

Universidad de las Ciencias Informáticas

Facultad 1



Desarrollo de la “Herramienta auxiliar para el procesamiento de datos de clientes” para un sistema de Correo Híbrido Masivo.

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autores: Yans Díaz García
Johan Barbosa Martínez

Tutores: Dra. Edelia García González
Ing. Johann Rodríguez Hernández

Ciudad de la Habana
Julio de 2008

"Si el presente es lucha, el futuro es nuestro"
Ernesto "Che" Guevara

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de ____ del ____.

Autores:

Tutores:

Yans Díaz García

Dra. Edelia García González

Johan Barbosa Martínez

Ing. Johann Rodríguez Hernández

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado Desarrollo de la "Herramienta auxiliar para el procesamiento de datos de clientes" para un Sistema de Correo Híbrido Masivo, fue realizado en la Universidad de las Ciencias Informáticas (UCI), como ejercicio académico dentro del polo de servicios postales de la facultad 1. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

- Totalmente
- Parcialmente

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes:

Y para que así conste, se firma la presente a los ____ días del mes de ____ del año ____.

Representante de la entidad

Cargo

Firma

Cuño

DEDICATORIA

A:

*Mami y papi que prácticamente se graduaron junto a mí y a todos
los que desearon siempre lo mejor para mí.*

Yans Díaz García

A:

*Mi mamá, a mi abuela, a Carlito, a mi abuelo (Pape), mis tíos
Raulito y Jorge y mis hermanas queridas Yoli y Carla.*

Johan Barbosa Martínez

AGRADECIMIENTOS

A mis padres por tener siempre un consejo y una solución a cada uno de mis problemas, gracias a ellos estoy aquí.

A mi hermana por inspirarme a ser su ejemplo.

A Yanelys mi novia y mejor amiga por su apoyo, su amor y su aguante.

A mis tíos, abuelos y primos por ser mi gran familia y por estar siempre pendientes y alegrarse por mis resultados.

A los tutores Edelia y Johann por su tiempo, por crecerse y enseñarnos a crecer ante las dificultades.

A cada maestro y profesor que desde la enseñanza primaria, hasta la UCI se esforzó por enseñarme algo nuevo.

A los amigos Dayron, Ricardo, Vladimir, Rolando, Darién y a las amigas Lurdes, Dole, Yadira, Maylin, Helen a todos ellos por hacerme vivir buenos momentos en la UCI.

A los compañeros del proyecto, Blanco, Diovis, Erick, Fernando, Liudnet por toda la ayuda que nos brindaron para la realización de este trabajo.

A mi compañero de tesis Johan por compartir esta dura tarea.

A todos los que se preocuparon y siempre estuvieron al tanto de esta tesis.

Yans Díaz García

AGRADECIMIENTOS

A mi Mamá porque siempre a estado a mi lado en todos los momentos y me ha convertido en el hombre que soy y porque no tengo palabras para decirle cuanto significa para mi.

A mi abuela porque desde niño la he querido con toda mi vida y siempre me ha apoyado y contribuido a mi formación como profesional.

A Carlito porque ha sido más que un padre para mi y mucho trabajo ha pasado conmigo y muchos viajes a La Habana por escoger Informática en al UCI.

A mis hermanas, a Pape, a mis tíos y todos mis familiares porque ellos siempre están donde los necesito.

A mi abuela Georgina, mi tía, mi Papá, mi hermano, mi abuelo y a mi prima.

A Carlos, Victoria, Marlene por ser parte de mi familia.

A David porque se que está orgulloso de mi.

A Maidiel, Tito, Eddy y Yanisley, Anniel, Frikky, Bombino, Aleanne, Andro, Arnaldito y Susel, Polito y todos mis amigos por ser solo eso (jejeje),mis amigos que ya con eso es suficiente tener que aguantarme constantemente en momentos buenos y malos.

A mi novia Meylin por estar a mi lado estos 5 años y ser siempre mi guía, que gracias a ella estoy aquí con los resultados que he tenido.

A mi compañero de tesis Yans, a Fernando, a Erick, Blanco y todos los que han contribuido a la realización de este trabajo de una u otra forma.

Muchas gracias a todos de verdad.

Johan (JBM)

RESUMEN

El Correo Híbrido (CH) es un servicio, o conjunto de servicios vinculados a la actividad postal que permite la transmisión electrónica de mensajes desde un cliente hasta un proveedor, el cual utilizando sistemas teleinformáticos y medios postales garantiza la terminación del producto y la entrega física del envío al destinatario final.

El Instituto Postal y Telegráfico de Venezuela (IPOSTEL) desea incluir el Correo Híbrido Masivo en su gama de servicios y para materializarlo, se desarrolló una aplicación Web de gestión. Esta aplicación garantiza la gestión del negocio en un sistema de plantas de Correo Híbrido Masivo (CHM) desde que el cliente suministra los datos a enviar a la planta, hasta que los datos son entregados al destinatario. Durante este proceso la información suministrada por el cliente puede ser objeto de ataques informáticos, o víctima de errores humanos, que atenten contra su integridad, por lo que se plantea la necesidad de fortalecer el sistema en este aspecto de la seguridad y otros aspectos relacionados con la clasificación y el compactado de la información a procesar.

El presente trabajo propone el desarrollo de una herramienta que implemente las funcionalidades necesarias para garantizar la seguridad de la información que será procesada en la planta de CHM. El estudio de las tendencias tecnológicas actuales sustenta la decisión de optar por desarrollarla con software libre, para ello se utilizan NetBeans como IDE y Java como lenguaje de programación. Para el modelado se utiliza la herramienta case *Visual Paradigm*, y todo el proceso de desarrollo esta basado en la metodología RUP. De esta manera surgió una aplicación de escritorio que soluciona la problemática planteada.

TABLA DE CONTENIDOS

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	6
1.1 Introducción.....	6
1.2 Conceptos asociados al dominio del problema.....	6
1.2.1 Los servicios postales.....	6
1.2.2 Los sistemas de Correo Híbrido.....	7
1.3 Posibles clientes del Correo Híbrido Masivo.....	7
1.4 Criptografía.....	7
1.4.1 Encriptación.....	8
1.4.2 Tipos de cifrado de información.....	8
1.4.3 Métodos y técnicas de encriptación.....	10
1.5 Niveles de seguridad en las herramientas de manejo de información.....	11
1.6 Análisis de herramientas de procesamiento de datos existentes.....	11
1.6.1 Agata Report.....	12
1.6.2 MSD Caja Fuerte Gestor de documentos y ficheros privados.....	12
1.7 Tecnologías de desarrollo propuestas para la solución.....	13
1.7.1 Software libre.....	13
1.7.2 Java como lenguaje de programación.....	13
1.7.3 NetBeans.....	14
1.8 Metodología utilizada.....	14
1.8.1 Lenguaje Unificado de Modelado (UML).....	15
1.8.2 Proceso Unificado de Rational (RUP).....	15
1.8.3 Visual Paradigm.....	17
1.9 Conclusiones.....	18
CAPÍTULO 2: MODELO DE NEGOCIO.....	19
2.1 Introducción.....	19
2.2 Flujo actual de los procesos.....	19
2.3 Análisis crítico de la ejecución de los procesos.....	19
2.4 Modelo del negocio.....	20
2.4.1 Definición de los actores del negocio.....	20
2.4.2 Definición de los trabajadores del negocio.....	20
2.4.3 Diagrama de casos de uso del negocio.....	21
2.4.4 Descripción textual de los casos de uso del negocio.....	21
2.4.5 Diagramas de actividades.....	23
2.4.6 Diagrama de clases del modelo de objetos.....	26
2.5 Conclusiones.....	26
CAPÍTULO 3: CARACTERÍSTICAS DEL SISTEMA.....	27
3.1 Introducción.....	27
3.2 Especificación de los requisitos de software.....	27
3.2.1 Definición de los requisitos funcionales.....	27
3.2.2 Definición de los requisitos no funcionales.....	28
3.3 Modelo del sistema.....	30
3.3.1 Definición de los actores del sistema.....	30

Tabla de Contenidos

3.3.2	Definición de los casos de uso del sistema.....	31
3.3.3	Diagrama de casos de uso del sistema.....	31
3.3.4	Descripción expandida de los casos de uso del sistema.....	32
3.4	Conclusiones.....	35
CAPÍTULO 4: ANÁLISIS Y DISEÑO DEL SISTEMA.....		36
4.1	Introducción.....	36
4.2	Análisis.....	36
4.2.1	Modelos de clases del análisis.....	36
4.3	Diseño.....	37
4.3.1	Diagramas de clases del diseño.....	38
4.3.2	Algunas clases, ficheros y sus funciones.....	47
4.3.3	Diagramas de interacción.....	49
4.3.4	Prototipo de interfaz.....	54
4.4	Conclusiones.....	54
CAPÍTULO 5: IMPLEMENTACIÓN Y PRUEBA.....		55
5.1	Introducción.....	55
5.2	Diagrama de despliegue.....	55
5.3	Diagrama de componentes.....	56
5.4	Modelo de prueba.....	56
5.4.1	Modelos de prueba de caja negra.....	57
5.4.2	Modelos de prueba de caja blanca.....	58
5.5	Conclusiones.....	60
CONCLUSIONES.....		61
RECOMENDACIONES.....		62
BIBLIOGRAFÍA.....		63
GLOSARIO.....		66
ANEXOS.....		67

ÍNDICE DE TABLAS

TABLA 1.	OPERACIONALIZACIÓN DE LAS VARIABLES.	4
TABLA 2.1.	DESCRIPCIÓN DE LOS ACTORES DEL NEGOCIO.	20
TABLA 2.2.	DESCRIPCIÓN DE LOS TRABAJADORES DEL NEGOCIO.	20
TABLA 2.3.	DESCRIPCIÓN TEXTUAL DEL CASO DE USO DEL NEGOCIO ENTREGAR DATOS A LA PLANTA. ..	22
TABLA 2.4.	DESCRIPCIÓN TEXTUAL DEL CASO DE USO DEL NEGOCIO EXTRAER DATOS PARA IMPRESIÓN.	23
TABLA 3.1.	DESCRIPCIÓN DE LOS ACTORES DEL SISTEMA.	31
TABLA 3.2.	DESCRIPCIÓN TEXTUAL DEL CASO DE USO DEL SISTEMA CONFIGURAR CONEXIÓN A BD.	34
TABLA 3.3.	DESCRIPCIÓN TEXTUAL DEL CASO DE USO DEL SISTEMA GENERAR ARCHIVOS CON DATOS.	35
TABLA 4.1.	DESCRIPCIÓN DEL FICHERO SOLICITUDES.TXT.	47
TABLA 4.2.	DESCRIPCIÓN DEL FICHERO USUARIOS.TXT.	47
TABLA 4.3.	DESCRIPCIÓN DEL FICHERO CONSULTAS.TXT.	47
TABLA 4.4.	DESCRIPCIÓN DE LA CLASE VALIDATE.	47
TABLA 4.5.	DESCRIPCIÓN DE LA CLASE ACCESSDATAABS.	48
TABLA 4.6.	DESCRIPCIÓN DE LA CLASE FILEHANDLER.	48
TABLA 4.7.	DESCRIPCIÓN DE LA CLASE ADMINISTERCONFIG.	48
TABLA 4.8.	DESCRIPCIÓN DE LA CLASE ENCRIPADORDES.	48
TABLA 4.9.	DESCRIPCIÓN DE LA CLASE ENCRIPADORTDES.	49
TABLA 5.1.	PRUEBA DE CAJA NEGRA AL CASO DE USO ADMINISTRAR USUARIOS CLIENTES.	57
TABLA 5.2.	PRUEBA DE CAJA NEGRA AL CASO DE USO DESENCRIPTAR.	58
TABLA 5.3.	PRUEBA DE CAJA NEGRA AL CU GENERAR RESUMEN DE PROPIEDADES DE FICHERO.	58
TABLA 3.4.	DESCRIPCIÓN TEXTUAL DEL CASO DE USO DEL SISTEMA AUTENTICAR USUARIO.	67
TABLA 3.5.	DESCRIPCIÓN TEXTUAL DEL CU DEL SISTEMA ADMINISTRAR USUARIOS CLIENTES.	69
TABLA 3.6.	DESCRIPCIÓN TEXTUAL DEL CASO DE USO DEL SISTEMA DESENCRIPTAR.	70
TABLA 3.7.	DESCRIPCIÓN TEXTUAL DEL CASO DE USO DEL SISTEMA GENERAR RESUMEN.	72

ÍNDICE DE FIGURAS

FIGURA 1.1.	CIFRADO SIMÉTRICO.....	9
FIGURA 1.2.	CIFRADO ASIMÉTRICO.....	9
FIGURA 1.3.	ALGORITMO TRIPLE DES.....	11
FIGURA 1.4.	FASES DEL RUP.....	16
FIGURA 2.1.	DIAGRAMA DE CASOS DE USO DEL NEGOCIO.....	21
FIGURA 2.2.	DIAGRAMA DE ACTIVIDAD: ENTREGAR DATOS A LA PLANTA.....	24
FIGURA 2.3.	DIAGRAMA DE ACTIVIDAD: EXTRAER DATOS PARA IMPRESIÓN.....	25
FIGURA 2.4.	DIAGRAMA DE OBJETOS.....	26
FIGURA 3.1.	DIAGRAMA DE CASOS DE USO DEL SISTEMA.....	32
FIGURA 4.1.	DIAGRAMA DE CLASES DEL ANÁLISIS PARA CONFIGURAR CONEXIÓN A BD.....	37
FIGURA 4.2.	DIAGRAMA DE CLASES DEL ANÁLISIS PARA GENERAR ARCHIVOS CON DATOS.....	37
FIGURA 4.3.	DIAGRAMA DE CLASES DEL DISEÑO PARA CONFIGURAR CONEXIÓN A BASE DE DATOS: SUPERIOR.....	39
FIGURA 4.4.	DIAGRAMA DE CLASES DEL DISEÑO PARA CONFIGURAR CONEXIÓN A BASE DE DATOS: CENTRAL.....	40
FIGURA 4.5.	DIAGRAMA DE CLASES DEL DISEÑO PARA CONFIGURAR CONEXIÓN A BASE DE DATOS: INFERIOR IZQUIERDO.....	41
FIGURA 4.6.	DIAGRAMA DE CLASES DEL DISEÑO PARA CONFIGURAR CONEXIÓN A BASE DE DATOS: INFERIOR DERECHO.....	42
FIGURA 4.1.	DIAGRAMA DE CLASES DEL DISEÑO PARA GENERAR ARCHIVOS CON DATOS: SUPERIOR.....	43
FIGURA 4.2.	DIAGRAMA DE CLASES DEL DISEÑO PARA GENERAR ARCHIVOS CON DATOS: CENTRAL IZQUIERDO.....	44
FIGURA 4.3.	DIAGRAMA DE CLASES DEL DISEÑO PARA GENERAR ARCHIVOS CON DATOS: CENTRAL DERECHO.....	45
FIGURA 4.4.	DIAGRAMA DE CLASES DEL DISEÑO PARA GENERAR ARCHIVOS CON DATOS: INFERIOR.....	46
FIGURA 4.5.	DIAGRAMA DE SECUENCIA PARA CONFIGURAR CONEXIÓN A BASE DE DATOS: IZQUIERDO.....	50
FIGURA 4.6.	DIAGRAMA DE SECUENCIA PARA CONFIGURAR CONEXIÓN A BASE DE DATOS: DERECHO.....	51
FIGURA 4.7.	DIAGRAMA DE SECUENCIA PARA GENERAR ARCHIVOS CON DATOS: IZQUIERDO.....	52
FIGURA 4.8.	DIAGRAMA DE SECUENCIA PARA GENERAR ARCHIVOS CON DATOS: DERECHO.....	53
FIGURA 4.9.	INTERFAZ DE USUARIO DEL SISTEMA.....	54
FIGURA 5.1.	DIAGRAMA DE DESPLIEGUE.....	55

