTaxCodeChange) y delegar la lógica.	119, 120, 127 - 150	-
magic_numbers	low	Uso de valores numéricos y strings hardcoded (p.ej., índices de línea `0`, `1`, `1000`) sin constantes descriptivas.	Definir constantes descriptivas para índices y tamaños de página.	262 - 290	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
security_issue	low	Se muestra al usuario el mensaje genérico `error_clnt` pero se registra el error completo en `console.log`, lo cual puede exponer información sensible en la consola del navegador.	Limitar la información registrada; usar `log.error` del módulo `N/log` y mostrar mensajes genéricos al usuario.	71, 73, 101, 104, 112, 114, 185, 186, 212, 214, 450, 452	-

Fix Plan

- Migrar todas las llamadas `nlapi*` a la API 2.x.
- Encapsular estado mutable en objetos de instancia o usar closures.
- Refactorizar `fieldChanged` en módulos más pequeños.
- Eliminar código muerto y exportar solo lo necesario.
- Reemplazar magic numbers por constantes descriptivas.
- Implementar manejo de errores con `N/log` y mensajes de usuario seguros.

PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrymxinbound/Client/LMRY_MX_Inbound_CLNT.js

Script type: ClientScript

Audit score (0-10): 7

Final score (0-100): 73

API version: 2.1

El script está escrito en SuiteScript 2.1 y sigue una arquitectura modular adecuada, delegando la lógica al handler. Sin embargo, presenta algunos problemas menores de robustez y limpieza que deben corregirse.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
null_handler_usage	medium	Las funciones fieldChanged y saveRecord asumen que la variable global 'handler' está inicializada por pageInit. Si el usuario dispara un evento antes de pageInit, handler será null y provocará un error de runtime.	Agregar una verificación de existencia de handler antes de usarlo o inicializarlo de forma segura en cada función.	22, 26	-
unused_dependency	low	El módulo 'N/url' (urlApi) se importa pero nunca se utiliza en el script.	Eliminar la importación o utilizarla si es necesaria.	8	-
debug_logging	low	Se deja un console.log(handler) que expone información interna en la consola del navegador.	Eliminar el console.log o reemplazarlo por un logger controlado por un flag de depuración.	18	-

Fix Plan

- Implementar validación de parámetros en toggleCheckBoxes (isApplied, sublistId).
- Agregar manejo de errores (try/catch) alrededor de operaciones de registro para evitar fallos silenciosos.
- Documentar claramente la expectativa de que pageInit se ejecute antes que cualquier otro evento.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Handlers/
LMRY_MX_InboundDetail_STLT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa características modernas, pero presenta graves problemas de arquitectura, seguridad y gobernanza. Los entry points son demasiado extensos, hay falta de validación de inputs, construcción insegura de código cliente y errores críticos en la generación de filtros de búsqueda.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	El método createForm supera los 200 líneas, mezcla lógica de UI, negocio y acceso a datos, y contiene múltiples bloques anidados.	Dividir la lógica en módulos de servicio (ej. FormBuilder, DataProvider) y delegar la creación de campos a funciones pequeñas.	77, 348	-
compatibility_break	critical	Construcción incorrecta del filterExpression: se inserta un array de filtros y strings "OR" directamente, lo que rompe la sintaxis esperada por search.create.	Construir el filtro como un array plano de expresiones, usando search.createFilter o combinando correctamente los operadores.	950, 989	-
security_issue	high	Los nombres de funciones cliente se construyen con interpolación directa de this.params.line sin sanitizar.	Validar y escapar los valores antes de incluirlos en strings de función, o usar métodos seguros de NetSuite (e.g., client script con parámetros).	311, 322, 332, 342	-
security_issue	medium	Se asignan valores de parámetros directamente a campos ocultos sin validación.	Validar y sanitizar todos los inputs del request antes de usarlos en la lógica del servidor.	125, 138, 151, 164, 297, 306	-
governance_risk	medium	Se ejecuta search.run().getRange(0, 1000) sin paginación adicional; si existen más de 1000 facturas, se omiten resultados.	Utilizar runPaged con paginación o iterar con while (remainingUsage) para procesar todos los resultados.	1182, 1194	-
obsolete_style	low	Uso de var y bucles for tradicionales en lugar de let/const y métodos de array.	Reescribir con const/let y usar .map/.forEach.	950, 956, 960	-
anti_pattern	low	No se captura excepción al usar search.lookupFields; si falla, el script lanzará error sin manejo.	Envolver la llamada en try/catch y registrar errores de forma controlada.	350, 374	-
bad_naming	low	Variables como FEAT_SUBS, FEAT_DPT, etc., usan mayúsculas sin const, lo que sugiere que son constantes pero se declaran con this.	Declarar como const dentro del constructor o usar nombres en camelCase.	40 - 42	-

Fix Plan

- Refactorizar createForm en varios módulos (FormBuilder, VendorInfo, WithholdingInfo).
- Corregir la generación de filtros en getItemDetailsByVendorCodes usando search.createFilter o filterExpression bien estructurada.
- Sanitizar todos los parámetros de entrada antes de usarlos en la UI o en llamadas a funciones cliente.
- Reemplazar var por let/const y usar métodos de array modernos para mayor claridad.
- Implementar paginación en fillManualBill y cualquier búsqueda que pueda superar 1000 resultados.
- Agregar manejo de errores (try/catch) en llamadas a lookupFields y parsing de XML.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Handlers/
LMRY_MX_Inbound_CLNT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa correctamente la arquitectura AMD, pero concentra demasiada lógica en una única clase cliente, viola SRP, contiene métodos muy extensos (fieldChanged >150 líneas) y repite patrones de carga de listas. Usa var y valores mágicos, construye URLs sin sanitizar y realiza búsquedas dentro de bucles que pueden impactar la gobernanza. Necesita una refactorización profunda.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	El método fieldChanged contiene más de 150 líneas, múltiples bucles anidados y lógica de UI, validación, construcción de URLs y apertura de ventanas emergentes.	Dividir la lógica en módulos de servicio: validación de UUID, construcción de URL, manejo de ventana emergente y actualización de sublista.	104, 255	-
anti_pattern	high	La clase combina responsabilidades de UI (alert, window), lógica de negocio (validaciones, creación de logs) y acceso a datos (search, query).	Separar en clases o módulos: UIHandler, ValidationService, DataService, LogService.	36, 381	-
obsolete_style	medium	Uso de var en varios lugares dentro de un script 2.1 donde let/const son preferibles.	Reemplazar var por const o let según corresponda.	132, 146, 158, 164, 169, 174, 179, 184, 190, 194	-
magic_numbers	medium	IDs de scripts, despliegues y valores de campos están escritos literalmente.	Externalizar a Script Parameters o constantes configurables.	32, 34, 35, 226, 227, 236 - 240	-
governance_risk	medium	Ejecuta una búsqueda de ScheduledScriptInstance en cada llamada a validateExecution, potencialmente dentro de bucles de guardado.	Cachear resultados o limitar la ejecución a una sola vez por request.	453, 475	-
security_issue	low	Se construyen URLs de Suitelet concatenando valores del registro sin sanitizar (xml, pdf, etc.).	Utilizar url.resolveScript con parámetros seguros y validar/escapar valores críticos.	225, 902	-
anti_pattern	low	Se almacena información de ventanas emergentes en el objeto global window, lo que puede colisionar con otras scripts.	Encapsular en un objeto interno o usar un módulo de estado.	253, 318, 322, 324, 327, 329	-

Fix Plan

- Refactorizar fieldChanged en varios servicios pequeños.
- Reemplazar var por let/const y eliminar código comentado.
- Externalizar IDs y strings a parámetros de script.
- Cachear resultados de validateExecution para evitar búsquedas repetidas.
- Sanitizar parámetros al construir URLs de Suitelet.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Handlers/
LMRY_MX_Inbound_STLT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa correctamente AMD, const/let y APIs modernas. Sin embargo, la clase LibraryHandler concentra demasiadas responsabilidades, contiene métodos muy extensos, usa valores mágicos y construye HTML sin sanitizar, lo que genera deuda técnica y riesgos de seguridad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	Método createFilterForm supera ampliamente las 50 líneas y mezcla lógica de UI, validaciones y generación de HTML, violando el principio de Responsabilidad Única.	Dividir en sub-módulos: generación de formulario, carga de listas, y manejo de mensajes. Cada sub-función no debe superar 30-40 líneas.	136, 411	-
magic_number	medium	Se usan códigos de estado hardcoded ('11','15','18',...) y tamaños de página fijos (1000) sin consts descriptivas.	Definir constantes enumeradas (e.g., STATUS_PENDING = '11') y una constante PAGE_SIZE = 1000.	1009, 1026, 1079, 1080, 1089	-
obsolete_style	low	Uso de var para departmentObj en lugar de const/let.	Reemplazar var por const.	1227	-
security_issue	medium	Se construye HTML mediante concatenación de valores externos (e.g., issuer_name, see_xml) sin sanitizar, lo que abre posibilidad de XSS si los datos provienen de fuentes no confiables.	Escapar/ sanitizar todos los valores antes de insertarlos en HTML o usar APIs de NetSuite que generan enlaces seguros.	150, 180, 681, 785	-
input_validation	medium	Los parámetros del request (subsidiary, startdate, enddate) se asignan directamente a campos del formulario sin validar su formato ni existencia.	Validar y sanitizar los parámetros antes de usarlos; aplicar checks de tipo y rango.	877, 902	-
redundant_code	low	Se llama a updateDisplayType dos veces consecutivas sobre el mismo campo.	Mantener una única llamada con el valor final deseado.	574, 576	-

Fix Plan

- Refactorizar LibraryHandler en varios módulos (FormBuilder, DataFetcher, Logger, MapReduceOrchestrator).
- Externalizar constantes de estado y tamaños de página.
- Implementar una función de escape HTML centralizada.
- Agregar validación de parámetros de entrada en setFormValues y en los métodos que reciben datos externos.
- Reemplazar var por const/let y eliminar código redundante.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Handlers/
LMRY_MX_VendorBillInbound_URET_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 7

Final score (0-100): 75

API version: 2.1

El script declara correctamente SuiteScript 2.1 y usa AMD, const/let y manejo de errores básico. Sin embargo presenta deuda técnica: número mágico, nombre de script incoherente, mezcla de responsabilidades en afterSubmit y oportunidades de optimizar búsquedas y evitar carga de registros innecesaria.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RIS
magic_number	high	Valor hardcodeado 18 para el campo custrecord_psg_ei_inbound_status.	Definir una constante o parámetro de script con el ID del estado y usarla en lugar del número literal.	58	-
bad_naming	medium	El @Name indica LMRY_PE_VendorBill_URET_HNDL.js mientras el archivo real es LMRY_MX_VendorBillInbound_URET_HNDL.js.	Actualizar el @Name para que coincida con el nombre del archivo o eliminarlo si no es necesario.	4	-
compatibility_risk	medium	Uso de String.replaceAll, característica ES2021, que puede no estar soportada en todas las versiones de GraalVM usadas por SuiteScript 2.1.	Reemplazar con replace usando expresión regular global o validar la compatibilidad del entorno.	104	-
governance_risk	high	Carga de registro (record.load) dentro de la función que se ejecuta por cada inbound record.	Utilizar search.lookupFields o cargar solo los campos necesarios; considerar cachear resultados si se procesan varios registros del mismo tipo.	123	-
spaghetti_code	medium	afterSubmit combina varias responsabilidades: extracción de XML, actualización de campos extra, y cambio de estado del registro.	Dividir la lógica en un servicio dedicado (e.g., VendorBillInboundService) y delegar cada acción a métodos separados.	40, 66	-
error_handling	low	No hay bloque try/catch alrededor de la lógica principal de afterSubmit.	Encapsular la lógica de afterSubmit en try/catch y registrar errores con nLog.error.	40, 66	-

Fix Plan

- Refactorizar afterSubmit en un módulo de servicio con métodos claros.
- Reemplazar valores mágicos por constantes o parámetros de script.
- Actualizar @Name para evitar confusión de identificación.
- Cambiar replaceAll por una alternativa compatible con ES2020.
- Optimizar acceso a registros usando lookupFields y caching.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/LMRY_MX_InboundPdf_STLT.js**

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.1

El script presenta múltiples fallos críticos: dependencia no importada (LMRY_EI_ENCRYPT), vulnerabilidad de open-redirect, variables globales implícitas, duplicación de módulos y parámetros, y lógica excesiva en el entry point. Necesita una re-escritura completa para cumplir con SuiteScript 2.1 y los principios SOLID.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
missing_dependency	critical	Se invoca LMRY_EI_ENCRYPT.B62.decode sin haber importado el módulo, provocando ReferenceError en tiempo de ejecución.	Importar el módulo correspondiente mediante define y agregarlo a los parámetros de la función.	53 - 55	-
duplicate_module	medium	El módulo "N/redirect" se incluye dos veces y el parámetro de función también está duplicado, generando confusión y posible sobrescritura.	Eliminar la duplicación del módulo y del parámetro; mantener una única referencia a "N/redirect".	19 - 22	-
global_variable	high	Variable "isSubsi" se asigna sin declaración (var/let/const), creando una variable global implícita.	Declarar la variable con const o let dentro del scope apropiado.	43 - 45	-
security_issue	high	Se construye una URL de redirección usando el encabezado Host provisto por el cliente, lo que permite un open-redirect y posible phishing.	Validar y sanitizar el valor de Host o usar una URL estática de confianza; evitar confiar en encabezados controlados por el cliente.	77 - 80	-
bad_practice	low	Se lanza una cadena de texto con throw en lugar de una instancia de Error.	Lanzar new Error('Access denied') o una clase de error personalizada.	49	-
typo	low	Variable "tipoDocuemnto" contiene un error tipográfico que dificulta la legibilidad.	Renombrar a "tipoDocumento" y actualizar todas sus referencias.	66	-

Fix Plan

- Reescribir el Suitelet siguiendo arquitectura modular: separar validación, carga de archivo y generación de respuesta en módulos independientes.
- Implementar inyección de dependencias mediante define y usar const/let para evitar variables globales.
- Agregar validación estricta de parámetros y sanitizar encabezados antes de usarlos en redirecciones.
- Incluir manejo de errores con objetos Error y registro estructurado.
- Eliminar código muerto y duplicado (módulo redirect duplicado).

