

Universidad de las Ciencias Informáticas

Facultad 2



Título: Módulo de carga de datos offline
del Sistema Integral de Análisis de Información.

Trabajo de Diploma para optar por el título de
Ingeniero Informático.

Autor: Pedro A Carbonell Figueredo.

Tutores: Ing. Erik Machado Cano¹
Ing. Osmani Acosta Cantero²

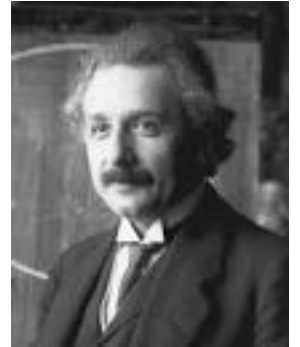
Co-tutor: MSc. Jorge Luis Olmedo Flores.

Junio del 2013.

Pensamiento

“¿Por qué esta magnífica tecnología científica, que ahorra trabajo y nos hace la vida más fácil, nos aporta tan poca felicidad? La respuesta es ésta simplemente: porque aún no hemos aprendido a usarla con tino.”

Albert Einstein



DECLARACIÓN DE AUTORÍA

Declaro ser el único autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Pedro Alejandro Carbonell Figueredo

Autor

Ing. Erik Machado Cano

Tutor

Ing. Osmani Acosta Cantero

Tutor

MsC. Jorge Luis Olmedo Flores

Co-tutor

DATOS DE CONTACTO

¹ Ingeniero en Ciencias Informáticas, graduado en la Universidad de Ciencias Informáticas en el año 2008. Trabaja en la Facultad 2 de la mencionada casa de altos estudios con categoría docente de Instructor. Pertenece al Centro de Telemática, específicamente al proyecto Sistema Integral de Análisis de Información con el rol de desarrollador.

Correo electrónico: emachado@uci.cu

² Ingeniero en Ciencias Informáticas, graduado en la Universidad de Ciencias Informáticas en el año 2011. Trabaja en la Facultad 2 de la mencionada casa de altos estudios. Pertenece al Centro de Telemática, donde trabajó un año en el proyecto Sistema Integral de Análisis de Información con el rol de desarrollador. Actualmente se desempeña laboralmente en el proyecto Centro de Datos.

Correo electrónico: ocosta@uci.cu

AGRADECIMIENTOS

“[Insertar agradecimientos (opcional)]”

DEDICATORIA

RESUMEN

Las herramientas para la extracción, transformación y carga de datos constituyen un nuevo paradigma en la gestión de la información. En este trabajo se presenta una herramienta informática que permite la carga de datos offline relacionados con comportamientos en el uso de la telefonía fija y móvil en Cuba. A lo largo de cuatro capítulos contenidos en el presente documento abordan temas de vital importancia para el desarrollo de dicha aplicación. Se propone la fundamentación acerca de las herramientas utilizadas, además de gran parte de los elementos necesarios para construir el Módulo de Carga de Datos Offline del SIAI cubriendo con el mismo las necesidades expuestas por la Oficina Nacional del Grupo de Investigación de la Dirección de Operaciones de Seguridad (DOPS) de ETECSA cliente principal del presente producto informático.

PALABRAS CLAVE: Extracción, Transformación, Carga, offline.

ÍNDICE

AGRADECIMIENTOS	1
DEDICATORIA	2
RESUMEN.....	3
INTRODUCCIÓN.....	7
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	12
1.1 Introducción	12
1.2 Inteligencia de Negocio – Herramientas para la Extracción, Transformación y Carga de datos ...	12
1.2.1 Características de las Herramientas ETL.....	13
1.2.2 Clasificación de las herramientas ETL.....	14
1.2.3 Herramientas ETL utilizadas a nivel mundial	16
1.2.4 Herramientas ETL en Cuba.....	21
1.2.5 Justificación de las herramientas ETL a utilizar.	21
1.3 Propuesta de lenguajes, metodologías, técnicas y herramientas a utilizar.....	22
1.3.1. Metodología de desarrollo	22
1.3.2. Lenguaje de Modelado	23
1.3.3. Técnica de modelación del negocio.....	23
1.3.4. Herramientas CASE.....	23
1.3.5. Lenguajes de programación.....	24
1.3.6. Entorno Integrado de Desarrollo (IDE)	24
1.3.7. Tecnologías para el diseño de la interfaz de usuario.....	24
1.3.8. Sistema Gestor de Base de Datos	25
1.4 Conclusiones del capítulo	25
CAPÍTULO 2: CARACTERÍSTICAS DE LA APLICACIÓN.....	26
2.1 Introducción	26
2.2 Propuesta del sistema	26
2.3 Modelo de negocio	26
2.4.1 Descripción de los procesos del negocio	29
2.5 Requisitos de software.....	30
2.6 Requisitos funcionales	30
2.7 Requisitos no funcionales	31

2.7.1	Requisitos de Seguridad.	32
2.7.2	Requisito de Usabilidad.	32
2.7.3	Requisito de Fiabilidad.....	32
2.7.4	Requisito de Eficiencia.....	33
2.7.5	Requisito de portabilidad.	33
2.7.6	Requisitos de Estándares de codificación.....	33
2.7.7	Requisito de Diseño.	33
2.8	Definición de los Casos de Uso	34
2.8.1	Definición de los actores del sistema	34
2.8.2	Diagrama de caso de uso del sistema.....	35
2.8.3	Descripción textual de caso de usos.....	35
2.9	Conclusiones del capítulo	42
CAPÍTULO 3: DISEÑO DEL SISTEMA.....		43
3.1	Introducción	43
3.2	Propuesta de arquitectura del sistema.....	43
3.2.1	Patrones de diseño aplicados	44
3.3.	Diagrama de Paquetes	47
3.4.	Diagrama de clases del Diseño	48
3.4.1	Diagrama de clases del caso de uso Administrar notificación.....	49
3.5.	Modelo de Datos.....	49
3.6	Diagrama de despliegue	50
3.6.1	Descripción de los nodos del Diagrama de despliegue	51
3.7	Diagramas de componentes	52
3.8	Conclusiones del capítulo	53
CAPÍTULO 4: PRUEBAS		54
4.1	Introducción	54
4.2	Estrategia de pruebas.....	54
4.2.1	Niveles de Pruebas.....	54
4.2.2	Métodos de Prueba	55
4.2.3	Ejecución de pruebas.....	56
Conclusiones del capítulo.....		62
CONCLUSIONES GENERALES.....		64
RECOMENDACIONES.....		66

REFERENCIAS	67
BIBLIOGRAFÍA.....	69
ANEXOS.....	70
ANEXO 1	70
ANEXO 2	¡Error! Marcador no definido.
ANEXO 3	¡Error! Marcador no definido.
GLOSARIO	¡ERROR! MARCADOR NO DEFINIDO.

INTRODUCCIÓN

Las redes de comunicación son cada vez más rápidas y eficientes según el sitio oficial de la Unión Internacional de Telecomunicaciones (UIT). La invención de los circuitos integrados, el ordenador, los satélites artificiales y la fibra óptica, entre otros, han permitido una genuina revolución de los canales de comunicación, trayendo consigo un amplio desarrollo en todas las ramas de las telecomunicaciones. Hoy en día las mismas constituyen uno de los sectores más desarrollados desde el punto de vista tecnológico brindando enormes ventajas a la sociedad.

Una rama de las telecomunicaciones ampliamente desarrollada en la actualidad es la telefonía, que en el marco del estudio sobre el impacto social de las tecnologías digitales, resulta de gran interés debido a su uso masivo ha dado lugar a notables transformaciones económicas y sociales. Por tanto abordar el fraude en una sociedad totalmente dependiente de las telecomunicaciones es algo de vital importancia, pues la telefonía en sus variantes tanto móvil como fija es propensa a ser víctima de diferentes acciones fraudulentas. Estas ocasionan cuantiosas pérdidas económicas a las empresas telefónicas.

En Cuba los servicios de telefonía son ofertados por la Empresa de Telecomunicaciones de Cuba S.A (ETECSA). La misma tiene como principal misión “Lograr en el período 2011-2015 una gestión efectiva que permita cada vez más brindar servicios de telecomunicaciones que satisfagan las necesidades de los usuarios y la población, así como respaldar los requerimientos de la defensa y del desarrollo socio-económico del país con resultados económicos que de la empresa demanda y espera el Estado cubano.”
(1)

Según datos oficiales de la Oficina Nacional del Grupo de Investigación de la Dirección de Operaciones de Seguridad (DOPS) de ETECSA, en el periodo comprendido entre los meses de enero y septiembre del año 2012 se han detectado en el país 705 casos de fraude telefónicos en sus diferentes variantes las cuales representan una pérdida económica sustancial para el país. Las mismas alcanzan los 12.909.054,84 millones de Pesos Cubanos Convertibles (CUC), reportándose una media mensual por concepto de pérdidas de 1.434.339,30 CUC.

A nivel mundial se han generado diferentes soluciones para hacer frente a estos comportamientos, como por ejemplo los Sistemas de Gestión de Fraude (FMS)¹, que son parte de las herramientas utilizadas en la

¹Del inglés: Fraud Management System

prevención y control del fraude en el campo de la telefonía. “Su funcionalidad clave consiste en realizar un proceso de gestión completamente automático, mediante el seguimiento de los patrones de comportamiento de los usuarios, regido por normas para el uso de los servicios con el fin de detectar una actividad inusual o fraudulenta”. (2)

En Cuba se han puesto en explotación por ETECSA diferentes FMS, estos en su gran mayoría cumplen con los requisitos básicos para la gestión del fraude. De los mismos resalta como característica principal los grandes volúmenes de datos generados. En estos momentos la entidad no cuenta con los medios físicos que posibiliten mantener integrados al sistema estas enormes cantidades de datos. Provocando que los analistas de fraude en ETECSA en sus investigaciones sobre los comportamientos en el uso de los servicios de telefonía fija y móvil no cuenten con un mecanismo que permita cargar los datos que se encuentren fuera de línea (offline), los cuales pudieran ser muy útiles en determinadas investigaciones, ya sea por su relación con sucesos actuales que se estén realizando o porque se encuentren analizando situaciones en sí de esos períodos.

Producto de ello no se puede, como parte de un proceso investigativo, solicitar acceso a información con caducidad mayor de un periodo de tiempo determinado, por lo que no pueden ser ubicados en un banco de datos, específicamente en una zona donde puedan ser preparados para ser interrogados atómicamente, tanto en consultas que los involucren a ellos solos como otras donde dichos datos y otros online puedan correlacionarse. Además no se cuenta con un catálogo descriptor del contenido de cada salva offline, lo cual facilitaría la localización con mayor precisión de la información offline que se desee cargar. Tal escenario conlleva a demoras en la clasificación de determinados comportamientos como anómalos para su inmediato seguimiento, provocando que pueda aumentar el número de fraudes y por consiguiente pérdidas económicas al país.

Partiendo de la problemática anterior se plantea como **problema a resolver**: ¿Cómo garantizar que la información con caducidad mayor de un período de tiempo que se encuentra offline pueda ser cargada en el Sistema Integral de Análisis de Información (SIAI) para ser utilizada en el proceso de investigación por parte de los analistas de fraude en ETECSA sobre el uso de los servicios de telefonía fija y móvil en Cuba?

A raíz del problema anterior se establece como **objeto de estudio**: los procesos de gestión de la información offline, para posibilitarla carga de elementos de interés en un banco de datos. Se tiene como **objetivo general** desarrollar una aplicación que permita cargar la información con caducidad mayor de un período de tiempo que se encuentra offline en el SIAI, para ser utilizada en el proceso de investigación por parte de los analistas de fraude en ETECSA sobre el uso de los servicios de telefonía fija y móvil en Cuba.

Las **tareas** para dar cumplimiento al objetivo general se mencionan a continuación:

Elaboración de la fundamentación teórica del trabajo.

- Estudio de sistemas que posean cargadores de datos para conocer su estado del arte y tendencias a nivel nacional e internacional.
- Estudio de mecanismos de carga de datos offline.
- Selección y estudio de herramientas a utilizar para desarrollar el módulo de carga de datos offline del SIAI.
- Análisis de la estructura de los datos caducados que pasan al estado offline.
- Identificación de los procesos y reglas del negocio.
- Elaboración del modelo de negocio de los procesos que tienen lugar en el organismo en cuestión, para la comprensión del flujo normal de funcionamiento para cargar los datos offline del SIAI.
- Descripción de los procesos representados en el modelo de negocio.
- Definición de requerimientos funcionales y no funcionales de la solución de software para cargar los datos offline del SIAI.
- Descripción de los requerimientos funcionales y no funcionales de la solución de software para cargar los datos offline del SIAI.
- Elaboración del modelo de diseño de la solución de software para cargar los datos offline del SIAI.
- Implementación de la solución de software para cargar los datos offline del SIAI, teniendo en cuenta aspectos fundamentales como el diseño y construcción de sub módulos que permitan:
 - La gestión y posterior atención de las solicitudes.
 - La carga y descarga de datos en el Banco de Datos.

- La creación y gestión de reportes relacionados con toda la actividad realizada por la aplicación.
- Realización de pruebas de funcionalidad a la solución de software.
- Realización de los manuales de usuarios necesarios para guiar a los futuros beneficiarios del módulo de carga de datos en un correcto uso y explotación de las diferentes funcionalidades del mismo.

Elaboración de una adecuada documentación de la solución en todas sus etapas (estudio del estado del arte, modelamiento del negocio, diseño, implementación y prueba).

Teniéndose como **campo de acción**: los sistemas de carga de datos para la gestión de la información relacionada con comportamientos en el uso de los servicios de telefonía fija y móvil en Cuba.

Métodos científicos utilizados:

Métodos teóricos:

- Analítico-Sintético: Se utilizó este método durante todo el proceso investigativo, ya que permitió la descomposición del problemas en diferentes partes, lo que garantizó una mejor comprensión del mismo.
- Histórico-lógico: Utilizado para establecer un estudio de los diferentes comportamientos y características específicas que brindan cada uno de los Sistemas especializados en la Carga de datos para la determinación y selección de la propuesta específica de las herramientas a utilizar.
- Modelación: Método utilizado principalmente en la investigación debido a que permitió la representación de diferentes ideas de forma gráfica, lo que posibilitó un mayor entendimiento, fundamentalmente en la etapa de implementación del sistema.

Métodos empíricos:

Entrevista: Método ampliamente ejecutado en el proceso de entendimiento del negocio y los requerimientos del sistema.

A continuación se presenta una breve descripción de los capítulos que componen el presente trabajo de diploma:

Capítulo 1: “Fundamentación Teórica”. Se tratan conceptos y definiciones fundamentales con el objetivo de formar la base teórica del objeto de estudio, se presentan características importantes de algunas aplicaciones informáticas que se utilizan para extracción, transformación y carga de datos, además se examinan, lenguajes, metodologías, técnicas y las herramientas a utilizar para el desarrollo de la herramienta.

Capítulo 2: “Características del sistema”. Se definen aspectos fundamentales para el desarrollo del sistema entre los cuales destacan la descripción de los procesos del negocio y la presentación de los requisitos funcionales y no funcionales que debe cumplir la solución de software propuesta.

Capítulo 3: “Diseño del sistema”. En el mismo se propone y describe la arquitectura empleada para el desarrollo del sistema y los patrones de diseño utilizados. Además se muestra una vista lógica y física del sistema con la presentación del diagrama de clases de diseño, diagrama de componentes, modelo de datos y diagrama de despliegue.

Capítulo 4: “Prueba”. Se presenta y ejecuta la estrategia de pruebas a desarrollar sobre el sistema, analizando los resultados obtenidos en cada iteración.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

La necesidad de contar con herramientas que faciliten la integración de los datos siempre ha constituido uno de los principales retos para el hombre. La informática brinda hoy las soluciones más eficientes y adaptables a los diferentes entornos, siendo estos delimitados fundamentalmente por el tipo de información a gestionar. Para el desarrollo del Módulo Cargador de Datos Offline del Sistema Integral de Análisis de Información (SIAI), se realiza un estudio teniendo en cuenta los principales conceptos en este ámbito, además de las principales tecnologías utilizadas a nivel mundial, en Cuba y en la UCI como centro para el desarrollo de la informática en el país. En este capítulo se analizan temas relacionados con las Herramientas Extract, Transform, Load (*del Inglés: Extraer, Transformar, Cargar*) (ETL), tendencias actuales en su uso, características, conceptos elementales así como las ventajas de su utilización.

