



Nombre del Proyecto

Consultoría de Migración de Aplicaciones

Cliente

Servicio Nacional
de Aduanas



Fecha

27-01-2016

Versión

1.2

Tipo de documento

Levantamiento situación aplicaciones

34 DipsViajeros

CONTROL DE DIFUSIÓN

Nombre y Apellidos
Gerardo Olmedo Nova
Eduardo Godoy Llanca
Antonio Gabriel González Casado
Rafael Vázquez

1. Nombre y versión

Nro. Catálogo: 34

Nombre del Sistema: DipsViajeros

Versiones identificadas:

- Fuentes aplicación y proyecto EAR en:
 - http://172.40.100.50/svn/sistemas/workshop/DIPSVI/cod_fuente/trunk
- **Vehiculos:** Etiquetado como último EAR de Producción en:
 - http://172.40.100.50/svn/tecnologia/Proyectos/Cons.Migr.Apps/Documentos/1%20Levantamiento/Despliegues_Produccion/WEBLOGIC_Curu%20-%20Antiguo%20Acuario/DipsViajerosBlazeDS.ear

2. Dependencias

2.1. Librerías

Java 1.4

KNA Libraries ([trunk/KNA/WEB-INF/lib](#)) [Usadas en tiempo de compilación y ejecución]

- avalon-framework-4.2.0.jar
- axis.jar
- axis-ant.jar
- backport-util-concurrent.jar
- batik-awt-util.jar
- batik-bridge.jar
- batik-gvt.jar
- batik-svg-dom.jar
- batik-util.jar
- cfgatewayadapter.jar
- commons-beanutils.jar
- commons-codec-1.3.jar
- commons-collections-2.1.jar
- commons-discovery-0.2.jar
- commons-fileupload-1.1.1.jar
- commons-httpclient-3.0.1.jar
- commons-io.jar
- commons-logging.jar
- commons-logging-1.0.4.jar
- concurrent.jar
- flex-messaging-common.jar
- flex-messaging-core.jar
- flex-messaging-opt.jar
- flex-messaging-proxy.jar
- flex-messaging-remoting.jar
- fop.jar
- hsqldb.jar
- itext-1.3.1.jar
- jasperreports-1.3.4.jar

- jasperreports-3.1.2.jar
- javaUtil.jar
- jaxrpc.jar
- libKna.jar
- libKna_tc_2008.12.18_16.34.jar
- log4j-1.2.8.jar
- menuPerfilado.jar
- msbase.jar
- mssqlserver.jar
- msutil.jar
- mysql-connector-java-3.1.11-bin.jar
- Opta2000.jar
- poi-2.0-final-20040126.jar
- saaj.jar
- spring.jar
- spring-jdbc.jar
- wsdl4j-1.5.1.jar
- xalan.jar
- xercesImpl.jar
- xml-apis.jar

Web App Libraries ([trunk/DipsViajeros/Aduanas/WebContent/WEB-INF/lib](#)) [Usadas en tiempo de compilación y ejecución]

- antlr.jar
- beanaction.jar
- commons-beanutils.jar
- commons-digester.jar
- commons-fileupload.jar
- hsqldb.jar
- ibatis-2.3.3.720.jar
- ibatis-2.3.3.x-build.723.jar
- ibatis-common-2.jar
- ibatis-dao-2.jar
- ibatis-sqlmap-2.jar
- jakarta-oro.jar

- ojdbc14.jar
- ojdbc14_g.jar
- struts.jar

2.2. Otros sistemas

No se han encontrado integración con otros sistemas.

2.3. Capas

2.3.1. Presentación

Vista en Flash mediante Flex

Java Server Pages (jsp)

NetUI Tag Library 1.0

Apache Struts Frameworks 1.1

2.3.2. Negocio

Envío de mails con Javax Mail

Generación de pdfs mediante la librería iText 1.3.1

HyperSQL DataBase

Generación de reportes con Jasper Reports 1.3.4

OPTA2000: NonStop Clock and Time Zone simulator.

Manipulación de documentos de Microsoft con Apache POI 2.0-final

Spring Frameworks 2.0.3

2.3.3. Integración

Patrón Data Access Object (DAO)

Oracle Java Database Connectivity (OJDBC)

Simple Object Access Protocol (SOAP)

Flex Messaging

3. Servicios Provistos / Consumidos

3.1. Provistos

9 servicios provistos.