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/LMRY_MX_InboundSAT_MPRD.js**

Script type: Map/Reduce

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa sintaxis moderna, pero contiene errores críticos de ejecución (referencia a variable inexistente, falta de validaciones nulas) y prácticas inseguras (URL http). Además, funciones monolíticas violan SRP y la lógica de manejo de errores es deficiente.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES
bug	critical	En el bloque catch se intenta usar la variable 'e' que no está definida (el error capturado se llama 'error'). Provoca ReferenceError y detiene la ejecución del Map/Reduce.	Reemplazar 'e' por 'error' en el objeto pasado a log.error o usar directamente log.error('title', error).	88 - 90
null_pointer	high	getSetupData puede devolver null si no se encuentra configuración, pero getInputData accede a sus propiedades sin validar, lo que genera TypeError.	Validar que setupData no sea null antes de usar sus campos; lanzar error controlado o devolver [] si falta configuración.	60 - 69, 405, 418 - 420
logic_error	medium	Al crear la subcarpeta, si no existe se llama a createPDFFolder('SuiteLatamReady', parentId) reutilizando el nombre del folder padre en lugar de crear la subcarpeta 'XML and PDF Inbound'.	Crear la subcarpeta con el nombre correcto usando createPDFFolder('XML and PDF Inbound', parentId).	322 - 325
security_issue	medium	La URL del servicio externo usa protocolo HTTP sin cifrado.	Cambiar a HTTPS y validar certificados; si el endpoint no soporta HTTPS, considerar un proxy seguro.	20
json_parse_error	high	Se hace JSON.parse(responseData.body) sin captura de excepción; si el cuerpo no es JSON válido el script lanzará error.	Envolver el parse en try/catch y manejar errores devolviendo [] o registrando el fallo.	40
unused_constant	low	Constante E_DOC_TYPE declarada pero nunca utilizada.	Eliminar la constante o usarla en la lógica correspondiente.	21
incorrect_error_count	low	Se asigna a errCount el resultado de iterator().each, que siempre devuelve true, no el número de errores.	Contar los errores dentro del each o usar summary.mapSummary.errors.length si está disponible.	96 - 99
style_issue	low	Uso de 'var' para objDocumentoCFDI en lugar de 'const' o 'let'.	Reemplazar por 'const objDocumentoCFDI = JSON.parse(context.value);'.	73
governance_risk	medium	Se ejecuta una búsqueda de vendor por cada registro procesado, lo que puede consumir rápidamente la gobernanza en lotes grandes.	Implementar caché de resultados de búsqueda por RFC o usar búsqueda batch antes del map.	75

Fix Plan

- Dividir getInputData en módulos: fetchService, filterExisting, enrichDocuments.

- Agregar validaciones nulas y manejo de errores robusto (try/catch alrededor de `JSON.parse` y `getSetupData`).
- Cambiar la URL del servicio a HTTPS y parametrizarla vía Script Parameter.
- Corregir el bug del catch en map y mejorar el logging estructurado.
- Implementar caché de búsqueda de vendors para reducir consumo de gobernanza.
- Revisar y corregir la lógica de creación de carpetas en `createPDFFile`.
- Contar correctamente los errores en `summarize` y exponer métricas útiles.
- Eliminar código muerto y constantes no usadas.
- Aplicar linting (ESLint) para uniformizar estilo (`const/let`, semicolons).

PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrymxinbound/LMRY_MX_InboundSAT_RSLT.js

Script type: RESTlet

Audit score (0-10): 7

Final score (0-100): 78

API version: 2.1

El Restlet está bien estructurado y cumple su objetivo de lanzar un Map/Reduce, pero presenta problemas de validación de entrada, uso de valores mágicos y dependencias no utilizadas.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
unused_import	low	El módulo 'N/record' se importa pero nunca se utiliza, generando código muerto y posible sobrecarga de carga.	Eliminar la dependencia o usarla si es necesaria.	6	-
magic_value	medium	Los IDs del script y despliegue del Map/Reduce están hardcoded, dificultando mantenimiento y reutilización.	Mover a constantes definidas o a parámetros de script.	15, 16	-
validation_missing	high	No se valida que requestBody y requestBody.processID existan ni su tipo, lo que puede provocar errores o vulnerabilidades.	Agregar validación explícita y devolver error 400 si falta o es inválido.	11	-
obsolete_style	low	Se utilizan declaraciones 'var' en lugar de 'const' o 'let', lo que puede generar hoisting inesperado.	Reemplazar por const/let según corresponda.	12, 14, 21	-

Fix Plan

- Agregar validación de requestBody.processID y devolver códigos HTTP apropiados.
- Reemplazar valores hardcoded por constantes o parámetros de script.
- Eliminar la dependencia N/record no utilizada.
- Actualizar declaraciones var a const/let para mayor claridad y seguridad.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/LMRY_MX_InboundSAT_SCHDL.js**

Script type: Scheduled

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script declara SuiteScript 2.1 pero contiene errores críticos de ejecución (uso de función antes de su definición), exposición de credenciales en logs y en cuerpo de petición, endpoints hardcodeados sin HTTPS y falta de manejo de gobernanza. Necesita una re-escritura completa.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se invoca getSATCredentials antes de su declaración (const arrow function). En JavaScript el temporal dead zone provoca ReferenceError.	Mover la definición de getSATCredentials antes de su uso o declararla como function tradicional.	35	-
security_issue	critical	Se registra en log el objeto bodyData que contiene rfc y password en texto plano.	Eliminar log.debug de bodyData o enmascarar los campos sensibles. Utilizar logging seguro.	39, 40	-
security_issue	high	Endpoints hardcodeados usan HTTP y direcciones IP públicas.	Convertir a HTTPS y mover la URL a un parámetro de script o registro de configuración.	15, 16	-
bad_naming	high	Constante declarada pero nunca utilizada.	Eliminar la constante o implementarla si es necesaria.	17	-
spaghetti_code	medium	El entry point supera las 50 líneas y contiene lógica de negocio, construcción de payload y llamada HTTP en un solo bloque.	Delegar la construcción del payload y la llamada HTTP a módulos de servicio separados.	28, 30, 34, 35, 37, 39, 42, 44, 53, 55	-
governance_risk	medium	No se verifica el consumo de gobernanza (runtime.getCurrentScript().getRemainingUsage()).	Agregar chequeos de uso restante y dividir lógica en etapas si es necesario.	30, 95	-
dead_code	low	Función definida pero nunca invocada en el script.	Eliminar o integrar la función donde sea requerida.	66, 90	-
anti_pattern	low	Si http.post lanza excepción, la variable response queda undefined y luego se accede a response.code, provocando TypeError.	Inicializar response con un objeto vacío o retornar dentro del catch, y validar su existencia antes de usarlo.	42, 49, 53, 56	-

Fix Plan

- Reordenar el código moviendo la definición de getSATCredentials antes de su uso.
- Externalizar URLs a parámetros de script y usar HTTPS.
- Eliminar logs que contienen credenciales y enmascarar información sensible.
- Dividir la función execute en servicios: obtención de credenciales, construcción de payload, envío HTTP.
- Implementar control de gobernanza y manejo seguro de errores de http.post.
- Eliminar código muerto (buscarVendorPorRFC, ENDPOINTS_SAVE).

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/LMRY_MX_InboundXML_STLT.js**

Script type: Suitelet

Audit score (0-10): 7

Final score (0-100): 75

API version: 2.1

El Suitelet está declarado correctamente para SuiteScript 2.1 y usa AMD. Implementa manejo de errores y parseo de XML, pero carece de validaciones de entrada, usa var en exceso y contiene código HTML embebido que dificulta el mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
security_issue	high	No se valida el parámetro inboundRecordId antes de usarlo en search.lookupFields, lo que puede generar errores o exposición a inyección de IDs no válidos.	Validar que inboundRecordId exista, sea numérico y pertenezca al tipo esperado antes de ejecutar la búsqueda; devolver mensaje de error amigable si la validación falla.	18 - 22	-
bad_naming	medium	Uso extensivo de var en lugar de const/let, lo que genera alcance de función y posibles sobrescrituras accidentales.	Reemplazar var por const cuando el valor no cambia y por let cuando sea mutable; aplicar const/let en todo el script.	24, 28, 32 - 34, 44 - 47, 49 - 59, 68 - 96	-
obsolete_style	medium	Se utilizan valores y IDs hardcodedados (tipo de registro, nombre de campo, HTML estático) en lugar de constantes o parámetros de script.	Definir constantes o parámetros de script para el tipo de registro, nombres de campos y fragmentos HTML reutilizables.	20 - 22, 45, 55	-
governance_risk	low	Uso de search.lookupFields dentro de un bucle no existe, pero no se verifica el consumo de gobernanza antes de la respuesta.	Agregar verificación de uso restante con runtime.getCurrentScript().getRemainingUsage() y manejar escenarios de límite excedido.	19 - 22	-
anti_pattern	low	Construcción manual de HTML mediante concatenación de strings, lo que es propenso a errores y difícil de mantener.	Utilizar serverWidget.Form y campos HTML predefinidos o plantillas externas para generar el contenido.	49 - 59	-

Fix Plan

- Validar y sanitizar inboundRecordId antes de usarlo.

- Reemplazar var por const/let y aplicar linting ES6.
- Externalizar HTML a plantillas o usar serverWidget para generar UI.
- Crear constantes o parámetros de script para tipos de registro y nombres de campos.
- Agregar verificación de uso de gobernanza y manejo de límites.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Module/
LMRY_MX_InboundDetail_STLT.js**

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.1

El script está declarado como SuiteScript 2.1 y sigue buenas prácticas de modularidad, pero presenta errores críticos que impiden su ejecución y riesgos de seguridad por falta de validación de entrada.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se invoca log.error sin haber importado el módulo N/log, lo que genera una excepción de referencia no definida.	Agregar "N/log" a la lista de dependencias del define y pasar el objeto log al callback.	47	-
security_issue	high	Los parámetros de la solicitud se pasan directamente al handler sin validación ni sanitización.	Validar y sanitizar todos los parámetros de entrada antes de utilizarlos o pasarlos al handler.	28, 30, 31	-
bad_naming	medium	Nombres de constantes en mayúsculas mezclados con camelCase (SCRIPT_NAME vs CLIENT_SCRIPT) y sin prefijo de módulo.	Adoptar una convención única, por ejemplo UPPER_SNAKE_CASE para constantes globales.	14, 15	-
incomplete_implementation	medium	El Suitelet solo maneja el método GET; no hay lógica para POST u otros métodos que podrían ser enviados.	Implementar manejo explícito de POST o devolver una respuesta clara indicando método no soportado.	52	-
anti_pattern	low	Construcción de HTML mediante concatenación de strings.	Utilizar plantillas o al menos separar la lógica de generación de HTML en una función reutilizable y segura.	62 - 72	-

Fix Plan

- Agregar la dependencia N/log y refactorizar la llamada a log.error.
- Implementar validación y sanitización de los parámetros de entrada.
- Completar el manejo de métodos HTTP distintos a GET.
- Unificar la convención de nombres de constantes.
- Reemplazar la concatenación de HTML por una plantilla segura.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Module/
LMRY_MX_InboundLogDetail_STLT.js**

Script type: Suitelet

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.

El script está estructurado como Suitelet 2.x pero contiene varias violaciones críticas de buenas prácticas: anotación de versión ambigua, fuga de variable global, búsquedas dentro de bucles que pueden superar los límites de gobernanza, falta de validación de parámetros y uso indiscriminado de var. El código funciona pero es frágil, difícil de mantener y presenta riesgos de rendimiento y seguridad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	medium	Se declara @NApiVersion 2.x, práctica prohibida. Debería especificarse 2.0 o 2.1 para evitar ambigüedad.	Cambiar a @NApiVersion 2.0 (o 2.1 si se usa ES6+).	10	-
global_variable_leak	high	Variable resultInboundEDoComplete se asigna sin declaración var/let/const, creando una variable global.	Declarar la variable con const o let dentro del ámbito del bucle.	228	-
governance_risk	high	Se ejecuta una búsqueda completa (search.create + run().getRange) y una búsqueda paginada de auditoría dentro de cada iteración del bucle.	Pre-cargar los datos necesarios fuera del bucle o usar búsquedas de tipo lookupFields / query con filtros IN y procesar los resultados en memoria.	204, 205, 210, 212, 228 - 230, 332	-
bad_naming	medium	Se reutiliza la variable 'i' declarada con var en varios bucles anidados, lo que genera sombra de alcance y posibles errores lógicos.	Usar let para cada bucle o nombres de variables diferentes.	188, 230	-
validation_missing	medium	No se valida que el parámetro 'logid' exista ni que sea numérico antes de usarlo en lookupFields. Tampoco se verifica que el parámetro de script 'LANGUAGE' esté definido antes de aplicar substring.	Agregar validaciones y manejo de errores antes de usar los parámetros.	104, 105, 108	-
security_issue	low	Se inserta HTML directamente en el valor de la sublista sin escape. Si el valor de 'refnum' proviene de datos externos, podría generar XSS.	Escapar o sanitizar cualquier dato que se inserte en HTML.	302	-
unused_variable	low	Variable 'inboundIds' declarada pero nunca utilizada.	Eliminar la declaración o usarla según sea necesario.	203	-
obsolete_style	low	Uso extensivo de var en lugar de const/let, contraviniendo las mejores prácticas de ES6.	Reemplazar var por const o let según corresponda.	32, 33, 37, 90	-

Fix Plan

- Refactorizar onRequest separando la lógica de negocio en servicios independientes.

- Consolidar búsquedas de inbound y audit trail en una única operación paginada antes del bucle.
- Reemplazar var por const/let y usar let para índices de bucle para evitar colisiones.
- Agregar validaciones robustas de parámetros de entrada y manejo de errores.
- Escapar datos antes de generar HTML para prevenir XSS.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Module/
LMRY_MX_InboundLog_CLNT.js**

Script type: ClientScript

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

Script cliente en SuiteScript 2.1 con lógica mínima pero contiene código muerto, nombres inconsistentes y falta de documentación en funciones exportadas.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
dead_code	low	Funciones pageInit, fieldChanged, validateField están definidas pero nunca exportadas ni utilizadas.	Eliminar funciones no usadas o exportarlas si son requeridas por la UI.	29, 44, 62, 75	-
bad_naming	low	Nombre 'funcionCancel' contiene error tipográfico y mezcla de idiomas.	Renombrar a 'functionCancel' o 'cancelNavigation' siguiendo la convención camelCase en inglés.	79	-
missing_documentation	low	La función exportada 'funcionCancel' carece de JSDoc que describa su propósito y parámetros.	Agregar bloque JSDoc con descripción y, si corresponde, parámetros de entrada.	79	-
commented_exports	low	Los handlers pageInit, fieldChanged y validateField están comentados en el objeto de retorno.	Decidir si se necesitan esos handlers; si no, eliminar su definición y los comentarios.	90 - 92	-

Fix Plan

- Eliminar código muerto y simplificar el módulo.
- Renombrar y documentar correctamente la función de cancelación.