1.2 Inteligencia de Negocio – Herramientas para la Extracción, Transformación y Carga de datos

Partiendo que “Inteligencia de Negocio (del Inglés: Business Intelligences) es la habilidad para transformar los datos en información, y la información en conocimiento, de forma que se pueda optimizar el proceso de toma de decisiones en los negocios. Asociándolo directamente con las tecnologías de la información, se puede definir inteligencia de negocio como el conjunto de metodologías, aplicaciones y tecnologías que permiten reunir, depurar y transformar datos de los sistemas transaccionales e información estructurada (interna y externa a la organización) en información estructurada, para su explotación directa (reporting (del Inglés: reporte),, análisis OLTP / OLAP, alertas...) o para su análisis y conversión en conocimiento, dando así soporte a la toma de decisiones sobre el negocio ”.²

En los proyectos Business Intelligence se destaca como componente imprescindible y quizás el más crítico, el proceso de integración de datos en un repositorio, este debe permitir almacenar la información para ser posteriormente explotada por herramientas de análisis. Para dar una solución eficiente al problema se incorporan los componentes ETL.

Para conocer que es una herramienta ETL se hace necesario definir qué es un proceso ETL:

Proceso ETL: “El sistema Extraer-Transformar-Cargar (ETL) es la base del almacén de datos. Un sistema ETL bien diseñado extrae datos de los sistemas fuente, hace cumplir las normas de calidad de los datos y

² Corporativo 2006

la consistencia, conforma datos de manera que las fuentes independientes se pueden utilizar juntas, y finalmente entrega los datos en un formato de presentación listos para que los desarrolladores puedan crear aplicaciones y los usuarios finales puedan tomar decisiones” (3)

“Un buen proceso ETL permite una integración consistente de los datos. Al culminar este proceso, los datos serán; precisos, completos, creíbles, rigurosos en el tiempo, interpretables, accesibles y con valor añadido. Este proceso no es una simple migración, pues asegura la calidad de los datos, los prepara, conforma, limpia, transforma, y organiza, brindando a quienes utilicen estos datos un conocimiento que facilitará la toma de decisiones.” (4)

1.2.1 Características de las Herramientas ETL

“Las herramientas para la Extracción, Transformación y Carga son las encargadas de desarrollar los procesos ETL, impulsando el flujo de datos, haciendo transformaciones intermedias y permitiendo una integración de datos exitosa. Es por eso que cada paso, desde su diseño de acuerdo a los requerimiento de cada negocio, hasta su eficiente puesta en marcha, precisa de la mayor atención y esfuerzo posible.” (5). Entre las principales características de una herramienta ETL se destacan:

“Conectividad: Se refiere a los distintos orígenes de datos que da soporte la herramienta, desde base de datos relaciones y no relacionales, distintos formatos de archivos, XML, aplicaciones empresariales (ERP, CRM y SCM), emails, herramientas de ofimática etc.

Capacidades de entrega de datos: Es la posibilidad que posee el software de brindar datos a otras aplicaciones, ya sea mediante proceso bash, o en tiempo real.

Capacidades de transformación de datos: Permite transformar datos, desde lo más simple como conversión del tipo de datos, o cálculos simples para mencionar algunos ejemplos, hasta transformaciones más complejas como agregaciones, sumalizaciones o lookups.

Capacidades de metadatos y modelado de datos: Es la recuperación de los modelos de datos desde sus respectivos orígenes de datos o aplicaciones, mapeo del modelo físico al lógico, documentación, y sincronización de los cambios en los distintos componentes de la herramienta.

Capacidades de diseño y entorno de desarrollo: Comprende la representación gráfica de los objetos del repositorio, modelos de datos, y flujos de datos, soporte para el test y debugging, también incluye a las capacidades para trabajo en equipo. ” (5)

¿Cómo está estructurada una herramienta ETL?

Según Kimball (Wiley 2004), los procesos ETL trabajan principalmente en dos partes, en la primera los usuarios podrán acceder a los datos para realizar diferentes tipos de análisis con las diversas herramientas que se posean. En la segunda área solo accede el personal especializado en la integración de datos, la misma está totalmente fuera del alcance de los usuarios, pues en este punto se realizan acciones sobre los datos, los mismos son extraídos, limpiados, depurados, normalizados, conformados, manipulados y finalmente preparados para su inserción en el banco de datos.

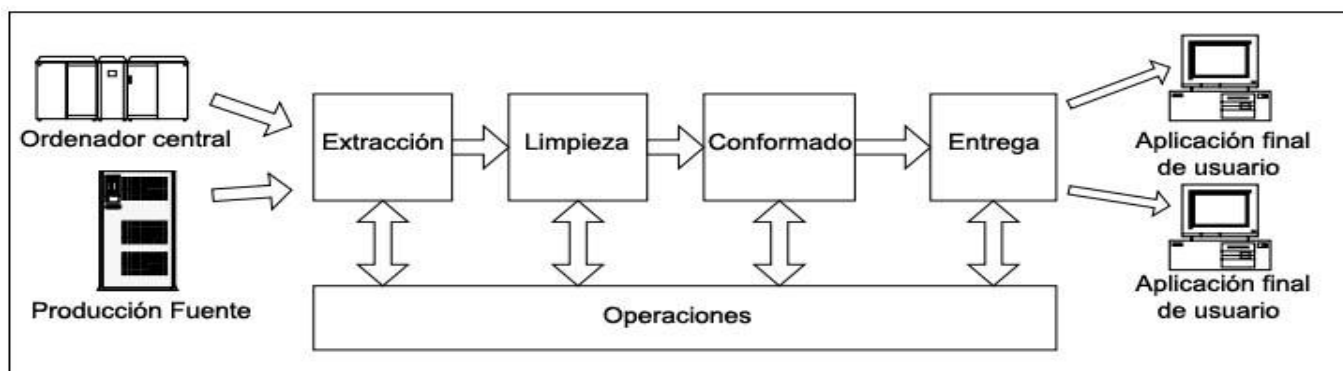


Fig. 1: Flujo de datos según Ralph Kimball³

1.2.2 Clasificación de las herramientas ETL

Una herramienta ETL correctamente diseñada debe extraer los datos de los sistemas fuentes, asegurar la calidad y consistencia de los datos, estandarizarlos de orígenes divergentes para su utilización de forma conjunta y para finalizar generar los datos en el formato establecido para su manejo.

Para conformar una herramienta ETL se deben tomar decisiones importantes acerca de la forma en que se va a construir el sistema ETL. Las mismas son clasificadas por su principio de funcionamiento en 2 grandes grupos:

Tool-Based ETL (De propósito general)

Herramientas ETL diseñadas de forma genérica para adaptarse a diferentes entornos, posibilitando su integración con muchos negocios.

Entre sus principales ventajas se encuentran:

³ Figura extraída de **Kimball, RLH.** *The Datawarehouse ETL Toolkit.* 2004

- “Son utilizadas eficientemente por personal técnico con amplios conocimientos del negocio y no necesariamente por programadores profesionales.
- Algunas tienen integrados repositorios de metadatos esto permite sincronizar metadatos de los sistemas fuentes y otras herramientas BI.
- La mayoría de estas herramientas generan automáticamente metadatos en cada paso del proceso, proporcionando una metodología consistente de control de metadatos.
- Poseen conectores pre construcción para la mayoría de las fuentes y sistemas objetivos. En un nivel más técnico de estas herramientas ya que son capaces de manejar todo tipo de conversiones complejas de tipos de datos.
- Usualmente proveen encriptación en línea y capacidad de compresión.
- Pueden manejar la carga balanceada en complejos escenarios a través de servidores evitando estancamientos.
- El aprovechamiento de estas herramientas puede ser aumentado con la inserción de módulos de procesamiento hechos a mano.(...)” (5)

Hand-Coded ETL (A medida)

Herramientas diseñadas para resolver una problemática específica, solamente pueden adaptarse al negocio para el cual fueron diseñadas. Las mismas en su gran mayoría no son publicadas para su utilización por terceros, pues son herramientas solicitadas por determinados clientes para ser insertadas como mecanismos auxiliares en sus respectivos negocios.

De la misma resaltan como características principales:

- “Los sistemas de codificación a medida cuentan con herramientas automáticas para la realización de pruebas de unidad.
- Las técnicas de programación orientadas a objetos ayudan a hacer reportes de errores, validaciones y actualizaciones de metadatos en las transformaciones.
- Se pueden gestionar directamente los metadatos en los sistemas codificados a medida y crear sus propias interfaces de metadatos.
- Un breve análisis de los requisitos de una herramienta ETL apunta con rapidez al procesamiento basado en ficheros, no a procedimientos de almacenados. El procesamiento

basado en ficheros es más directo, los mismos son simplemente codificados, fáciles de probar y entender.

- Se verá limitada solamente por las habilidades del programador y el lenguaje de programación.
- Las Herramientas ETL codificadas a medida brindan flexibilidad ilimitada.(...)” (5)

1.2.3 Herramientas ETL utilizadas a nivel mundial

“Existen componentes ETL que proveen las herramientas BI y hasta empresas que se dedican exclusivamente a desarrollar este tipo de aplicaciones. En los últimos 5 años ha crecido el segmento de herramientas ETL más del 50 % en lo que se refiere a las ventas de licencias de productos y de servicios asociados a nivel mundial.” (14) Entre ellos se destacan:

- “Almacén de Datos: Casi el 80% del uso está relacionado con esta arquitectura. Incluyendo Mercados de Datos que dan soporte a un proyecto.
- Tareas de Bases de datos: También se utilizan para consolidar, migrar y sincronizar bases de datos operativas.
- CDI: El Customer Data Integration es una función que permite unificar y homogenizar la cartera de clientes de las grandes corporaciones, disueltas y repetidas en diferentes fuentes.
- Gobierno: Algunos gobiernos consolidan información transaccional de todos los procesos magnéticos, electrónicos en un ODS (Almacén Operacional de Datos) para controlar en línea ciertas operaciones.” (14)

Las herramientas ETL además pueden ser clasificadas atendiendo al tipo de licencia fijada por su vendedor, por un lado se encuentran las Privativas y por otro las de Código abierto. A continuación se exponen aspectos importantes de los principales sistemas ETL que se encuentran actualmente en el mercado.

Herramientas ETL de código abierto:

La comunidad de código abierto se encuentra continuamente en evolución ofreciendo una alternativa tecnológica viable para las empresas. Partes del mercado mundial actualmente desarrollan alternativas de código abierto, ejemplo de ello se tiene:

Pentaho Data Integration

“Pentaho es la empresa líder en Commercial OpenSource Business Intelligence & DataIntegration. “Pentaho Data Integration es parte de Pentaho Open Source Business intelligence suite. Incluye software para todas las áreas de soporte para realizar decisiones del negocio, utilidades para la gestión de almacenes de datos, herramientas para la integración y análisis de datos y software para administradores y herramientas de minería de datos.” (11).

Esta nueva versión se integra en un único entorno de desarrollo la posibilidad de confeccionar las ETL, el modelar los metadatos y la visualización de datos, ya sea a través del navegador OLAP o por medio de reportes. Esta forma de construir soluciones de BI es mucho más rápida. La misma presenta las siguientes ventajas:

- “Facilita la comunicación entre los stakeholders y los desarrolladores. Estos últimos obtienen de forma más rápida prototipos funcionales (incluso en minutos) de cubos OLAP, reportes, etc. y a su vez, los stakeholders pueden entregar su retroalimentación a los desarrolladores en base a un prototipo que se ve exactamente como sería en un ambiente productivo.
- Permite llegar a un resultado final de forma más efectiva.
- Reduce riesgos y costos de implementación.
- Permite probar de forma empírica y temprana la arquitectura de la aplicación BI es decir que los datos necesarios son los correctos, que el tiempo de ejecución es aceptable, que los reportes muestran la información que los usuarios necesitan, que el diseño del cubo OLAP satisface las necesidades de información de los analistas, etc).
- Permite a los desarrolladores enfocarse en funcionalidad permitiendo el uso de transacciones.
- En proyectos de gran envergadura se evita elaborar documentos de diseño que meses después puedan resultar obsoletos. A cambio permite enfrentar los detalles “En tiempo”. “ (12)

Talend Integrator Suite

“Producto insignia de Talend, Talend Open Studio, es la más abierta, innovadora y poderosa solución para la integración de datos en el mercado. Talend Open Studio provee capacidades avanzadas que mejoran dramáticamente la productividad del diseño de trabajo de integración de datos y escalabilidad probada para asegurar una ejecución óptima.” (10)

“Talend Open Studio incluye 3 aplicaciones: Business Modeler, Job Designer and Metadata Manager, ellas son herramientas para soporte del negocio y no requieren conocimientos técnicos particulares. Las herramientas tienen también funciones incorporadas para monitorear los trabajos realizados y analizar problemas ocurridos.” (10). Entre sus principales ventajas destacan:

- La suite proporciona capacidades para monitorear datos, esto provee una visión centralizada de todo el proceso de integración.
- Además incorpora poderosas capacidades para administrar toda la integración de datos, desde los trabajos más simples hasta los más complejos.
- Balanceador de CPU proporciona el más alto grado de paralelismo en los procesos” (10)

Jasper ETL

“JasperETL es considerada como una de las soluciones más sencillas para la integración, limpieza, transformación y movimiento de datos en el mercado. Es una plataforma de integración de datos de alto desempeño, puede ser utilizada por cualquier organización. JasperETL no es solamente una herramienta de integración de datos, pues es parte de Jaspersoft Business Intelligence Suite. Sus principales usos están orientados a la inserción de grandes volúmenes de datos de diferentes fuentes, la ampliación de una solución de BI para incluir a los Data Warehouses y Data Marts, incremento y perfeccionamiento de las consultas y el análisis de los sistemas. Destacando como ventajas principales:

- Utiliza el modelador del negocio para ganar acceso a vistas no técnicas del flujo de trabajo de la información.
- Permite visualizar y editar el proceso ETL usando herramientas editoras de gráficos.
- Está capacitado para generar código portable en Java o Perl, pudiendo ejecutar este código en cualquier otra máquina.
- Permite la inserción y salida desde diferentes fuentes utilizando ficheros planos, ficheros XML, servicios web, bases de datos y servidores con múltiples conexiones
- Utiliza la Activity Monitoring Console⁴ (AMC) para monitorear volúmenes de datos, tiempos de ejecución y eventos de trabajo.” (10)

Herramientas ETL Privativas:

⁴Del inglés: Consola de Monitoreo de la Actividad

Las soluciones privadas juegan un papel importante en el desarrollo de las herramientas ETL, haciendo grandes incursiones sobre todo en el mercado de las herramientas a medidas. A continuación se describen algunas de las herramientas ETL privadas existentes en el mercado:

IBM (Information Server Infosphere Platform)

“IBM InfoSphere Information Server es una plataforma de integración de datos líder en el Mercado, que lo ayuda a entender, limpiar, transformar y entregar información confiable para sus iniciativas críticas del negocio como análisis del negocio, almacenamiento de datos, gestión de datos y aplicaciones para la migración.

InfoSphere Information Server provee todo lo que se necesita para integrar información heterogénea a través de su sistema para asegurar que la información es consistente, representada acertadamente, transformada fiablemente y regularmente actualizada.” (10). Encontrándose entre sus ventajas principales:

- “Fuerte visión del mercado, flexible.
- Progreso encaminado a la plataforma común de metadatos.
- Alto nivel de sofisticación para el cliente y gran variedad de iniciativas.” (10)

Informatica PowerCenter

“Informatica Power Center maximiza el valor de los datos del negocio. PowerCenter unifica la plataforma de integración de datos para el acceso, descubrimiento e integración de datos para cualquier sistema de negocio, en cualquier formato.” (10). Destacando entre sus ventajas principales:

- “Grabación consistente de datos, tecnología sólida, curva de aprendizaje sencillo, habilidad de direccionar en tiempo real los diferentes escenarios de integración de datos.
Esta herramienta está altamente especializada en procesos ETL y la Integración de Datos, centrándose principalmente en estos dos temas no en BI como un todo.” (10)

Oracle

“Oracle Data Integrator Enterprise Edition 11g perfecciona el movimiento y transformación de datos a través de plataformas empresariales con sus arquitecturas abiertas e integradas. Incluye conectividad integrada a todas las principales bases de datos, almacenes de datos, servicios orientados a arquitecturas, BI. Oracle Data Integrator Enterprise Edition ayuda a reducir el costo total de propiedad

entre arquitecturas centradas en datos. Oracle Data Integrator Enterprise Edition está totalmente integrado con Oracle Fusion Middleware, Oracle Database, y Exadata para poner los datos en el centro de su empresa.” (10) Citando entre sus ventajas más relevantes:

- “Basado en Oracle Warehouse Builder y Oracle Data integrator dos herramientas muy poderosas.
- Brinda una conexión firme entre todas las aplicaciones de Oracle Datawarehousing.
- Tendencia de integrar todas las herramientas en una aplicación y un ambiente definido para las operaciones sobre datos”.(10)

SAS ETL Studio

“SAS Data Integration Studio le da “El Poder de Conocer” cuán rápido se puede alcanzar y gestionar datos consistentes y confiables en toda la organización. Se ofrece una completa solución de integración de datos esto provee organización, flexibilidad, confiabilidad y agilidad para responder rápidamente con nuevos requerimientos de datos, consolidar vendedores, estandarizar en solución de integración y reducir el costo total por integración de datos.” (10), dentro de sus principales ventajas se tiene:

- “Puede trabajar sobre muchos sistemas operativos y recopilar datos a través de varias fuentes, muy flexible.
- Gran soporte para las compañías de clase empresarial, también para las de medio y pequeño calibre“ (10)

Microsoft SQL Server Integration Services (SSIS)

“SSIS – SQL Server Integration Services conecta y transforma diferentes tipos de fuentes de datos. Ofrece una plataforma escalable para la integración de datos, para extraer, transformar, cargar (ETL) y con capacidades de integración usando Microsoft SQL Server R2 Integration Services” (10), citando entre sus principales ventajas:

- “Amplia documentación y soporte, herramienta construida utilizando las mejores prácticas en almacenes de datos.
- Fácil y rápida implementación.
- Integración de datos estandarizada.
- La herramienta posee amplias capacidades para el procesamiento de datos en tiempo real.