Índice de Figuras

FIGURA 5.2.	DIAGRAMA DE COMPONENTES.....	56
FIGURA 4.10.	DIAGRAMA DE CLASES DEL ANÁLISIS PARA AUTENTICAR USUARIO.	73
FIGURA 4.11.	DIAGRAMA DE CLASES DEL ANÁLISIS PARA ADMINISTRAR USUARIOS CLIENTES.....	73
FIGURA 4.12.	DIAGRAMA DE CLASES DEL ANÁLISIS PARA DESENCRIPTAR.....	73
FIGURA 4.13.	DIAGRAMA DE CLASES DEL ANÁLISIS PARA GENERAR RESUMEN.	74
FIGURA 4.14.	DIAGRAMA DE CLASES DEL DISEÑO PARA AUTENTICAR USUARIO: SUPERIOR.	75
FIGURA 4.15.	DIAGRAMA DE CLASES DEL DISEÑO PARA AUTENTICAR USUARIO: INFERIOR.....	76
FIGURA 4.16.	DIAGRAMA DE CLASES DEL DISEÑO PARA ADMINISTRAR USUARIOS CLIENTES: SUPERIOR.	77
FIGURA 4.17.	DIAGRAMA DE CLASES DEL DISEÑO PARA ADMINISTRAR USUARIOS CLIENTES: INFERIOR. .	78
FIGURA 4.18.	DIAGRAMA DE CLASES DEL DISEÑO PARA DESENCRIPTAR: SUPERIOR.....	79
FIGURA 4.19.	DIAGRAMA DE CLASES DEL DISEÑO PARA DESENCRIPTAR: INFERIOR.	80
FIGURA 4.20.	DIAGRAMA DE CLASES DEL DISEÑO PARA GENERAR RESUMEN DE PROPIEDADES.	81
FIGURA 4.21.	DIAGRAMA DE SECUENCIA PARA AUTENTICAR USUARIO.....	82
FIGURA 4.22.	DIAGRAMA DE SECUENCIA PARA ADMINISTRAR USUARIOS CLIENTES.....	83
FIGURA 4.23.	DIAGRAMA DE SECUENCIA PARA DESENCRIPTAR.....	84
FIGURA 4.24.	DIAGRAMA DE SECUENCIA PARA GENERAR RESUMEN DE PROPIEDADES DE FICHERO.....	85

INTRODUCCIÓN

La revolución científico-tecnológica de las últimas décadas significó una transformación en los hábitos de la sociedad. Las personas con acceso a Internet, lo ven como una forma fácil, rápida y en ocasiones económica, para acceder a la información sobre productos o servicios de interés, desde el confort, conveniencia y privacidad de los hogares (SAVARIS 2008).

Basadas en esta idea las empresas han tomado conciencia de que Internet puede llegar a ser un negocio, gracias a la informática, que incluye diversas técnicas y actividades relacionadas con el tratamiento automático y lógico de la información. Esta disciplina alcanza un enorme desarrollo debido a las computadoras, pues tienen gran capacidad de memoria y el acceso a los datos e informaciones se realiza de manera sencilla y rápida. Gran cantidad de empresas, instituciones, y otras entidades tienen presencia en Internet, no sólo con el objetivo de promocionarse sino también con el objetivo de ofertar productos y servicios (CIENCIA 2007).

Las administraciones postales de diferentes países, como Brasil, Chile y otros, también han hecho uso de las Tecnologías de la información (TICs) para aumentar su presencia y servicios en Internet. Uno de los servicios que utiliza de forma amplia la tecnología, es el servicio de Correo Híbrido Masivo (CHM) para grandes clientes.

El Instituto Postal y Telegráfico de Venezuela (IPOSTEL) desde su privilegiada posición, a cargo la mayoría de los movimientos de paquetería y mensajería que realizan las empresas estatales a nivel nacional, y tener como misión satisfacer las necesidades de servicios en comunicaciones postales y telegráficas de sus clientes, contando para esto con la infraestructura y la logística necesarias para llegar a cualquier rincón del país, ofrecen la cobertura más amplia del servicio postal venezolano con un despliegue de más de 300 oficinas y 2197 carteros distribuidos a nivel nacional, que garantizan la entrega oportuna hasta en aquellos puntos donde la competencia no llega. Decide implementar la prestación del servicio de Correo Híbrido Masivo a posibles clientes como son los Bancos, cadenas televisivas y otras empresas de soporte vital a la población como la electricidad y el agua.

Para brindar este servicio a grandes clientes la institución necesita disponer de información valiosa de las empresas contratantes. La protección de la información es parte integral de la cultura del Correo y se debe garantizar por tanto, que los procesos que involucran la correspondencia y la información de facturación se harán con la máxima confidencialidad y seguridad para el cliente (IPOSTEL 2007).

Para dar soporte informático y gestionar el proceso de este servicio, se desarrolló una solución Web para la gestión del negocio de un sistema de plantas de Correo Híbrido Masivo, siendo aún una debilidad, no contar con las funcionalidades para garantizar la integridad de la información, compactar

Introducción

y clasificar de acuerdo a especificidades del negocio. De esta forma queda definido como **problema** a resolver, la siguiente interrogante:

¿Cómo procesar los datos provistos por los clientes al sistema de Correo Híbrido Masivo de IPOSTEL, garantizando la integridad de los mismos?

Como **objeto de estudio** se tiene al Sistema de Correo Híbrido Masivo de IPOSTEL.

Para dar solución al problema anterior se plantea como **objetivo general**:

Implementar una herramienta auxiliar dentro del sistema de Correo Híbrido Masivo de IPOSTEL que permita procesar los datos de los clientes garantizando su protección.

Como **tareas** a cumplir se tienen:

- La captura de los requisitos de la herramienta auxiliar.
- Estudio de las herramientas de procesamiento de datos a nivel internacional.
- La argumentación de las tecnologías a usar en el desarrollo de la herramienta auxiliar.
- Selección de los métodos de encriptación para la primera etapa de desarrollo de la herramienta.
- La realización del análisis, diseño e implementación de la herramienta auxiliar basados en la metodología RUP.
- Diseñar e implementar las estructuras necesarias para lograr que la herramienta se conecte a varios gestores de bases de datos con cualquier estructura donde se puedan utilizar diferentes métodos de encriptación de compactación y descompactación que garanticen el desarrollo de una herramienta flexible y segura.

Por tanto el presente trabajo centra su **campo de acción** en los procesos de recepción de la información de los clientes y de inserción de datos para la producción en un sistema de plantas de Correo Híbrido Masivo.

Se parte de la **hipótesis** que: con la implementación de una aplicación de escritorio, configurable a las necesidades de un sistema de Correo Híbrido Masivo y sus clientes, se garantizará el procesamiento de los datos provistos al negocio, logrando confidencialidad e integridad de la información.

Dada la hipótesis se plantean las siguientes **variables de la investigación**:

Variable independiente: implementación de una aplicación de escritorio.

Variable dependiente: confidencialidad e integridad de la información.

Variable dependiente: configurable a las necesidades de un SCHM y sus clientes.

Operacionalización de las variables:

Variable Conceptual	Dimensión	Indicadores	UM
Implementación de una aplicación de escritorio	Factibilidad	Tiempo de desarrollo	Extenso Moderado Breve
		Costo	Costoso Moderado Abaratado
		Esfuerzo	Alto Moderado Despreciable
Confidencialidad e integridad de la información	Eliminación de Vulnerabilidades	Físicas	Alto Medio Básico
		De Hardware	Alto Medio Básico
		Naturales	Alto Medio Básico
		Humanas	Alto Medio Básico
		De Software	Alto Medio Básico
		De Almacenamiento	Alto Medio Básico
		De Comunicación	Alto Medio Básico
Configurable a las necesidades de un SCHM y sus clientes	Satisfacción del cliente y la	Clasificación por plantas	Si__ No__
		Encriptar Información	Si__ No__
		Desencriptar Información	Si__ No__

Introducción

	empresa.	Clasificación del peso de los envíos.	Si__ No__
		Conexión a varios Gestores de bases de Datos	Si__ No__

Tabla 1. Operacionalización de las variables.

Los **métodos científicos** utilizados en la investigación fueron:

Métodos Teóricos:

- Método hipotético deductivo: Es el que permite la detección de un problema y el planteamiento de una hipótesis que posteriormente se demuestra con la práctica.
- Análisis y la síntesis: Se analiza la bibliografía utilizada y se realiza síntesis de la misma.
- Modelación: Se modelan diagramas en *Visual Paradigm* (VP) para un mejor entendimiento e implementación de la aplicación.

Métodos Empíricos:

- Encuestas: El empleo de encuestas con la realización de preguntas, se utilizan como comienzo para el diagnóstico inicial del objeto de estudio y específicamente del problema a resolver.
- Entrevistas: Se realizan entrevistas con el fin de precisar el problema a resolver, así como las principales deficiencias hacia las cuales se debe dirigir la solución que se implementará.

Resultados esperados:

Con el trabajo que se lleva a cabo se espera:

Una herramienta capaz de generar un fichero con datos encriptados, comprimidos y clasificados de acuerdo a criterios específicos del negocio. Realizará además tareas tanto en una planta (*online*) como en el cliente (*offline*).

El presente documento consta de cinco capítulos:

Capítulo 1: Fundamentación teórica: Es el respaldo teórico de los temas tratados en el informe, necesarios para el entendimiento correcto de la propuesta de solución planteada. Se describen los conceptos fundamentales asociados al dominio del problema y el objeto de estudio, haciéndose un análisis de la situación actual. Se presenta la fundamentación de las tecnologías utilizadas para el diseño del sistema y las propuestas para su implementación y desarrollo. Se abordan los principales problemas que originan la realización de este trabajo.

Introducción

Capítulo 2: Modelo de negocio: Se describen las actividades del flujo de procesos actuales a través de un modelo de negocio, el cual sirve de base para determinar qué es lo que se va a desarrollar. En el modelo del negocio se presentan los actores y trabajadores, los diagramas de los casos de uso y su correspondiente modelo de objeto, así como los diagramas de actividades de cada caso de uso del negocio según lo propone la metodología RUP.

Capítulo 3: Características del sistema: Se presentan los requerimientos funcionales y no funcionales con los que debe cumplir el sistema propuesto, así como sus actores y diagrama de casos de uso con la descripción expandida de cada uno de ellos.

Capítulo 4: Análisis y diseño del sistema: Se muestran los diagramas de clases del análisis y del diseño para cada caso de uso del sistema, junto con los correspondientes diagramas de interacción, además de otras restricciones del diseño.

Capítulo 5: Implementación y prueba: Se muestra el modelo de implementación además del diagrama de despliegue y se lleva a cabo pruebas de "Caja blanca" y "Caja negra" para obtener una mayor seguridad del sistema.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En este capítulo se presentará una visión general de los aspectos relacionados con el procesamiento de los datos de los clientes de una planta de Correo Híbrido Masivo (CHM), se dará una noción de quienes pudieran ser potenciales clientes del servicio de CHM. Como proteger sus respectivas informaciones mediante la criptografía. Además se describirán los principales conceptos asociados al dominio de problema.

La segunda parte del capítulo se centra en la investigación realizada para el estudio de herramientas de procesamiento de datos y la selección de la tecnología y metodología propuestas para desarrollar el software, acompañado de un resumen de las principales características de las herramientas seleccionadas.

1.2 Conceptos asociados al dominio del problema

Para una mejor comprensión de las funciones e importancia de esta investigación es necesario dominar ciertos temas y conocer al detalle el significado de los conceptos que están directamente relacionados con la problemática planteada. ¿Qué son los servicios postales?, dentro de ellos ¿Qué es el Correo Híbrido y quienes se pudieran servir de él?, ¿Qué nivel de seguridad deben tener las herramientas que procesan datos y cómo garantizarlo?, a continuación se brinda información indispensable que ayudará al entendimiento de la solución propuesta.

1.2.1 Los servicios postales

Con el desarrollo de la civilización el hombre buscó medios para establecer una comunicación a distancia de una forma cada vez más rápida y eficiente. En esta búsqueda surge el correo postal y con ello los servicios postales. El correo postal tiene como fin la comunicación entre individuos, es una institución de primer orden desde el punto de vista económico y social, creado para brindar de una forma eficiente los servicios postales. Comprenden las actividades de recepción, transporte y entrega de mensajería y paquetería. Son un medio de cohesión social y un instrumento de presencia del estado para facilitar el acceso a la información y el intercambio.

Capítulo 1: Fundamentación teórica

Entre los servicios postales existentes se encuentran la imposición de cartas, impresos, pequeños paquetes, bultos, cecogramas, postales y aerogramas, estos son conocidos como servicios tradicionales (CASTELLANO y LLANO 2007).