- Considerar exportar `pageInit`, `fieldChanged` y `validateField` si la UI los requiere.
- Aplicar constantes para valores estáticos y evitar strings mágicos.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Module/
LMRY_MX_InboundLog_STLT.js**

Script type: Suitelet

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.0

El Suitelet funciona pero presenta graves problemas de arquitectura y gobernanza. El punto de entrada onRequest supera ampliamente los 50 líneas, carece de control de uso de gobernanza y no valida parámetros críticos. Además, hay hardcodeo de IDs y ausencia de paginación para resultados mayores a 1000.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	L
spaghetti_code	high	El punto de entrada onRequest contiene más de 200 líneas de lógica directa, violando el principio SRP y dificultando el mantenimiento.	Refactorizar extrayendo lógica a módulos de servicio (ej. InboundLogService.js) y mantener la función onRequest por debajo de 50 líneas.	1 4
governance_risk	high	No se verifica el consumo de unidades de gobernanza ni se implementa manejo de límites.	Agregar verificación de runtime.getCurrentScript().getRemainingUsage() dentro del bucle y dividir la carga en lotes o usar runPaged().	3 3
bad_naming	medium	Se asume que el parámetro 'LANGUAGE' siempre está presente; si es null, .substring() lanzará excepción.	Validar que el parámetro exista antes de usarlo y proporcionar un valor por defecto.	3 4
anti_pattern	medium	Se limita la obtención a los primeros 1000 registros sin paginación.	Utilizar search.runPaged() con un tamaño de página adecuado y procesar todas las páginas.	3 3
hardcoded_value	low	Los IDs de script y deployment están hardcoded.	Mover los IDs a parámetros de script o a constantes configurables.	4 4
inconsistent_style	low	Se usa el string literal 'textarea' en lugar de serverWidget.FieldType.TEXTAREA.	Reemplazar por serverWidget.FieldType.TEXTAREA.	2
obsolete_style	low	Uso extensivo de var en lugar de const/let, aunque permitido en 2.0, no sigue buenas prácticas modernas.	Migrar a const para valores que no cambian y let donde sea necesario.	3 3 3 4 4 1 3

Fix Plan

- Refactorizar onRequest en varios módulos (service, UI builder, utils).
- Implementar control de gobernanza y uso de runPaged para búsquedas masivas.
- Validar y sanitizar parámetros de entrada.
- Externalizar IDs de scripts y despliegues a parámetros de script.
- Actualizar declaraciones a const/let para mayor claridad.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Module/LMRY_MX_Inbound_MPRD.js**

Script type: Map/Reduce

Audit score (0-10): 2

Final score (0-100): 20

API version: 2.0

El script declara @NApiVersion 2.0 pero usa sintaxis ES6 (const) y contiene errores críticos de ejecución (uso de variables antes de su definición, variables implícitas). Además hay duplicación de funciones, carga masiva de records dentro de bucles y violaciones de SOLID que hacen el código altamente frágil y de bajo rendimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Uso de `const` en SuiteScript 2.0 (Nashorn) que solo soporta ES5. Provocará error de sintaxis en tiempo de carga.	Reemplazar `const` por `var` o actualizar a @NApiVersion 2.1 si se desea usar ES6.	1213	-
runtime_error	critical	Se construye `logParams` usando `entityId` antes de que la variable sea asignada (se define en línea 194). `entityId` será undefined.	Definir `entityId` antes de crear `logParams` o mover la construcción del objeto después de la asignación.	188 - 192, 194	-
global_leak	critical	Variable `conversionError` se asigna sin declaración (`var/let/const`), creando una variable global implícita.	Declarar la variable con `var conversionError;` antes de su uso.	310 - 315	-
duplicate_function	high	Funciones `areItemAmountsEqual` y `filterPOLinesWithInvoiceItems` están definidas dos veces, la segunda sobrescribe la primera.	Eliminar duplicados y consolidar la lógica en una única implementación.	573, 592, 595, 696, 698, 717, 720, 827	-
governance_risk	high	Se carga cada registro de ítem (`record.load`) dentro de un bucle que puede iterar sobre muchos ítems.	Cachear resultados o usar búsquedas masivas (`search.lookupFields`) para obtener datos de ítems en lote.	511 - 515	-
governance_risk	high	Dentro de bucles se crean líneas de sublista usando `record.selectNewLine` y `record.setCurrentSublistValue` para cada ítem, lo que incrementa significativamente el consumo de gobernanza.	Optimizar la creación de líneas mediante batch o validar la necesidad de crear cada línea individualmente.	1034 - 1051	-
magic_numbers	medium	Los códigos de estado (`14`, `15`, `16`) están hardcoded sin explicación ni enumeración descriptiva.	Definir un enum o constantes descriptivas con nombres claros.	33 - 35	-
error_handling	medium	En caso de error se intenta actualizar `paramIdDocuments` que podría no estar definido si la excepción ocurre antes de su asignación.	Validar la existencia de `paramIdDocuments` antes de usarlo o mover la lógica de actualización fuera del bloque catch.	74 - 87	-

Fix Plan

- Actualizar a SuiteScript 2.1 para usar sintaxis moderna y eliminar incompatibilidades.
- Refactorizar funciones críticas separando responsabilidades (p.ej., crear servicios de conversión, utilidades de búsqueda).
- Eliminar duplicados y consolidar lógica común.
- Implementar caché o búsquedas en lote para reducir carga de records dentro de bucles.
- Declarar todas las variables con var/let/const y evitar globals.
- Reemplazar códigos mágicos por constantes descriptivas.
- Agregar validaciones de parámetros antes de usarlos en bloques catch.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Module/LMRY_MX_Inbound_STLT.js**

Script type: Suitelet

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

El script declara correctamente SuiteScript 2.1 y usa módulos AMD, pero el punto de entrada Suitelet está sobrecargado, contiene lógica duplicada y registra parámetros sin sanitizar, lo que genera deuda técnica y riesgos de mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	El método onRequest supera las 50 líneas y combina múltiples responsabilidades (validación, construcción de formularios, redirecciones, ejecución de Map/Reduce).	Dividir la lógica en funciones auxiliares o servicios (ej. buildForm, handleGet, handlePost) y delegar a módulos externos.	24 - 26, 28 - 34, 36 - 39, 41 - 46, 48 - 59, 61, 63 - 65, 67, 71 - 85, 88, 89, 91	-
duplicate_logic	medium	Los bloques para status === 1 y status === 2 son idénticos, generando código redundante.	Unificar la lógica en una única rama que se ejecute cuando status sea 1 o 2, o extraerla a una función reutilizable.	48 - 59	-
security_issue	low	Se registra directamente el objeto params sin sanitizar, lo que puede exponer datos sensibles en los logs.	Sanitizar o eliminar datos sensibles antes de loguear, o usar log.debug con información mínima.	29	-
hardcoded_path	low	La ruta del client script está hardcodeada como '../Client/LMRY_MX_Inbound_CLNT.js'.	Externalizar la ruta a un parámetro de script o a una constante configurada en un archivo de configuración.	15	-
suboptimal_variable_declaration	low	Se usan 'let' donde la variable nunca se reasigna (por ejemplo, params, handler).	Cambiar a 'const' para variables inmutables.	28, 31 - 34	-

Fix Plan

- Refactorizar onRequest en funciones pequeñas: handleGet, handlePost, buildForm, redirectIfNeeded.
- Eliminar código duplicado para status 1 y 2 mediante una función común.
- Sanitizar los parámetros antes de registrarlos en los logs.
- Reemplazar 'let' por 'const' donde corresponda para mejorar la claridad del código.
- Externalizar rutas y valores configurables a parámetros de script o archivo de configuración.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Plugin/
LMRY_MX_Inbound_Email_Capture.js**

Script type: UserModule.js

Audit score (0-10): 2

Final score (0-100): 25

Script escrito en SuiteScript 1.0 con graves incompatibilidades: uso de expresiones de filtro estilo 2.x, llamadas a métodos inexistentes (fromAddress.getEmail()), variable global sin declarar, parámetro erróneo en nlapiSendEmail y referencia a variable no definida. La arquitectura está muy acoplada, con lógica de negocio mezclada en el entry point y múltiples violaciones a SOLID.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se utiliza una expresión de filtro estilo SuiteScript 2.x (array de condiciones) en un script 1.0 que solo soporta nlobjSearchFilter objects.	Reemplazar filterExpression por un array de nlobjSearchFilter objetos.	196 - 200	-
compatibility_break	critical	Uso de expresión de filtro estilo 2.x para buscar clientes.	Construir filtros con nlobjSearchFilter en lugar de arrays de condiciones.	240 - 246	-
compatibility_break	critical	Segunda expresión de filtro estilo 2.x para buscar por email del remitente.	Convertir a nlobjSearchFilter objects.	262 - 268	-
bad_naming	critical	Llamada a fromAddress.getEmail() cuando fromAddress es una cadena de texto.	Usar directamente fromAddress (es una cadena) o extraer el email con lógica adecuada.	373	-
bad_naming	critical	Mismo error: fromAddress.getEmail() en identificación de cliente.	Eliminar .getEmail() y pasar la cadena directamente.	402	-
anti_pattern	high	Variable innerXML asignada sin declaración (var/let/const), creando una variable global implícita.	Declarar con const innerXML = updateXMLContent(xmlContent);	327	-
compatibility_break	high	Se construye un array 'records' y se pasa como séptimo parámetro de nlapiSendEmail, que corresponde a attachments, no a registros de autor.	Eliminar el uso de 'records' y pasar null o un array de archivos según la firma correcta.	169, 170, 172 - 180	-
undefined_variable	high	Referencia a variable edocStandardId que nunca se declara.	Reemplazar por la variable correcta (por ejemplo, edocPackage) o definir edocStandardId.	394	-
magic_number	medium	ADMIN_ROLE está hardcodeado como 3.	Obtener el ID del rol mediante búsqueda o parámetro de script.	24	-
hardcoded_string	medium	Nombre de carpeta PDF está hardcodeado.	Mover a parámetro de script o registro de configuración.	25	-

Fix Plan

- Migrar a SuiteScript 2.1 para aprovechar ES6+, promesas y mejor manejo de módulos.
- Reescribir las búsquedas usando nlobjSearchFilter en lugar de expresiones de filtro 2.x.
- Separar la lógica de envío de email en un módulo independiente y usar inyección de dependencias.
- Convertir variables globales en constantes de módulo o parámetros de script.
- Implementar manejo de errores más robusto y evitar referencias a variables no definidas.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Plugin/
LMRY_MX_Inbound_PLGNIMP.js**

Script type: PlugInImplementation

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.0

El script declara SuiteScript 2.0 pero contiene errores críticos de lógica, alta complejidad en el punto de entrada y varias malas prácticas que comprometen mantenibilidad y fiabilidad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
logic_error	high	Asignación en condición `if (isMultipleRetentions = true)` convierte siempre la condición en true y sobrescribe el valor original.	Reemplazar por `if (isMultipleRetentions) {` o `if (isMultipleRetentions === true) {`.	77	-
variable_shadowing	high	Se declara `var withholdingTaxCode` dentro del bucle, ocultando la variable externa declarada en la línea 98.	Eliminar la declaración interna y asignar directamente a la variable externa.	153	-
fat_entry_point	high	La función `inject` supera los 250 líneas y combina extracción XML, lógica de negocio, búsquedas, validaciones y construcción de respuesta.	Dividir la función en módulos de servicio: parser XML, cálculo de retenciones, validación SAT, construcción de data source.	34, 279	-
governance_risk	medium	Uso de `search.lookupFields` dentro de bucles anidados para cada tasa de retención.	Pre-cargar los códigos de retención en un mapa mediante una única búsqueda y reutilizarlo dentro del bucle.	145, 155, 170	-
security_issue	medium	Construcción de la petición SOAP mediante concatenación de valores sin sanitizar.	Escapar o validar los parámetros antes de incluirlos en el XML; usar plantillas seguras.	533, 538	-
anti_pattern	medium	Llamada a `withholdingItems.pop();` elimina arbitrariamente el último elemento sin justificación.	Revisar la necesidad de esta operación; eliminarla o reemplazar por filtrado explícito.	195	-
dead_code	low	Función `pad` nunca es utilizada.	Eliminar o reutilizar la función.	514, 515	-
dead_code	low	Función vacía sin implementación ni uso.	Eliminar o implementar la lógica requerida.	518, 520	-
style_issue	low	Uso de snake_case (`obj_injection`) en lugar de camelCase, inconsistente con el resto del código.	Renombrar a `objInjection` y actualizar todas sus referencias.	75 - 77	-
unused_variable	low	Variable `tagsJSON` se declara y se llena, pero algunos campos nunca se usan fuera de la función.	Eliminar campos no necesarios o documentar su uso futuro.	398	-

Fix Plan

- Refactorizar `inject` en módulos separados (XMLParser, RetentionCalculator, SATValidator, DataSourceBuilder).
- Corregir la condición de asignación y el shadowing de `withholdingTaxCode`.
- Reemplazar bucles con búsquedas por mapas pre-cargados para reducir consumo de gobernanza.
- Implementar sanitización de parámetros en la petición SOAP a SAT.
- Eliminar código muerto (pad, getSetupValidationSAT) y variables no usadas.

PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrymxinbound/Plugin/ LMRY_MX_Inbound_PLGNTYPE.js

Script type: PlugInImplementation

Audit score (0-10): 8

Final score (0-100): 85

API version: 2.0

El script está correctamente declarado como plugintypeimpl en SuiteScript 2.0, usa AMD y respeta las limitaciones de la versión. No hay problemas de compatibilidad, gobernanza ni seguridad evidentes. La única observación es la ausencia de uso de los parámetros recibidos, lo que indica lógica incompleta o potencialmente redundante.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
unused_parameter	low	Los parámetros 'params' recibidos nunca se utilizan dentro de la función, lo que genera código muerto y dificulta la mantenibilidad.	Validar y utilizar los objetos params (inboundEDocRec, templateRec, entityRec) o eliminar el parámetro si no es necesario.	20, 21	-

Fix Plan

- Implementar la lógica real de inyección usando los parámetros proporcionados.
- Agregar validación de entrada y manejo de errores para robustez.
- Documentar claramente el propósito del DataSource 'custom' y considerar externalizar constantes.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Router/LMRY_MX_Inbound_ROUT.js**

Script type: UserModule.js

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara SuiteScript 2.1 pero contiene un error crítico de lógica al cargar dinámicamente módulos con require asíncronico y devolver el resultado antes de que se haya cargado. Además carece de validación de entrada y manejo adecuado de errores.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se utiliza require([...], callback) de forma asíncrona dentro de una función que devuelve inmediatamente el valor. El callback asigna a 'result' después de que la función ya ha retornado, por lo que siempre se devuelve una cadena vacía.	Reemplazar la carga dinámica por require('modulePath') sin callback (carga síncrona) o devolver una Promise y adaptar los consumidores. Ejemplo: const UserEvent_HNDL = require(userEventRouter[type]); return UserEvent_HNDL;	23, 24, 28	-
bad_naming	high	El nombre sugiere un router, pero su contenido es un simple objeto de mapeo estático.	Renombrar a 'USER_EVENT_MAP' y declararlo como constante fuera de la función.	17	-
security_issue	medium	No se valida que 'type' sea una cadena esperada; cualquier valor puede provocar una búsqueda de módulo inexistente.	Validar contra una lista blanca de tipos permitidos y lanzar un error descriptivo si no coincide.	15	-
obsolete_style	low	Inicializar con "" es innecesario cuando la función debería devolver directamente el módulo o lanzar excepción.	Eliminar la variable y retornar directamente el módulo cargado.	21	-

Fix Plan

- Reescribir getUserEventRouter usando carga síncrona o Promise.
- Externalizar el mapa de tipos a un módulo de constantes.
- Agregar validación de entrada y manejo de errores robusto.
- Documentar claramente el contrato del router y sus tipos soportados.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Setup/
LMRY_MX_InboundSetup_CLNT.js**

Script type: ClientScript

Audit score (0-10): 5

Final score (0-100): 45

API version: 2.1

El script declara SuiteScript 2.1 y usa características modernas, pero presenta funciones infladas, código duplicado y falta de manejo de errores críticos que comprometen mantenibilidad y robustez.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	La función pageInit supera los 50 líneas y contiene múltiples responsabilidades (carga de configuración, mapeo de campos, manejo de sublista).	Refactorizar extrayendo la lógica de búsqueda y mapeo a módulos de servicio separados; mantener pageInit como orquestador.	10, 124	-
spaghetti_code	high	La función fieldChanged también supera los 50 líneas y replica gran parte de la lógica de pageInit.	Crear una función reutilizable que reciba la subsidiaryId y realice la búsqueda y mapeo; invocarla desde pageInit y fieldChanged.	127, 258	-
anti_pattern	medium	Se llama a JSON.parse sin manejo de excepciones; datos corruptos pueden lanzar errores en tiempo de ejecución.	Envolver cada JSON.parse en try/catch y registrar/loggear errores de forma segura.	65, 70, 75, 181, 186, 191	-
bad_naming	medium	Uso de var en lugar de const/let en un entorno 2.1 donde ES6 está disponible.	Reemplazar var por const o let según corresponda.	11, 12, 128, 129	-
magic_number	low	Se asigna directamente el valor 1 a custpage_subsiary cuando la característica de subsidiarias está desactivada.	Definir una constante o parámetro de script para el ID de subsidiaria predeterminado.	15, 19	-
missing_error_handling	low	No se captura posible excepción al ejecutar la búsqueda.	Envolver la ejecución de la búsqueda en try/catch y manejar errores de forma controlada.	42, 158	-

Fix Plan

- Extraer la lógica de búsqueda y mapeo a un módulo de servicio reutilizable.
- Implementar manejo de errores robusto para JSON.parse y búsquedas.
- Actualizar variables a const/let y eliminar código duplicado.
- Externalizar IDs y valores mágicos a constantes o parámetros de script.

**PROYECTOS INBOUND/sdf-mexico-inbound/
com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrymxinbound/Setup/
LMRY_MX_InboundSetup_STLT.js**

Script type: Suitelet

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script usa SuiteScript 2.1 y está modularizado, pero el punto de entrada onRequest contiene lógica excesiva, hay errores de UI (grupos duplicados), variables de característica no inicializadas y riesgos de gobernanza y seguridad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	high	Se añaden dos field groups con el mismo id 'inboundgroup', provocando conflicto de UI y posible sobrescritura.	Eliminar la definición duplicada o usar ids únicos para cada group.	23, 28	-
bug	high	La variable FEAT_SUBSI nunca se inicializa antes de la solicitud GET, por lo que la condición de filtrado de subsidiarias mexicanas nunca se ejecuta.	Inicializar FEAT_SUBSI al inicio del GET (similar a FEAT_DPT, FEAT_LOC, FEAT_CLASS) o pasarla como parámetro.	332	-
spaghetti_code	high	El entry point supera ampliamente las 50 líneas, combina lógica de UI, validación, búsquedas y persistencia en un solo bloque.	Dividir la lógica en servicios: UIBuilder, RequestProcessor, RecordHandler y reutilizar módulos externos.	13, 322	-
governance_risk	medium	Se invoca search.lookupFields dentro del bucle each, generando una llamada de gobernanza por cada resultado.	Reemplazar lookupFields por una búsqueda conjunta (join) o cachear resultados fuera del bucle.	409, 414	-
security_issue	high	La contraseña del SAT se guarda en texto plano en un registro custom, aunque se evita sobrescribir si está enmascarada.	Encriptar la contraseña antes de guardarla o usar un vault seguro (SuiteVault) y nunca almacenar texto plano.	290, 311	-
magic_numbers	low	Se usan strings de tipo de registro y IDs internos directamente en el código.	Definir constantes o parámetros de script para estos valores.	258, 386, 390	-

Fix Plan

- Refactorizar onRequest en módulos separados (UIBuilder, DataProcessor, RecordService).
- Corregir los field groups duplicados y asegurar IDs únicos.
- Inicializar todas las variables de característica antes de usarlas.
- Optimizar fillWhtTaxCodeSelect usando joins o cache para evitar múltiples lookupFields.
- Implementar encriptación para la contraseña del SAT o usar SuiteVault.
- Externalizar strings y IDs a constantes o parámetros de script.

PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.lmrymxinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrymxinbound/Translator/ LMRY_MX_Inbound_Translator.js

Script type: UserModule.js

Audit score (0-10): 7

Final score (0-100): 78

API version: 2.1

El módulo declara correctamente SuiteScript 2.1 y sigue buenas prácticas de arquitectura, pero contiene claves duplicadas que sobrescriben valores y recrea el objeto de traducciones en cada invocación, generando riesgo de comportamiento inesperado y ligera ineficiencia.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
duplicate_key	high	La clave 'pending' está declarada dos veces en cada bloque de idioma, lo que provoca que el primer valor sea sobrescrito sin advertencia.	Eliminar la duplicación y mantener una única definición por clave. Revisar todo el objeto para asegurar que no existan más claves repetidas.	41, 109, 150, 218	-
performance_inefficiency	low	El objeto 'labels' se crea en cada llamada a la función, consumiendo memoria y CPU innecesariamente.	Definir 'labels' fuera de la función (const LABELS = {...}) y devolver LABELS[language] o una copia inmutable si se requiere.	9	-
missing_documentation	low	La función carece de JSDoc que describa parámetros y valor de retorno.	Agregar /** @param {string} language - Código de idioma (es en) @returns {Object} Mapa de traducciones */ antes de la declaración.	9	-

Fix Plan

- Externalizar el objeto de traducciones a un archivo JSON estático y cargarlo una sola vez.

- Implementar un mecanismo de fallback más robusto (p.ej., validar que el idioma exista antes de acceder).
- Agregar pruebas unitarias que verifiquen la ausencia de claves duplicadas y la correcta devolución de traducciones.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Client/
LMRY_PE_InboundDetail_CLNT.js**

Script type: ClientScript

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara @NApiVersion 2.1 y usa AMD, pero mezcla APIs legacy 1.0 (nlapi*) dentro de un cliente 2.1, contiene código muerto, funciones infladas y alta mutabilidad global. Riesgo de incompatibilidad y fallos en runtime.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Uso de APIs legacy 1.0 (nlapiSetLineItemValue, nlapiGetFieldValue, nlapiGetLineItemCount) dentro de un script declarado como SuiteScript 2.1. Incompatible con el runtime GraalVM y provocará errores de ejecución.	Reemplazar todas las llamadas nlapi* por sus equivalentes 2.x (N/record, currentRecord, etc.) o mover esa lógica a un Suitelet/RESTlet si es necesario.	463, 496	-
fat_entry_point	high	pageInit contiene más de 50 líneas de lógica directa, incluye parsing de XML, bucles, llamadas a búsquedas y manipulación de UI.	Descomponer pageInit en servicios auxiliares (carga de datos, configuración de UI, validaciones) y delegar la lógica a módulos separados.	54, 91	-
dead_code	high	Después de `return false;` el código que modifica `isNotRunning` y actualiza `itemValues` es inalcanzable.	Eliminar el código inalcanzable o reordenar la lógica para que se ejecute antes del return.	311 - 319	-
global_state	medium	Uso extensivo de variables mutables globales (`purchaseOrderList`, `bills`, `itemList`, `isNotRunning`, etc.) que son modificadas por múltiples funciones.	Encapsular el estado dentro de objetos o pasar como parámetros; evitar mutabilidad global.	17 - 27	-

logging_in_production	medium	Uso extensivo de console.log para depuración en código productivo.	Eliminar o envolver los logs en una condición de entorno (debug mode).	80, 84, 85, 386, 387, 400, 408, 410, 414, 416, 418, 420, 422, 424, 426, 428, 430, 432, 434, 436, 438, 440, 442, 444, 446, 448, 450, 452, 454, 456, 458, 460, 462, 464, 466, 468, 470, 472, 474, 476, 478, 480, 482, 484, 486, 488, 490, 492, 494, 496, 498, 500, 502, 504, 506, 508, 510, 512, 514, 516, 518, 520, 522, 524, 526, 528, 530, 532, 534, 536, 538, 540, 542, 544, 546, 548, 550, 552, 554, 556, 558, 560, 562, 564, 566, 568, 570, 572, 574, 576, 578, 580, 582, 584, 586, 588, 590, 592, 594, 596, 598, 600, 602, 604, 606, 608, 610, 612, 614,	-
-----------------------	--------	--	--	---	---

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
				616, 618, 620, 622, 624, 626, 628, 630, 632, 634, 636, 638, 640, 642, 644, 646, 648, 650, 652, 654, 656, 658, 660, 662, 664, 666, 668, 670, 672, 674, 676, 678, 680, 682, 684, 686, 688, 690, 692, 694, 696, 698, 700, 702, 704, 706, 708, 710, 712, 714, 716, 718, 720, 722, 724, 726, 728, 730, 732, 734, 736, 738, 740, 742, 744, 746, 748	
hardcoded_strings	low	Cadenas de texto y IDs de campos están hardcodeados.	Mover constantes a un módulo de configuración o usar Script Parameters.	1, 3 - 6, 31 - 39, 42 - 51	-

Fix Plan

- Reemplazar todas las llamadas nlapi* por APIs 2.x.
- Refactorizar pageInit y otras funciones largas en módulos de servicio.
- Eliminar código muerto y reducir variables globales.
- Implementar un mecanismo de logging controlado por entorno.
- Externalizar mensajes y constantes a archivos de configuración.

PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrypeinbound/Client/LMRY_PE_Inbound_CLNT.js

Script type: ClientScript

Audit score (0-10): 8

Final score (0-100): 80

API version: 2.1

Script cliente en SuiteScript 2.1 bien estructurado, delega lógica a un handler y usa sintaxis moderna. Presenta pequeñas áreas de mejora como validaciones de null y eliminación de console.log.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
null_reference	medium	El script asume que 'handler' está inicializado; si pageInit no se ejecuta antes, se producirá un error de referencia nula.	Agregar verificación de existencia (if (handler) {...}) o lanzar error controlado.	22, 26	-
debug_logging	low	Uso de console.log en producción genera ruido en los logs y puede exponer información interna.	Eliminar o reemplazar por un logger configurado con nivel apropiado.	18	-
performance_suboptimal	low	Uso de selectLine + setCurrentSublistValue dentro de bucle; puede ser menos eficiente que setSublistValue.	Reemplazar por recordObj.setSublistValue({ sublistId, fieldId: 'apply', line: i, value: isApplied }) para evitar selección de línea.	35, 37 - 39	-

Fix Plan

- Implementar manejo de errores con try/catch en los métodos delegados.
- Añadir validaciones de entrada en toggleCheckBoxes (sublistId y isApplied).
- Documentar el contrato del handler para asegurar consistencia entre versiones.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Handlers/
LMRY_PE_InboundDetail_STLT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa AMD correctamente, pero la clase LibraryHandler concentra demasiada lógica (UI, parsing XML, búsquedas, reglas de negocio) violando SRP y generando un punto de entrada gordo. Hay problemas de gobernanza (búsqueda duplicada sin uso), riesgos de seguridad (logueo de XML y generación de HTML sin sanitizar) y estilo obsoleto (uso de var). Necesita una refactorización profunda.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	El método createForm supera los 200 líneas, mezcla creación de UI, lógica de negocio y llamadas a servicios, violando el principio KISS y dificultando el mantenimiento.	Extraer la generación de campos y botones a módulos/servicios independientes (FormBuilder), mantener createForm como orquestador mínimo.	85, 292	-
anti_pattern	high	La clase LibraryHandler agrupa más de 20 responsabilidades (parsing XML, búsqueda de datos, construcción de formularios, lógica de negocio, generación de sublistas, redirecciones).	Dividir la clase en varios servicios: XmlParser, DataProvider, FormBuilder, SublistFiller, y un controlador ligero que coordine.	36, 1929	-
governance_risk	medium	Se ejecuta una búsqueda rawBillSearch (líneas 1198-1210) que nunca se utiliza, consumiendo gobernanza innecesariamente.	Eliminar la búsqueda rawBillSearch o reutilizar sus resultados; mantener solo la búsqueda billSearch.	1198, 1210	-
security_issue	high	Se registra el contenido XML completo en log.debug y se inserta directamente en un campo LONGTEXT sin sanitizar, exponiendo datos sensibles y potencialmente permitiendo XSS en el cliente.	Eliminar o sanitizar los logs de XML; escapar o validar el contenido antes de asignarlo a campos que se renderizan en el cliente.	236, 237, 1432, 1443	-
obsolete_style	low	Uso de la palabra clave var en lugar de const/let.	Reemplazar var por const o let según corresponda.	658	-
hardcoded_id	low	Los IDs de script y deployment están hardcodeados.	Mover los IDs a constantes o a parámetros de script.	353, 359	-