- Relativamente de bajo costo, excelente soporte y modelo de distribución.” (10)

1.2.4 Herramientas ETL en Cuba

En las empresas cubanas el avance de tecnologías de la información y las comunicaciones en lo que se refiere a conocimiento y desarrollo, tiene como objetivo fundamental el uso racional de los recursos, tratar de obtener la mayor productividad posible y los productos cada vez con una mejor calidad. Por eso se ha pensado desde ya, en el uso de los almacenes y mercados de datos, puesto que los mismos aportarían una solución innovadora a las empresas cubanas capaces de aventurarse en el mundo de la Inteligencia del Negocio.

“La Universidad de las Ciencias Informáticas está enfrascada en lograr que los proyectos cada día se hagan con una mayor calidad, y está tomando medidas para que esto se haga realidad. De ahí que hay que ganar experiencia en el uso de herramientas ETL para poderlas utilizar después en el diseño de grandes almacenes de datos. Anteriormente había surgido el tema del Data Warehouse en la facultad 2, específicamente en el proyecto SINSEC. Es importante destacar también, que la facultad 4 no ha querido quedarse atrás en esta nueva fase de desarrollo tecnológico, por lo que estudiantes pertenecientes a la misma, desarrollaron un Data Mart para la gestión del conocimiento, algo que ha influido mucho en los nuevos seguidores del avance informático, ya que estos le pueden ofrecer apoyo y experiencia en su trabajo.” (26)

En la UCI se han desarrollado diferentes investigaciones contenidas principalmente en tesis en las cuales se abordan el tema de las herramientas ETL, pero vale destacar que el desarrollo de las ETL en la UCI y en general en el país se encuentra en una etapa embrionaria, pues muchas de las variantes antes existentes para el tratamiento de datos se están trabajando en la actualidad bajo los conceptos ETL.

1.2.5 Justificación de las herramientas ETL a utilizar.

Inicialmente se llevó a cabo un estudio de las principales herramientas ETL libres de propósito general, para su inserción como parte de la solución a desarrollar. Para ello se tuvieron en cuenta aspectos fundamentales que dichas herramientas deberían cumplir para su selección, ejemplo de ellos se tienen:

- Nivel de integración con los mecanismos de comunicación existentes en el SIAI.
- Interacción con dispositivos de almacenamiento extraíbles.
- Gestión de catálogos de información.

- Control de reportes de actividad.
- Orientada al negocio de la gestión de software.

Luego del análisis realizado a cada una de las herramientas, se puede decir que en su mayoría no poseen el nivel de integración necesario con el negocio. Detectándose deficiencias en el cumplimiento de algunas de las características expuestas con anterioridad, específicamente en la integración con los mecanismos de comunicación existentes en el SIAI, en la gestión de los catálogos de información y en su orientación a la gestión del fraude. Se continuó el estudio con las Herramientas ETL privativas de propósito general con el objetivo de adquirir una visión acerca de sus principales características y principios de funcionamiento de las mismas para brindarle una mayor visión al equipo de desarrollo sobre sus potencialidades.

Se concluye que es necesario desarrollar un sistema ETL a medida para la solución del negocio en cuestión, incluyendo de forma directa en el sistema una plataforma para la administración de las Notificaciones de solicitud de carga de datos (NSCD) y de Carga de datos realizadas (NCDR), además de funcionalidades necesarias para la generación de catálogos de información, así como mecanismos para la gestión de las actividades realizadas por el sistema. Dicho sistema debe estar totalmente integrado con los mecanismos de comunicación existentes en el SIAI y totalmente orientado a la gestión del fraude en el entorno cubano.

1.3 Propuesta de lenguajes, metodologías, técnicas y herramientas a utilizar.

1.3.1. Metodología de desarrollo

Proceso Unificado del Software (RUP): *“RUP constituye una forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo (quién hace qué, cuándo y cómo), y tiene como objetivo asegurar la producción de software de calidad dentro de plazos y presupuestos predecibles. Dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental (...)” (2)*

RUP como metodología de desarrollo permite un seguimiento detallado de cada una de las fases por las que transita el producto y un dominio elevado del estado real del mismo. El proyecto SIAI tiene concebido el desarrollo sobre esta metodología porque el mismo posee un gran tiempo de duración y envergadura, ya que en el mismo converge personal de diferentes entidades, en su mayoría especialistas (negocio y desarrollo de software) de ETECSA así como estudiantes y profesores de la UCI. Debido a la diversidad del personal y teniendo en cuenta la incorporación de nuevos integrantes al proyecto se hace imprescindible una detallada documentación en cada fase de desarrollo para una mejor comprensión por

parte de todos los implicados. Por estas características se toma RUP como metodología a poner en práctica para el desarrollo del SIAI, incluyendo el módulo de Carga de Datos Offline del SIAI (SIAIODL).

1.3.2. Lenguaje de Modelado

Lenguaje Unificado de Modelado (UML Lenguaje Unificado de Modelado (UML): *“UML es un lenguaje para construir modelos; no guía al desarrollador en la forma de realizar el análisis y diseño orientados a objetos ni le indica cual proceso de desarrollo adoptar. UML utiliza los diagramas gráficos para obtener distintos puntos de vista de un sistema.”* (28)

Se selecciona a UML para facilitar la comprensión de procesos en la construcción del software, brindando complementos visuales para modelar sistemas. UML se adapta a muchas metodologías de desarrollo de software y en especial RUP hace uso del mismo. Además provee muchos artefactos para la comprensión y análisis de las fases involucradas en la construcción de un software.

1.3.3. Técnica de modelación del negocio

Definición de la Integración para la Modelización de las Funciones (IDEF0): *“Es un método diseñado para modelar las decisiones, acciones y actividades de una organización o sistema (...).”* (29) IDEF0 organiza el análisis de los sistemas permitiendo una excelente comunicación entre el analista y cliente. Es de gran utilidad a la hora de establecer el alcance del análisis, principalmente para un análisis funcional, además de que ayuda al modelador a identificar cuáles son las funciones que se desarrollan y lo necesario para desempeñarlas. Se propone la utilización de IDEF0 como una alternativa para la descripción del negocio, permitiendo el ahorro de tiempo y esfuerzo.

1.3.4. Herramientas CASE

Microsoft Viso: Esta herramienta permite la creación de diagramas con apariencia profesional que ayudan a comprender, documentar y analizar información. Mejorando en gran medida la comprensión de sistemas y procesos. Herramienta sencilla y fácil de utilizar, ofrece un diagrama predefinido para modelar los procesos de IDEF0 permitiendo la realización de diagramas de procesos del negocio con una alta calidad visual.

Visual Paradigm 5.0: *“Visual Paradigm para UML (VP-UML) es una herramienta de diseño UML y herramienta CASE UML diseñada para ayudar al desarrollo de software. VP-UML soporta los principales estándares de la industria tales como (UML), ofrece un conjunto completo de herramientas utilizadas por los equipos de desarrollo de software para la captura de requisitos, planificación de software, la planificación de controles, el modelado de clases, modelado de datos.”* (34) Por razones de familiarización del personal en el diseño de diagramas de las diferentes fases del desarrollo del software, se propone Visual Paradigm para el diseño de artefactos.

1.3.5. Lenguajes de programación

Python 2.7: Potente lenguaje de programación, simple, sencillo de aprender y libre. Se propone su utilización pues brinda muchas funcionalidades para aplicaciones de escritorio que trabajen a un nivel bien cercano al sistema operativo, además posee mecanismos cómodos y eficientes para su integración con bases de datos y la utilización de Servicios Web.

1.3.6. Entorno Integrado de Desarrollo (IDE)

Eclipse (Helios): Entorno de desarrollo multilenguaje, de código abierto y libre que se ejecuta sobre la máquina virtual de Java, con una comunidad muy activa de desarrolladores. Posee una arquitectura basada en plugins que permite añadir soporte para varios lenguajes, entre los cuales se encuentra Python el lenguaje de programación seleccionado para desarrollar el sistema. Además posibilita la integración de muchas librerías utilizadas por dicho lenguaje. Por todas estas características se propone la utilización de Eclipse.

1.3.7. Tecnologías para el diseño de la interfaz de usuario

QT Designer 4.7.4: Herramienta multiplataforma para el diseño de interfaz gráfica de usuarios que permite trabajar sobre una gran gama de sistemas operativos como: Windows, Mac OS X, GNU/Linux y la mayoría de las versiones comerciales de Unix. Qt es una herramienta libre y de código abierto ampliamente utilizada para el diseño de las interfaces de usuario. Ofrece gran cantidad de componentes visuales y funcionalidades para el desarrollo de aplicaciones, razones por la cual se propone para su uso en el diseño de la interfaz de usuario de la aplicación..

1.3.8. Sistema Gestor de Base de Datos

Oracle 11g: *“Con más de 400 funcionalidades, 15 millones de horas de test y 36.000 meses-hombre de esfuerzo de desarrollo, la Base de datos Oracle 11g es el producto más innovador y de mayor calidad que Oracle ha anunciado hasta la fecha.”* (32) Ofrece varias mejoras altamente poderosas para la gestión de bases de datos con Oracle. Oracle 11g es compatible para sistemas operativos sobre plataforma GNU/Linux, proporciona herramientas que permiten y garantizan la solución de problemas en complejos entornos empresariales, asegurando su alto rendimiento, fiabilidad y sobre todo seguridad. Actualmente todos los sistemas desplegados en ETECSA trabajan utilizando como sistema gestor de base de datos a Oracle, motivo principal por el cual el cliente solicita la utilización del mismo. Además tanto el personal vinculado al proceso de desarrollo como el cliente están plenamente familiarizados con Oracle y sus herramientas. Razones por las cuales se propone Oracle como sistema gestor de Base de Datos de la solución. .

1.4 Conclusiones del capítulo

Durante este capítulo fueron abordados temas relacionados con el objeto de estudio, se realizó una profunda investigación con el objetivo principal de poder contar con elementos sólidos que permitan la selección de una o varias alternativas adecuadas para emplear en la realización del Módulo Cargador de Datos Offline del proyecto SIAI. Se desarrolló un estado del arte de los principales sistemas ETL utilizados mundialmente, para la selección de un diseño base de una herramienta ETL que satisfaga el negocio. Se seleccionaron y argumentaron las metodologías, tecnologías y herramientas necesarias para desarrollar eficientemente la solución propuesta.

CAPÍTULO 2: CARACTERÍSTICAS DE LA APLICACIÓN

2.1 Introducción

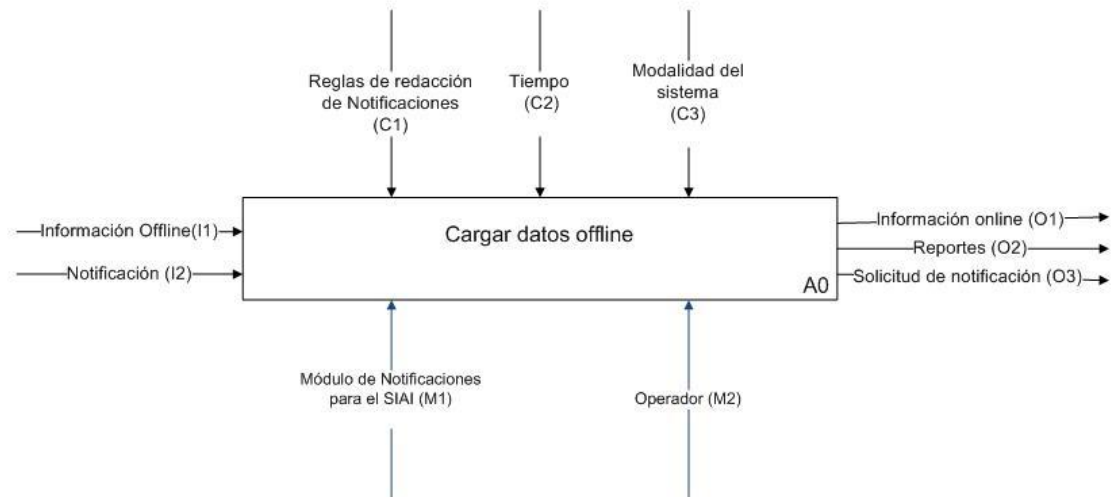
Después de seleccionar las herramientas y metodología a utilizar en el desarrollo de la aplicación, se realizará el levantamiento de los requisitos, para dar una solución factible al problema que se investiga. En el presente capítulo se abordarán las características de la aplicación que se pretende desarrollar. Se definen los requisitos funcionales y no funcionales, modelo de dominio, entidades y actores que intervienen, así como diagrama el de caso de uso del sistema.

2.2 Propuesta del sistema

La carencia de un mecanismo que permita la carga de datos que se encuentran fuera de los sistemas utilizados por ETECSA para la gestión de datos relacionados con comportamientos anómalos en el uso de los servicios de telefonía fija y móvil en Cuba, expone como necesidad la creación de una aplicación que permita a los analistas del fraude de ETECSA poder solicitar la carga de datos correspondientes a determinados períodos de tiempo. Por lo que el sistema deberá gestionar la atención a todas estas solicitudes realizadas. Además debe realizar catálogos descriptores de los diferentes datos almacenados en los dispositivos de almacenamiento extraíbles en este caso el cliente propone la utilización de DVD aunque deberá interactuar con otros dispositivos, brindando así la posibilidad de atender de forma manual o automática las solicitudes capturadas por el sistema. La aplicación debe garantizar también que los datos insertados sean eliminados del banco de datos transcurrido un período de tiempo configurable por el usuario. Así como admitir la gestión reportes correspondientes a toda la actividad realizada por la aplicación.

2.3 Modelo de negocio

Descripción del modelo de negocio, se hace uso de IDEF0 para capturar y modelar las actividades, entradas, controles, mecanismos y salidas del Módulo de carga de datos offline del SIAI. Teniéndose como proceso principal del negocio: La carga de datos offline relacionados con comportamientos en el uso de los servicios de telefonía fija y móvil en Cuba.



NODE: A-0	TITLE: Cargar datos offline relacionados con comportamientos en el uso de los servicios de telefonía fija y móvil en Cuba.	NO.: 1
-----------	--	--------

Fig. 2: Diagrama del proceso general A0 Cargar datos offline relacionados con comportamientos en el uso de los servicios de telefonía fija y móvil en Cuba.