1.2.2 Los sistemas de Correo Híbrido

El Correo Híbrido consiste en la unión de dos tecnologías: el procesado de datos desde un formato (generalmente digital) a otro (generalmente físico). Las funciones del Correo Híbrido, son transformar datos desde el formato: digital al formato físico. Posibilita que personas individuales o empresas envíen electrónicamente sus cartas, informes y facturas al servicio de correos, y éste se encargue de imprimir y compaginar los documentos, ponerlos en un sobre, cerrarlo, franquearlo y enviarlo.

Las administraciones postales cuentan con infraestructura para distribuir mensajería y paquetería desde y hacia la mayoría de los lugares. Las empresas que generan grandes volúmenes de información destinada a sus clientes, no están preparadas para asumir su distribución. Es por ello que el Correo Híbrido es una opción de continuar su desarrollo y expansión, ya que la responsabilidad de impresión y traslado de la información recae sobre la administración postal. La empresa contratante solo tiene que preocuparse de entregar la información digital a la planta de CHM con la que establezca el contrato, esto le ahorra recursos en materiales de impresión y distribución porque solo tiene que trasladar datos digitales hasta un solo lugar y no gran cantidad de información impresa hasta cada uno de sus clientes (RODRÍGUEZ 2007).

1.3 Posibles clientes del Correo Híbrido Masivo

Toda empresa necesita tener comunicación con su entorno, ya sea internamente (entre trabajadores o asociados) o externamente, con sus clientes o mercado. El envío de comunicados, invitaciones, cartas, facturas y publicidad, entre otros, es un elemento importante para que esta comunicación se mantenga estable y haga de la empresa una organización exitosa. Por lo que el Correo Híbrido es un servicio útil para todo tipo de empresa que genere grandes volúmenes de envíos.

1.4 Criptografía

La criptología surge a raíz de la necesidad de poder enviar mensajes de forma que solo fueran entendidos por sus receptores y emisores, hizo que se crearan sistemas de cifrado, de forma que un mensaje después de un proceso de transformación, que se denomina cifrado, solo pudiera ser leído si se conocía el mismo proceso pero a la inversa, denominado descifrado.

Capítulo 1: Fundamentación teórica

El surgimiento de redes de comunicación, en particular de Internet, ha abierto nuevas posibilidades para el intercambio de información. Al mismo tiempo, son cada vez mayores las amenazas a la seguridad de la información que se transmite. Es necesario entonces, crear diferentes mecanismos, dirigidos a garantizar la confidencialidad y autenticidad de los documentos electrónicos, todo ello es parte de una nueva tecnología denominada Criptografía (MARRERO 2003).

La palabra criptología proviene de las palabras griegas krypto y logos y significa estudio de lo oculto. Una rama de la criptología es la criptografía (de las palabras krypto y graphos que significa descripción), que se ocupa del cifrado de mensajes. Esta se basa en que el emisor crea un mensaje en claro, que es tratado mediante un cifrador con la ayuda de una clave, para crear un texto cifrado. Este texto cifrado, por medio del canal de comunicación establecido, llega al descifrador que convierte el texto cifrado, apoyándose en otra clave, para obtener el texto en claro original. Las dos claves implicadas en el proceso de cifrado/descifrado pueden ser o no iguales dependiendo del sistema de cifrado utilizado.

La criptografía, de acuerdo con la Real Academia Española, es "la técnica de escribir con claves secretas o de un modo enigmático". En informática se define como "el arte y la ciencia de mantener seguros archivos y mensajes". Se usan los términos codificar/descodificar y cifrar/descifrar como sinónimos de encriptar/desencriptar.

En los procesos de almacenamiento y transmisión de la información es de primordial importancia la seguridad. En el almacenamiento, el peligro latente suele ser el robo del soporte del mensaje o simplemente el acceso no autorizado a esa información, mientras que en las transmisiones lo es la intervención del canal (MORENO 2008).

1.4.1 Encriptación

La encriptación es un conjunto de técnicas que intentan hacer inaccesible la información a personas no autorizadas. Por lo general, la encriptación se basa en una clave, sin la cual la información no puede ser descifrada. Encriptar datos tiene dos principales objetivos: la confidencialidad, para mantener la información en secreto, y la integridad, para evitar que la información se destruya o sea corrompida.

1.4.2 Tipos de cifrado de información

Existen dos tipos de cifrado el simétrico y el asimétrico. El cifrado simétrico realiza cálculos más sencillos que el asimétrico lo que generalmente lo hace más rápido, y usualmente utiliza claves no

Capítulo 1: Fundamentación teórica

perdurables. El cifrado asimétrico realiza cálculos costosos, no se usa para grandes volúmenes de información, utiliza claves con validez para varios años.

Cifrado Simétrico

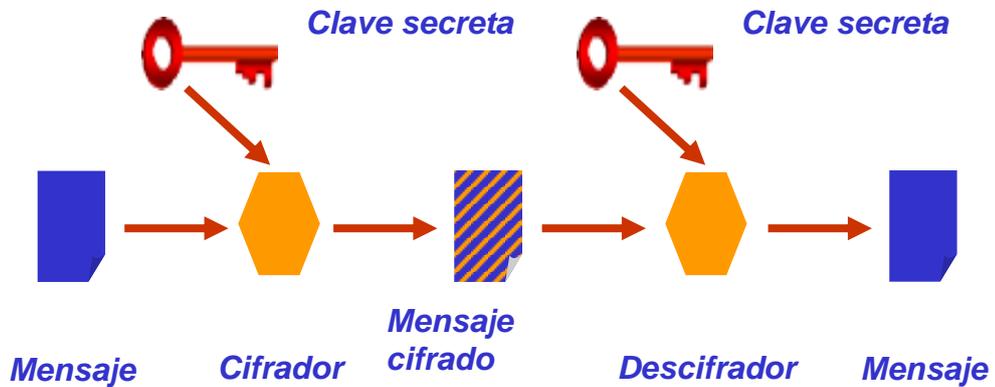


Figura 1.1. Cifrado simétrico.

Cifrado Asimétrico

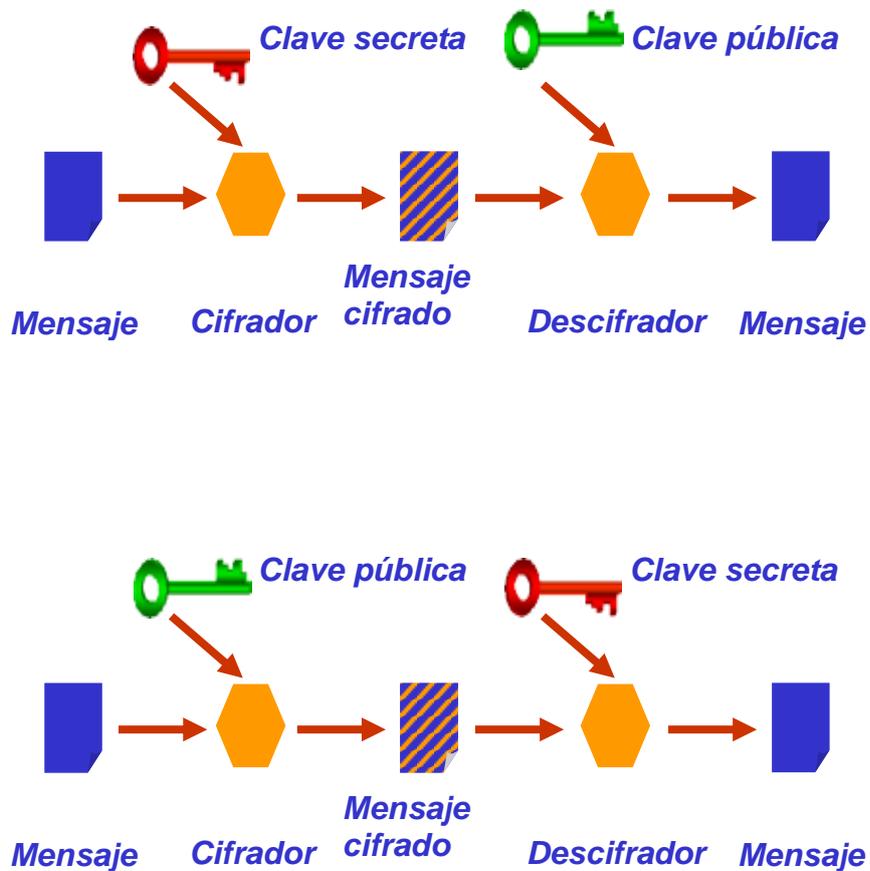


Figura 1.2. Cifrado asimétrico.

Dependiendo de estas características para ambos tipos de cifrado se decide restringir la selección a los algoritmos que emplean el cifrado simétrico. Esto se debe a que se prevé el procesamiento de grandes volúmenes de información en las diferentes empresas cliente y evitando posibles problemas de rendimiento de su hardware (VIRUEGA 2006).

1.4.3 Métodos y técnicas de encriptación

➤ DES

El Algoritmo de encriptación DES trabaja con clave simétrica, fue desarrollado en 1977 por la empresa IBM, se basa en un sistema monoalfabético, con un algoritmo de cifrado consistente en la aplicación sucesiva de varias permutaciones y sustituciones. Inicialmente el texto a cifrar se somete a una permutación, con bloque de entrada de 64 bits (o múltiplo de 64), para posteriormente ser sometido a la acción de dos funciones principales, una función de permutación con entrada de 8 bits y otra de sustitución con entrada de 5 bits, en un proceso que consta de 16 etapas de cifrado.

En general, DES utiliza una clave simétrica de 64 bits, de los cuales 56 son usados para la encriptación, mientras que los 8 restantes son de paridad, y se usan para la detección de errores en el proceso. Es rápido y fácil de implementar (GUMPERT 2007).

➤ Triple DES

Al algoritmo que hace triple cifrado del DES también es conocido como TDES o 3DES, fue desarrollado por IBM en 1978. El Triple DES con una clave de 128 bits y que es compatible con el DES y mucho más robusto. Este nuevo algoritmo toma una clave de 128 bits y la divide en dos de 64 bits cada una, de la siguiente forma:

Se le aplica al documento a cifrar un primer cifrado mediante la primera clave, C1.

Al resultado (denominado ANTIDES) se le aplica un segundo cifrado con la segunda clave, C2.

Y al resultado se le vuelve a aplicar un tercer cifrado con la tercera clave, C3 (GUMPERT 2007).

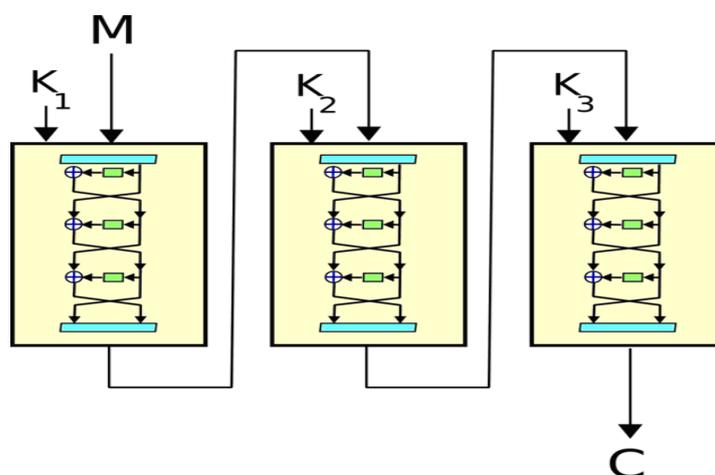


Figura 1.3. Algoritmo Triple DES.

1.5 Niveles de seguridad en las herramientas de manejo de información

Las empresas que se dedican a la creación y comercialización de Software, desarrollan sus productos cumpliendo con modelos de calidad como SPICE, CMM, CMMI. La aplicación de estos estándares en los procesos de desarrollo de software incorporan pruebas al software que aseguran la calidad y el cumplimiento de las necesidades del cliente, esto garantiza que el producto final tenga un elevado nivel de seguridad en dependencia de los niveles críticos del software (CAMACHO y FUENTES 2007).

Los sistemas de Correo Híbrido deben garantizar la seguridad de los datos que los clientes ponen en sus manos ya que el impositor debe enviar a la entidad que brinda el servicio los datos de sus usuarios que son en la mayoría de las ocasiones el “corazón” de su negocio (como facturas, datos personales de sus clientes, etc.), es por ello que se requiere un alto nivel de seguridad y control en la transmisión y manipulación de los mismos, de modo que se pueda garantizar al cliente, que el destinatario final recibirá una información fiel a la que suministró.

1.6 Análisis de herramientas de procesamiento de datos existentes

Las empresas que manejan grandes e importantes volúmenes de información, con frecuencia suelen crear Centros de Procesamiento de Datos (CPD) con el objetivo de tener acceso a la información necesaria para sus operaciones. Por ejemplo, un banco puede tener un CPD con el propósito de almacenar todos los datos de sus clientes y las operaciones que estos realizan sobre sus cuentas. Prácticamente todas las compañías que son medianas o grandes tienen algún tipo de CPD, mientras que las más grandes llegan a tener varios. En estos términos se hace muy importante la

Capítulo 1: Fundamentación teórica

protección física de los equipos informáticos o de comunicaciones implicados, así como servidores de bases de datos que puedan contener información crítica (WIKIPEDIA 2008a).

Por tal motivo existen aplicaciones informáticas para gestionar los datos de estos CPD que por el carácter confidencial de la información con la que trabajan deben cumplir una serie de requisitos. A continuación se describen características de algunas aplicaciones de este tipo.

1.6.1 Agata Report

Agata Report: Conjunto de herramientas para extraer lo más sustantivo de las bases de datos. En versión libre, permite extraer datos de PostgreSQL, MySQL, Oracle, DB2, MS-SQL, Informix, InterBase, Sybase, o Frontbase y exportarlas a PostScript, texte, HTML, XML, PDF, o CSV.

Entre otras funcionalidades interesantes, Agata Report da la posibilidad de generar un completo diagrama ER en formato DIA, proporciona un módulo para crear gráficas en curvas o en bastones y permite cruzar informes con otras bases de datos. Además, el programa soporta parámetros temporales de ejecución. Este programa no cuenta con métodos o funcionalidades que protejan la información (REPORT 2007).