Fix Plan

- Refactorizar LibraryHandler en módulos especializados (Parser, Service, UI Builder).
- Eliminar búsquedas duplicadas y optimizar filtros para reducir consumo de gobernanza.
- Sanitizar y limitar la exposición de datos sensibles en logs y campos HTML.
- Reemplazar var por const/let y aplicar linting para estilo consistente.
- Externalizar IDs y textos estáticos a constantes o a parámetros de script.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Handlers/
LMRY_PE_Inbound_CLNT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa sintaxis moderna, pero concentra demasiadas responsabilidades en una única clase, contiene funciones infladas (>50 líneas), mezcla lógica de UI, validación y persistencia, y presenta varios errores de implementación y malas prácticas que comprometen mantenibilidad y seguridad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
anti_pattern	critical	Violación del Principio de Responsabilidad Única (SRP). La clase maneja UI, validaciones, búsquedas, creación de registros y lógica de navegación, lo que genera un código espagueti y dificulta pruebas unitarias.	Dividir la clase en módulos especializados: UIHandler, ValidationService, DataService, LogService y utilizar inyección de dependencias.	35, 58, 100, 230, 279, 386, 438, 487, 537, 587, 657	-
spaghetti_code	high	Método con más de 120 líneas, anidación profunda y lógica compleja para construir URL y abrir ventanas emergentes.	Extraer la lógica de construcción de parámetros y apertura de ventana a funciones auxiliares; limitar la longitud a <60 líneas.	101, 162, 188, 189, 217	-
spaghetti_code	high	Método extenso (>70 líneas) que combina iteración de sublist, parsing JSON, creación y guardado de registro custom.	Separar la generación del payload y la persistencia en servicios independientes; reutilizar funciones de validación.	658, 666, 672, 680, 688, 702, 710	-
bad_naming	medium	Uso de nombres genéricos y abreviados (e.g., 'filter', 'xml', 'pdf') sin contexto claro; además, línea 158 usa getValue con sintaxis incorrecta.	Renombrar variables con prefijos descriptivos y corregir la llamada a getValue({fieldId: 'custpage_filter'}).	158, 190 - 203	-
security_issue	medium	Se exponen mensajes de error mediante alert() y console.log, potencialmente revelando información interna al usuario.	Implementar un mecanismo centralizado de logging que registre errores en un registro interno y muestre mensajes genéricos al usuario.	90, 92, 93, 223 - 225, 269 - 271	-
obsolete_style	low	Uso de 'var' en varios lugares dentro de un script 2.1 donde 'let' y 'const' son preferibles.	Reemplazar 'var' por 'let' o 'const' según corresponda.	116, 122, 128, 133, 138, 143, 148, 154, 158	-
governance_risk	low	Uso de getRange sin paginación en potenciales listas grandes; aunque limitado a 1000, podría superar límites en entornos con muchos periodos.	Utilizar runPaged() con manejo de páginas para asegurar escalabilidad.	414, 466, 470	-

Fix Plan

- Refactorizar ClnHandler en varios servicios (UI, Validation, Data, Logging).
- Implementar manejo centralizado de errores y logging seguro.
- Reemplazar 'var' por 'let'/'const' y aplicar linting para consistencia.
- Optimizar búsquedas usando runPaged() o SuiteQL donde sea posible.
- Eliminar código muerto y comentarios innecesarios para reducir ruido.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Handlers/
LMRY_PE_Inbound_STLT_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 5

Final score (0-100): 45

API version: 2.1

El script está declarado como SuiteScript 2.1 y usa sintaxis moderna, pero presenta graves problemas de arquitectura (SRP violado, métodos gigantes), lógica inconsistente (índices desalineados entre XML y listas), y falta de validaciones de seguridad en datos externos. La gobernanza es aceptable pero podría exceder límites en entornos de alto volumen.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
fat_entry_point	high	Método con más de 200 líneas, mezcla creación de UI, lógica de negocio y generación de URLs.	Dividir en sub-módulos: un builder de formulario, un generador de URLs y un validador de parámetros.	130, 402	-
spaghetti_code	high	Método extenso con múltiples bucles anidados, llamadas a url.resolveScript dentro del bucle y lógica condicional compleja.	Extraer la generación de enlaces a funciones auxiliares y pre-calcular datos fuera del bucle.	578, 752	-
logic_bug	medium	Se asume que los índices de vendorList, statusList, pdfList, xmlList coinciden con los documentos XML procesados, pero se filtraron registros antes de construir la lista, provocando desalineación de datos.	Construir un mapa por folio (como estaba comentado) o mantener la asociación durante la extracción.	960, 1000	-
governance_risk	medium	Llamadas a url.resolveScript dentro del bucle para cada fila, lo que incrementa el consumo de gobernanza en listas grandes.	Pre-generar URLs por registro o usar caché de resultados.	724, 737	-
security_issue	low	Se inserta HTML sin escape de contenido externo; aunque el texto proviene de traducciones internas, cualquier cambio futuro podría introducir XSS.	Escapar siempre los valores antes de insertarlos en HTML.	158, 176	-
anti_pattern	low	Variable anysubsidiary siempre inicializada a true y nunca puesta a false, lógica redundante.	Inicializar a false y asignar true sólo cuando haya al menos una subsidiaria activa.	74, 99	-

Fix Plan

- Refactorizar createFilterForm y loadSublist en módulos independientes.
- Implementar un mapa de datos por folio en getResponse para garantizar consistencia.
- Cachear URLs generadas por resolveScript para reducir llamadas repetitivas.
- Aplicar escape HTML a cualquier contenido dinámico en campos INLINEHTML.
- Revisar lógica de areThereSubsidiaries para devolver false cuando no haya subsidiarias.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Handlers/
LMRY_PE_VendorBillInbound_URET_HNDL.js**

Script type: UserModule.js

Audit score (0-10): 7

Final score (0-100): 70

API version: 2.1

El script está escrito en SuiteScript 2.1 con uso adecuado de módulos AMD y sintaxis ES6+. La lógica está razonablemente separada en funciones pequeñas, pero existen problemas de compatibilidad (replaceAll), falta de control de gobernanza y algunos valores hardcodeados que generan deuda técnica.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	high	Se utiliza String.replaceAll, que no está garantizado en el runtime de SuiteScript 2.1 (GraalVM ES2019). Provocará RuntimeError en entornos donde no esté soportado.	Reemplazar replaceAll por replace con expresión regular global, e.g., xmlData.fechaEmision.replace(/-/g, '/');	145	-
governance_risk	medium	No se verifica el consumo de unidades de gobernanza antes de ejecutar búsquedas y creación de registros dentro de afterSubmit.	Agregar verificación de remaining usage con runtime.getCurrentScript().getRemainingUsage() y dividir lógica en etapas o usar Map/Reduce si el volumen es alto.	44, 46, 50, 61	-
hardcoded_value	low	Los nombres de carpetas ('SuiteLatamReady', 'XML and PDF Inbound') están hardcodeados.	Definir constantes o parámetros de script para los nombres de carpetas.	66, 72	-
anti_pattern	low	Falta de validación de existencia de nodos antes de acceder a textContent, lo que puede lanzar errores si el XML no contiene los elementos esperados.	Agregar comprobaciones de null/undefined antes de acceder a propiedades de nodos.	158, 165, 170, 171	-
obsolete_style	low	Uso mixto de var y const/let; var es innecesario en un entorno ES6+.	Reemplazar todas las declaraciones var por const o let según corresponda.	66, 72, 81, 98, 116, 137, 153, 205	-

Fix Plan

- Reemplazar replaceAll por replace con regex para garantizar compatibilidad.

- Implementar verificación de remaining usage y, si es necesario, migrar a Map/Reduce para procesamiento masivo.
- Externalizar nombres de carpetas y otros valores configurables a parámetros de script.
- Unificar estilo de declaración de variables usando const/let exclusivamente.
- Añadir validaciones de existencia de nodos XML antes de acceder a sus valores.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/LMRY_PE_InboundPdf_STLT.js**

Script type: Suitelet

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara @NApiVersion 2.1 pero contiene errores críticos de arquitectura y compatibilidad que impedirán su ejecución en el runtime de SuiteScript 2.1.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Módulo "N/redirect" está listado dos veces en la definición AMD y el parámetro de función también está duplicado, provocando colisión de dependencias y fallo de carga.	Eliminar la duplicación y asegurar que la lista de módulos y los parámetros coincidan uno-a-uno.	18, 20	-
plugin_error	critical	Se inyectan dependencias en variables globales (UI, SEARCH, etc.) y se sobrescribe REDIRECT, creando fugas de ámbito y dificultando pruebas.	Mantener las dependencias dentro del scope de la función y devolver un objeto con los handlers sin usar variables globales.	20	-
missing_dependency	critical	Referencia a LMRY_EI_ENCRYPT sin haber sido importado ni definido.	Importar el módulo correspondiente o eliminar la lógica de decodificación.	53	-
bad_naming	high	Variables 'ActiveSubsi' y 'isSubsi' se usan sin declaración explícita (isSubsi sin var) y nunca se utilizan después.	Declarar con const/let y remover variables no usadas.	35, 42	-
spaghetti_code	medium	El manejador onRequest contiene lógica de validación, carga de archivo, decisiones de tipo y escritura de respuesta en una sola función (>50 líneas).	Dividir en servicios: validación de usuario, carga de archivo, generación de respuesta.	32, 39, 71, 74, 85	-
security_issue	medium	Construcción de URL de redirección usando el header 'host' del request, que puede ser manipulado por el cliente.	Validar que el host coincida con el dominio esperado o usar URL absoluta del archivo mediante N/file.getUrl().	76, 77	-
obsolete_style	low	Se lanzan strings con throw en lugar de objetos Error.	Utilizar `throw new Error('mensaje')`.	48, 62	-
anti_pattern	low	Comparación de número con cadena (`currentuser != '-5'`).	Comparar con número o convertir explícitamente.	47	-

Fix Plan

- Eliminar la duplicación de módulos en la definición AMD.

- Importar y validar la dependencia LMRY_EI_ENCRYPT o remover su uso.
- Reestructurar el Suitelet en módulos separados: validación de usuario, carga de archivo, generación de respuesta.
- Usar ``log.debug`` en lugar de ``log.error`` para información de depuración.
- Implementar manejo de errores con objetos Error y capturar excepciones de forma consistente.

PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrypeinbound/LMRY_PE_InboundXML_STLT.js

Script type: Suitelet

Audit score (0-10): 6

Final score (0-100): 65

API version: 2.1

El script declara SuiteScript 2.1 y usa AMD correctamente, pero presenta varios problemas de calidad y seguridad que requieren refactor.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
obsolete_style	low	Módulos N/encode y N/url son importados pero nunca utilizados.	Eliminar los imports no usados para reducir la huella del script.	8 - 13	-
security_issue	high	No se valida el parámetro inboundRecordId recibido vía query string.	Validar que inboundRecordId exista, sea numérico y pertenezca a un registro accesible; devolver error 400 si es inválido.	15, 18 - 23	-
bad_naming	low	Uso extensivo de var en lugar de let/const y funciones anónimas tradicionales.	Reescribir la función usando const/let y arrow functions; eliminar variables no necesarias.	68 - 96	-
minor_inconsistency	low	El título del formulario menciona 'MX Inbound XML' mientras el script es para Perú.	Actualizar el título a 'LatamReady - PE Inbound XML'.	44, 45	-
anti_pattern	medium	El Suitelet solo contempla el método GET; cualquier otro método queda sin respuesta.	Agregar manejo explícito para métodos no soportados (p.ej., devolver 405 Method Not Allowed).	16	-
performance	low	Algoritmo manual de formateo XML con múltiples bucles y concatenaciones de strings.	Utilizar la opción indent:true del método xml.Parser.toString o una librería de formateo nativa para simplificar.	68, 70, 71, 73 - 95	-

Fix Plan

- Eliminar imports innecesarios (N/encode, N/url).
- Implementar validación robusta de inboundRecordId y manejo de errores de entrada.
- Refactorizar formatXMLString usando const/let y arrow functions o delegar a xml.Parser.indent.
- Separar la lógica de negocio (lookup, formateo) en un módulo de servicio para cumplir SRP.
- Agregar manejo de métodos HTTP no soportados con respuesta 405.

PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrypeinbound/Libraries/LMRY_PE_Message_LBRY.js

Script type: UserModule.js

Audit score (0-10): 7

Final score (0-100): 70

API version: 2.1

El módulo está escrito en SuiteScript 2.1 con buenas prácticas de ES6+, sin uso de gobernanza ni búsquedas, pero presenta riesgos de inyección y una función de plantilla excesivamente larga.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
security_issue	high	El valor del parámetro `id` se inserta directamente en HTML y en el selector del script sin sanitización, lo que permite inyección de código si el id proviene de una fuente no confiable.	Validar y escapar el `id` antes de usarlo en plantillas; limitar a caracteres alfanuméricos y guiones bajos.	16, 21	-
security_issue	medium	Los placeholders `{entity.companyname}` y `{transaction.tranid}` se insertan sin escape, lo que puede generar XSS si los datos contienen contenido malicioso.	Escapar o sanitizar los valores antes de interpolarlos en la plantilla.	64, 65, 69, 70	-
spaghetti_code	high	La función `getExample` contiene más de 70 líneas de HTML estático, violando la regla de <50 líneas para entry points y dificultando el mantenimiento.	Externalizar la plantilla a un archivo HTML o usar un motor de plantillas; cargarla mediante `N/file` o `N/template` y devolverla como string.	44 - 115	-

Fix Plan

- Sanitizar todos los valores interpolados en plantillas para evitar XSS.
- Refactorizar `getExample` a una plantilla externa o usar un motor de plantillas.
- Implementar validación estricta del parámetro `id` en `ckEditorField`.
- Añadir pruebas unitarias que verifiquen la correcta escapada de datos.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Module/
LMRY_PE_InboundDetail_STLT.js**

Script type: Suitelet

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script declara SuiteScript 2.1 pero omite importar el módulo N/log, provocando ReferenceError en tiempo de ejecución. Además contiene código muerto, falta de validación de parámetros y ausencia de manejo de métodos HTTP distintos a GET.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se invoca log.error sin haber importado el módulo N/log, lo que genera ReferenceError y detiene la ejecución del Suitelet.	Agregar "N/log" al arreglo de dependencias de define y pasar el módulo como parámetro (e.g., function(serverWidget, runtime, redirect, librarySuitelet, log) { ... }).	47	-
dead_code	high	Variable 'params' declarada pero nunca utilizada.	Eliminar la declaración o utilizarla para validar los parámetros de entrada.	28	-
security_issue	high	Se pasa directamente scriptContext.request.parameters al handler sin validación ni sanitización.	Implementar validación estricta de los parámetros esperados antes de delegar al handler.	30 - 33	-
anti_pattern	medium	No se maneja el caso de métodos HTTP distintos a GET (p.ej., POST).	Agregar un bloque else que devuelva un mensaje de método no soportado o implementar lógica para POST si corresponde.	54	-
hardcoded_path	low	Ruta del client script está hardcodeada con ruta relativa.	Utilizar un módulo de configuración o constantes definidas en un archivo central.	15	-

Fix Plan

- Importar N/log y usarlo correctamente.
- Eliminar variables no usadas y refactorizar para mayor claridad.
- Implementar validación y sanitización de parámetros de entrada.
- Agregar manejo explícito de métodos HTTP no GET.
- Externalizar rutas y constantes a un módulo de configuración.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Module/
LMRY_PE_InboundLogDetail_STLT.js**

Script type: Suitelet

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script está escrito en SuiteScript 2.1 y usa buenas prácticas de modularidad, pero presenta graves problemas de lógica, gobernanza y seguridad que comprometen su funcionamiento y mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES
spaghetti_code logic_error	high	El índice de línea del sublist se reinicia a 0 en cada iteración del while, sobrescribiendo filas anteriores y perdiendo datos mostrados al usuario.	Calcular el índice de línea como contadorGlobal + i, donde contadorGlobal se incrementa por cada bloque de 900 registros procesados.	221, 224, 229, 236, 240, 246, 254, 260, 266, 272, 282, 295, 324, 326, 328
governance_risk	high	No se verifica el consumo de unidades de gobernanza dentro del bucle, lo que puede provocar SSS_USAGE_LIMIT_EXCEEDED en procesos con gran número de registros.	Agregar verificación de runtime.getCurrentScript().getRemainingUsage() en cada iteración y dividir la lógica en batches más pequeños o usar Map/Reduce.	194 - 203, 219 - 221, 332 - 338
security_issue input_validation	medium	El parámetro 'logid' se asume válido y se usa directamente en search.lookupFields sin validar su tipo ni existencia.	Validar que log_id sea numérico y exista antes de usarlo; manejar errores con try/catch y devolver mensaje amigable.	99, 100
error_handling	medium	Se parsean strings JSON sin capturar excepciones; si el contenido no es JSON válido el script fallará.	Envolver JSON.parse en try/catch y registrar errores; considerar fallback a objeto vacío.	177, 184
bad_naming anti_pattern	low	Variables declaradas con var son de alcance funcional y se reutilizan en distintos bucles, lo que dificulta el mantenimiento y puede generar colisiones inesperadas.	Cambiar a let/const y usar nombres de variables descriptivos por bloque.	179, 221
security_issue xss	medium	Se inserta HTML con enlaces contruidos a partir de datos externos sin sanitizar, lo que abre la posibilidad de XSS si los valores son manipulados.	Escapar o validar los valores antes de concatenarlos; usar sublist.setSublistValue con tipo URL seguro o generar enlaces mediante N/ui/message.	295 - 308

Fix Plan

- Implementar verificación de unidades de gobernanza y migrar a Map/Reduce para procesamiento masivo.
- Refactorizar onRequest en servicios separados: UI Builder, Data Provider, Link Generator.