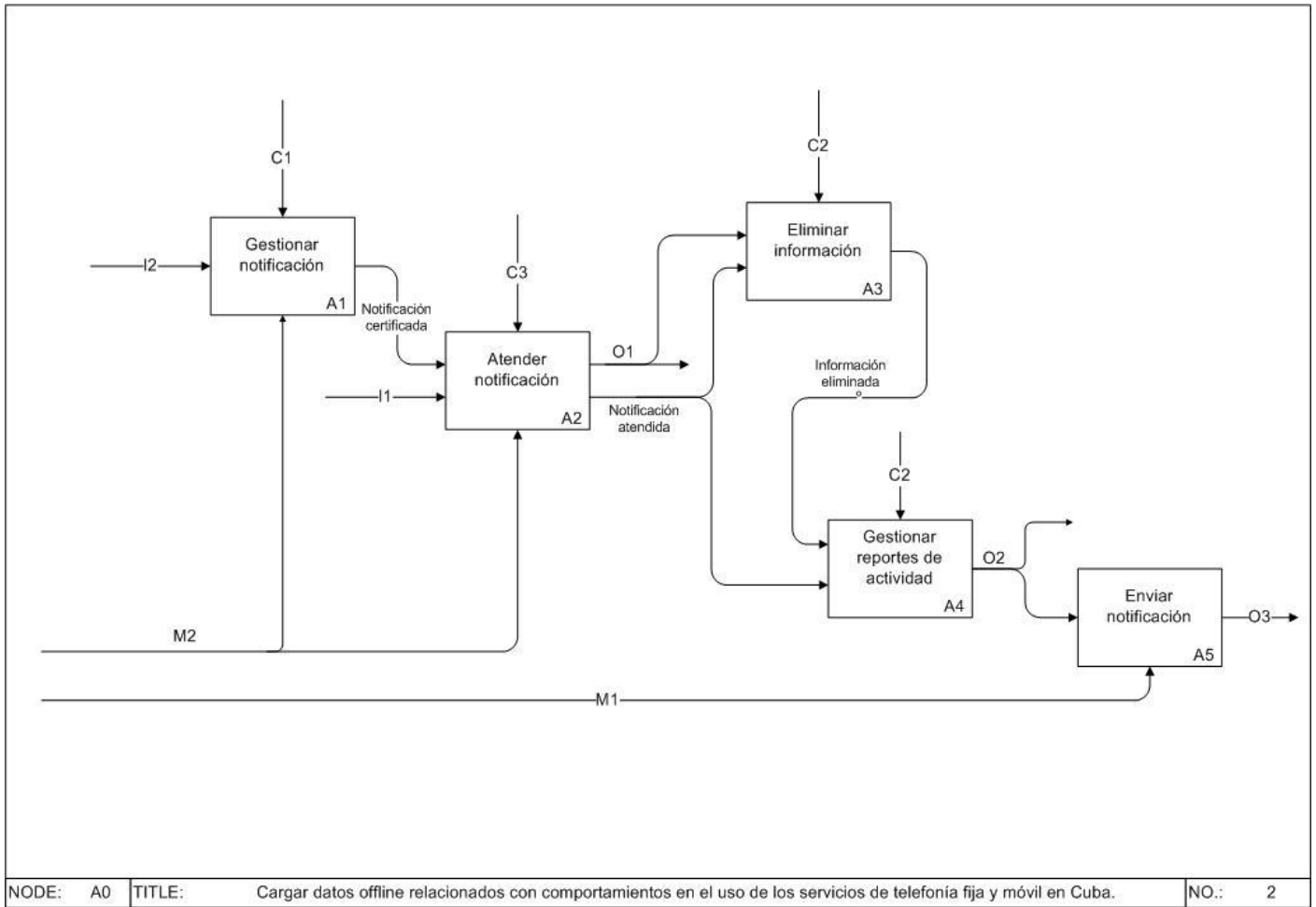


Fig. 3: Diagrama detallado del proceso general A0 Cargar datos offline relacionados con comportamientos en el uso de los servicios de telefonía fija y móvil en Cuba.

Tabla 1: Ficha del proceso A0.

Ficha de Proceso		
Proceso	Cargar datos offline.	
Notación:	A0	
Entradas:	I1	Información offline relacionados con comportamientos en el uso de los servicios de telefonía fija y móvil en Cuba.
	I2	Notificación de solicitud para carga de datos.
Controles:	C1	Reglas para la correcta redacción de las notificaciones.
	C2	Tiempo.
	C3	Modalidad del sistema (Personalizada/Automática).

Salidas:	O1	Información online.
	O2	Reportes de actividad del sistema.
	O3	Solicitud de notificación al usuario solicitante.
Mecanismos:	M1	Módulo de comunicaciones para el SIAI.
	M2	Operador.

2.4.1 Descripción de los procesos del negocio

A0: Es el proceso principal que hace referencia a las actividades que se llevan a cabo para la correcta carga de datos offline relacionados con comportamientos en el uso de los servicios de telefonía fija y móvil en Cuba, tomando como punto de partida las informaciones proporcionadas por las entradas (información offline, notificación). Estas informaciones de entrada son procesadas por los mecanismos existentes como son el módulo de comunicaciones del SIAI y el operador. Todas las actividades estarán reguladas por el cumplimiento de las restricciones que imponen los controles que determinan cómo, cuándo y si una actividad se ejecuta o no, con el objetivo de ajustar y establecer pautas para el desempeño de cada actividad. Producto del resultado de la interacción de los factores mencionados se obtienen como salidas información online, reportes de actividad del sistema y solicitudes de notificación.

El proceso principal además incluye en un nivel más interno, un conjunto de procesos que a continuación se detallan:

A1: Es el encargado de gestionar las NSCD a partir de las notificaciones entrantes. Puede ser realizado por el operador. Es controlado por las reglas para la redacción de notificaciones. Como resultados se pueden obtener las notificaciones certificadas por el sistema y el operador. En el mismo se pueden realizar operaciones como, listar y eliminar notificación, además de mostrar un resumen de datos asociados a las notificaciones.

A2: Es el encargado de atender las NSCD a partir de las notificaciones certificadas y la información online. El proceso es realizado por el operador. Es controlado por modalidad de atención a notificaciones del sistema, lo cual permite seleccionar la forma de atender a dichas notificaciones, siendo de forma automática o personalizada proporcionando entre otras cosas mecanismos avanzados para la selección

de la información solicitada por las notificaciones. Como resultados se pueden obtener las notificaciones atendidas e información online.

A3: Es el encargado de eliminar la información que se encuentra almacenada en el banco de datos según si período de caducidad, el mismo se desarrolla a partir de la información online y de las notificaciones atendidas, el proceso es controlado por el tiempo. Produciendo como salida una cierta cantidad de información eliminada del banco de datos.

A4: Es el encargado de gestionar los reportes de actividad del sistema a partir de las notificaciones atendidas y la información eliminada del banco de datos. Es controlado por el tiempo y como resultados se pueden obtener los reportes de actividad del sistema. Permitiendo que dichos reportes sean listados facilitando la selección y visualización de contenido.

A5: Es el encargado de enviar la notificación al usuario originador de la NSCD, a partir de los reportes de actividad del sistema, originando una NCDR. Utilizando el servidor de notificaciones del SIAI (SIAINTFS).

2.5 Requisitos de software

Los requisitos funcionales y no funcionales son de gran importancia en el desarrollo de software producto a que ambos deben ser tenidos en cuenta para obtener un producto acabado y con la calidad requerida; donde los primeros están enfocados en garantizar las funciones que una aplicación será capaz de realizar, como el comportamiento interno de la misma, y los segundos se encargarán de especificar los criterios que son empleados en la calificación o evaluación del funcionamiento de determinado sistema.

2.6 Requisitos funcionales

RF1: Autenticar usuario.

RF2: Verificar conexión con la base de datos.

RF3: Configurar conexión con la base de datos

RF4: Enviar notificaciones al cargador de datos

RF5: Listar notificaciones

RF6: Administrar notificación

RF6.1: Seleccionar notificación

RF6.2: Eliminar notificación

RF6.3: Actualizar notificaciones a demanda

RF7: Atender notificaciones de forma automática

RF8: Atender notificaciones de forma personalizada

RF9: Listar usuario originador de la notificación.

RF10: Actualizar nivel de completamiento de las notificaciones.

RF11: Listar archivos solicitados por una notificación.

RF12: Listar ficheros de un directorio.

RF13: Listar fichero de unidad DVD.

RF14: Mostrar propiedades de fichero.

RF15: Insertar contenido de los ficheros en el banco de datos.

RF16: Listar ficheros a insertar en el banco de datos.

RF17: Eliminar ficheros.

RF18: Mostrar resumen de datos a insertar en el banco de datos.

RF19: Administrar reporte.

RF19.1: Listar reportes.

RF19.2: Mostrar datos de reporte.

RF19.3: Eliminar reporte.

RF20: Enviar notificación de carga de datos realizada.

RF21: Mostrar ayuda.

RF22: Eliminar información de la base de datos.

2.7 Requisitos no funcionales

Los requisitos no funcionales que se exponen a continuación fueron definidos siguiendo los requerimientos planteados para el buen funcionamiento del SIAI y se adaptaron en algunos casos a las

peculiaridades del Módulo de carga de datos offline del SIAI, teniéndose en cuenta las expectativas de los clientes con el fin de establecer un equilibrio entre estos sistemas.

2.7.1 Requisitos de Seguridad.

El módulo de Carga de Datos Offline del SIAI se ajusta a los requisitos de seguridad del SIAI .Ver planilla de Requisitos de software Módulo de plataforma web del SIAI, sección Requisitos no funcionales de seguridad.

2.7.2 Requisito de Usabilidad.

La aplicación puede ser utilizada por cualquier persona que posea conocimientos elementales en el trabajo con sistemas informáticos, debido a la facilidad de su interfaz. Motivo por el cual se establecen algunos requisitos adicionales con el propósito de optimizar el trabajo con el sistema por parte de los usuarios como son:

- **Requisitos de Idioma**

El objetivo primordial es garantizar la facilidad de uso por personas que hablen el idioma español.

- **RNF Habilidades mínimas de usuarios**

Garantizar que el sistema pueda ser utilizado por personas con un mínimo de conocimientos sobre el manejo del ordenador.

- **RNF Entrenamiento a usuarios**

Impartir curso sobre el manejo de la herramienta de carga de datos es una opción para mejorar las habilidades de los usuarios finales en el uso de esta aplicación.

2.7.3 Requisito de Fiabilidad.

La aplicación como sistema desktop, el tiempo que determine el operador del mismo, no presentando razones para interrumpir su funcionamiento.

En caso de errores o fallos propio de este módulo podrá restaurarse en un tiempo máximo de 1 hora.

2.7.4 Requisito de Eficiencia.

La eficiencia de la aplicación va a depender del volumen de información que esta deba procesar, así como de la velocidad de conexión del entorno en el que se desarrolla y la cantidad de personal que se encontrarán haciendo peticiones al sistema.

- **RNF Capacidad.**

El número de clientes concurrentes haciendo peticiones al sistema como máximo debe ser 25 personas con el objetivo de mantener el rendimiento equilibrado.

2.7.5 Requisito de portabilidad.

El código basado en Python es portable a cualquier plataforma de sistema operativo, pero el sistema solo podrá ser operado sobre sistemas operativos GNU/LINUX pues muchas de las librerías utilizadas por el mismo solo funcionan sobre sistemas operativos GNU/LINUX.

2.7.6 Requisitos de Estándares de codificación.

Los estándares de programación a seguir se rigen por las pautas y especificaciones establecidas por los desarrolladores del SIAI, para el lenguaje de programación Python.

2.7.7 Requisito de Diseño.

- RNF Interfaces de usuario

Las interfaces de usuarios deben ser simples de utilizar, contribuyendo a un estilo profesional.

Interfaz de usuario amigable y fácil de entender.

El usuario puede acceder a las diferentes opciones sin tener que acceder a muchos puntos intermedios.

- RNF Interfaces hardware

Estaciones cliente:

- ✓ Conexión 100 mbps.
- ✓ 1 GB de memoria RAM.
- ✓ Procesador 2.0 MHz.
- ✓ Espacio en disco disponible de 10 GB.

Servidores:

- ✓ Conexión 100 mbps.

- ✓ Procesador Intel 2.0 GHz (x4).
- ✓ RAM de 16 GB.
- ✓ Espacio en disco disponible (*Por determinar con el cliente*).

- RNF Interfaces software

El sistema operativo a utilizar debe ser cualquier distribución de GNU/LINUX. Se debe contar con la Máquina virtual de Java 1.6 instalada.

Estaciones cliente:

- ✓ Máquina virtual (OpenJDK Development Kit).
- ✓ Sistemas operativos GNU/Linux.
- ✓ JayDeBeApi-0.1.3.
- ✓ Ojdbc6
- ✓ Suds.
- ✓ M2Crypto

Servidores

- ✓ Oracle 11gR2.

2.8 Definición de los Casos de Uso

Una vez identificados los requisitos funcionales los mismos son agrupados en casos de usos funcionalidades que el sistema deberá cumplir. Los Casos de Uso Autenticar usuario, Atender Notificación, Atender Notificación de forma automática ,Verificar conexión con la base de datos, Administrar Reporte, Configurar conexión con la base de datos y Mostrar ayuda pueden ser encontrados en el [Anexo 1](#).

2.8.1 Definición de los actores del sistema

Tabla 2: Actor del sistema.

Actor	Descripción
Operador	Es el encargado del seguimiento y control de las notificaciones de solicitud de carga de datos sobre el comportamiento anómalo en las comunicaciones.

2.8.2 Diagrama de caso de uso del sistema

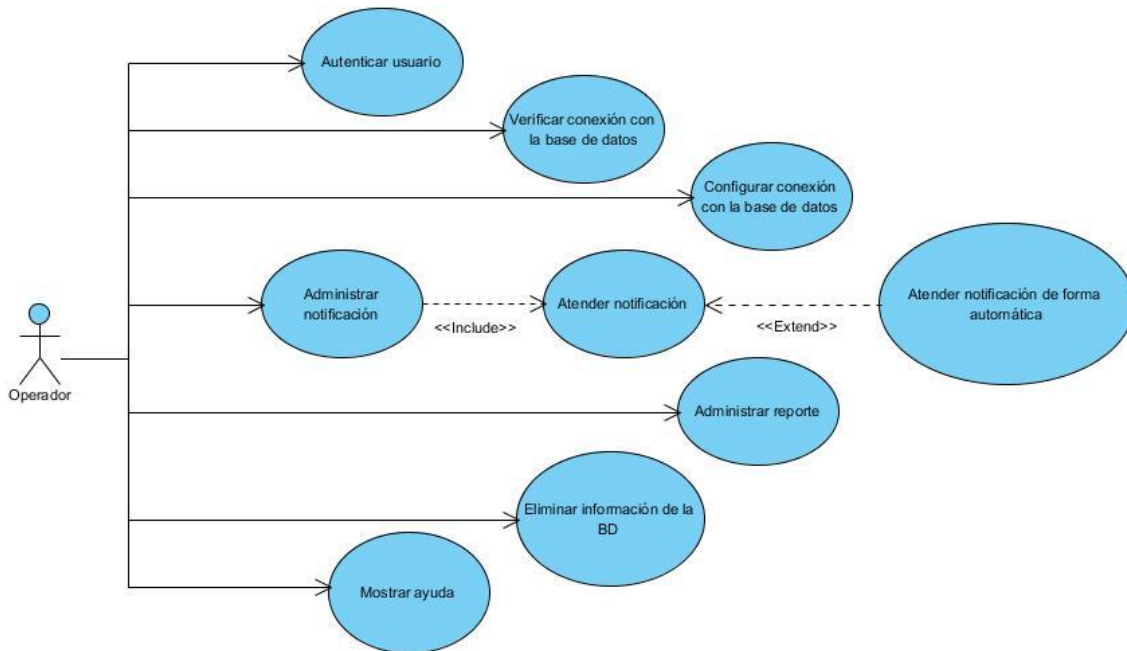


Fig. 4: Diagrama de Casos de uso del sistema.

2.8.3 Descripción textual de caso de usos

A continuación se presenta la descripción de uno de los casos de uso críticos en la realización del sistema, además dicho caso de uso será tomado como referencia en el resto del documento en los diagramas que sea necesario.

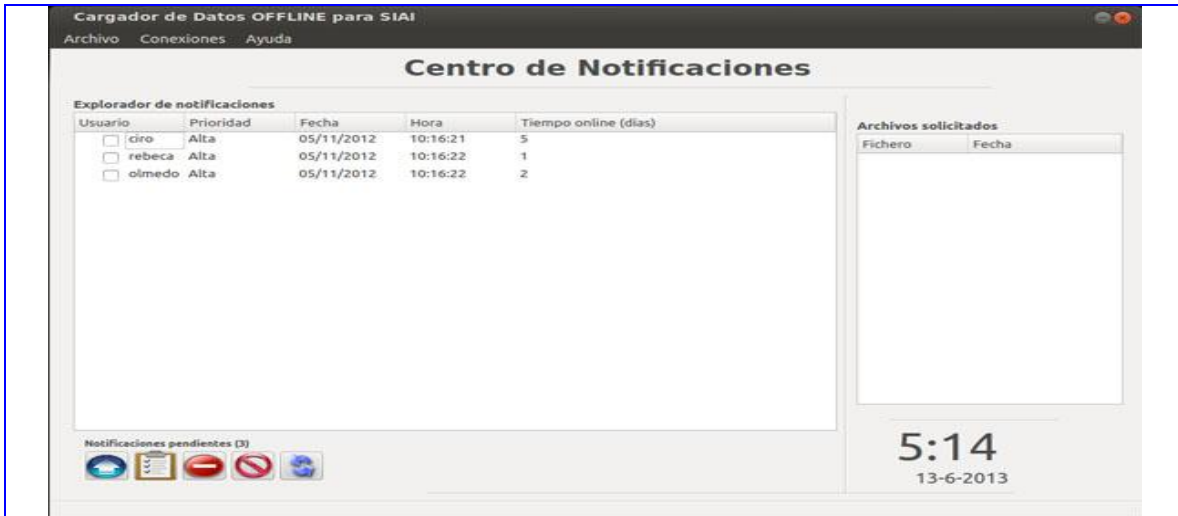
CU Administrar notificación

Descripción de Casos de Uso

Tabla 3: Descripción del Caso de uso Administrar notificación

Caso de Uso:	Administrar notificación
Actores:	Operador (Inicia)
Resumen:	El caso de uso se inicia cuando el operador se autentica en el sistema.
Precondiciones:	1. El usuario debe estar autenticado por el sistema

Referencias	RF4, RF5, RF6, RF6.1, RF6.2, RF6.3
Prioridad	Crítico
Flujo Normal de Eventos	
Sección “Administrar notificación”	
Acción del Actor	Respuesta del Sistema
	<p>1. El sistema muestra la interfaz “Centro de notificaciones”, listando las NSCD mostrando de ellas:</p> <ul style="list-style-type: none"> • Usuario • Prioridad • Fecha • Hora • Tiempo online en el BD <p>Y se brindan las opciones:</p> <ul style="list-style-type: none"> • Enviar notificación notificaciones al cargador de datos • Seleccionar todas las notificaciones • Eliminar notificación • Actualizar notificación a demanda • Listar archivos solicitados por una notificación • Centro de reportes (Se invoca al CU Administrar reporte)
Prototipo de Interfaz	



Flujo Alternos

Acción del Actor	Respuesta del Sistema

Prototipo de Interfaz

Flujo Normal de Eventos

Sección "Enviar notificaciones al cargador de datos"

Acción del Actor	Respuesta del Sistema
1. Selecciona alguna de las notificaciones listadas y selecciona la opción "Enviar notificaciones al cargador de datos".	2. Verifica que existan notificaciones seleccionadas.
	3. Envía las notificaciones seleccionadas al cargador de datos.