1.6.2 MSD Caja Fuerte Gestor de documentos y ficheros privados

Algunas de las principales características de esta herramienta: Garantiza la confidencialidad y seguridad de los datos, mediante clave de acceso y encriptación, la información se protege durante la ejecución del programa, al guardar la información en el disco tras la ejecución y al realizar las copias de seguridad. Si no se conoce la clave con la que se encriptaron los ficheros, no hay forma de recuperarlos.

Este programa utiliza varias técnicas para garantizar la integridad de los datos guardados en una base de datos. Por ejemplo, si el usuario decide borrar un fichero del disco después de guardarlo en la base de datos, el programa comprueba que es capaz de extraer el fichero completamente y que el fichero extraído es idéntico al fichero original antes de borrarlo del disco. Además, si se guardan ficheros en la base de datos durante una sesión, el programa solicitará la realización de una copia de seguridad antes de cerrarse.

Cuenta con múltiples claves de encriptación, por defecto, el programa encripta los ficheros guardados en la base de datos con la clave de acceso. Si el usuario lo desea, puede utilizar una clave distinta para encriptar cada fichero que guarde en el programa. Además comprime los ficheros antes de encriptarlos para que ocupen menos espacio en la base de datos.

Capítulo 1: Fundamentación teórica

Durante la ejecución del programa: se solicita una contraseña para abrirlo, para abrir los módulos o para restaurar el programa desde la bandeja del sistema. Después de cerrar el programa, es empaquetada toda la información en un único fichero protegido por una contraseña (SEBARES 2008).

1.7 Tecnologías de desarrollo propuestas para la solución

Por la documentación consultada en Internet, se pudieron conocer las principales características de las diferentes herramientas libres existentes para el desarrollo de software; logrando conocer de esta manera las principales ventajas de cada una de ellas; esta búsqueda brindó argumentos que permitieron escoger las herramientas.

1.7.1 Software libre

El Software libre es aquel en que el usuario tiene la libertad de una vez que lo obtiene, distribuirlo con o sin modificaciones. El Software libre tiene esa denominación, debido a que posee cuatro libertades básicas:

- La libertad de usar el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y adaptarlo a sus necesidades (libertad 1).
- El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2) (FOUNDATION 2008).

1.7.2 Java como lenguaje de programación

Java es un lenguaje de programación orientado a objetos. Las aplicaciones Java están típicamente compiladas en un bytecode, aunque la compilación en código máquina nativo también es posible. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel como punteros.

Una de sus mejores cualidades es que un programa desarrollado y compilado en una plataforma pueda ejecutarse en otra sin modificaciones.

Java es un lenguaje de programación de alto nivel que tiene las siguientes características:

- Orientado a objetos.
- Distribuido y dinámico.
- Robusto.

- Seguro.
- Multitarea.
- Portable (ROMÁN 2007).

Otra de las utilidades de la programación en Java es el desarrollo de aplicaciones, que son programas que se ejecutan en forma independiente, es decir con la programación Java, se pueden realizar aplicaciones como un procesador de palabras, una hoja que sirva para cálculos, una aplicación gráfica, etc. En resumen cualquier tipo de aplicación se puede realizar con ella.

La programación en Java, permite el desarrollo de aplicaciones bajo el esquema de Cliente Servidor, como de aplicaciones distribuidas, lo que lo hace capaz de conectar dos o más computadoras u ordenadores, ejecutando tareas simultáneamente, y de esta forma logra distribuir el trabajo a realizar (PROGRAMACIÓN 2006).

1.7.3 NetBeans

Se propone como plataforma para el desarrollo de una aplicación de escritorio usando Java por ser un entorno integrado de desarrollo (IDE). Esta plataforma permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos.

La plataforma ofrece servicios comunes a las aplicaciones de escritorio, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. Entre las características de la plataforma están:

- Administración de las interfaces de usuario (ej. menús y barras de herramientas).
- Administración de las configuraciones del usuario.
- Administración del almacenamiento (guardando y cargando cualquier tipo de dato).
- Administración de ventanas.
- Framework basado en asistentes (diálogo paso a paso) (WIKIPEDIA 2008b).

1.8 Metodología utilizada

Para desarrollar cualquier software es necesario guiar el proceso a través de una metodología, la cual será la encargada de elaborar “el plano” sobre el cual se apoyará el equipo de desarrollo. En la actualidad existen diferentes metodologías por las que se guía el desarrollo del software entre ellas se encuentra: RUP como la más utilizada. RUP utiliza el UML como lenguaje de representación visual.

1.8.1 Lenguaje Unificado de Modelado (UML)

El UML es un lenguaje para visualizar, especificar, construir y documentar los elementos que componen un sistema con gran cantidad de software y tecnología orientada a objetos. Fue creado por los estudiosos de la Ingeniería de Software: Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1995. Desde sus comienzos se convirtió en el estándar internacional para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos.

Actualmente UML es el estándar para el diseño orientado a objetos, ya que es el resultado de la unión de las mejores cualidades de los tres lenguajes existentes que le dieron paso por el trabajo en conjunto de sus autores.

A partir del surgimiento de UML, muchas de las metodologías existentes han sido adaptadas para utilizar este lenguaje, como es el caso de la Metodología de Análisis y Diseño Orientado a Objetos de Sistemas Informáticos en su versión 5.0 y en otras como el Proceso Unificado de Desarrollo se concibió desde sus inicios utilizar UML.

La decisión de utilizar UML como lenguaje para modelar el sistema de software se debe a que se ha convertido en un estándar con las siguientes características:

- Permite modelar sistemas utilizando técnicas orientadas a objetos (OO).
- Permite especificar las decisiones de análisis y diseño, construyéndose modelos precisos y completos.
- Está compuesto por diversos elementos gráficos que se combinan para conformar diagramas, además cuenta con reglas para combinar dichos elementos.
- Es independiente del lenguaje de programación y de las características de los proyectos, ya que fue diseñado para modelar cualquier tipo de proyecto.
- Es un lenguaje suficientemente expresivo para manejar los conceptos que se originan en un sistema moderno, tales como la distribución física, así como también los mecanismos de la ingeniería de software, como son el encapsulamiento y los componentes.
- Integra las mejores prácticas de los lenguajes de modelación existentes.
- A pesar de ser un lenguaje potente, es fácil de aprender y de usar.
- Permite documentar los artefactos de un proceso de desarrollo (MARQUEZ).

1.8.2 Proceso Unificado de Rational (RUP)

El RUP es una metodología de desarrollo de software cuyo objetivo y tendencia actual es su utilización para la construcción de sistemas más grandes, más completos y de más calidad. Se

Capítulo 1: Fundamentación teórica

considera un proceso para el desarrollo de software orientado a objeto que hace uso del UML como lenguaje de modelado. RUP fue creado en el año 1998 por el mismo grupo de trabajo que creó el UML, Ivar Jacobson, Grady Booch y James Rumbaugh, bajo el nombre de Proceso Unificado de Rational, de ahí las siglas con las que se identifica a este proceso de desarrollo (BARRIENTOS).

En RUP se han agrupado las actividades en grupos lógicos definiéndose 9 flujos de trabajo principales, los 6 primeros son conocidos como flujos de ingeniería (Modelado del negocio, Requerimientos, Análisis y diseño, Implementación, Prueba, Instalación) y los tres últimos de apoyo (Administración del proyecto, Administración de configuración y cambios, Gestión de entorno).

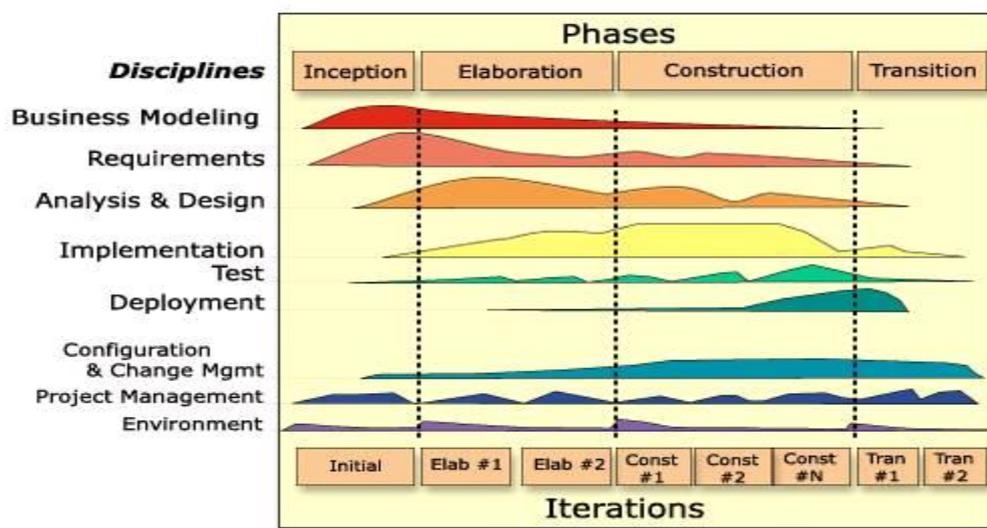


Figura 1.4. Fases del RUP.

Un proyecto realizado siguiendo RUP se divide en cuatro fases: Inicio (puesta en marcha), Elaboración (definición, análisis, diseño), Construcción (implementación) y Transición (fin del proyecto y puesta en producción). En cada fase se ejecutarán una o varias iteraciones (de tamaño variable según el proyecto), dentro de cada una de ellas seguirá un modelo de cascada para los flujos de trabajo con sus respectivas actividades.

RUP se caracteriza por ser un proceso dirigido por casos de uso (que avanza a través de los flujos de trabajo que parten de los casos de uso y estos son el instrumento para validar la arquitectura del software y extraer los casos de prueba), está centrado en la arquitectura (los modelos son proyecciones del análisis y el diseño, constituye la arquitectura del producto a desarrollar) y es iterativo e incremental (durante todo el proceso de desarrollo se producen versiones incrementales del producto en desarrollo, que se acercan al producto terminado).

Capítulo 1: Fundamentación teórica

Se decide utilizar RUP como metodología para desarrollar el software ya que presenta las siguientes características:

- Unifica los mejores elementos de las restantes metodologías.
- Preparado para desarrollar grandes y complejos proyectos.
- Orientado a Objetos.
- Utiliza el UML como lenguaje de representación visual.
- Muy organizado y con documentación.
- Define las actividades, roles y artefactos para cada flujo de trabajo.
- Iterativo incremental, dirigido por casos de uso y centrado en la arquitectura.

Además cubre el ciclo de vida de desarrollo de un proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de software, las cuales se muestran a continuación:

- Desarrollo de software en forma iterativa.
- Manejo de requerimientos.
- Utiliza arquitectura basada en componentes.
- Modela el software visualmente.
- Verifica la calidad del software.
- Controla los cambios.

1.8.3 Visual Paradigm

Visual Paradigm es una herramienta fácil de usar que soporta la última notación UML, ingeniería inversa, generación de código, exportación/importación XML, generador de informes, editor de figuras, plug-in, integración IDE con Eclipse, NetBeans y otros. Entre sus características se incluyen el modelado colaborativo con CVS y Subversion, interoperabilidad con modelos UML2 a través de XMI. Cuenta con una licencia gratuita y Comercial.

Características:

- Producto de calidad.
- Generación de código para Java.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.
- Es una herramienta diseñada para desarrollar software que utilizan la Programación Orientada a Objetos.

Capítulo 1: Fundamentación teórica

- Busca reducir la duración de un ciclo de desarrollo brindando ayuda a arquitectos, analistas, diseñadores y desarrolladores.
- Navegación intuitiva entre el modelo visual y el código.
- Poderosa herramienta de generación de PDF/HTML/DOC a partir de diagramas UML.
- Sincronización entre el código fuente y el modelo en tiempo real.
- Soporte para toda la notación UML.
- Presenta un diseño basado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Ofrece capacidades de ingeniería directa e inversa.
- Se integra al IDE NetBeans.

1.9 Conclusiones

Habiendo dominado los conceptos asociados al problema se da paso al modelamiento del negocio. Además la investigación llevada a cabo en este capítulo permitió seleccionar la metodología de desarrollo y el conjunto de herramientas que se utilizarán en la creación de la solución propuesta de la siguiente manera: Siguiendo la metodología RUP, realizando el análisis y diseño de la herramienta con los diagramas correspondientes generados con la herramienta case Visual Paradigm. La implementación se hará con el lenguaje de programación Java utilizando el IDE NetBeans.

CAPÍTULO 2: MODELO DE NEGOCIO

2.1 Introducción

Partiendo de los conocimientos adquiridos en el capítulo anterior se procede a modelar el negocio, el cual está enmarcado en dos procesos, la recepción de la información de los clientes, y la extracción de los datos antes de entrar en el proceso de impresión. Ambos procesos suceden en una planta de Correo Híbrido.

Este capítulo contiene la descripción sobre cómo se trabaja en dicha PCHM y en las empresas clientes. Se ilustra el negocio actual mediante un diagrama de casos de usos del negocio, identificándose los actores, trabajadores y las actividades a ser automatizadas.

2.2 Flujo actual de los procesos

Una vez que el negocio de Correo Híbrido Masivo establece un contrato con alguna empresa se le especifica al cliente que datos son necesarios para brindar el servicio con la calidad requerida:

En el acto son capturadas por el SCHM las direcciones de los clientes, el tipo de empresa que es, su objeto social etc. Para recibir el servicio, los clientes deberán entregar información que evidenciará la estructura de datos que usa, además de información básica de su negocio, que pudieran ser nombres, direcciones postales, cuentas, facturas, créditos entre otros. Estos datos podrían tener cualquier estructura y formato y serían diferentes para cada cliente.

Luego de capturada toda la información necesaria, se almacena en la planta por un período de tiempo que depende de la fecha de entrega fijada por el cliente. La información pasa por un proceso manual de clasificación para ser enviada a la planta más cercana al destino antes de continuar con el proceso de impresión.

2.3 Análisis crítico de la ejecución de los procesos

La forma en que los procesos son llevados a cabo en estos momentos alarga considerablemente el tiempo que emplea el cliente para determinar que datos debe llevar a la planta y el tiempo de procesamiento de los datos ya en la planta. Otro aspecto negativo es que media poca seguridad en cuanto al almacenamiento y traslado de los datos, lo que posibilita en muchos casos la pérdida de integridad de la información, o que sea susceptible a cualquier tipo de ataque informático, que restan prestigio a la institución y causan grandes pérdidas de clientes y económicas.