3.2. Consumidos

N/A

4. Funcionalidades base

AduanasWS

- obtenerAduanas
- obtenerAvanzadasAduana

ComunasWS

- obtenerComuna

ConsultasWS

- buscarDipsViajeroPorRangoFechas
- buscarDipsViajeroPorId

DipsViajerosWS

- anularDips
- validarAnularDips
- validarNroDeFolioUtilizado
- validarEliminarDips
- eliminarDips
- obtenerAdValoren
- obtenerAdValorenAcuerdoComercial
- obtenerTipoCambio
- obtenerGlosaRegimenImportacion
- obtenerOtrosImpuestos
- obtenerAduanaPorNroDips

NumerosDeFoliosWS

- ingresarRangoNroFolio
- bloquearRangoNroFolio

RetiroMercanciasWS

- marcarRetiroMercancia

TiposOperacionesWS

- obtenerTiposOperacion

ViajerosWS

- solicitarDatosViajeroNacional
- solicitarDatosViajeroExtranjero
- ingresarDatosViajeroNacional
- ingresarDatosViajeroExtranjero

- modificarDatosViajero
- anularViajero

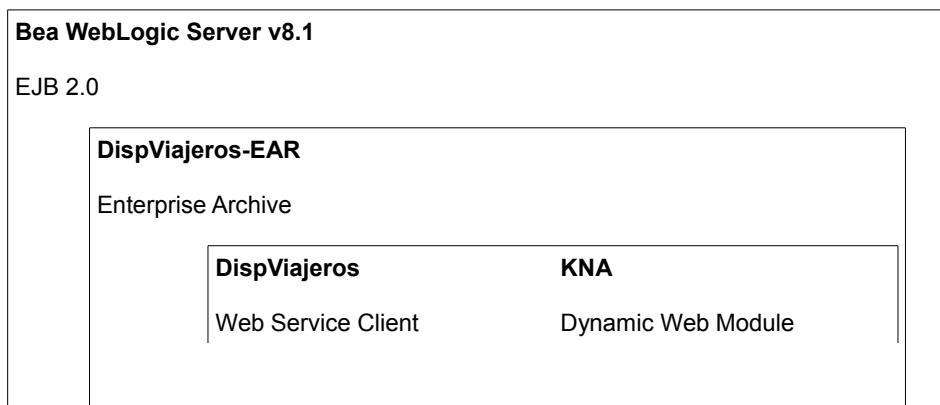
KnaWsInvoke

5. Infraestructura, arquitectura y seguridad

5.1. Modelo de Despliegue

El último ear desplegado se llama DipsViajerosBlazeDS.ear pero este nombre no ha sido encontrado en ninguno de los archivos de los proyectos.

La definición para generar el ear parece encontrarse en [trunk/DipsViajeros_8.1.6/Aduana/.beabuild.txt](#)



5.2. Seguridad

5.2.1. RetiroMercanciasWS

Utiliza el componente Kerberos como protocolo de autenticación que está incluido en la librería sun.security.krb5

5.2.2. Acceso Login

La primera pantalla de la aplicación es un login de control de acceso a la misma mediante usuario y password.

El login de la aplicación lo gestiona la librería NetUi.

5.2.3. Cross-domain

Cross-domain para el intercambio de mensajes y componentes entre la vista (flash) y el negocio.

6. Documentación

6.1. Documentación Descripción General

Existen una gran variedad de documentos con documentación general del sistema en la carpeta [doc_externos](#)

6.2. Casos de Uso

No se ha encontrado documentación.

6.3. Especificación de Requisitos

[1923_Analisis y Diseño DipsViajero.rar](#)

6.4. Documento de Diseño

[1923_Analisis y Diseño DipsViajero.rar](#)

[1923_DIPS_Viajeros_PP_Plan_Despliegue_T1G-SF_1.1.pdf](#)

6.5. Plan de pruebas

[1923_DIPS_Viajeros_PP_Plan_Pruebas_T1G-SF.pdf](#)

[1923_DIPS_Viajeros_PP_Plan_Aceptacion_T1G-SF_1.1.doc](#)

6.6. Manual de Usuario

[Manual_Usuario_DipsViajeros.doc](#)

6.7. Manual del Sistema

[Revision de la instalacion.docx](#)

7. Resumen Estadístico

7.1. Sistema distribuido

Tipo de elemento	Numero
Total archivos una vez distribuido	1220 archivos, 141 carpetas.
JARs distribuidos	37
EARs generados desde el código fuente	1
WARs generados desde el código fuente	0
Páginas JSP	2

7.2. Entorno de desarrollo

Tipo de elemento	Numero
Total fuentes .java	87
Total ficheros .jpf	0
Total ficheros .jcx	0
Total ficheros .jws	11
Esquemas XSD	33
Ficheros .css	5
Ficheros .js	15
PageFlow Tags	0
NetUi Tags	10

7.3. Sentencias SQL

El script de análisis de sentencias `sql sql_aduana_analyzer_v1.1.pl` no es capaz de encontrar ninguna sentencia porque estas se encuentran en archivos .xml.

Tipo de elemento	Numero
call/PL-SQL	93
sqlQuery.insert	1
sqlQuery.select	35
sqlQuery.update	3*
sqlQuery.delete	1

* Corresponden a ejemplos de Flex incluidos en el proyecto KNA

Los comandos utilizados para obtener estas sentencias han sido:

- `grep -ri "".*select.*from"" . --exclude-dir=""svn""`
- `grep -ri "".*update.*set"" . --exclude-dir=""svn""`
- `grep -ri ""{.*call\\s.*"" . --exclude-dir=""svn""`
- `grep -ri "".*delete.*from.*where"" . --exclude-dir=""svn""`

- `grep -ri "".*insert.*into"" . --exclude-dir=""svn""`