Flujo Alternos

Acción del Actor	Respuesta del Sistema

Prototipo de Interfaz

Flujo Normal de Eventos

Sección “Seleccionar todas las notificaciones”

Acción del Actor	Respuesta del Sistema
1. Selecciona la opción “Seleccionar todas las notificaciones”.	2. Selecciona todas las notificaciones.
	3. Brinda la opción deseleccionar todas las notificaciones

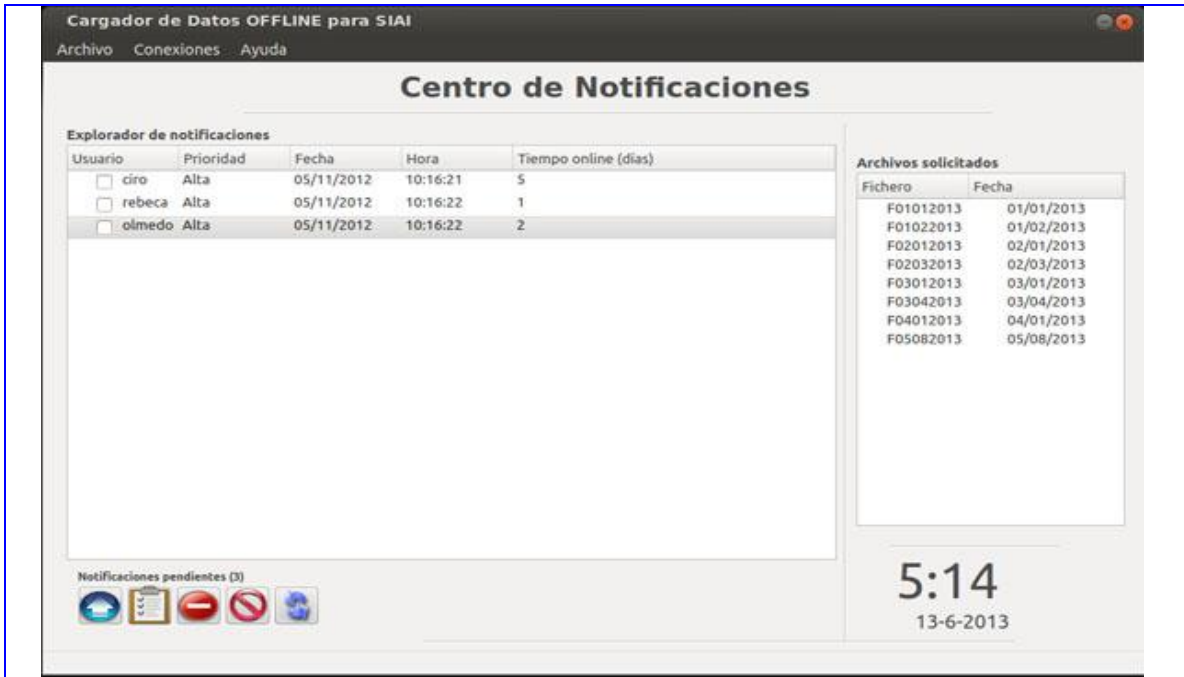
Prototipo de Interfaz



Flujo Alternos

Acción del Actor	Respuesta del Sistema
4. Selecciona la opción “deseleccionar todas las notificaciones”.	5. Se desmarcan todas las notificaciones

Prototipo de Interfaz



Flujo Normal de Eventos

Sección "Eliminar notificación"

Acción del Actor	Respuesta del Sistema
1. Selecciona alguna de las notificaciones listadas y selecciona la opción "Eliminar notificación".	2. Verifica que existan notificaciones seleccionadas.
	3. Elimina las notificaciones seleccionadas.

Flujo Alternos

Acción del Actor	Respuesta del Sistema

Flujo Normal de Eventos

Sección "Actualiza notificación a demanda"

Acción del Actor	Respuesta del Sistema
1. Selecciona la opción “Actualiza notificación a demanda “.	2. Verifica la conexión con el Servidor de comunicaciones del SIAI.
	3. Se conecta al Servidor de comunicaciones de comunicaciones del SIAI y descarga las NSCD que se encuentren en la cola.
	4. Se actualiza la interfaz “Centro de notificaciones” con las nuevas notificaciones descargadas.

Prototipo de Interfaz



Flujo Alternos

Acción del Actor	Respuesta del Sistema
	1.1 No se puede establecer la conexión con el servidor de comunicaciones del SIAI por lo que se muestra el mensaje

“Imposible establecer comunicación con el servidor de comunicaciones del SIAI”.

Prototipo de Interfaz



Sección “Listar archivos solicitados por una notificación”

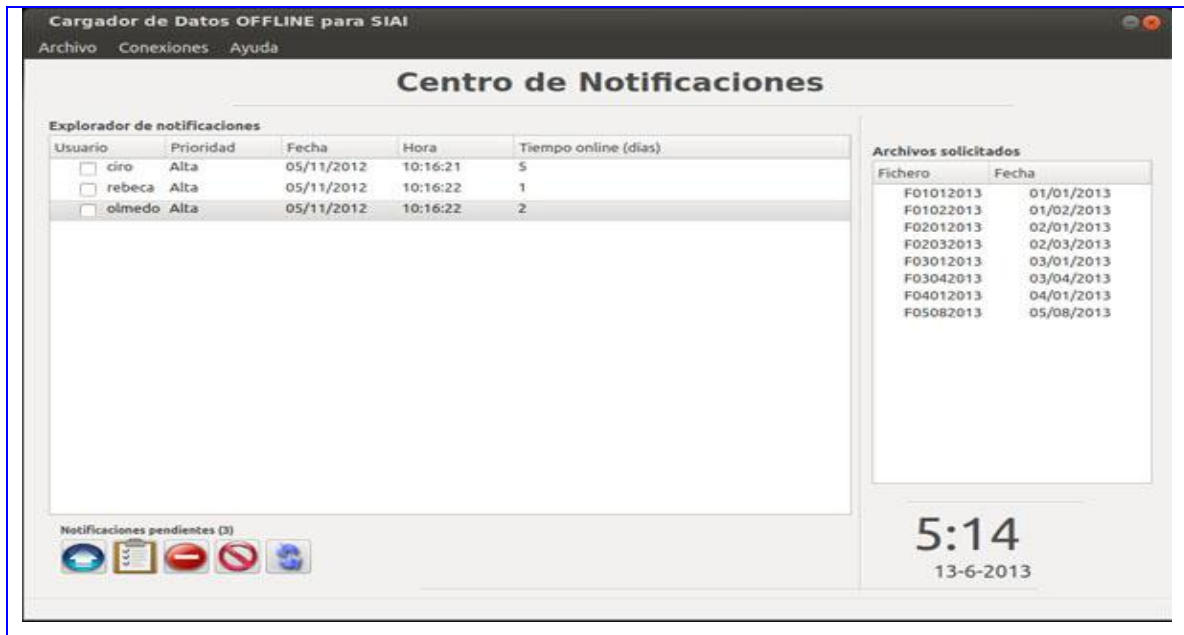
Acción del Actor	Respuesta del Sistema
1. Selecciona una notificación	2. Lista los archivos solicitados por una notificación. Mostrando de los mismos: <ul style="list-style-type: none"> • Fecha • Nombre del Fichero

Prototipo de Interfaz

Flujo Alternos

Acción del Actor	Respuesta del Sistema

Prototipo de Interfaz



2.9 Conclusiones del capítulo

En el presente capítulo se realizó un estudio de los procesos del negocio a automatizar por la herramienta Módulo de Carga de Datos Offline del SIAI. Se identificaron los requisitos funcionales y no funcionales, además se diseñaron y describieron los casos de uso que guiarán el desarrollo de la solución propuesta mostrando los prototipos de interfaces.

CAPÍTULO 3: DISEÑO DEL SISTEMA

3.1 Introducción

En este capítulo se detallan los patrones de diseño, que serán tenidos en cuenta en la creación del diagrama de clases del diseño, así como una guía que tendrá como propósito estructurar y organizar la implementación del sistema. Se presentan además los modelos físicos y lógicos de la base de datos describiendo cada uno de los atributos presentes en ellos. El diseño debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades, de forma flexible y orientada a objeto.

3.2 Propuesta de arquitectura del sistema

Diseñar la arquitectura del sistema permite definir estrategias y soluciones que abarquen los requisitos solicitados por el usuario, garantizando así el cumplimiento de las funcionalidades a poseer por el mismo. Además define los componentes que van a integrar el sistema, estableciéndose la relación que existirá entre ellos y finalmente proporcionando el marco de trabajo idóneo que servirá de plataforma para la construcción de la aplicación informática.

Patrón arquitectónico

Específica la estructura para un sistema de software, los cuales a su vez se componen de subsistemas con responsabilidades. También tienen una serie de directivas para organizar los componentes del mismo sistema, con el objetivo de facilitar las tareas del diseño.

Se propone el desarrollo del sistema sobre una arquitectura basada en un estilo arquitectónico en capas como se muestra en la Fig.12. Para ello se propone la utilización de 3 capas, entre las que se encuentran:

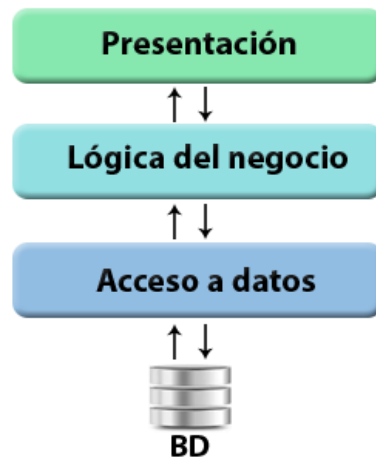


Fig. 5: Arquitectura del sistema.

- La capa de presentación es la encargada de gestionar las vistas de la aplicación, permitiendo mostrar y capturar información al usuario.
- La capa encargada de la lógica del negocio contiene las reglas del negocio establecidas para dar cumplimiento a las funcionalidades que debe realizar el sistema.
- La capa de acceso a datos va a contar con las entidades del dominio necesarias para la comunicación entre la lógica del negocio, la base de datos y el servidor de comunicaciones del SIAI (SIAINTFS).

3.2.1 Patrones de diseño aplicados

Los patrones de diseño son el esqueleto de las soluciones a problemas comunes en el desarrollo de software. En otras palabras, brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares. (18) A continuación se muestran los distintos patrones aplicados en el diseño del sistema:

3.2.3.1 Experto

Problema: ¿Cuál es el principio fundamental en virtud por el cual se asignan las responsabilidades a los objetos?

Solución: Asignar una responsabilidad a la clase que tiene la información necesaria para cumplirla. (19)

Problema a solucionar:

Se utiliza en todo el desarrollo de la aplicación, delegando las responsabilidades a los componentes que poseen la información necesaria para realizar determinadas acciones, conservando el encapsulamiento. Por ejemplo la clase “FileParser” es la encargada de parsear (acción de leer un fichero, realizando un análisis sobre el mismo para la extracción de información) los ficheros de textos asociados a la aplicación.

3.2.3.2 Creador

Problema: ¿Quién debería ser responsable de crear una nueva instancia de alguna clase?

Solución: Asignarle a la clase B la responsabilidad de crear una instancia de la clase A en uno de los siguientes casos:

- B agrega los objetos A.
- B contiene los objetos A.
- B registra las instancias de los objetos A o
- B utiliza especialmente los objetos A.
- B tiene los datos de inicialización que serán transmitidos a A cuando este objeto sea creado (así que B es un Experto respecto a la creación de A). B es un creador de los objetos A. (19)

Problema a solucionar:

Es necesario conocer cuál dentro de la estructura de clases de la aplicación es la idónea para instanciar los datos que brinden la solución esperada. Se utiliza a lo largo de todo el proceso de desarrollo del sistema al instanciar otras clases para su utilización. Por ejemplo en la clase “DataLoaderControler” se crea un objeto de la clase “Connection” para obtener acceso a las funcionalidades que esta brinda sobre la base de datos.

3.2.3.3 Bajo Acoplamiento

Problema: ¿Cómo dar soporte a una dependencia escasa y a un aumento de la reutilización?

El acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases, con que las conoce y con que recurre a ellas.

Solución: Asignar una responsabilidad para mantener bajo acoplamiento. (19)

Problema a solucionar:

Se utiliza en gran medida para evitar las dependencias excesivas o alto acoplamiento entre las clases, llegando a constituir un problema. El sistema deberá mantener una estructura reutilizable permitiendo la inserción de nuevas funcionalidades o la modificación de algunas ya existentes. Se evidencia su utilización en el sistema en las clases “Connection” y “FileParser”, usualmente muy utilizadas en conjunto pero con sus responsabilidades bien definidas para mantener un bajo margen de dependencias.

3.2.3.4 Alta Cohesión

Problema: *¿Cómo mantener la complejidad dentro de límites manejables?*

La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase.

Solución: *Asignar una responsabilidad de modo que la cohesión siga siendo alta. (19)*

Problema a solucionar:

Se utiliza en gran medida para evitar el exceso de responsabilidades sobre una misma clase evitando comportamientos innecesarios, otorgando de esta forma una gran claridad y facilidad para comprender el funcionamiento de la aplicación, permitiendo un alto grado de reutilización de los objetos. Este patrón es aplicado en el sistema específicamente sobre las clases “WS_Server” y “Connection” encargadas de controlar la mayoría de las acciones de acceso a datos.

3.2.3.5 Controlador

Problema: *¿Quién debería encargarse de atender un evento del sistema?*

Un evento del sistema es un evento de alto nivel generado por un actor externo; es un evento de entrada externa. Se asocia a operaciones del sistema: las que emite en respuesta a los eventos del sistema.

Solución: *Asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase que represente una de las siguientes opciones:*

- *El “sistema” global (controlador de fachada).*
- *La empresa u organización global (controlador de fachada).*
- *Algo en el mundo real que es activo (por ejemplo, el papel de una persona) y que pueda participar en la tarea (controlador de tareas).*
- *Un manejador artificial de todos los eventos del sistema de un caso de uso, generalmente denominados “Manejador<Nombre Caso de Uso>” (controlador de casos de uso). (19)*

Problema a solucionar:

Se utiliza ampliamente para asignar qué clases deben tener la responsabilidad de controlar el funcionamiento del sistema en general. Ampliamente utilizado en la asignación de responsabilidades en las clases “NotifyCenterControler”, “DataLoaderControler” y demás clases controladoras de la aplicación.

3.2.3.6 Fachada/Facade

Tipo: Estructura, a nivel de objetos.

Propósito: Proporcionar una interfaz unificada de alto nivel que, representando a todo un subsistema, facilite su uso. La “fachada” satisface a la mayoría de los clientes, sin ocultarlas funciones de menor nivel a aquellos que necesiten acceder a ellas. (20)

Problema a solucionar:

Una solución factible aportada por este patrón se encuentra al realizar una abstracción entre las clases y librerías utilizadas para efectuar las operaciones sobre la base de datos y la aplicación, para ello se realiza una clase conexión, la cual es la encargada de agrupar las operaciones hacia la base de datos.

3.3. Diagrama de Paquetes

El diagrama de paquetes muestra cómo está dividido un sistema en agrupaciones lógicas, representando las dependencias entre esas agrupaciones. Normalmente un paquete está pensado como un directorio, los diagramas de paquetes proporcionan una descomposición de un sistema. A continuación se muestra el diagrama de paquetes del Módulo de Carga de Datos Offline del SIAI.