2.4 Modelo del negocio

El modelo del negocio describe el negocio en términos de casos de usos del negocio, que corresponde a lo que generalmente se le llama procesos.

La descripción del negocio propuesto en detalle tendrá entre sus actividades principales la identificación de los procesos de negocio, delimitación del modelo de casos de uso del negocio, la especificación de los casos de uso del negocio, la identificación de trabajadores y entidades del negocio que ejecutan las realizaciones de los casos de uso del negocio y detallar la definición de las entidades del negocio y las responsabilidades de los trabajadores del negocio.

2.4.1 Definición de los actores del negocio

Actor	Descripción
Cliente	Es la persona natural o jurídica que contrata el servicio con la entidad encargada de brindarlo. Es el generador de los datos que serán procesados en la planta, interactúa con el negocio, y trae los datos a la planta para su procesamiento.

Tabla 2.1. Descripción de los actores del negocio.

2.4.2 Definición de los trabajadores del negocio

Trabajador	Descripción
Gestor	Persona encargada de la interacción entre la PCH y sus clientes, sus responsabilidades son: coordinar las acciones necesarias para que el cliente reciba el servicio adecuado, para ello debe recepcionar los datos que trae el cliente y brindar cualquier información sobre el servicio solicitado.
Sistema de Correo Híbrido Masivo (SCHM).	Es el sistema que se encarga de procesar e imprimir los datos proporcionados por los clientes, para su posterior ingreso a la red de distribución del correo.

Tabla 2.2. Descripción de los trabajadores del negocio.

2.4.3 Diagrama de casos de uso del negocio

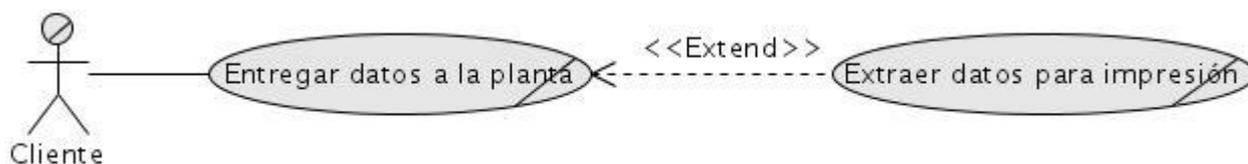


Figura 2.1. Diagrama de casos de uso del negocio.

2.4.4 Descripción textual de los casos de uso del negocio

Caso de Uso:	Entregar datos a la planta	
Actor:	Cliente (inicia)	
Trabajador:	Gestor	
Propósito:	Este caso de uso tiene como objetivo que los datos sean recepcionados en la PCHM.	
Resumen:	El CU inicia cuando el cliente extrae la información de sus bases de datos para solicitar el servicio de CHM, y los almacena en un dispositivo externo. El gestor los recibe si están correctos y los ingresa al SCHM.	
Caso de Uso asociado:	Extraer datos para impresión (extendido).	
Precondiciones:	Que el cliente se presente en la planta con el contrato y los datos que desee enviar.	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Negocio
	1. Extrae los datos de la BD de su negocio.	
	2. Crea los ficheros con la información que desee enviar a la Planta.	
	3. Traslada los datos a la planta de CHM.	4. El Gestor recibe los datos.
		5. El Gestor verifica si el contrato del cliente pertenece a la planta, si pertenece continúa con la acción 6 del flujo básico, si no pertenece ver flujo alterno 1.
		6. El Gestor verifica que los datos estén es buen estado para poder brindar el servicio con calidad, si los datos están correctos continúa con la acción 7 del flujo básico, si no ver flujo alterno 1.
		7. El Gestor elabora la Ficha, que contiene información sobre los ficheros de datos, el formato de la BD y la fecha de

Capítulo 2: Modelo de negocio

	entrega.
	8. El Gestor ingresa los datos a los servidores de la planta. 9. El SCHM acepta los datos. 10. El SCHM informa que la acción se ejecutó con éxito.
	11. El Gestor entrega comprobante e informa que la operación se realizó con éxito.
12. Recibe el comprobante y la información y se retira de la planta. 13. Finaliza la ejecución del caso de uso.	
Flujo Alterno 1: “No se brinda el servicio”	
Acción del Actor	Respuesta del Negocio
	a. El Gestor informa de los errores ocurridos.
	b. El Gestor devuelve los datos al cliente.
c. Recibe información y se retira de la planta. d. Finaliza la ejecución del caso de uso.	
Poscondiciones:	Los datos quedan almacenados en la BD de la PCHM

Tabla 2.3. Descripción textual del caso de uso del negocio Entregar Datos a la planta.

Caso de Uso:	Extraer datos para impresión
Caso de Uso Base:	Entregar datos a la planta
Actor:	Cliente(inicia)
Trabajador:	Gestor
Propósito:	Este caso de uso tiene como objetivo extraer los datos para impresión.
Resumen:	El CU inicia cuando el cliente solicita a la PCHM que imprima la información que entregó, atiende la solicitud el Gestor que extrae de los servidores de la PCHM la información para procesarla.
Precondiciones:	Que los datos estén almacenados en la BD de la PCHM.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1. El cliente solicita a la PCHM que imprima su información.	2. El Gestor extrae de la BD de la PCHM los datos a procesar.
	3. El Gestor verifica la integridad de los datos comparándolos con la ficha que contiene información de los datos. Si están correctos continúa con la acción 3 del flujo básico, si no ver

Capítulo 2: Modelo de negocio

	flujo alternativo 1.
	4. El Gestor clasifica manualmente los datos, según su criterio de dirección destino.
	5. El Gestor envía los datos a las plantas correspondientes para que continúe el proceso de impresión.
	6. El Gestor notifica al cliente el estado del proceso.
7. El cliente recibe la notificación y finaliza la ejecución del caso de uso.	
Flujo Alterno 1: “Informe de errores”	
Acción del Actor	Respuesta del Negocio
	3. El Gestor notifica al cliente de los problemas encontrados.
4. El cliente recibe la notificación y se presenta en la planta y finaliza la ejecución del caso de uso.	
Poscondiciones:	Los datos son enviados a las PCHM donde continuarán el proceso de impresión.

Tabla 2.4. Descripción textual del caso de uso del negocio Extraer datos para impresión.

2.4.5 Diagramas de actividades

Los casos de uso del negocio consisten en secuencias de actividades, que en conjunto producen algún resultado importante para el actor del negocio. El proceso consiste en un flujo básico de una o más alternativas de flujos. La estructura del flujo se describe gráficamente con la ayuda de un diagrama de actividad.

Un Diagrama de actividades ha sido diseñado para mostrar una visión simplificada de lo que ocurre durante una operación o proceso. Es un caso especial de un diagrama de estados en el cual casi todos los estados son estados de acción (identifican que acción se ejecuta al estar en él) y casi todas las transiciones son enviadas al terminar la acción ejecutada en el estado anterior.

El diagrama de actividades es un grafo de acciones que contiene los estados en que puede hallarse una actividad, puede contener bifurcaciones y describe un proceso que explora el orden de las tareas o actividades que logran los objetivos del negocio.

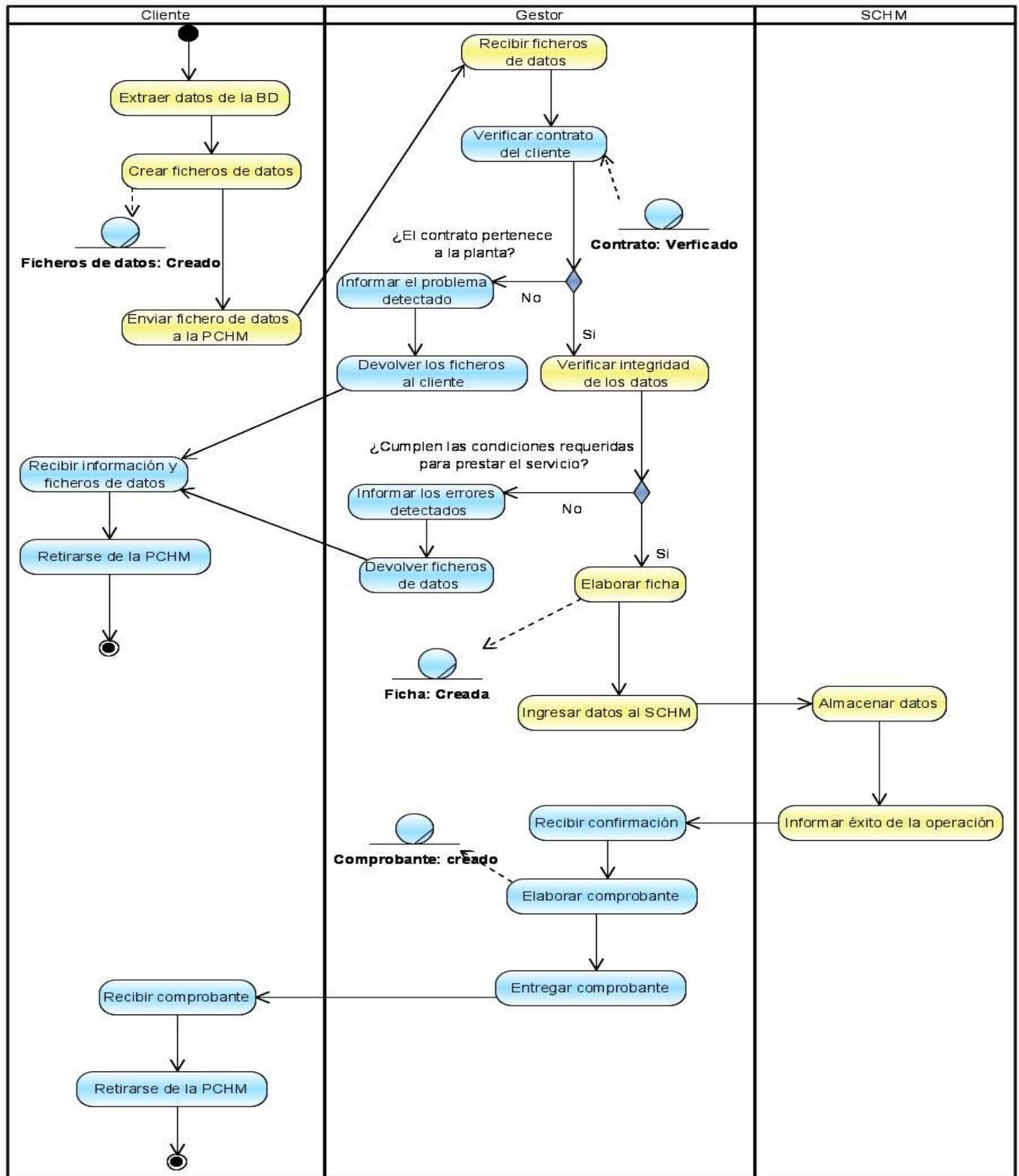


Figura 2.2. Diagrama de actividad: Entregar Datos a la planta.

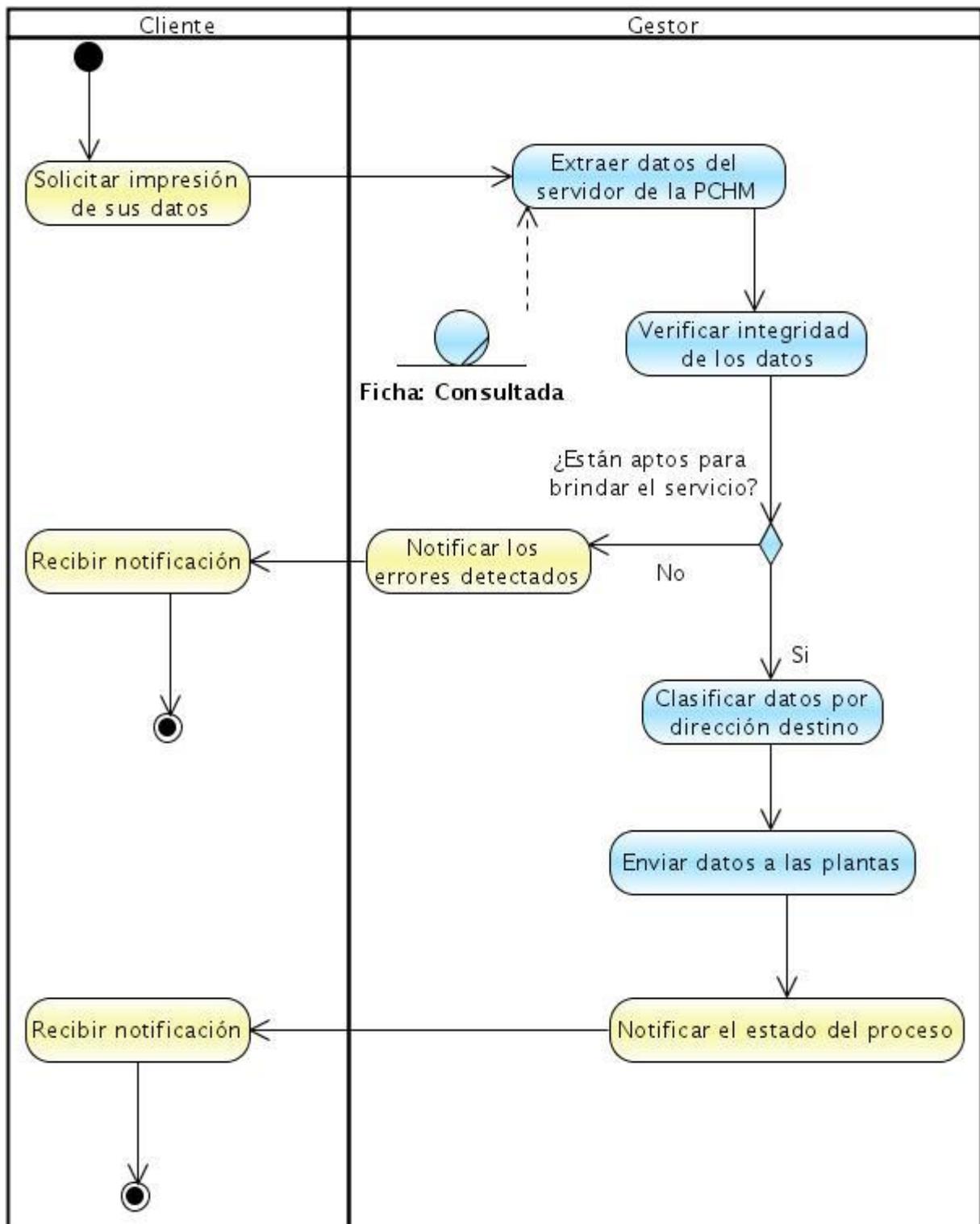


Figura 2.3. Diagrama de actividad: Extraer datos para impresión.