- Reemplazar var por let/const y evitar reutilización de variables.
- Añadir validación y sanitización de parámetros de entrada y de datos externos antes de generar HTML.
- Utilizar N/query (SuiteQL) para obtener datos de auditoría y evitar búsquedas anidadas.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Module/
LMRY_PE_InboundLog_CLNT.js**

Script type: ClientScript

Audit score (0-10): 2

Final score (0-100): 15

API version: 2.1

El script declara SuiteScript 2.1 pero contiene errores críticos de alcance de variables que provocarán fallos en tiempo de ejecución, además de violaciones de buenas prácticas modernas y lógica excesiva en saveRecord.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Variable 'subsi_OW' se declara dentro de pageInit (alcance local) y se usa en saveRecord, provocando ReferenceError.	Declarar 'subsi_OW' como variable global del módulo o pasar su valor a saveRecord como parámetro.	148, 149	-
bad_naming	high	Nombre de función en español 'funcionCancel' rompe la convención de nomenclatura en inglés y dificulta la consistencia.	Renombrar a 'cancelFunction' o 'handleCancel'.	296	-
anti_pattern	high	saveRecord contiene más de 50 líneas y combina validación, lógica de negocio y creación de registros, violando el principio de Fat Entry Point.	Dividir la lógica en módulos de servicio (e.g., ValidationService, InboundRecordFactory) y delegar desde saveRecord.	137, 272	-
obsolete_style	medium	Uso extensivo de 'var' en lugar de 'let' o 'const' en un script declarado como SuiteScript 2.1.	Reemplazar 'var' por 'const' para valores inmutables y 'let' para variables que cambian.	21, 23, 66, 75, 100, 122, 137, 281, 296, 317	-
style	low	Coma final innecesaria en la lista de dependencias del define.	Eliminar la coma final.	11	-

Fix Plan

- Refactorizar saveRecord en servicios separados (validación, creación de registro).
- Utilizar let/const para mejorar la claridad del alcance de variables.
- Corregir el alcance de subsi_OW para evitar ReferenceError.
- Renombrar funciones y variables a inglés para mantener consistencia.
- Implementar manejo de errores más robusto (registro de auditoría, notificaciones).
- Considerar mover la lógica de creación de registros a un Suitelet o Map/Reduce para mejor gobernanza.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Module/
LMRY_PE_InboundLog_STLT.js**

Script type: Suitelet

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.0

El script declara SuiteScript 2.0 pero contiene un entry point inflado (>250 líneas), lógica de presentación mezclada con lógica de negocio, falta de control de gobernanza, uso de valores mágicos y vulnerabilidades de seguridad potenciales.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RIS
spaghetti_code	high	El entry point supera ampliamente el límite recomendado de 50 líneas, mezclando generación de UI, lógica de negocio y manejo de datos en un solo bloque.	Refactorizar extrayendo la generación del formulario, la búsqueda de registros y la población del sublist a módulos/ servicios independientes.	161, 455	-
compatibility_break	critical	Se utiliza una cadena literal 'textarea' en lugar del enum serverWidget.FieldType.TEXTAREA, lo que puede fallar en tiempo de ejecución bajo SuiteScript 2.0.	Reemplazar con serverWidget.FieldType.TEXTAREA.	268, 270	-
governance_risk	high	Se recuperan hasta 1000 resultados sin paginación ni control de uso de gobernanza, lo que puede provocar SSS_USAGE_LIMIT_EXCEEDED en entornos con gran volumen.	Implementar paginación con runPaged() y validar remaining usage en cada iteración.	295	-
bad_naming	medium	Nombres de variables como LMRY_script, subsi_OW, Language, jsonLanguage no siguen la convención camelCase y son poco descriptivos.	Renombrar siguiendo camelCase y usar prefijos claros (e.g., scriptName, isSubsidiaryEnabled, languageCode).	23, 24, 28, 33, 39	-
magic_numbers	medium	Los valores de estado '0', '1', '2' están hardcoded y no están documentados ni centralizados.	Definir constantes descriptivas (e.g., STATUS_PENDING = '0') y utilizarlas en el switch.	386, 401, 415	-
security_issue	low	Se concatena directamente el parámetro id en la URL sin sanitizar ni codificar.	Utilizar encodeURIComponent al agregar parámetros a la URL.	429	-
obsolete_style	low	Uso extensivo de var en lugar de const/let, lo que reduce la claridad del alcance y puede generar bugs de hoisting.	Reemplazar var por const o let según corresponda.	23, 24, 28, 33, 39, 161 - 455	-

Fix Plan

- Dividir onRequest en funciones: crearFormulario, cargarDatos, poblarSublist.
- Implementar paginación con search.runPaged() y verificar remaining usage.
- Reemplazar var por const/let y aplicar naming conventions.
- Externalizar constantes de estado y IDs de script en un módulo de configuración.
- Utilizar serverWidget.FieldType enums de forma consistente.
- Sanitizar parámetros al construir URLs.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Module/LMRY_PE_Inbound_MPRD.js**

Script type: Map/Reduce

Audit score (0-10): 4

Final score (0-100): 45

API version: 2.1

El script funciona pero presenta graves violaciones de arquitectura, alta complejidad en los entry points y problemas de gobernanza que pueden provocar errores de límite de uso y mantenimiento costoso.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
global_state	high	Variables globales mutables (objContext, hasSubsidiariesFeature, hasMultiVendor, actualUser) son compartidas entre etapas del Map/Reduce, rompiendo aislamiento y pudiendo causar datos corruptos en ejecuciones concurrentes.	Eliminar variables globales; pasar contexto como parámetros o crear módulos de servicio que obtengan la información en cada función.	24, 26, 28, 29	-
fat_entry_point	high	La función map supera los 50 líneas y contiene lógica de carga de registros, búsquedas, transformación, envío de correos y manejo de plugins.	Dividir la lógica en servicios: carga de datos, validación, conversión, envío de email. Cada servicio debe ser una función independiente.	88, 187	-
plugin_load_inefficiency	high	Se invoca plugin.loadImplementation por cada registro procesado, generando sobrecarga innecesaria.	Cargar la implementación del plugin una sola vez al inicio del script y reutilizarla (caching).	253, 256, 375, 379	-
input_validation	high	Se parsea JSON de context.value sin validar su estructura ni capturar errores de parseo.	Validar que context.value sea JSON válido y que contenga los campos esperados antes de usarlo.	101	-
governance_risk	medium	Se carga y guarda un registro de ítem por cada ítem del XML dentro de bucles, lo que puede superar los límites de uso.	Batch cargar ítems, usar search.lookupFields cuando solo se necesitan valores, y guardar cambios fuera del bucle o mediante bulk update.	904, 917, 921, 945, 960	-
governance_risk	medium	Se realizan búsquedas (searchItemsByVendorCode) y carga de registros dentro de bucles para cada ítem.	Pre-cargar todos los ítems necesarios antes del bucle y reutilizar los resultados; evitar record.load dentro de iteraciones.	985, 1000, 1025, 1030	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
bad_naming	low	Objeto customPluginMasive usa nombres en mayúsculas y valores hardcodeados.	Mover a Script Parameters o a un registro de configuración.	42, 43	-
duplicate_declaration	low	Se redeclara var conversionResult dentro del bloque else, ocultando la variable externa.	Eliminar la redeclaración y usar la variable ya declarada.	144, 145	-
obsolete_style	low	Se utiliza var en todo el código a pesar de estar en SuiteScript 2.1 que soporta const/let.	Reemplazar var por const o let según corresponda.	1, 300	-

Fix Plan

- Refactorizar map en varios servicios (carga, validación, conversión, notificación).
- Cachear la implementación del plugin y reutilizarla en todas las etapas.
- Reemplazar var por const/let y eliminar variables globales.
- Implementar validación robusta del JSON de entrada.
- Optimizar búsquedas: usar lookupFields o query en lugar de search.create dentro de bucles.
- Externalizar configuraciones (IDs de plugins, códigos de error) a parámetros de script.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Module/LMRY_PE_Inbound_STLT.js**

Script type: Suitelet

Audit score (0-10): 5

Final score (0-100): 45

API version: 2.1

El script declara SuiteScript 2.1 y usa sintaxis moderna, pero el punto de entrada onRequest supera los 50 líneas, contiene lógica duplicada, números mágicos y falta de validación de parámetros, lo que lo clasifica como código espagueti de riesgo medio.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
spaghetti_code	high	El método onRequest contiene más de 50 líneas y combina lógica de routing, construcción de formularios y ejecución de procesos, violando el principio SRP.	Dividir onRequest en sub-módulos: router, FormBuilder y postProcessor. Cada sub-módulo debe ser una función o clase independiente.	24, 96	-
duplicate_logic	medium	Bloques repetidos para los valores de status (0,1,2) crean código redundante.	Unificar la lógica en una función que reciba el status y ejecute los pasos comunes, usando un mapa de configuraciones.	42, 49, 55, 60	-
magic_number	low	Se usan valores numéricos (0,1,2) sin constantes descriptivas.	Definir constantes descriptivas (e.g., STATUS_NONE = 0, STATUS_FILTER = 1, STATUS_RESULT = 2) y utilizarlas en lugar de números literales.	39, 49, 55, 74, 76	-
security_issue	medium	Los parámetros de la request se convierten a Number sin validación previa, lo que permite valores inesperados o ataques de inyección de tipo.	Validar que los parámetros existan y sean del tipo esperado antes de usarlos; lanzar errores controlados si la validación falla.	28, 74	-
anti_pattern	low	Construcción de HTML mediante concatenación de strings, lo que aumenta el riesgo de XSS si en el futuro se inserta contenido dinámico.	Utilizar template literals o cargar el HTML desde un archivo estático; evitar concatenaciones manuales.	107, 118	-

Fix Plan

- Refactorizar onRequest en un controlador con métodos separados para GET y POST.
- Crear un módulo de constantes para los códigos de estado.
- Implementar validación de parámetros usando schemas (e.g., Joi) o validaciones manuales.
- Reemplazar la generación de HTML por template literals o archivos estáticos.
- Documentar y probar unitariamente cada sub-módulo para mejorar la cobertura y la mantenibilidad.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Plugin/
LMRY_PE_Inbound_PLGNIMP.js**

Script type: PlugInImplementation

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.0

El script declara @NApiVersion 2.0 pero emplea sintaxis ES6 (const) prohibida, genera conflictos de nombres y contiene lógica propensa a fallos en tiempo de ejecución. La arquitectura es modular pero la calidad del código es baja, con deuda técnica significativa y riesgos de gobernanza.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	
compatibility_break	critical	Uso de const y sintaxis ES6 en un script declarado @NApiVersion 2.0 (Nashorn) que solo soporta ES5.1.	Reemplazar const por var y evitar ES6; o cambiar la versión a @NApiVersion 2.1 si se desea usar ES6.	523, 525, 529	
variable_shadowing	critical	Declaración duplicada de la variable inboundSubsiSetupFields (var inboundSubsiSetupFields = {} y luego var inboundSubsiSetupFields = getSetupSubsiFields(...)).	Eliminar la declaración redundante y usar un único nombre de variable.	55, 79	
naming_conflict	critical	El catch captura la variable 'error' que colisiona con el módulo N/error importado, ocultando la referencia al módulo.	Renombrar el parámetro del catch a e o err y usar el módulo error con otro alias.	114, 115	
logic_error	high	Condición if(inboundSubsiSetupFields[inboundSubsiSetupFields]) es lógica incorrecta y siempre evalúa a true si el objeto tiene alguna propiedad, provocando un error intencional.	Revisar la condición; probablemente se quiso validar la existencia de un tax code específico.	219 - 225	
governance_risk	medium	Uso de search.lookupFields dentro de un bucle for que itera sobre la tabla de detracción; cada iteración consume unidades de gobernanza.	Pre-cargar los códigos de retención en un mapa fuera del bucle o usar una búsqueda batch.	119, 125 - 133, 135 - 142	
hardcoded_values	low	URLs contienen marcadores de posición '-CREDSUNAT-' y '-RUC-' sin mecanismo de sustitución seguro.	Almacenar URLs y credenciales en Script Parameters o en un registro de configuración seguro.	29, 30	
missing_return	low	Si la búsqueda no devuelve resultados, la función no retorna nada (undefined).	Agregar un return explícito (null o valor por defecto) al final de la función.	274 - 277	

Fix Plan

- Migrar a SuiteScript 2.1 para aprovechar ES6 y eliminar incompatibilidades.
- Refactorizar la función inject en múltiples servicios (currencyService, taxService, detractionService, xmlParser).

- Reemplazar `const/let` por `var` o actualizar la versión del script.
- Eliminar variables duplicadas y renombrar parámetros de `catch` para evitar colisiones.
- Optimizar búsquedas dentro de bucles mediante `cache` o búsquedas `batch`.
- Externalizar URLs y credenciales a `Script Parameters` o a un registro de configuración.

PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrypeinbound/Plugin/ LMRY_PE_Inbound_PLGNTYPE.js

Script type: PlugInImplementation

Audit score (0-10): 7

Final score (0-100): 75

API version: 2.1

El script está declarado correctamente como plug-in type 2.1, usa AMD y devuelve la estructura esperada. No hay problemas de gobernanza ni de compatibilidad, pero carece de validación de parámetros, contiene código comentado y usa var en lugar de const/let.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
obsolete_style	low	Código comentado que no aporta valor y genera ruido.	Eliminar los comentarios de código muerto o moverlos a documentación externa.	20 - 22	-
security_issue	medium	No se valida la existencia ni la integridad de los objetos recibidos en params.	Agregar validaciones explícitas de los campos inboundEDocRec, templateRec y entityRec, lanzando errores claros si faltan.	19	-
bad_naming	low	Se declara con var en SuiteScript 2.1, donde const/let son la práctica recomendada.	Cambiar a const customObj = {};	25	-
anti_pattern	low	Falta manejo de excepciones; cualquier error interno provocará una excepción sin control.	Envolver la lógica en try/catch y devolver un objeto de error estructurado si ocurre una excepción.	19	-

Fix Plan

- Reemplazar var por const/let para alinearse con ES6+.
- Eliminar código comentado y documentar la intención en JSDoc.
- Implementar validación de parámetros de entrada y manejo de errores.
- Considerar extraer la construcción del objeto de datos a un módulo de dominio separado para mayor reutilización.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Plugin/
LMRY_PE_Inbound_mail_PLGIMP.js**

Script type: UserModule.js

Audit score (0-10): 2

Final score (0-100): 25

El script está escrito en SuiteScript 1.0 con múltiples violaciones de buenas prácticas: uso extensivo de variables globales, IDs y valores mágicos hardcoded, falta de validaciones de resultados de búsquedas, errores de tipo (llamadas a métodos inexistentes), exposición de detalles de error y ausencia de control de gobernanza. Necesita una re-escritura completa hacia una arquitectura modular y segura.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	No se declara @NapiVersion; el script usa napi* indicando SuiteScript 1.0, pero contiene código ES6 (template literals) y arrow functions en comentarios que podrían confundirse con versiones posteriores.	Agregar JSDoc `@NapiVersion 1.0` y migrar a 2.x si se desea usar ES6.	33	-
bad_naming	critical	Variable `fromAddress` es una cadena, pero luego se llama `fromAddress.getEmail()` (línea 456), provocando TypeError.	Usar directamente la cadena `fromAddress` o extraer el email correctamente antes de pasar a `identifyVendor`.	380, 456	-
null_check_missing	high	No se verifica si `napiSearchRecord` devuelve null antes de iterar sobre `searchResults`, lo que genera error cuando no hay resultados.	Agregar `if (searchResults) { ... }` antes del bucle.	97, 104	-
hardcoded_values	high	IDs y tipos de registro están hardcodedos (ej. ADMIN_ROLE = 3, VENDOR_BILL_TYPE = 17).	Mover a Script Parameters o a un archivo de configuración.	13 - 15, 24, 30, 31	-
governance_risk	medium	Se llama a `getActiveAdmins()` en cada envío de email sin cache, consumiendo gobernanza innecesariamente.	Cachear la lista de administradores durante la ejecución del script.	121, 122	-
security_issue	medium	Se envía al usuario el código y detalle completo del error (`e.getDetails()`), exponiendo información interna.	Sanitizar el mensaje de error y evitar exponer detalles internos.	568, 582, 583	-
anti_pattern	low	Construcción manual de nombre de archivo usando concatenación de strings.	Utilizar template literals o una función de generación de nombres.	598, 600	-
spaghetti_code	low	Función `process` supera 150 líneas con lógica de negocio, búsquedas, validaciones y manejo de errores mezclados.	Dividir en módulos: parser XML, identificación de entidad, selección de plantilla, creación de registro.	395, 424, 440, 452, 456, 470, 492, 520, 544	-

Fix Plan

- Migrar a SuiteScript 2.1 para aprovechar ES6+, promesas y mejor modularidad.
- Externalizar constantes y configuraciones a Script Parameters.
- Implementar manejo de errores centralizado y sanitización de mensajes.
- Refactorizar `process` en varios módulos con pruebas unitarias.
- Agregar validaciones de null después de cada búsqueda para evitar excepciones.

PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrypeinbound/Router/LMRY_PE_Inbound_ROUT.js

Script type: UserModule.js

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El script declara SuiteScript 2.1 y usa sintaxis moderna, pero la lógica de carga dinámica de módulos es incorrecta: se utiliza require asíncrono y se devuelve el resultado antes de que se cargue, provocando un retorno vacío y potenciales fallos en tiempo de ejecución.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
compatibility_break	critical	Se emplea require([modulePath], callback) de forma asíncrona y se devuelve 'result' inmediatamente, lo que siempre retorna una cadena vacía. En SuiteScript 2.1 la carga debe ser sincrónica o la función debe devolver una Promise.	Reemplazar la carga asíncrona por const UserEvent_HNDL = require(userEventRouter[type]); y devolver directamente el módulo, o bien convertir getUserEventRouter en async y devolver una Promise que resuelva con el handler.	23 - 25, 28	-
validation_missing	high	No se valida que el parámetro 'type' sea una cadena no vacía ni que corresponda a una clave válida del mapa.	Agregar validación explícita del parámetro y lanzar una excepción controlada si el tipo no está soportado.	22, 27	-
obsolete_style	low	Se incluye la extensión '.js' en la ruta del módulo, lo cual no es necesario en SuiteScript y puede generar inconsistencias al mover archivos.	Eliminar la extensión '.js' de la ruta del módulo.	18	-

Fix Plan

- Convertir getUserEventRouter en una función async que devuelva una Promise.
- Utilizar require de forma sincrónica dentro de SuiteScript 2.1.
- Implementar validación de entrada y manejo de errores estructurado.
- Eliminar la extensión '.js' de las rutas de módulos para mayor consistencia.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Setup/
LMRY_PE_InboundSetup_CLNT.js**

Script type: ClientScript

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

Script declara SuiteScript 2.1 pero contiene errores críticos de API, inconsistencias de campos y funciones monolíticas que superan los límites de mantenimiento.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION
compatibility_break	critical	Uso incorrecto de record.setValue: se pasan dos argumentos en lugar de un objeto {fieldId, value}. En SuiteScript 2.x la firma es setValue(options).	Reemplazar por record.setValue({ fieldId: 'custfield_email', value: ck5Library ? NetSuite.RTEManager.getInstance('custfield_email_body')._editor: CKEDITOR.instances['custfield_email_body'].getData() });
compatibility_break	critical	Inconsistencia de nombres de campos: pageInit escribe en 'custpage_taxcode19' mientras fieldChanged limpia y escribe en 'custpage_taxcode16'.	Unificar el nombre del campo (ej. 'custpage_taxcode16') en ambos.
spaghetti_code	high	Función pageInit supera 50 líneas y contiene lógica de UI, búsquedas, mapeo de campos y manejo de sublistas.	Dividir en funciones auxiliares: cargarConfiguración(), poblarCampos(), inicializarEditor(), limpiarCampos().
spaghetti_code	high	Función fieldChanged también supera 50 líneas y replica gran parte de la lógica de pageInit.	Extraer lógica de búsqueda y mapeo a un módulo reutilizable y llamarlo desde ambos eventos.
bad_naming	medium	Uso extensivo de 'var' en un script 2.1 donde se prefiere 'let' o 'const'.	Reemplazar por const/let según necesidad.
security_issue	low	Parseo directo de valores JSON sin manejo de excepciones; datos corruptos pueden lanzar errores.	Envolver cada JSON.parse en try/catch y registrar errores de forma controlada.
governance_risk	low	Se ejecuta una búsqueda dentro de un evento de cliente; aunque solo se recupera un registro, no hay control de uso de gobernanza.	Cachear resultados cuando sea posible o validar uso restante con runtime.getCurrentScript().getRemainingUsage().

Fix Plan

- Refactorizar pageInit y fieldChanged en módulos reutilizables.
- Corregir la firma de record.setValue en saveRecord.
- Unificar nombres de campos de código de impuesto.
- Reemplazar var por const/let y aplicar linting.
- Agregar manejo de errores para JSON.parse.
- Implementar caché de configuración para reducir búsquedas repetidas.

**PROYECTOS INBOUND/sdf-peru-inbound/
com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/
com.latamready.lmrypeinbound/Setup/
LMRY_PE_InboundSetup_STLT.js**

Script type: Suitelet

Audit score (0-10): 3

Final score (0-100): 30

API version: 2.1

El Suitelet está escrito en SuiteScript 2.1 pero presenta graves problemas de arquitectura y gobernanza. Usa variables globales mutables, un entry point monolítico >200 líneas, falta de validación de entrada y uso ineficiente de búsquedas (lookupFields dentro de bucles). Necesita refactor urgente.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
global_state	critical	Variables FEAT_DPT, FEAT_LOC, FEAT_CLASS y FEAT_SUBSI se declaran a nivel de módulo y se sobrescriben en cada request, provocando posible contaminación entre ejecuciones concurrentes.	Eliminar variables globales; obtener los flags dentro de onRequest o pasarlos como parámetros a las funciones auxiliares.	9 - 12	-
spaghetti_code	high	El método onRequest contiene más de 200 líneas y maneja UI, lógica de negocio, validación y persistencia en un solo bloque.	Dividir en módulos: UIBuilder, RequestProcessor, RecordService. Cada uno con responsabilidad única.	13, 226	-
security_issue	high	Los parámetros recibidos del cliente se usan directamente para crear/ actualizar registros sin validación ni sanitización.	Validar tipos, longitud y valores permitidos; sanitizar textos antes de guardarlos; usar try/ catch para manejar errores.	235, 251	-
governance_risk	high	Dentro del bucle each se ejecuta search.lookupFields por cada registro, consumiendo unidades de gobernanza de forma exponencial.	Reemplazar por una búsqueda que incluya el campo deseado mediante join o usar N/query para obtener todos los datos en una sola llamada.	425, 432	-
anti_pattern	medium	Condición `if (FEAT_SUBSI recordType != 'subsidiary')` es confusa y puede producir lógica invertida.	Revisar la lógica; probablemente se quiso usar `&&` en lugar de ` `.	344	-
style	low	Se usa la cadena literal `select` en lugar del enum `ui.FieldType.SELECT`.	Reemplazar por `ui.FieldType.SELECT`.	176	-
hardcoded_id	low	Los IDs del script y despliegue están hardcodeados.	Utilizar parámetros de script o constantes definidas en un archivo de configuración.	330, 332	-

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
style	low	Uso extensivo de `var` en lugar de `let` o `const`.	Reemplazar por `let` / `const` según corresponda.	15, 31, 45, 52, 59, 78, 92, 106, 119, 127, 132, 139, 144, 149, 160, 174, 182, 183, 195, 203, 207, 214, 218, 224, 236 - 250, 252 - 255, 257, 259, 261, 262, 267, 273, 278, 280, 281, 284, 286 - 299, 301, 304, 305, 308 - 322, 324	-

Fix Plan

- Eliminar variables globales y pasar flags como parámetros.
- Dividir onRequest en módulos separados (UIBuilder, Processor, Service).
- Agregar validación y sanitización de todos los parámetros de entrada.
- Reemplazar search.lookupFields dentro de bucles por una única búsqueda o query.
- Usar const/let en lugar de var y aplicar normas de estilo ES6+.
- Implementar manejo de errores con try/catch y registro de logs.
- Utilizar N/query (SuiteQL) para búsquedas complejas y evitar inyección SQL.
- Externalizar IDs de scripts y constantes en un archivo de configuración.

PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.lmrypeinbound/src/FileCabinet/SuiteApps/ com.latamready.lmrypeinbound/Translator/ LMRY_PE_Inbound_Translator.js

Script type: UserModule.js

Audit score (0-10): 8

Final score (0-100): 85

API version: 2.1

Módulo de traducciones bien estructurado, usa SuiteScript 2.1 con sintaxis moderna y cumple con la arquitectura AMD. No hay problemas de compatibilidad ni de gobernanza. Se detectan pequeñas oportunidades de mejora en mantenibilidad.

Hotspots

TITLE	SEVERITY	DESCRIPTION	RECOMMENDATION	LINES	RISK
duplicate_key	low	Las claves 'pending' aparecen duplicadas en ambos objetos de idioma, lo que provoca que la segunda definición sobrescriba a la primera.	Eliminar duplicados o consolidar bajo una única clave.	43, 113, 154, 224	-
maintainability	low	El objeto de traducciones es muy extenso y está embebido en el código fuente, dificultando su mantenimiento y localización.	Externalizar las traducciones a archivos JSON o a un recurso de configuración y cargarlos dinámicamente.	10, 121	-
missing_documentation	low	La función carece de JSDoc que describa parámetros y valor de retorno.	Agregar JSDoc con @param {string} language y @returns {Object} mapa de traducciones.	9	-

Fix Plan

- Externalizar el catálogo de traducciones a archivos JSON para facilitar la internacionalización.
- Implementar pruebas unitarias que verifiquen la presencia de todas las claves esperadas en cada idioma.

- Agregar JSDoc y comentarios descriptivos para mejorar la auto-documentación.