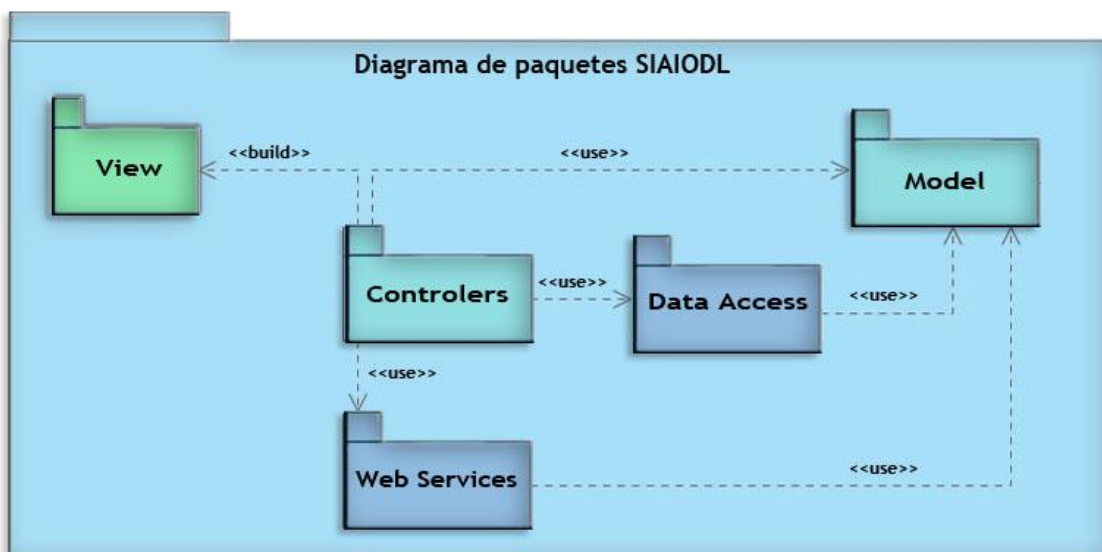


Fig. 6: Diagrama del Módulo de Carga de Datos del SIAI.

Descripción de los paquetes:

- **View:** Paquete que contiene todas las interfaces del sistema y los recursos necesario para el correcto funcionamiento de las mismas, tales como los iconos.
- **Model:** En el mismo se almacenan las clases entidades encargadas de contener el formato para almacenar objetos en la aplicación.
- **Controlers:** Contiene las clases controladoras y la estructura para el almacenamiento de ficheros necesarios para el funcionamiento de la aplicación.
- **DataAccess:** Paquete en el cual se almacenan las clases encargadas del acceso a la base de datos y al parseo de ficheros. Además de poseer los ficheros de configuración para el acceso a la base de datos.
- **Web Services:** En el mismo se almacenan los componentes necesarios para establecer la comunicación con el Servidor de Comunicaciones del SIAI (SIAINTFS). Además se encuentran los ficheros de configuración para el cliente del SIAINTFS.

3.4. Diagrama de clases del Diseño

El diagrama de clases del diseño se realiza para describir las especificaciones de las clases y las interfaces de una aplicación. A continuación se muestra el diagrama de clases del diseño del caso de uso Administrar notificación. Los restantes diagramas correspondientes a los casos de uso: Autenticar usuario, Atender notificación, Atender notificación de forma automática, Eliminar información de la base de datos, Administrar reporte, Configurar conexión con la base de datos y Verificar conexión con la base de datos se encuentran en el [anexo 2](#).

3.4.1 Diagrama de clases del caso de uso Administrar notificación

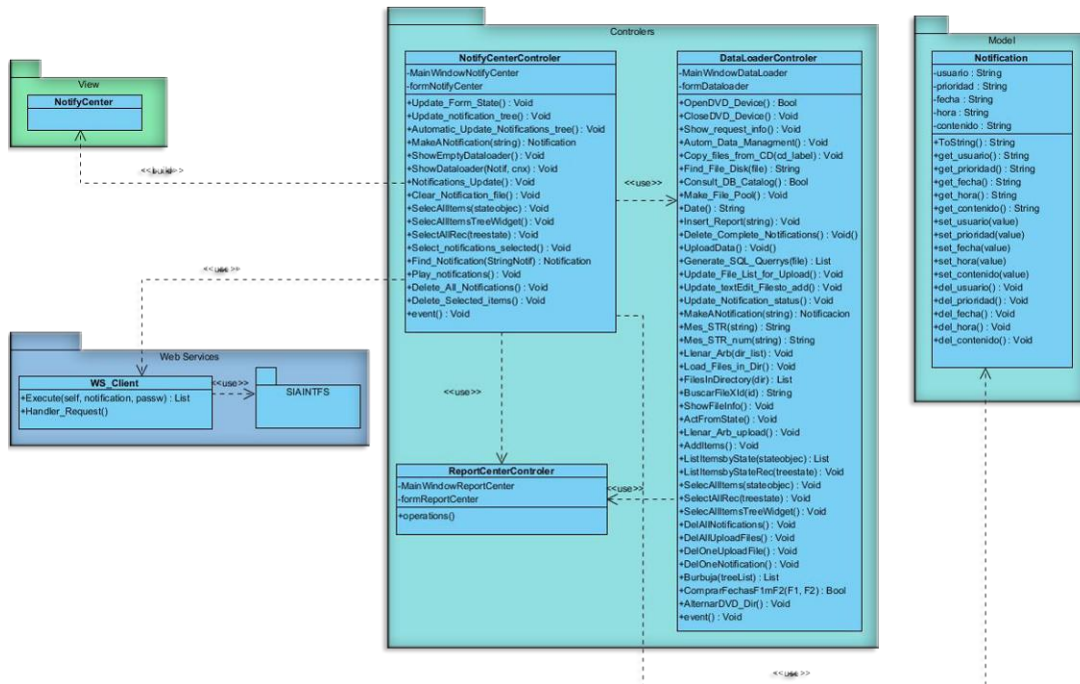


Fig. 7: Diagrama de clases del diseño del paquete View

3.5. Modelo de Datos

A continuación se presenta y describe el Modelo de Datos, en el mismo se describen las representaciones físicas de los datos persistentes utilizados en la herramienta y que serán almacenados en la base de datos.



Fig. 8: Modelos Físico de Datos

El SIAIODL ejecutará sus acciones sobre una estructura paralela al BD, totalmente similar a la original con la peculiaridad que cada tabla poseerá un último campo denominado “ODLREQUESTUSER”, mecanismo implementado para controlar el acceso de los usuarios a la información almacenada.

Tabla 4: Descripción de la tabla ODLCATALOG

ENTIDAD	Descripción
ODLCATALOG	Entidad en la cual se almacena la información relacionada a las salvas de información offline realizadas en dispositivos de almacenamiento extraíbles (DVD).
CAMPOS	
ID	Identificador de la tupla en la tabla.
start_date	Fecha de inicio del período de la salva.
end_date	Fecha de fin del período de la salva.
CD_number	Número del DVD en que almacenó la información.

Tabla 5: Descripción de la tabla ODLUSERS

Entidad	Descripción
ODLUSERS	Entidad en la cual se almacena la información relacionada a los usuarios calificados para la utilización del SIAIODL.
CAMPOS	
ID	Identificador de la tupla en la tabla.
Name	Nombre del usuario.
LastName	Apellidos del usuario.
Passw	Contraseña del usuario para acceder al SIAIODL.

3.6 Diagrama de despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. Los nodos se interconectan mediante soportes bidireccionales que pueden a su vez estereotiparse. Esta vista permite determinar las consecuencias de la distribución y la asignación de recursos. Las instancias de los nodos pueden

contener instancias de ejecución, como instancias de componentes y objetos. El modelo puede mostrar dependencias entre las instancias y sus interfaces, y también modelar la migración de entidades entre nodos u otros contenedores. (21). A continuación se muestra el diagrama de despliegue correspondiente al Módulo de Carga de Datos Offline del SIAI (SIAIODL).

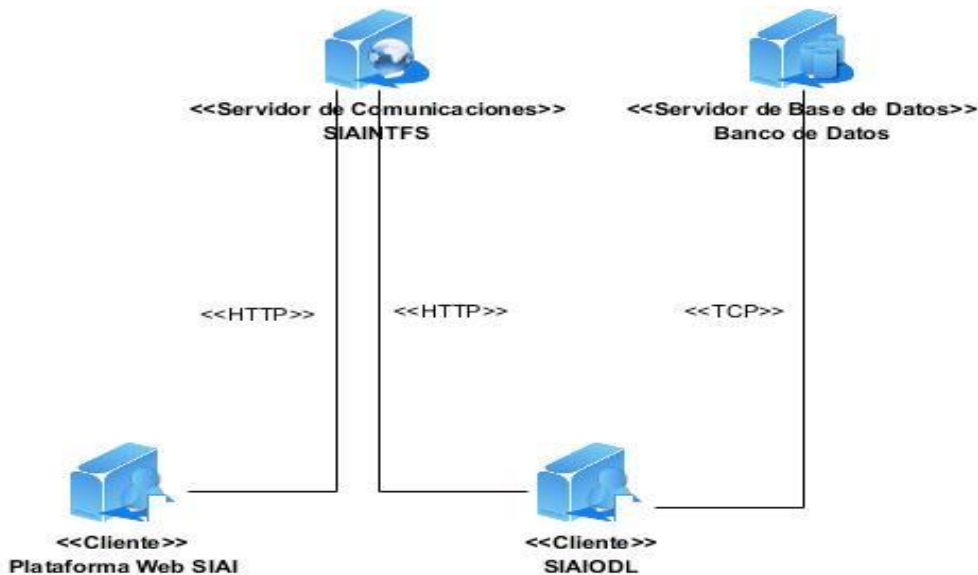






Fig. 9: Diagrama de despliegue del SIAIODL

3.6.1 Descripción de los nodos del Diagrama de despliegue

A continuación se describen los nodos que componen el diagrama de despliegue:

Tabla 6: Descripción del diagrama de despliegue.

Nodo	Requerimientos
 <p data-bbox="261 1692 440 1745"><<Cliente>> Plataforma Web SIAI</p>	<p data-bbox="561 1478 1511 1549">Cliente establecido en la plataforma Web del SIAI desde el cual se enviarán las solicitudes de carga de datos hacia el Servidor de Notificaciones del SIAI.</p> <p data-bbox="561 1568 987 1598">Requisitos mínimos de hardware:</p> <ul data-bbox="573 1614 967 1793" style="list-style-type: none"> ✓ Conexión 100 mbps. ✓ 1GB de memoria RAM. ✓ Procesador 2.0 MHz. ✓ 10 GB de almacenamiento

 <p align="center"><<Cliente>> SIAIODL</p>	<p>Cliente al cual serán enviadas y procesadas las solicitudes de carga de datos procedentes del Servidor de Comunicaciones del SIAI.</p> <p>Requisitos mínimos de hardware:</p> <ul style="list-style-type: none"> ✓ Conexión 100 mbps. ✓ 1 GB de memoria RAM. ✓ Procesador Intel 2.0 MHz. ✓ 10GB de almacenamiento (Requeridos por el sistema)
 <p align="center"><<Servidor de Comunicaciones>> SIAINTFS</p>	<p>Servidor de Comunicaciones que estará encargado de gestionar todas las comunicaciones en el SIAI.</p> <p>Requisitos mínimos de hardware:</p> <ul style="list-style-type: none"> ✓ Conexión 100 mbps. ✓ 2 GB de memoria RAM. ✓ Procesador Intel Core I3 3.10GHz. ✓ 80 GB de almacenamiento
 <p align="center"><<Servidor de Base de Datos>> Banco de Datos</p>	<p>Servidor de Base de Datos donde estará contenido el Banco de Datos y la Base de datos del SIAIODL</p> <p>Requisitos mínimos de hardware:</p> <ul style="list-style-type: none"> ✓ Conexión 100 mbps. ✓ 16 GB de memoria RAM. ✓ Procesador 2.0 GHz (x4).

3.7 Diagramas de componentes

“Un diagrama de componentes muestra las dependencias lógicas entre componentes de software, sean éstos componentes fuentes, binarios o ejecutables, ilustran las piezas del software, controladores embebidos, etc. Los diagramas de Componentes prevalecen en el campo de la arquitectura de software pero pueden ser usados para modelar y documentar cualquier arquitectura de sistema.” (22) A continuación se realiza una breve descripción del diagrama de componentes general del Módulo de Carga de Datos Offline del SIAI. Los restantes diagramas de componentes correspondientes a los casos de uso: Autenticar usuario, Atender notificación, Atender notificación de forma automática, Eliminar información de

la base de datos, Administrar reporte, Configurar conexión con la base de datos y Verificar conexión con la base de datos se encuentran en el [anexo 3](#).

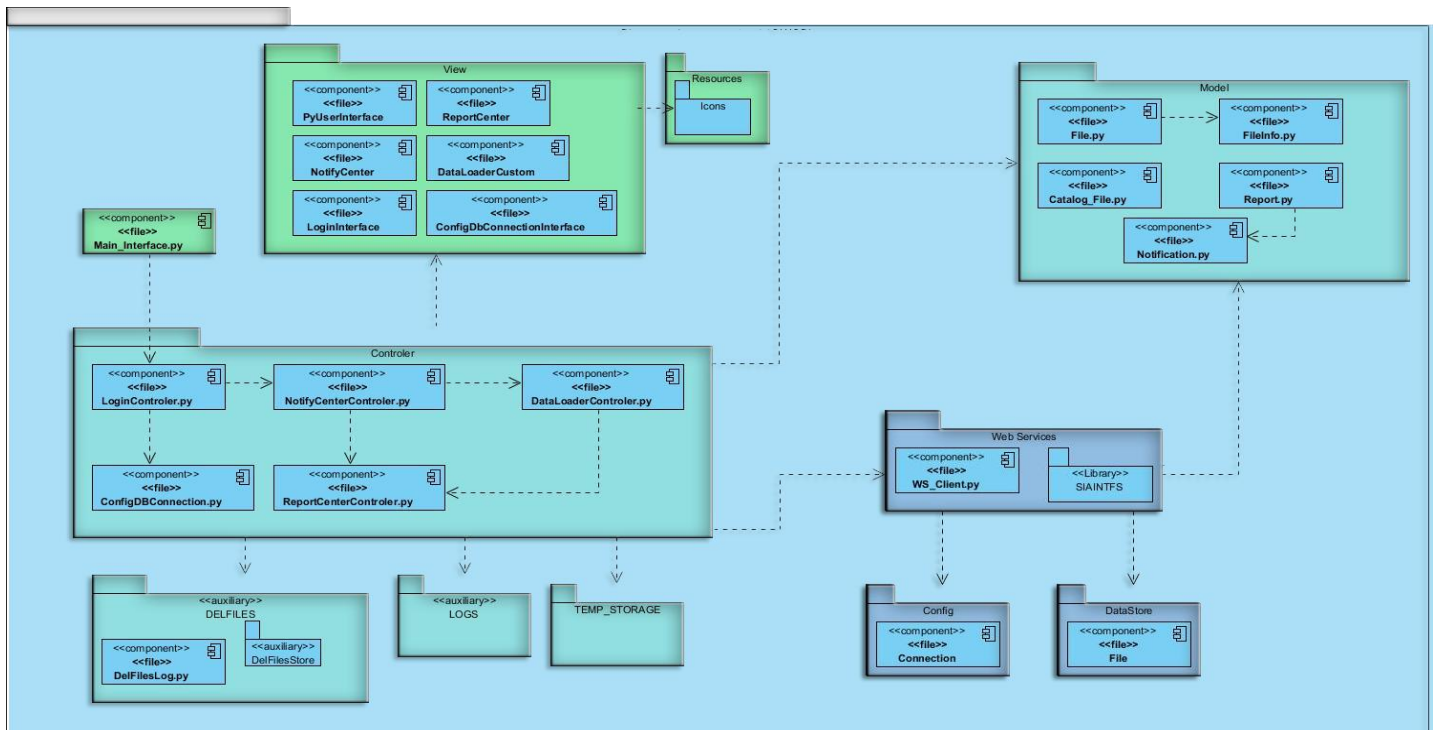


Fig. 10: Diagrama de componentes del caso de uso Administrar notificación

3.8 Conclusiones del capítulo

En este capítulo quedó reflejada tanto la arquitectura como los patrones utilizados en la implementación del Módulo de carga de Datos Offline del SIAI, otorgándole mayor robustez a la aplicación. Todo con el fin de darle cumplimiento a las funcionalidades definidas. Fueron expuestos los diagramas de clases del diseño para comprender como van a relacionarse los subsistemas, se presentó un modelo de datos para ver la estructura de la información que va a estar almacenada en la base de datos. Por último se presentaron los diagramas de Despliegue y de Componentes obteniendo con estos una vista más detallada del funcionamiento del sistema.

CAPÍTULO 4: PRUEBAS

4.1 Introducción

Como objetivos principales de la realización de pruebas a un software se tienen obtener información sobre los errores, defectos o fallas que tiene el sistema, así se realizan las correcciones pertinentes, según el caso y se asegura la calidad del producto que se está entregando al cliente. Además de ser un indicador de la calidad que posee el producto informático.