2.4.6 Diagrama de clases del modelo de objetos

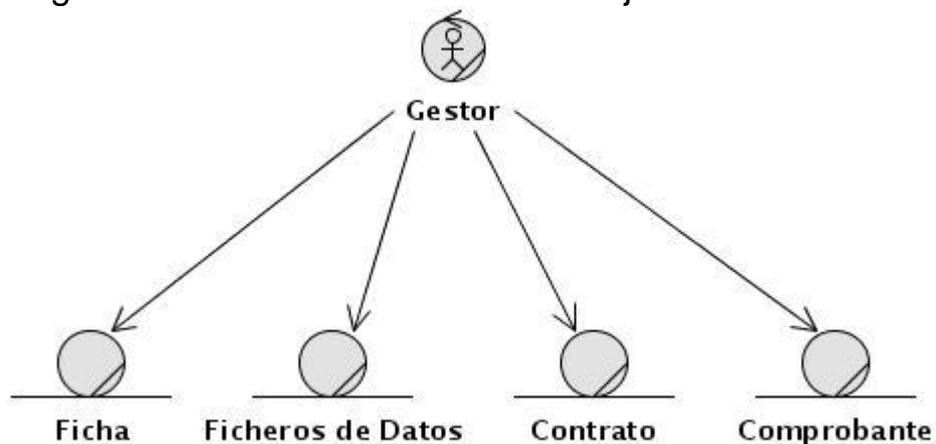


Figura 2.4. Diagrama de Objetos.

2.5 Conclusiones

En el capítulo dos se logró detallar a fondo el flujo actual de los procesos. Se realizaron los diagramas de casos de uso del negocio y los diagramas de actividades quedando definidos de esta forma los actores y trabajadores que intervienen en su ejecución, para conformar un conjunto de información que será la base para la detección de los requisitos contenidos en la propuesta de solución.

CAPÍTULO 3: CARACTERÍSTICAS DEL SISTEMA

3.1 Introducción

En este punto de la investigación se conoce la necesidad de contar con una aplicación de escritorio portable de fácil instalación ya que se prevé que funcione en la planta donde se brindará el servicio o se instale directamente en los negocios de aquellos clientes que así lo deseen. Por tal motivo la herramienta debe ser capaz de conectarse a varios gestores de bases de datos para extraer la información que se procesará. Existe gran número de Sistemas Gestores de Bases de Datos hoy en el mundo. Algunos libres como por ejemplo: PostgreSQL, MySQL, Firebird, SQLite, Sybase ASE. Otros propietarios como son: Microsoft Access, MySQL, IBM Informix, PervasiveSQL, Microsoft SQL Server, Oracle, entre muchos otros. Se decide trabajar en base a conexiones a SQL Server, Oracle, PostgreSQL y MySQL ya que sus características, los convierten en los más utilizados por los clientes a los que se les pretende brindar el servicio de Correo Híbrido Masivo.

3.2 Especificación de los requisitos de software

Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir. Mientras que los requerimientos no funcionales son las propiedades o cualidades que el producto debe tener, debe pensarse en estos atributos como las características que hacen al producto atractivo, usable, rápido o confiable.

3.2.1 Definición de los requisitos funcionales

- 1 Proporcionar los permisos necesarios para cada operario de la aplicación según sus roles y ubicación.
- 2 Ser capaz de conectarse a varios sistemas gestores de bases de datos (SGBD).
 - 2.1 Los SGBD serán:
 - PostgreSQL.
 - MySQL.
 - Oracle.
 - SQL Server.
 - 2.2 Configurar conexiones a los SGBD.
 - 2.3 Cargar una conexión a SGBD existente.

Capítulo 3: Características del sistema

- 3 Crear nuevos usuarios y sus contraseñas para el uso en los negocios de clientes
 - 3.1 Asignar roles a los usuarios creados.
- 4 Generar archivos con los datos que serán procesados en las Plantas de Correo Híbrido Masivo (PCHM).
 - 4.1 Permitir la selección del rango entre fechas (fecha inicio y fecha fin) en que se desee obtener los datos.
 - 4.2 Generar un archivo para cada PCHM que contenga el negocio atendiendo al campo región de las direcciones proporcionadas por el cliente.
 - 4.3 Encriptar los archivos generados, utilizando algún algoritmo de encriptación estándar.
 - 4.4 Compactar los archivos encriptados.
- 5 Obtener datos planos a partir de un fichero encriptado y compactado.
 - 5.1 Descompactar y desencriptar cualquier fichero generado previamente en la propia herramienta.
- 6 Generar un resumen con las propiedades de un archivo generado.
 - 6.1 Listar las propiedades:
 - Nombre del fichero.
 - Tamaño del fichero.
 - Cantidad de líneas del fichero.
 - Tipo de fichero (si es generado o no por la aplicación).
 - 6.2 Exportar el resumen a un archivo de texto.
- 7 Gestionar diferentes archivos de configuración necesarios para que la aplicación se pueda usar tanto en la PCHM (*online*) como en las ubicaciones físicas de cada cliente individual (*offline*).

3.2.2 Definición de los requisitos no funcionales

Apariencia o interfaz externa

- Todas las interfaces de usuario que se definan para la herramienta, respetarán los patrones de diseño establecidos para la organización.
- Las ventanas de la herramienta contendrán claro y bien estructurados los datos, y al mismo tiempo permitirán la interpretación correcta e inequívoca de la información.
- Mostrar mensajes de errores en la introducción de datos de una forma sencilla y explicativa, la entrada de datos incorrecta será detectada claramente por la herramienta.
- Mostrar todos los textos y mensajes en pantalla en español.
- Diseñar su funcionamiento de modo que sea intuitivo, y requiera de información mínima.

Capítulo 3: Características del sistema

Portabilidad

- La herramienta está construida para operar en cualquier plataforma, realizado con un lenguaje libre.

Seguridad

- En las PCHM la herramienta accederá a la Base de Datos del sistema web, para realizar la autenticación de los usuarios que correspondan a los roles establecidos (jefe de la planta, diseñador, gestor).
- En los clientes la herramienta se autenticará a partir de un fichero previamente configurado en el negocio de CHM.

Confidencialidad

- La información manejada por la herramienta estará protegida de acceso no autorizado y divulgación.

Integridad

- La información manejada por la herramienta será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos.

Confiabilidad

- Se garantizará la consistencia de los datos, se realizarán comprobaciones y validaciones cada vez que sea necesaria la validación de los datos.

Ayuda y documentación

- Entregar documentos técnicos y las guías de usuario, que incluyen presentaciones realizadas en cada tema.
- Entregar carpeta del proyecto, con la documentación técnica generada en el desarrollo para la especificación del sistema.
- Se deberá entregar el código fuente al propietario del negocio CHM.

Hardware

- Mínimo PC Pentium 3, 256 Mb de Ram, 2GB de espacio libre en disco duro.

Software

- SO Windows XP o superior
- SO linux (probado en Ubuntu 7.10)
- Máquina Virtual de Java (JDK 1.6 o superior)

3.3 Modelo del sistema

Es una representación gráfica del entorno del sistema (actores) y su funcionalidad principal (casos de uso). Un diagrama de casos de uso muestra, por tanto, los distintos requisitos funcionales que se esperan de una aplicación o sistema y cómo se relaciona con su entorno (usuarios u otras aplicaciones).

3.3.1 Definición de los actores del sistema

El actor es una entidad externa del sistema que de alguna manera participa en la historia del caso de uso. Por lo regular estimula el sistema con eventos de entrada o recibe algo de él (LARMAN 1999).

Actor	Justificación
Usuario	Actor abstracto, generalización utilizada para autenticarse en el sistema del que heredan el resto de los usuarios de la aplicación.
Gestor	Persona encargada de la interacción entre la PCH y sus clientes, sus responsabilidades son: Generar los ficheros con datos, encriptarlos, desencriptarlos y generar un resumen de las propiedades del fichero.
Súper Cliente	Persona perteneciente a la entidad que contrata el servicio, con permisos de administración de la aplicación.
Administrador de Planta	Trabajador perteneciente a la administración postal, con permisos de administración de la aplicación.
Operador Avanzado	Actor abstracto, generalización utilizada para dar funcionalidades de administración a los actores que heredan de él: <ul style="list-style-type: none">➤ Súper cliente.➤ Administrador de la planta.
Técnico	Persona perteneciente a la entidad que contrata el servicio

Capítulo 3: Características del sistema

	con las mismas responsabilidades que el Gestor.
Administrador de BD	Persona perteneciente a la entidad que contrata el servicio con la capacidad de realizar las mismas acciones que el Gestor y el Técnico y además la responsabilidad de configurar la conexión a la base de datos del cliente.
Operador Simple	Actor abstracto, generalización utilizada para dar las funcionalidades de generar los ficheros con datos, encriptarlos, desencriptarlos y generar un resumen de las propiedades del fichero a los actores que heredan de él: <ul style="list-style-type: none"> ➔ Gestor ➔ Técnico

Tabla 3.1. Descripción de los actores del sistema.

3.3.2 Definición de los casos de uso del sistema

Un caso de uso es un fragmento de funcionalidad del sistema que proporciona al usuario un resultado significativo.

Listado de casos de uso del sistema:

CU #0 Autenticar Usuario: Otorga permisos a los usuarios según su rol.

CU #1 Administrar usuarios clientes: Permite crear usuarios con diferentes responsabilidades.

CU #2 Configurar conexión a BD: Permite configurar la conexión de la Base de datos del cliente.

CU #3 Desencriptar: Desencripta los datos.

CU #4 Generar archivos con datos: Genera ficheros de datos clasificados, compactados y encriptados.

CU #5 Generar resumen de propiedades de ficheros: Muestra las propiedades de un fichero seleccionado.

3.3.3 Diagrama de casos de uso del sistema

Los casos de uso representan los requisitos funcionales. Todos los casos de uso juntos constituyen el modelo de casos de uso el cual describe la funcionalidad total del sistema. A continuación se muestra el diagrama de casos de uso del sistema, siguiendo el criterio de agrupamiento por

Capítulo 3: Características del sistema

funcionalidad de los actores, estos no son más que los casos de usos requeridos para dar soporte a un determinado actor del sistema.

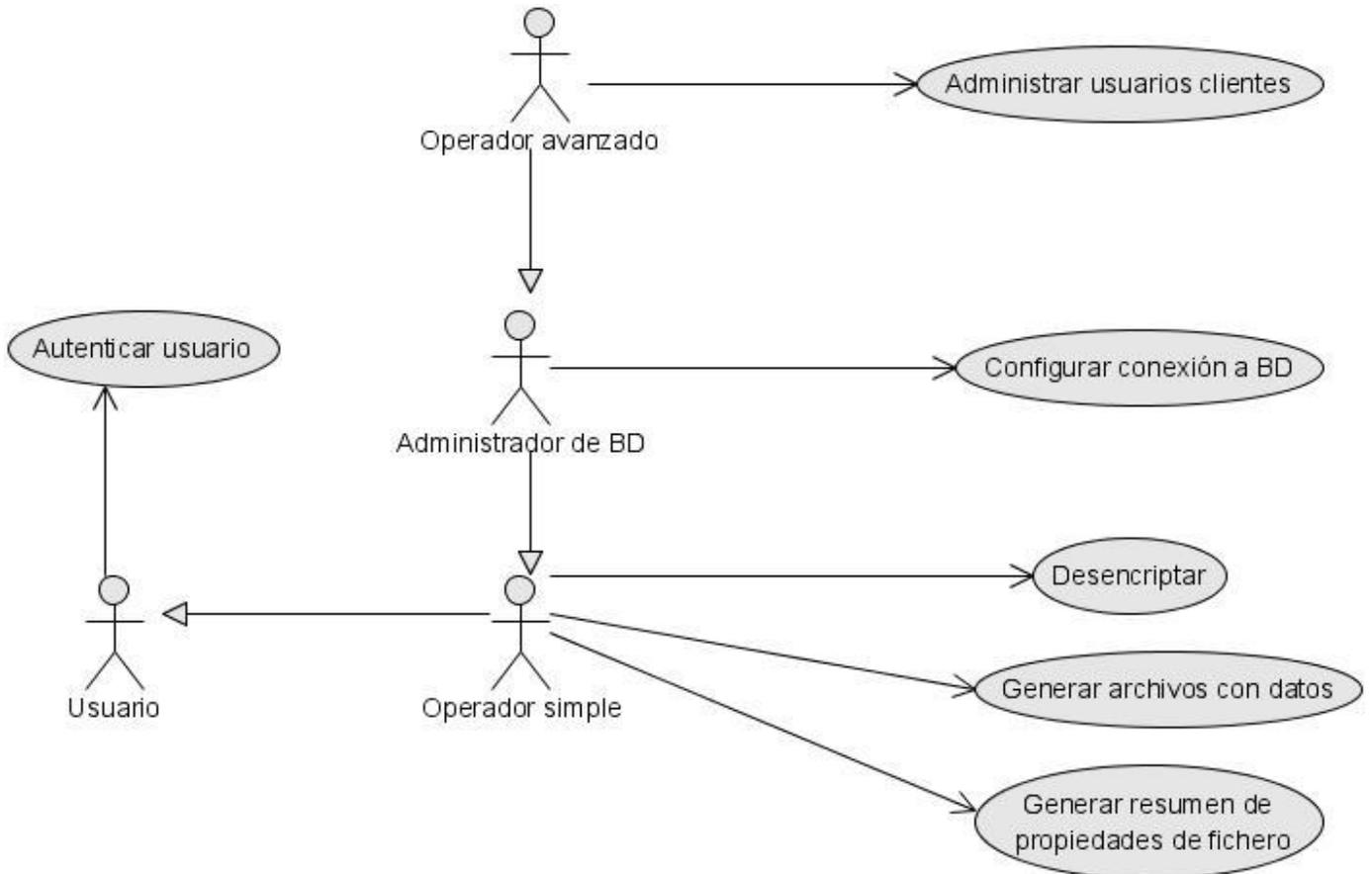


Figura 3.1. Diagrama de Casos de uso del sistema.