Generado automáticamente 2026-01-23 18:55 UTC

ÍNDICE DE DOCUMENTACIÓN

Job 2026-01-23-174258-PROYECTOS-INBOUND · 107 documentos

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
1	PROYECTOS INBOUND/Inbound_CL/Client/ LMRY_CL_Inbound_CLNT.pdf	LMRY_CL_Inbound_CLNT.pdf		10-11
2	PROYECTOS INBOUND/Inbound_CL/Client/ LMRY_CL_Inbound_Detail_CLNT.pdf	LMRY_CL_Inbound_Detail_CLNT.pdf		12-16
3	PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_CLNT_HNDL.pdf	LMRY_CL_Inbound_CLNT_HNDL.pdf		17-20
4	PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_Detail_STLT_HNDL.pdf	LMRY_CL_Inbound_Detail_STLT_HNDL.pdf		21-29
5	PROYECTOS INBOUND/Inbound_CL/Handlers/ LMRY_CL_Inbound_STLT_HNDL.pdf	LMRY_CL_Inbound_STLT_HNDL.pdf		30-36
6	PROYECTOS INBOUND/Inbound_CL/ LMRY_CL_Inbound_XML_STLT.pdf	LMRY_CL_Inbound_XML_STLT.pdf		37-38
7	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_Detail_STLT.pdf	LMRY_CL_Inbound_Detail_STLT.pdf		39-40
8	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_Log_Detail_STLT.pdf	LMRY_CL_Inbound_Log_Detail_STLT.pdf		41-43
9	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_Log_STLT.pdf	LMRY_CL_Inbound_Log_STLT.pdf		44-45
10	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_MPRD.pdf	LMRY_CL_Inbound_MPRD.pdf		46-50
11	PROYECTOS INBOUND/Inbound_CL/Module/ LMRY_CL_Inbound_STLT.pdf	LMRY_CL_Inbound_STLT.pdf		51-53
12	PROYECTOS INBOUND/Inbound_CL/Transaltor/ LMRY_CL_Inbound_Translator.pdf	LMRY_CL_Inbound_Translator.pdf		54-55
13	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Client/ LMRY_AR_InboundDetail_CLNT.pdf	LMRY_AR_InboundDetail_CLNT.pdf		56-60
14	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Client/ LMRY_AR_Inbound_CLNT.pdf	LMRY_AR_Inbound_CLNT.pdf		61-62
15	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Handlers/ LMRY_AR_InboundDetail_STLT_HNDL.pdf	LMRY_AR_InboundDetail_STLT_HNDL.pdf		63-69
16	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Handlers/ LMRY_AR_Inbound_CLNT_HNDL.pdf	LMRY_AR_Inbound_CLNT_HNDL.pdf		70-75
17	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Handlers/ LMRY_AR_Inbound_STLT_HNDL.pdf	LMRY_AR_Inbound_STLT_HNDL.pdf		76-81

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
18	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Handlers/ LMRY_AR_VendorBillInbound_URET_HNDL.pdf	LMRY_AR_VendorBillInbound_URET_HNDL.pdf		82-84
19	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundAFIP_MPRD.pdf	LMRY_AR_InboundAFIP_MPRD.pdf		85-87
20	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundAFIP_RSLT.pdf	LMRY_AR_InboundAFIP_RSLT.pdf		88-89
21	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundAFIP_SCHDL.pdf	LMRY_AR_InboundAFIP_SCHDL.pdf		90-91
22	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundPdf_STLT.pdf	LMRY_AR_InboundPdf_STLT.pdf		92-93
23	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ LMRY_AR_InboundXML_STLT.pdf	LMRY_AR_InboundXML_STLT.pdf		94-96
24	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_InboundDetail_STLT.pdf	LMRY_AR_InboundDetail_STLT.pdf		97-98
25	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_InboundLogDetail_STLT.pdf	LMRY_AR_InboundLogDetail_STLT.pdf		99-101
26	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_InboundLog_CLNT.pdf	LMRY_AR_InboundLog_CLNT.pdf		102-102
27	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_InboundLog_STLT.pdf	LMRY_AR_InboundLog_STLT.pdf		103-105
28	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_Inbound_MPRD.pdf	LMRY_AR_Inbound_MPRD.pdf		106-110
29	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Module/ LMRY_AR_Inbound_STLT.pdf	LMRY_AR_Inbound_STLT.pdf		111-112

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
30	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Plugin/ LMRY_AR_Inbound_Email_Capture.pdf	LMRY_AR_Inbound_Email_Capture.pdf		113-116
31	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Plugin/ LMRY_AR_Inbound_PLGNIMP.pdf	LMRY_AR_Inbound_PLGNIMP.pdf		117-118
32	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Plugin/ LMRY_AR_Inbound_PLGNTYPE.pdf	LMRY_AR_Inbound_PLGNTYPE.pdf		119-119
33	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Router/ LMRY_AR_Inbound_ROUT.pdf	LMRY_AR_Inbound_ROUT.pdf		120-121
34	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Setup/ LMRY_AR_InboundSetup_CLNT.pdf	LMRY_AR_InboundSetup_CLNT.pdf		122-124
35	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/Setup/ LMRY_AR_InboundSetup_STLT.pdf	LMRY_AR_InboundSetup_STLT.pdf		125-128
36	PROYECTOS INBOUND/sdf-argentina-inbound/ com.latamready.Imryarinbound/src/FileCabinet/ SuiteApps/com.latamready.Imryarinbound/ Translator/LMRY_AR_Inbound_Translator.pdf	LMRY_AR_Inbound_Translator.pdf		129-129
37	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Client/ LMRY_CO_Inbound_CLNT.pdf	LMRY_CO_Inbound_CLNT.pdf		130-131
38	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Client/ LMRY_CO_Inbound_Detail_CLNT.pdf	LMRY_CO_Inbound_Detail_CLNT.pdf		132-137
39	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Handlers/ LMRY_CO_Inbound_CLNT_HNDL.pdf	LMRY_CO_Inbound_CLNT_HNDL.pdf		138-143
40	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Handlers/ LMRY_CO_Inbound_Detail_STLT_HNDL.pdf	LMRY_CO_Inbound_Detail_STLT_HNDL.pdf		144-153
41	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Handlers/ LMRY_CO_Inbound_STLT_HNDL.pdf	LMRY_CO_Inbound_STLT_HNDL.pdf		154-160

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
42	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Handlers/ LMRY_CO_VendorBillInbound_URET_HNDL.pdf	LMRY_CO_VendorBillInbound_URET_HNDL.pdf		161-164
43	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/ LMRY_CO_Inbound_Pdf_STLT.pdf	LMRY_CO_Inbound_Pdf_STLT.pdf		165-166
44	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/ LMRY_CO_Inbound_XML_STLT.pdf	LMRY_CO_Inbound_XML_STLT.pdf		167-169
45	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_InboundLogDetail_STLT.pdf	LMRY_CO_InboundLogDetail_STLT.pdf		170-172
46	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_InboundLog_CLNT.pdf	LMRY_CO_InboundLog_CLNT.pdf		173-174
47	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_InboundLog_STLT.pdf	LMRY_CO_InboundLog_STLT.pdf		175-177
48	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_Inbound_Detail_STLT.pdf	LMRY_CO_Inbound_Detail_STLT.pdf		178-179
49	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_Inbound_MPRD.pdf	LMRY_CO_Inbound_MPRD.pdf		180-183
50	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Module/ LMRY_CO_Inbound_STLT.pdf	LMRY_CO_Inbound_STLT.pdf		184-185
51	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Plugin/ LMRY_CO_Inbound_Email_Capture.pdf	LMRY_CO_Inbound_Email_Capture.pdf		186-189
52	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Plugin/ LMRY_CO_Inbound_Email_Capture_STLT.pdf	LMRY_CO_Inbound_Email_Capture_STLT.pdf		190-191
53	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Plugin/ LMRY_CO_Inbound_PLGNIMP.pdf	LMRY_CO_Inbound_PLGNIMP.pdf		192-193

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
54	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Plugin/ LMRY_CO_Inbound_PLGNTYPE.pdf	LMRY_CO_Inbound_PLGNTYPE.pdf		194-194
55	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Router/ LMRY_CO_Inbound_ROUT.pdf	LMRY_CO_Inbound_ROUT.pdf		195-196
56	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Setup/ LMRY_CO_InboundSetup_CLNT.pdf	LMRY_CO_InboundSetup_CLNT.pdf		197-199
57	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Setup/ LMRY_CO_InboundSetup_STLT.pdf	LMRY_CO_InboundSetup_STLT.pdf		200-203
58	PROYECTOS INBOUND/sdf-colombia-inbound/ com.latamready.coinbound/src/FileCabinet/ SuiteApps/com.latamready.coinbound/Translator/ LMRY_CO_Inbound_Translator.pdf	LMRY_CO_Inbound_Translator.pdf		204-205
59	PROYECTOS INBOUND/sdf-latam-inbound/ com.latamready.Imrylataminbound/src/FileCabinet/ SuiteApps/com.latamready.Imrylataminbound/ Handlers/LMRY_Inbound_Uret_HNDL.pdf	LMRY_Inbound_Uret_HNDL.pdf		206-207
60	PROYECTOS INBOUND/sdf-latam-inbound/ com.latamready.Imrylataminbound/src/FileCabinet/ SuiteApps/com.latamready.Imrylataminbound/Urets/ LMRY_Inbound_URET.pdf	LMRY_Inbound_URET.pdf		208-209
61	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Client/ LMRY_MX_InboundDetail_CLNT.pdf	LMRY_MX_InboundDetail_CLNT.pdf		210-215
62	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Client/ LMRY_MX_Inbound_CLNT.pdf	LMRY_MX_Inbound_CLNT.pdf		216-217
63	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/ Handlers/LMRY_MX_InboundDetail_STLT_HNDL.pdf	LMRY_MX_InboundDetail_STLT_HNDL.pdf		218-225
64	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/ Handlers/LMRY_MX_Inbound_CLNT_HNDL.pdf	LMRY_MX_Inbound_CLNT_HNDL.pdf		226-233
65	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/ Handlers/LMRY_MX_Inbound_STLT_HNDL.pdf	LMRY_MX_Inbound_STLT_HNDL.pdf		234-241

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
66	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ Handlers/ LMRY_MX_VendorBillInbound_URET_HNDL.pdf	LMRY_MX_VendorBillInbound_URET_HNDL.pdf		242-245
67	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundPdf_STLT.pdf	LMRY_MX_InboundPdf_STLT.pdf		246-247
68	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundSAT_MPRD.pdf	LMRY_MX_InboundSAT_MPRD.pdf		248-250
69	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundSAT_RSLT.pdf	LMRY_MX_InboundSAT_RSLT.pdf		251-252
70	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundSAT_SCHDL.pdf	LMRY_MX_InboundSAT_SCHDL.pdf		253-254
71	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/ LMRY_MX_InboundXML_STLT.pdf	LMRY_MX_InboundXML_STLT.pdf		255-257
72	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_InboundDetail_STLT.pdf	LMRY_MX_InboundDetail_STLT.pdf		258-259
73	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_InboundLogDetail_STLT.pdf	LMRY_MX_InboundLogDetail_STLT.pdf		260-262
74	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_InboundLog_CLNT.pdf	LMRY_MX_InboundLog_CLNT.pdf		263-264
75	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_InboundLog_STLT.pdf	LMRY_MX_InboundLog_STLT.pdf		265-267
76	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_Inbound_MPRD.pdf	LMRY_MX_Inbound_MPRD.pdf		268-271
77	PROYECTOS INBOUND/sdf-mexico-inbound/ com.litamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrymxinbound/Module/ LMRY_MX_Inbound_STLT.pdf	LMRY_MX_Inbound_STLT.pdf		272-274

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
78	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Plugin/ LMRY_MX_Inbound_Email_Capture.pdf	LMRY_MX_Inbound_Email_Capture.pdf		275-278
79	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Plugin/ LMRY_MX_Inbound_PLGNIMP.pdf	LMRY_MX_Inbound_PLGNIMP.pdf		279-281
80	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Plugin/ LMRY_MX_Inbound_PLGNTYPE.pdf	LMRY_MX_Inbound_PLGNTYPE.pdf		282-282
81	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Router/ LMRY_MX_Inbound_ROUT.pdf	LMRY_MX_Inbound_ROUT.pdf		283-284
82	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Setup/ LMRY_MX_InboundSetup_CLNT.pdf	LMRY_MX_InboundSetup_CLNT.pdf		285-289
83	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/Setup/ LMRY_MX_InboundSetup_STLT.pdf	LMRY_MX_InboundSetup_STLT.pdf		290-293
84	PROYECTOS INBOUND/sdf-mexico-inbound/ com.latamready.Imrymxinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrymxinbound/ Translator/LMRY_MX_Inbound_Translator.pdf	LMRY_MX_Inbound_Translator.pdf		294-295
85	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Client/ LMRY_PE_InboundDetail_CLNT.pdf	LMRY_PE_InboundDetail_CLNT.pdf		296-303
86	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Client/ LMRY_PE_Inbound_CLNT.pdf	LMRY_PE_Inbound_CLNT.pdf		304-305
87	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Handlers/ LMRY_PE_InboundDetail_STLT_HNDL.pdf	LMRY_PE_InboundDetail_STLT_HNDL.pdf		306-313
88	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Handlers/ LMRY_PE_Inbound_CLNT_HNDL.pdf	LMRY_PE_Inbound_CLNT_HNDL.pdf		314-318
89	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Handlers/ LMRY_PE_Inbound_STLT_HNDL.pdf	LMRY_PE_Inbound_STLT_HNDL.pdf		319-327

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
90	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Handlers/ LMRY_PE_VendorBillInbound_URET_HNDL.pdf	LMRY_PE_VendorBillInbound_URET_HNDL.pdf		328-331
91	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/ LMRY_PE_InboundPdf_STLT.pdf	LMRY_PE_InboundPdf_STLT.pdf		332-333
92	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/ LMRY_PE_InboundXML_STLT.pdf	LMRY_PE_InboundXML_STLT.pdf		334-335
93	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Libraries/ LMRY_PE_Message_LBRY.pdf	LMRY_PE_Message_LBRY.pdf		336-337
94	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Module/ LMRY_PE_InboundDetail_STLT.pdf	LMRY_PE_InboundDetail_STLT.pdf		338-339
95	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Module/ LMRY_PE_InboundLogDetail_STLT.pdf	LMRY_PE_InboundLogDetail_STLT.pdf		340-342
96	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Module/ LMRY_PE_InboundLog_CLNT.pdf	LMRY_PE_InboundLog_CLNT.pdf		343-347
97	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Module/ LMRY_PE_InboundLog_STLT.pdf	LMRY_PE_InboundLog_STLT.pdf		348-350
98	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Module/ LMRY_PE_Inbound_MPRD.pdf	LMRY_PE_Inbound_MPRD.pdf		351-354
99	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Module/ LMRY_PE_Inbound_STLT.pdf	LMRY_PE_Inbound_STLT.pdf		355-356
100	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Plugin/ LMRY_PE_Inbound_PLGNIMP.pdf	LMRY_PE_Inbound_PLGNIMP.pdf		357-359
101	PROYECTOS INBOUND/sdf-peru-inbound/ com.litamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.litamready.Imrypeinbound/Plugin/ LMRY_PE_Inbound_PLGNATYPE.pdf	LMRY_PE_Inbound_PLGNATYPE.pdf		360-360

#	ARCHIVO	TÍTULO	TIPO	PÁGINAS
102	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Plugin/ LMRY_PE_Inbound_mail_PLGIMP.pdf	LMRY_PE_Inbound_mail_PLGIMP.pdf		361-363
103	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Router/ LMRY_PE_Inbound_ROUT.pdf	LMRY_PE_Inbound_ROUT.pdf		364-365
104	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Setup/ LMRY_PE_InboundSetup_CLNT.pdf	LMRY_PE_InboundSetup_CLNT.pdf		366-370
105	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/Setup/ LMRY_PE_InboundSetup_STLT.pdf	LMRY_PE_InboundSetup_STLT.pdf		371-374
106	PROYECTOS INBOUND/sdf-peru-inbound/ com.latamready.Imrypeinbound/src/FileCabinet/ SuiteApps/com.latamready.Imrypeinbound/ Translator/LMRY_PE_Inbound_Translator.pdf	LMRY_PE_Inbound_Translator.pdf		375-376
107	audit_report.pdf	audit_report.pdf		377-640

GENERADO AUTOMÁTICAMENTE 2026-01-23 18:56 UTC