4.2 Estrategia de pruebas.

La realización de las pruebas a un software es una tarea de vital importancia pues permite la detección de errores en diferentes áreas de la aplicación en dependencia del tipo de prueba que se esté efectuando, a continuación se propone la estrategia diseñada para realizar las pruebas al Módulo de Carga de Datos Offline del SIAI.

4.2.1 Niveles de Pruebas

Las pruebas se aplican en diferentes niveles de profundidad, a continuación se establecen y detallan los niveles propuestos para la realización de las pruebas al SIAIODL.

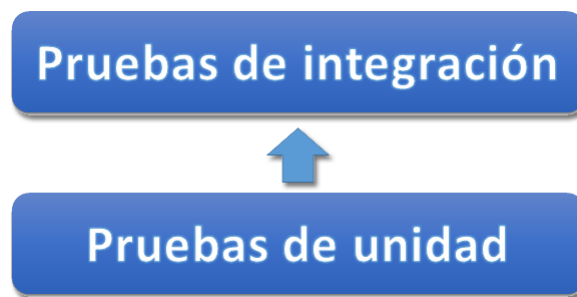


Fig 11: Niveles de pruebas a desarrollar

Pruebas de Unidad

Las pruebas de unidad o unitarias son una forma de probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado. Las

mismas se realizaran por los desarrolladores inmediatamente después de haber implementado la aplicación.

Es la mejor forma de detectar errores en tempranas etapas de desarrollo y ayudan a definir requerimientos y responsabilidades de cada método en cada clase. Aunque no garantizan detectar todos los errores, por tanto deben ejecutarse en conjunto con otras pruebas.

Pruebas de Integración

El proceso de integración del sistema implica construir este a partir de sus componentes y probar el sistema resultante para encontrar problemas que puedan surgir debido a la integración de los componentes. Los componentes que se integran pueden ser componentes comerciales, componentes reutilizables que han sido adaptados a un sistema en particular o componentes nuevos desarrollados.

Es vital realizar las pruebas de Integración pues el SIAIODL necesita integrarse con el SIAINTFS para llevar a cabo todo el proceso de comunicación con los nodos clientes a través de los servicios que brinda el mismo.

4.2.2 Métodos de Prueba

Existen diferentes métodos para la realización de las pruebas a los diferentes niveles propuestos, a continuación se detallan algunos de los principales:

Prueba de caja blanca: Comprueba los caminos lógicos del software proponiendo casos de prueba que se ejerciten conjuntos específicos de condiciones y/o bucles. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado o mencionado.

Pruebas de caja negra: “Se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene. Esta prueba examina algunos aspectos del modelo fundamentalmente del sistema sin tener mucho en cuenta la estructura interna del software.” (16)

4.2.3 Ejecución de pruebas

Una vez establecidas las bases para el desarrollo de las pruebas se propone el desarrollo de las pruebas.

4.2.3.1 Realización de las pruebas de unidad

Tabla 7: Estrategia para la realización de las pruebas de unidad.

Nivel de prueba:	Unidad
Tipo de Prueba:	Funcionalidad función
Método de prueba:	Caja Blanca

Para la ejecución de esta prueba se utilizó la técnica automática empleando la herramienta PyUnit la cual forma parte de la librería estándar de Python desde la versión 2.1. En su mayoría las pruebas fueron realizadas sobre clases críticas en el funcionamiento del sistema y que eran de vital importancia conocer el resultado devuelto por las diferentes funcionalidades de las mismas. A continuación se detallan los resultados obtenidos durante las 3 iteraciones realizadas.

Iteración #1

En la primera iteración de las pruebas de unidad se realizaron las pruebas a 5 funcionalidades de la clase "Connection" detectándose 2 errores, siendo solucionados por el equipo de desarrollo. Véase Fig 23

```
-----  
Ran 5 tests in 0.011s  
  
FAILED (failures=2)
```

Fig 12: Resultados de la Iteración #1 de las pruebas de unidad

Iteración #2

En la segunda iteración de las pruebas de unidad sobre 7 funcionalidades se detectó la aparición de un nuevo error, siendo solucionado por el equipo de desarrollo. Véase Fig 24

```
-----  
Ran 7 tests in 0.021s  
  
FAILED (failures=1)
```

Fig 13: Resultados de la Iteración #1 de las pruebas de unidad

Iteración #3

En la tercera iteración de las pruebas de unidad sobre 10 funcionalidades no se detectaron errores. Véase Fig 25

```
-----  
Ran 10 tests in 0.160s  
  
OK
```

Fig 14: Resultados de la Iteración #1 de las pruebas de unidad

Resultados de las Pruebas de Unidad

Durante las 3 iteraciones de las pruebas de unidad realizadas al sistema se detectaron un total de 3 errores en el análisis de 10 funcionalidades Véase Fig 27. Entre los principales errores se destaca la incompatibilidad entre los tipos de datos devueltos por las funcionalidades y errores en la lectura de ficheros almacenados en el sistema.

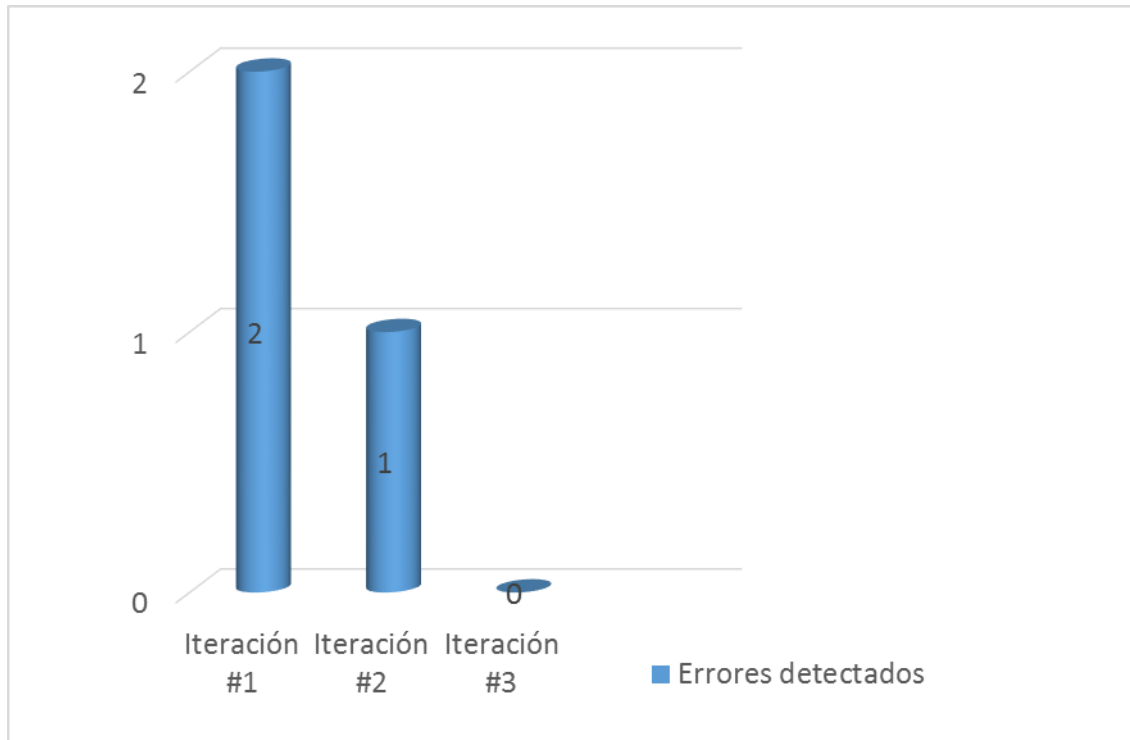


Fig 15: Resultados de las pruebas de unidad

4.2.3.2 Realización de las pruebas de integración

Tabla 8: Estrategia para la realización de las pruebas de integración.

Nivel de prueba:	Integración
Tipo de Prueba:	Funcionalidad función
Método de prueba:	Caja Negra

A continuación se detalla la tabla de integración para los módulos con los que se interactúa.

Tabla 9: Pruebas de integración iteración # 1.

Iteración # 1

Módulo actual	Módulo integrado	Funcionalidad	Condiciones de ejecución	Escenarios de prueba	Resultado previsto	Resultado real

Módulo de Carga de Datos Offline del SIAI (SIAIODL)	Módulo Servidor de notificaciones del SIAI (SIAINTFS)	Crear una conexión con el SIAINTFS	El SIAINTFS debe estar en ejecución y el Módulo actual debe estar ejecutándose correctamente.	Crear una conexión con el SIAINTFS.	Crear un objeto que manipule la conexión y demás funcionalidades con el SIAINTFS.	Se realizó la conexión con el SIAINTFS satisfactoriamente.
Módulo de Carga de Datos Offline del SIAI (SIAIODL).	Módulo Servidor de notificaciones del SIAI (SIAINTFS).	Recibir NSCD.	El SIAINTFS debe estar en ejecución y el Módulo actual debe estar ejecutándose correctamente y solicitar la descarga de las NSCD.	Obtener correctamente la NSCD	Obtener correctamente la NSCD sin ningún tipo de error en los datos que componen las mismas.	Se recibió satisfactoriamente la solicitud de NSCD. Pero se detectó un error al quedarse vacío el servidor de colas de mensajes, el mismo interrumpía la comunicación con el SIAIODL.
Módulo de Carga de Datos Offline del SIAI (SIAIODL).	Módulo Servidor de notificaciones del SIAI (SIAINTFS).	Enviar NCDR.	El SIAINTFS debe estar en ejecución y el Módulo actual debe estar ejecutándose correctamente.	Enviar correctamente la NCDR.	Enviar correctamente la NCDR.	Se envió satisfactoriamente al SIAINTFS la NCDR.

			e y solicitar el envío de una NCDR.			
Módulo de Carga de Datos Offline del SIAI (SIAIODL).	Módulo Servidor de notificaciones del SIAI (SIAINTFS).	Verificar la conexión con el SIAINTFS	El SIAINTFS debe estar en ejecución y el Módulo actual debe estar ejecutándose correctamente.	Verificar el estado de la conexión con el SIAINTFS	Se debe recibir el estado de la conexión entre el SIAIODL y el SIAINTFS	Se recibe satisfactoriamente el estado de la conexión entre el SIAIODL y el SIAINTFS.

Tabla 10: Pruebas de integración iteración # 2.

Iteración # 2

Módulo actual	Módulo integrado	Funcionalidad	Condiciones de ejecución	Escenarios de prueba	Resultado previsto	Resultado real
Módulo de Carga de Datos Offline del SIAI (SIAIODL)	Módulo Servidor de Comunicaciones del SIAI (SIAINTFS)	Crear una conexión con el SIAINTFS	El SIAINTFS debe estar en ejecución y el Módulo actual debe estar ejecutándose correctamente.	Crear una conexión con el SIAINTFS.	Crear un objeto que manipule la conexión y demás funcionalidades con el SIAINTFS.	Se realizó la conexión con el SIAINTFS satisfactoriamente.

Módulo de Carga de Datos Offline del SIAI (SIAIODL).	Módulo Servidor de Comunicaciones del SIAI (SIAINTFS).	Recibir NSCD.	El SIAINTFS debe estar en ejecución y el Módulo actual debe estar ejecutándose correctamente y solicitar la descarga de las NSCD.	Obtener correctamente NSCD.	Obtener correctamente la NSCD sin ningún tipo de error en los datos que componen las mismas.	Se recibió satisfactoriamente la NSCD.
Módulo de Carga de Datos Offline del SIAI (SIAIODL).	Módulo Servidor de Comunicaciones del SIAI (SIAINTFS).	Enviar NCDR.	El SIAINTFS debe estar en ejecución y el Módulo actual debe estar ejecutándose correctamente y solicitar el envío de una NCDR.	Enviar correctamente la NCDR.	Enviar correctamente la NCDR.	Se envió satisfactoriamente al SIAINTFS la NCDR
Módulo de Carga de Datos Offline del SIAI (SIAIODL).	Módulo Servidor de Comunicaciones del SIAI (SIAINTFS).	Verificar la conexión con el SIAINTFS	El SIAINTFS debe estar en ejecución y el Módulo actual debe estar ejecutándose correctamente	Verificar el estado de la conexión con el SIAINTFS	Se debe recibir el estado de la conexión entre el SIAIODL y el SIAINTFS	Se recibe satisfactoriamente el estado de la conexión entre el SIAIODL y el SIAINTFS.

Resultados de las Pruebas de Integración

Al realizar las pruebas de integración entre el SIAIODL y el SIAINTFS se detectaron y rectificaron diferentes errores en el análisis de una de las principales funcionalidades del SIAINTFS. Véase Fig 28

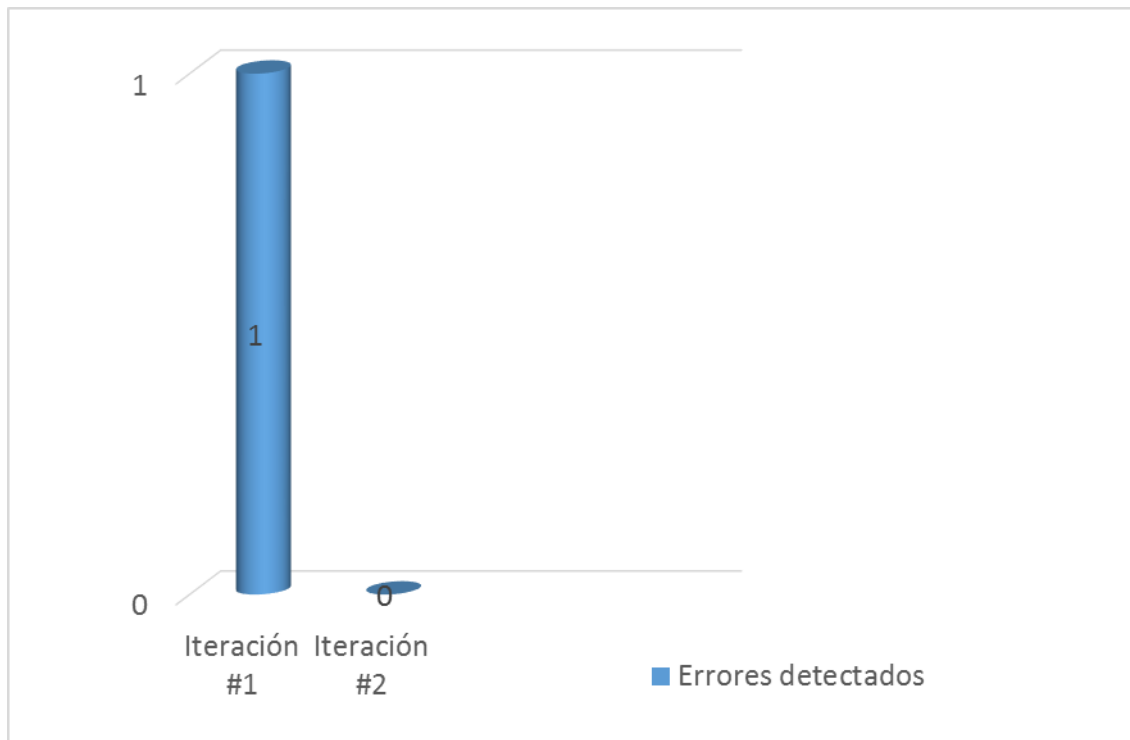


Fig 16: Resultados de las pruebas de integración.

Conclusiones del capítulo

Se logró en el presente capítulo un mejor entendimiento de la estrategia de pruebas ejecutadas sobre el sistema con el objetivo de comprobar su calidad, presentado una descripción bien detallada de los artefactos generados en esta fase de pruebas. Los resultados obtenidos a través de las pruebas llevadas a cabo ejemplifican la calidad del trabajo efectuado en la implementación del SIAIODL, mostrando que las

funcionalidades desarrolladas no presentan anomalías en su funcionamiento ya que realizan un correcto tratamiento de los datos que manejan. Por lo que se concluye que el resultado de las pruebas realizadas ha sido satisfactorio.

CONCLUSIONES GENERALES

El resultado principal de este trabajo de investigativo es la obtención de una aplicación que permita cargar la información con caducidad mayor de un período de tiempo que se encuentra offline en el SIAI, para ser utilizada en el proceso de investigación por parte de los analistas de fraude en ETECSA sobre el uso de los servicios de telefonía fija y móvil en Cuba. Para dar cumplimiento al objetivo general asentado sobre la problemática expuesta, se desarrollaron cada una de las tareas que fueron perfiladas al comienzo de la investigación.