3.3.4 Descripción expandida de los casos de uso del sistema

Para entender la funcionalidad asociada a los casos de uso no es suficiente con la representación gráfica del Diagrama de casos de uso. La descripción extendida brinda los detalles de la secuencia de las acciones, las precondiciones como estado inicial, los posibles estados finales como poscondiciones, además de cuando comienza y termina el caso de uso. Describe explícitamente que debe hacer el sistema, separando las responsabilidades del sistema y la de los actores.

A continuación se muestra la descripción expandida de dos casos de uso fundamentales del sistema, el resto se encuentra en el **ANEXO 1**.

Capítulo 3: Características del sistema

Caso de uso:	Configurar conexión a BD (CU-2)
Actor:	Administrador de BD (inicia)
Propósito:	Permitir al administrador configurar la conexión a BD.
Resumen:	El caso de uso se inicia cuando el usuario elige la opción “Acceso a datos” la herramienta muestra un formulario con los datos a llenar, el usuario llena los campos con los datos y presiona el botón “Guardar” terminando de esta forma el caso de uso.
Referencias:	RF 2
Precondiciones:	El administrador de BD debe estar registrado en el sistema.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario escoge la opción “Acceso a datos”.	2. Opciones: a. Si no existe una configuración guardada muestra el formulario para crear una configuración. Con los campos: <ul style="list-style-type: none"> ➤ Gestor de base de datos. ➤ Nombre o IP del servidor. ➤ Puerto. ➤ Nombre de la BD. ➤ Usuario de la BD. ➤ Contraseña de usuario. Seguido de los botones “Guardar”, “Probar conexión”, “Cancelar”. b. Si ya existe una configuración guardada ir al Flujo Alternativo 1
3. Llena los campos y selecciona la opción deseada.	4. Gestiona la selección : a. Si selecciona “Guardar” verifica la validez de los datos, si son válidos muestra un mensaje (“la configuración se ha guardado satisfactoriamente”) guarda la configuración, ir al paso 5 del flujo básico, si no ir al Flujo Alternativo 2. b. Si selecciona “Probar conexión” verifica la validez de los datos si son válidos trata de conectarse a la BD. c. Si selecciona “Cancelar” va al paso 5 del flujo básico.
	5. Termina el caso de uso.
Flujo Alternativo 1: “Configuración guardada”	
Acción del Actor	Respuesta del Sistema

Capítulo 3: Características del sistema

	2. Muestra un mensaje (“Existe una configuración ¿Desea cargarla?”) y los botones “Si”, “No” y “Cancelar”.
3. Selecciona la opción deseada.	4. Gestiona la selección: a. Si seleccionó “Si” Muestra una Form con los datos de la configuración, y los botones “Editar” y “Aceptar”. b. Si selecciona “No” ir al paso 1 del Flujo básico. c. Si selecciona “Cancelar” ir al paso 5 Flujo básico.
5. Selecciona la opción deseada.	6. Gestiona la selección. a. Si selecciona “Aceptar” ir al paso 5 del Flujo básico. b. Si selecciona “Editar” ir al paso 1 del Flujo básico.
Flujo Alterno 2: “Validar datos”	
Acción del Actor	Respuesta del Sistema
	4. Muestra mensajes de error según el error que exista en los datos. Ejemplos: “Debe llenar todos los campos”, “El nombre o IP del servidos es incorrecto”, “El número de puerto es incorrecto”. Ir al paso 4 del Flujo básico.
Poscondiciones de Éxito:	Se configuró el acceso a datos.
Poscondiciones de Fallo:	No se configuró el acceso a datos.

Tabla 3.2. Descripción textual del caso de uso del sistema Configurar Conexión a BD.

Caso de uso:	Generar archivos con datos (CU-4)
Actor:	Operador simple (inicia)
Propósito:	Permitir generar archivos con datos.
Resumen:	El caso de uso se inicia cuando el operador simple escoge la opción generar ficheros con datos, escoge el método de encriptación y encripta dichos datos concluyendo de esta forma el caso de uso.
Referencias:	RF 4
Precondiciones:	Que el operador simple esté autenticado.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción “Generar ficheros con datos”.	2. Muestra el formulario para crear los script de datos con las opciones de seleccionar la fecha inicial, fecha final y la solicitud deseada. Seguido de los botones “Aceptar” y “Cancelar”.
3. Selecciona las fechas de inicio y fin y la	4. Ejecuta una de las siguientes acciones:

Capítulo 3: Características del sistema

solicitud y presiona el botón deseado.	<ul style="list-style-type: none"> a. Si selecciona la opción “Aceptar” seguir el flujo básico. b. Si selecciona la opción “Cancelar” ir al paso 11 del flujo básico.
	5. Muestra un combobox con los métodos de encriptación.
6. Selecciona el método de encriptación deseado.	7. Muestra los campos a llenar con las contraseñas en dependencia del método escogido. Seguido de los botones “Encriptar” y “Cancelar”.
8. Presiona el botón deseado.	9. Ejecuta una de las siguientes acciones: <ul style="list-style-type: none"> a. Si selecciona el botón “Aceptar” sigue en el flujo básico. b. Si selecciona el botón “Cancelar” regresa al paso 4 del flujo básico.
10. Introduce la contraseña para encriptar los datos.	11. Verifica que las contraseñas cumplan con los requisitos. <ul style="list-style-type: none"> a. Si cumple con los requisitos entonces, encripta, clasifica por plantas y comprime los datos, ir al paso 11 del flujo básico. b. No cumple con los requisitos entonces ir al Flujo alterno 1.
	12. Termina el caso de uso.
Flujo Alterno 1: “Contraseña incorrecta”	
Acción del Actor	Respuesta del Sistema
	11. Muestra un mensaje de error en un formulario “Las contraseñas deben tener ocho caracteres” y regresa al paso 1 del flujo básico.
Poscondiciones de Éxito:	Se generan ficheros de datos compactados y encriptados.
Poscondiciones de Fallo:	No se generan ningún archivo.

Tabla 3.3. Descripción textual del caso de uso del sistema Generar archivos con datos.

3.4 Conclusiones

En este capítulo se capturaron los requisitos funcionales y no funcionales de la herramienta auxiliar. Se identificó además los actores del sistema y se dio una descripción de cada uno y la relación entre ellos, los diagramas y especificaciones de sus casos de usos. Quedando planteadas de esta forma, las condiciones y características de la solución propuesta.

CAPÍTULO 4: ANÁLISIS Y DISEÑO DEL SISTEMA

4.1 Introducción

En el presente capítulo se describe como debe ser el sistema, realizando su análisis y diseño. Se muestran los diagramas de clases del análisis, con las interfaces, controladoras, entidades y las relaciones entre las mismas. Del diseño se presentan los diagramas de clases y de interacción correspondientes a cada caso de uso del sistema.

4.2 Análisis

El análisis consiste en obtener una visión del sistema que se preocupa de ver qué hace, de modo que sólo se interesa por los requisitos funcionales. Es descrito en el lenguaje de los desarrolladores y analiza con profundidad los requisitos funcionales. Esboza de forma clara cómo llevar a cabo la funcionalidad dentro del sistema incluida la funcionalidad significativa para la arquitectura; sirve como una primera aproximación al diseño.

4.2.1 Modelos de clases del análisis

El modelo de clases del análisis está estructurado por clases y paquetes estereotipados que proporcionan la estructura de la vista interna del sistema. Es utilizado fundamentalmente por los desarrolladores para comprender cómo debería darse forma al sistema, es decir, cómo debería ser diseñado e implementado. Este modelo define realizaciones de casos de uso, y cada una de ellas representa el análisis de un caso de uso del modelo de casos de uso.

A continuación se muestran los diagramas de clases del análisis de dos casos de uso fundamentales del sistema, el resto se encuentra en el **ANEXO 2**.

Capítulo 4: Análisis y diseño del sistema

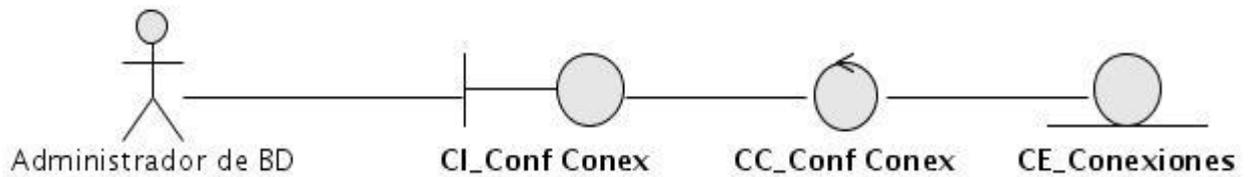


Figura 4.1. Diagrama de clases del análisis para Configurar conexión a BD.

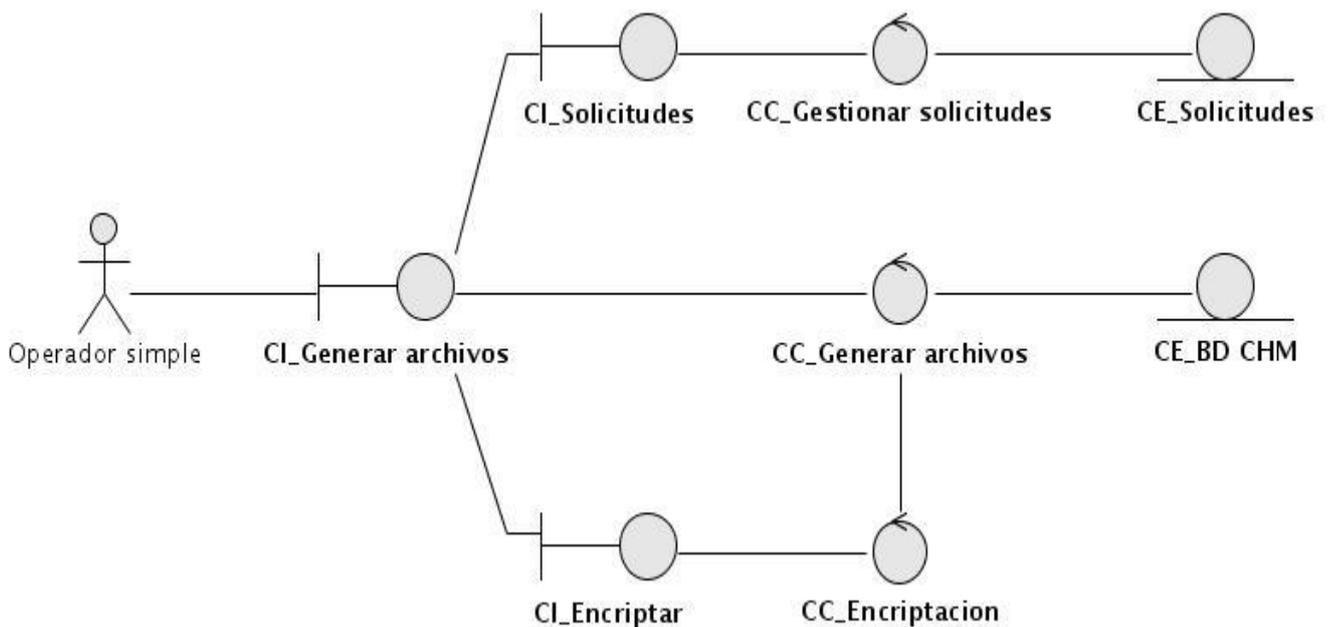


Figura 4.2. Diagrama de clases del análisis para Generar archivos con Datos.

4.3 Diseño

El diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, en definitiva cómo cumple el sistema sus objetivos. El diseño debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades. En el diseño se modela el sistema, para que soporte los requisitos, incluyendo los no funcionales y las restricciones que se le suponen.

4.3.1 Diagramas de clases del diseño

Los diagramas de clases del diseño muestran un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Gráficamente, un diagrama de clases es una colección de nodos y arcos.

Una clase de diseño y de ese modo, los subsistemas que contienen las clases de diseño a menudo participan en la realización de varios casos de uso. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema principalmente, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas. Los diagramas de clases son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa.

A continuación se muestran los diagramas de clases del diseño de dos casos de uso fundamentales del sistema, el resto se encuentra en el **ANEXO 3**.