- Se realizó un estudio de las tendencias actuales en el marco de las herramientas para la Extracción, transformación y carga de datos. Se tuvieron en cuenta las principales herramientas ETL “libres” para su inserción como parte de la solución a desarrollar continuando posteriormente con las herramientas ETL “privativas” con el objetivo de adquirir una mayor visión acerca de sus principales características y principios de funcionamiento, integrando estas ideas en la solución a desarrollar, un sistema ETL a resolviera el negocio en cuestión, incluyendo de forma directa en el sistema una plataforma para la administración de las NSCD y de Carga de datos realizadas, así como las funcionalidades necesarias para la generación de catálogos y mecanismos para la gestión de las actividades realizadas por el sistema.
- El proceso de desarrollo de software fue guiado a través del uso de RUP como metodología por generar abundante documentación y artefactos, algo de vital importancia para el funcionamiento del proyecto SIAI en general por su gran envergadura y complejidad.
- Se realizó la modelación del negocio para una mejor comprensión de los procesos vinculados a la carga de datos offline asociados a comportamientos en el uso de la telefonía fija y móvil en Cuba. Además se definieron los requisitos funcionales y no funcionales de acuerdo a las necesidades del sistema, fijando de esta forma un punto de encuentro entre los desarrolladores y el cliente final, en este caso ETECSA, para el desarrollo de la aplicación.
- Se realizó el análisis y diseño del sistema haciendo uso de la arquitectura en capas y de patrones de diseño GRASP y GOF como soporte para la correcta implementación del sistema. Arrojanando diferentes artefactos que brindaron una vista física y lógica de la aplicación, entre los cuales se

encuentran: los diagramas de clases del diseño, diagramas de componentes, modelo de datos y el diagrama de despliegue.

- Fue definida la estructura de ficheros necesaria para la gestión de las salvas de información del banco de datos.
- Fueron implementados los componentes necesarios para el correcto funcionamiento del sistema en general, ejemplo de ello se tiene:
 - ✓ Autenticar usuario: sección en la cual se gestiona el acceso y seguridad de la aplicación, permitiendo controlando de esta forma el acceso a la misma solo del personal autorizado.
 - ✓ Centro de Notificaciones: sección de la aplicación encargada de administrar las Notificaciones de NSCD recibidas por el sistema.
 - ✓ Cargador de Datos: sección del sistema que permite la atención de las notificaciones a tratar por la aplicación, díganse las NSCD, soportando dicha operación de forma personalizada y automática.
 - ✓ Centro de Reportes: Sección de la aplicación en la cual se pueden ver reflejadas todas las operaciones realizadas por el sistema.
 - ✓ Ejecución de operaciones de eliminación de información en la base de datos: operaciones ejecutadas por el sistema para eliminar información en el banco de datos que ha cumplido el tiempo de caducidad determinado por el usuario.
- Se diseñó y efectuó la estrategia de pruebas, asegurando de esta forma la calidad del sistema implementado. Durante el desarrollo de las mismas fueron detectados y resueltos diferentes errores asociados al código fuente y a la integración con el SIAINTFS.

Luego del análisis anterior se puede afirmar que la presente investigación alcanzó los objetivos propuestos.

RECOMENDACIONES

Teniendo en cuenta los resultados obtenidos con la realización del presente trabajo de diploma se hace necesario reflejar algunas recomendaciones:

- Creación de una herramienta con el mismo principio de funcionamiento del SIAIODL, pero que la misma se encargue de la realización y modificación de las salvadas de información de los bancos de datos, realizando además las actualizaciones pertinentes en los catálogos de información almacenados en la base de datos asociada al SIAIODL.
- Incluir en la plataforma Web del SIAI una sección para el envío de las NSCD, permitiéndoles a los usuarios del SIAI la solicitud de carga de datos.
- Continuar el desarrollo de la aplicación para poder relacionarse con otros sistemas gestores de bases de datos, pues en esta versión inicial solo es soportado ORACLE.

REFERENCIAS

1. **ETECSA.SA.** Empresa de Telecomunicaciones de Cuba. *ETECSA.SA.* [En línea] [Citado el: 10 de 8 de 2012.] www.etecsa.cu. 1.
2. **SIAI.** *Estado del arte del producto a desarrollar.* 2010. 2.
3. **Diaz, Yohanlena Hartman.** *Implementación del proceso de extracción, transformación y carga en un almacén de datos operacional para CIMEX.* 2009. 3.
4. **ETL-Tools.** Business Intelligence - Data warehousing - ETL. [En línea] 2006. <http://etl-tools.info/>. 4.
5. **Kimball, RLH.** *The Datawarehouse ETL Toolkit.* 2004. 5.
6. **GRAVITAR.** GRAVITAR. [En línea] 2010. [Citado el: 20 de 10 de 2012.]
7. **KRAUTHAMER, LIC. DIEGO.** *Ventajas del uso de herramientas de ETL sobre ANSI SQL.* 2009.
8. **Juan D Peña Rivera, Jesús A Suárez Daza.** *Utilización de información histórica para decisiones empresariales.* 2005.
9. **BI.** Rincón del BI. [En línea] 2009. [Citado el: 12 de 10 de 2012.]
10. **ETLTools company.** ETL Enterprise Data Integration. [En línea] [Citado el: 22 de 11 de 2012.] <http://www.etltools.net>.
11. **Pentaho .** Pentaho. [En línea] [Citado el: 24 de 10 de 2012.] <http://www.pentaho.com/explore/>.
12. **Cognus.** Cognus, empresa para la consultoria en negocios. [En línea] [Citado el: 1 de 12 de 2012.] <http://www.cognus.cl/content/view/794163/Pentaho-Data-Integration-4-0.html>.
13. **DS-Talend_Integration_Suite.** http://www.azinta.com/Services/DS-Talend_Integration_Suite_EN-A4-oct08.pdf. s.l. : azinta, 2008.
14. **Mundo Business Intelligence.** Herramienta ETL. [En línea] [Citado el: 3 de 12 de 2012.] <http://mundobi.wordpress.com/2007/06/24/herramientas-etl-%E2%80%A6-mundo-etl/>.
15. **Neyaris Pelegrín Tamayo, Virgen Casaña Vinagera.** *Proceso de la migración de datos hacia un Data Warehouse para el módulo Análisis Químico del proyecto LIMS Control de Calidad.* 2009.
16. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El proceso unificado de desarrollo de software.* Madrid : Pearson Adisson-Wesley. 2000.
17. **SIAI.** *Descripción de la metodología de desarrollo_SIAI.* 2010.
18. **Microsoft.** MSDN. [En línea] 2010. [Citado el: 1 de 1 de 2013.] <http://msdn.microsoft.com/es-es/library/bb972240.aspx>.
19. **Marcello Visconti, Hernán Astudillo.** *Fundamentos de Ingeniería deSoftware.* s.l. : Departamento de Informática - Universidad Técnica Federico Santa María., 2009.
20. **Facultad de Informática – Universidad Politécnica de Madrid.** *Patrones del “Gang of Four” .* s.l. : Facultad de Informática – Universidad Politécnica de Madrid.
21. **Marca Huallpara, Hugo Michael, Quisbert Limachi, Nancy Susana.** *Diagramas de Despliegue.* 2010.
22. **Ramírez, Elisa Arizaca.** *Diagrama de componentes.* ACHACHICALA : UNIVERSIDAD SALESIANA DE BOLIVIAINGENIERIA DE SISTEMASCAMPUS.
23. **Peña, Juan Manuel Fernández.** *Pruebas de sistema.* 2010.
24. **Pasionned Group.** Pasionned Group. [En línea] [Citado el: 10 de 2 de 2013.] <http://www.pasionned.com/>.
25. **Mundo Business Intelligence.** *Herramienta ETL (.o Mundo ETL).* [En línea] [Citado el: 04 de 05 de 2009.] <http://mundobi.wordpress.com/2007/06/24/herramientas-etl-%E2%80%A6-mundo-etl/>.
26. **Neyaris Pelegrín Tamayo Virgen Casaña Vinagera** *Proceso de la migración de datos hacia un Data Warehouse para el módulo Análisis Químico del proyecto LIMS Control de Calidad.* **2009**

27. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El proceso unificado de desarrollo de software.* Madrid : Pearson Adisson-Wesley, 2000.
28. **gdis.** *gdis.* [En línea] [Citado el: 21 de Noviembre de 2011.]
29. **idef.** *idef.* [En línea] [Citado el: 21 de Noviembre de 2011.] <http://www.idef.com/IDEF0.htm>.
30. **projects.gnome.** [En línea] [Citado el: 18 de Noviembre de 2011.] <http://projects.gnome.org/dia/>.
31. **Visual Paradigm.** *Visual Paradigm.* [En línea] [Citado el: 26 de Noviembre de 2011.] <http://www.visual-paradigm.com/product/vpuml/>.
32. **Alapati, Sam.** *OCP Oracle Administration 11g.* Mexico : Oracle Press, 2008. ISBN 0-07-149682-3.
33. *Strategic Management: A Stakeholder Approach* (1984) Boston, MA: **Pittman.**

BIBLIOGRAFÍA

1. *Journal Architect No. 6*. 2006.
2. *Mastering Data Warehouse Aggregates*. Adamson, Christopher. 2006.
3. *Pentaho Solutions. Business Intelligence and DataWarehousing with Pentaho and MySQL*. Roland Bouman, Jos Van Dongen. 2010.
4. *Building the Data Warehouse , Fourth Edition*. Inmon, W. H. 2005.
5. *Oracle Database 10g Data Warehousing*. Lilian Hobbs, Susan Hillson, Shilpa Lawande, Pete Smith. 2005.
6. *Intro to Python*. Hognl, Hubert. 2011.
7. *Inmersión en Python*. 2005.
8. *Python para todos*. 2008.
9. *Qt 4 manual introductorio*. Corporativo. 2012.
10. *Interfaces Graficas en Python*. Fernández, David López. 2010.

ANEXOS

ANEXO 1

CU Atender notificación

Descripción de Casos de Uso

Tabla 11: Descripción del Caso de uso Atender notificación

Caso de Uso:	Atender notificación
Actores:	Operador (Inicia)
Resumen:	El caso de uso se inicia cuando el operador decide atender las notificaciones enviadas al cargador de datos.
Precondiciones:	<ol style="list-style-type: none">1. El usuario debe estar autenticado por el sistema.2. Deben haber NSCD en el cargador de datos.
Referencias	RF8, RF9, RF10, RF11, RF12, RF13, RF14, RF15, RF16, RF17, RF18
Prioridad	Crítico
Flujo Normal de Eventos	
Sección “Atender notificación”	
Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none">1. El sistema muestra la interfaz “Cargador de datos” visualizando los siguientes datos:<ul style="list-style-type: none">• Listado de las solicitudes.Además visualiza las siguientes opciones:<ul style="list-style-type: none">• Atender solicitudes de forma automática. (Se invoca al CU Atender notificación de forma automática)• Eliminar una solicitud. (Ir a la sección Eliminar solicitud)• Eliminar todas las solicitudes. (Ir a la sección Eliminar solicitudes)• Centro de reportes (Se invoca al CU Administrar reporte)
2. Selecciona la solicitud.	<ol style="list-style-type: none">3. Muestra en la venta “Información de solicitud” los archivos asociados a la solicitud y en el menú de herramientas las siguientes opciones:<ul style="list-style-type: none">• Cargar de unidad DVD

	<ul style="list-style-type: none"> • Cargar de directorio • Expulsar bandeja de unidad DVD. (Ir a la sección Expulsar) • Mostrar ayuda. (Se invoca al CU Mostrar ayuda) <p>Además muestra las siguientes opciones en el menú de herramientas del explorador de ficheros:</p> <ul style="list-style-type: none"> • Añadir ficheros al cargador de ficheros. • Seleccionar todo los ficheros. (Selección de ficheros)
4. Selecciona la opción "Cargar de directorio ".	5. Carga los archivos del directorio.
	6. Muestra el árbol de ficheros contenidos en el directorio.
7. Seleccionar los ficheros previamente solicitados. Selecciona la opción "Añadir a cargador de ficheros".	<p>8. Muestra en la ventana "Cargador de ficheros", un resumen de los ficheros, con la siguiente información></p> <ul style="list-style-type: none"> • Ficheros. • Fecha. <p>Además muestra en la Barra de herramientas del cargador de fichero las opciones:</p> <ul style="list-style-type: none"> • Eliminar fichero. (Ir a la sección Eliminar fichero) • Eliminar todos los ficheros. (Ir a la sección Eliminar todos los fichero)
	<p>9. Actualiza el contenido a adicionar en la BD, con la siguiente información></p> <ul style="list-style-type: none"> • Archivo • Tabla • Tupla
	10. Actualiza el nivel de completamiento de las notificaciones afectadas por los archivos cargados para insertar en la base de datos.
11. Selecciona la opción	12. Verifica conexión con la BD.

“Enviar contenido de fichero”.	
	13. Inserta en la BD, el contenido de los ficheros.
	14. Genera automáticamente un reporte de la solicitud atendida.
	15. Envía notificación al usuario correspondiente con el siguiente formato: <ul style="list-style-type: none"> • Fecha • Hora • Detalles

Prototipo de Interfaz



Flujo Alternos

Acción del Actor	Respuesta del Sistema
4.1. Introduce el disco en la unidad DVD. Selecciona la opción “Cargar de unidad DVD”.	4.2. Continúa con la acción 5 del flujo normal de eventos.
4.1.1. Selecciona la opción “Cargar de unidad DVD”.	4.1.2. Muestra el siguiente mensaje de error: “Introduzca el DVD correspondiente en la unidad”. Fig #17

	12.1. Existen problemas de conexión con la BD. Fig #18
	13.1. Muestra el siguiente mensaje de error: “Imposible conectar con la Base de Datos”. Fig #19
	15.1. Existen problemas de comunicación con el servidor de comunicación del SIAI. No se envía la notificación. Se inserta un reporte de fallo.

Prototipo de Interfaz

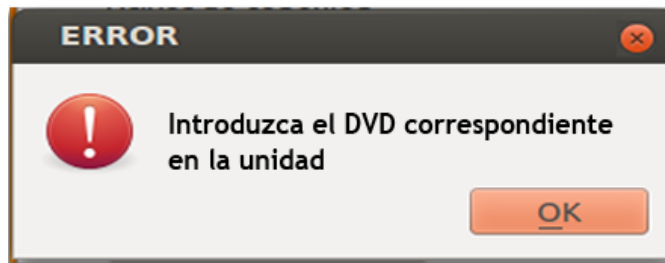


Fig #17 Mensaje emitido por el sistema

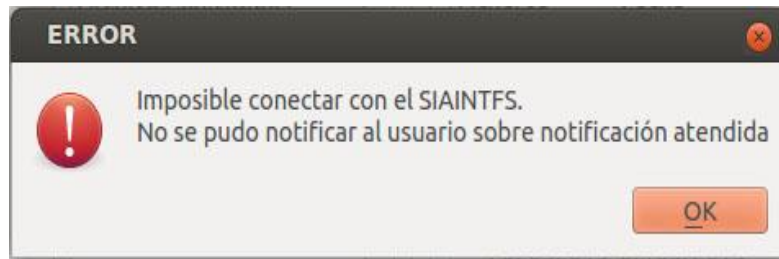


Fig #18 Mensaje emitido por el sistema

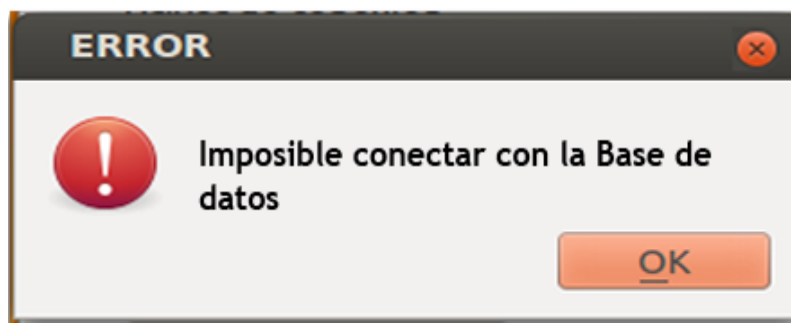


Fig #19 Mensaje emitido por el sistema

Flujo Normal de Eventos

Sección “Selección de ficheros”

Acción del Actor	Respuesta del Sistema
-------------------------	------------------------------

1. Selecciona la opción “Seleccionar todos los ficheros”.

2. Selecciona o deselecciona todos los ficheros en dependencia de la última acción registra.

Prototipo de Interfaz



Flujo Alternos

Acción del Actor	Respuesta del Sistema

Prototipo de Interfaz

Flujo Normal de Eventos

Sección “Eliminar notificación”

Acción del Actor	Respuesta del Sistema
1. Selecciona una notificación accede a la opción “Eliminar notificación”.	2. Elimina la notificación del sistema.

Prototipo de Interfaz

Flujo Alternos

Acción del Actor	Respuesta del Sistema

Prototipo de Interfaz



Sección “Eliminar todas las notificaciones”

Acción del Actor	Respuesta del Sistema
1. Selecciona la opción “Eliminar todas las notificaciones”	2. Elimina todas las notificaciones listadas en el cargador de datos.

Prototipo de Interfaz



Flujo Alternos

Acción del Actor	Respuesta del Sistema

Prototipo de Interfaz