Capítulo 4: Análisis y diseño del sistema

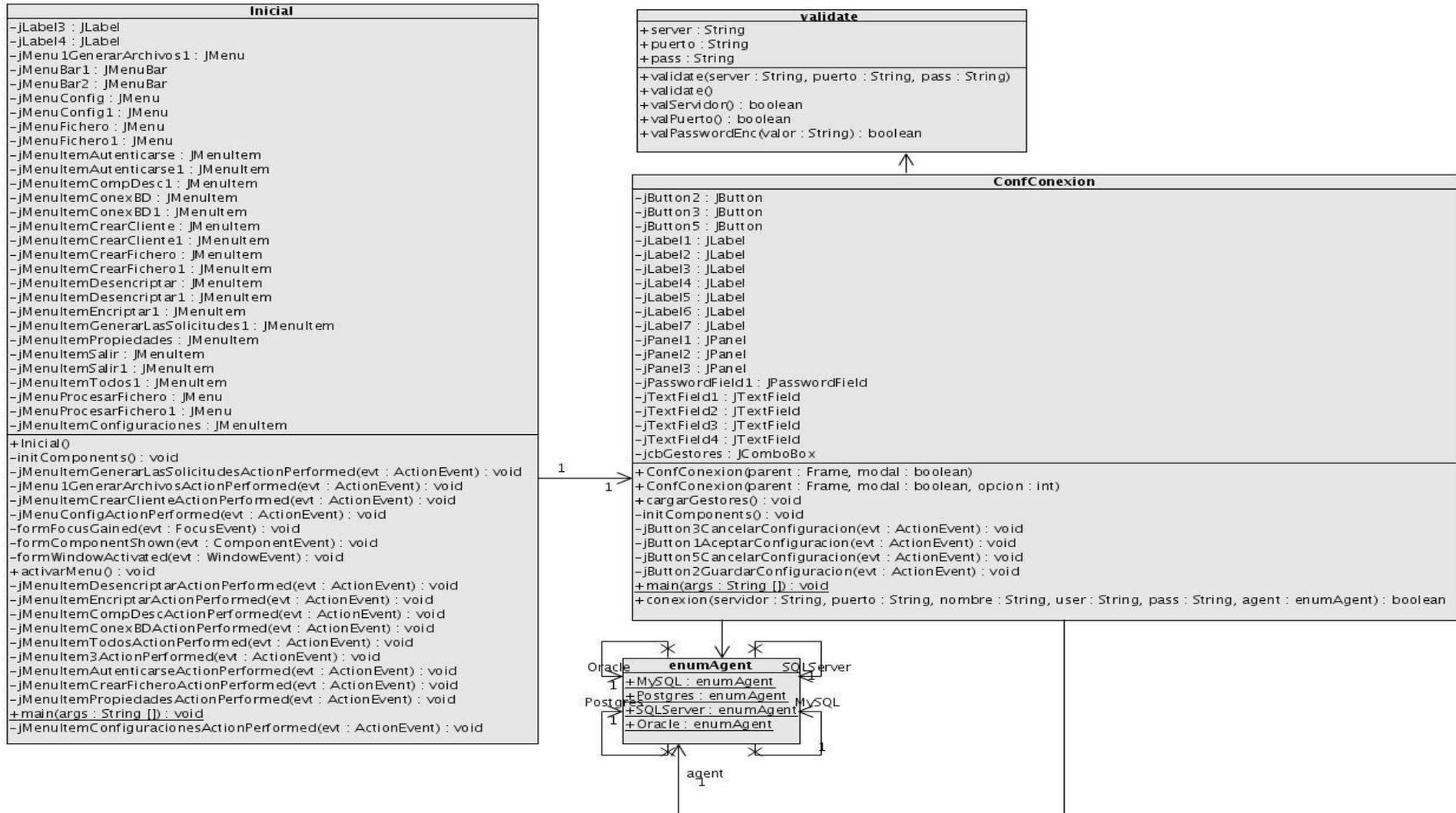


Figura 4.3. Diagrama de clases del diseño para Configurar conexión a base de datos: superior.

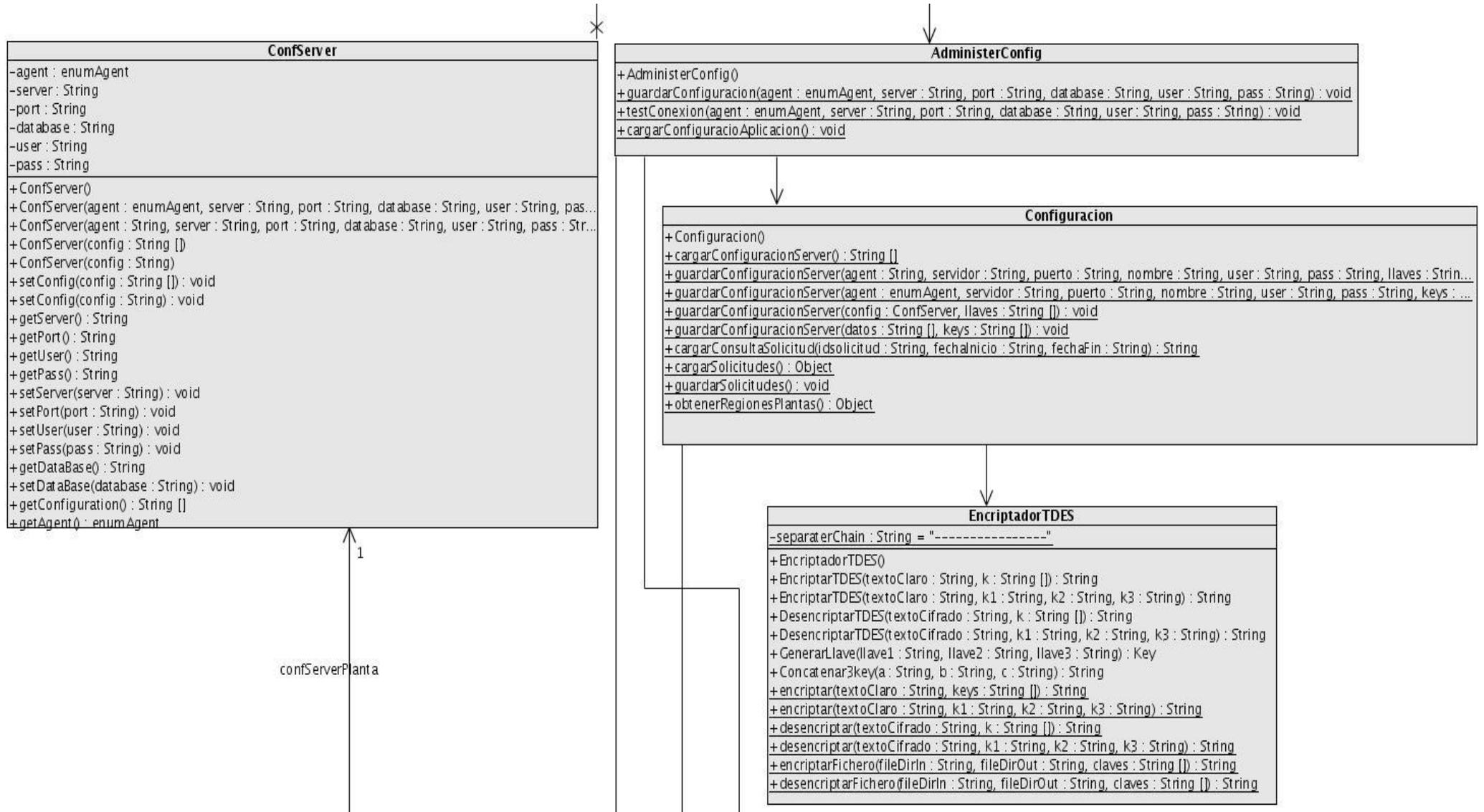


Figura 4.4. Diagrama de clases del diseño para Configurar conexión a base de datos: central.

Capítulo 4: Análisis y diseño del sistema

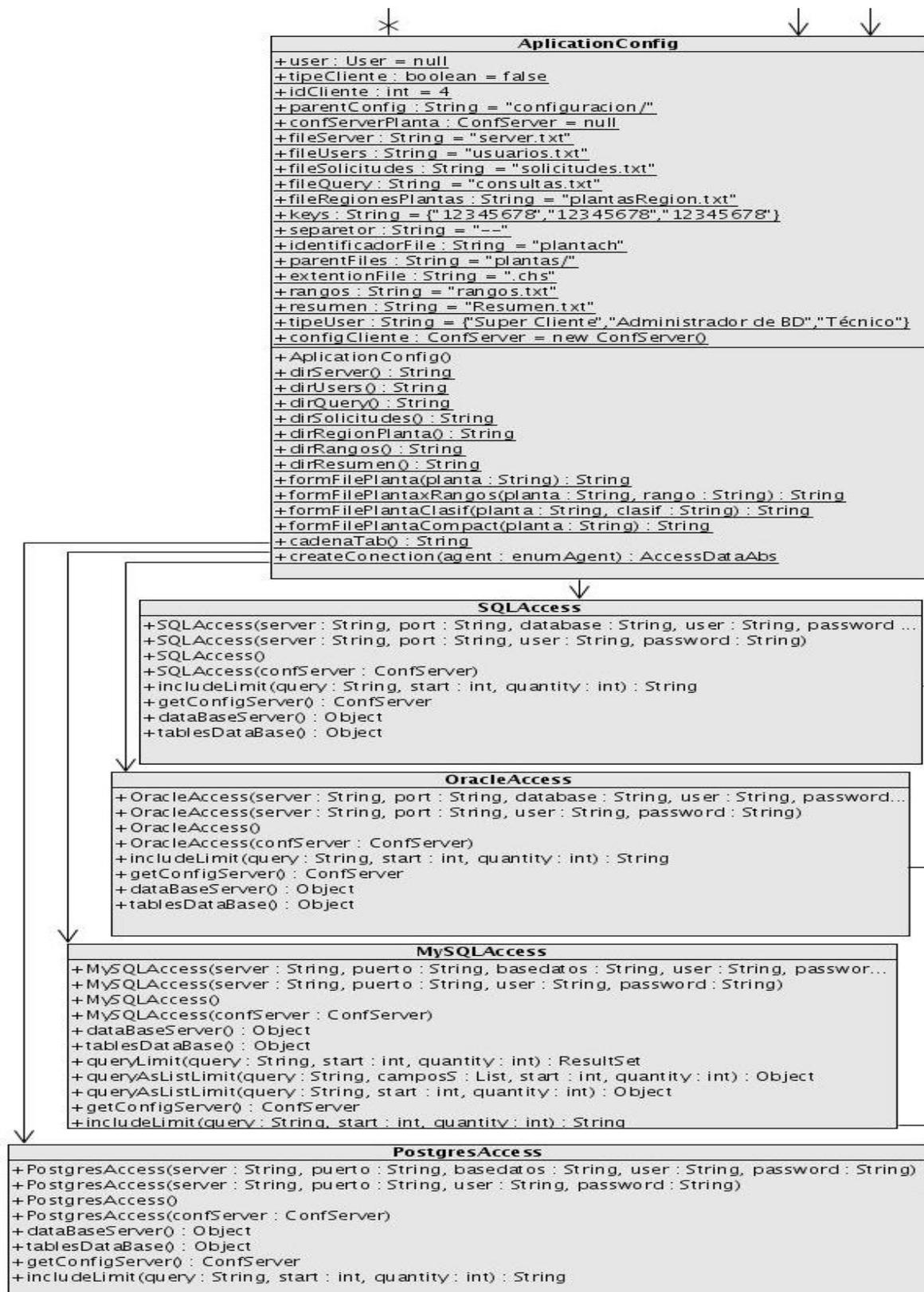


Figura 4.5. Diagrama de clases del diseño para Configurar conexión a base de datos: inferior izquierdo.



Figura 4.6. Diagrama de clases del diseño para Configurar conexión a base de datos: inferior derecho.

Capítulo 4: Análisis y diseño del sistema



Figura 4.1. Diagrama de clases del diseño para Generar archivos con datos: superior.

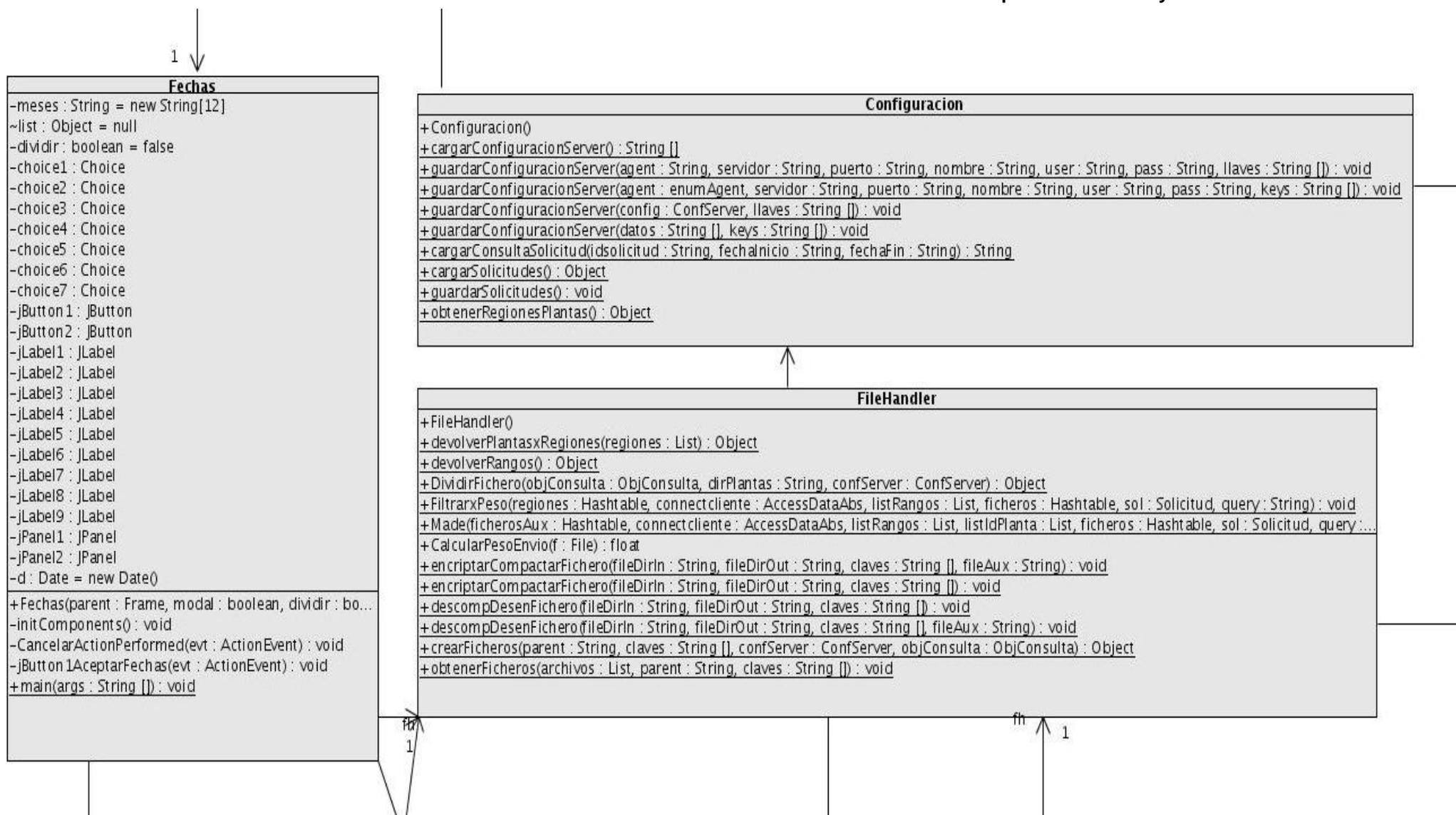


Figura 4.2. Diagrama de clases del diseño para Generar archivos con datos: central izquierdo.

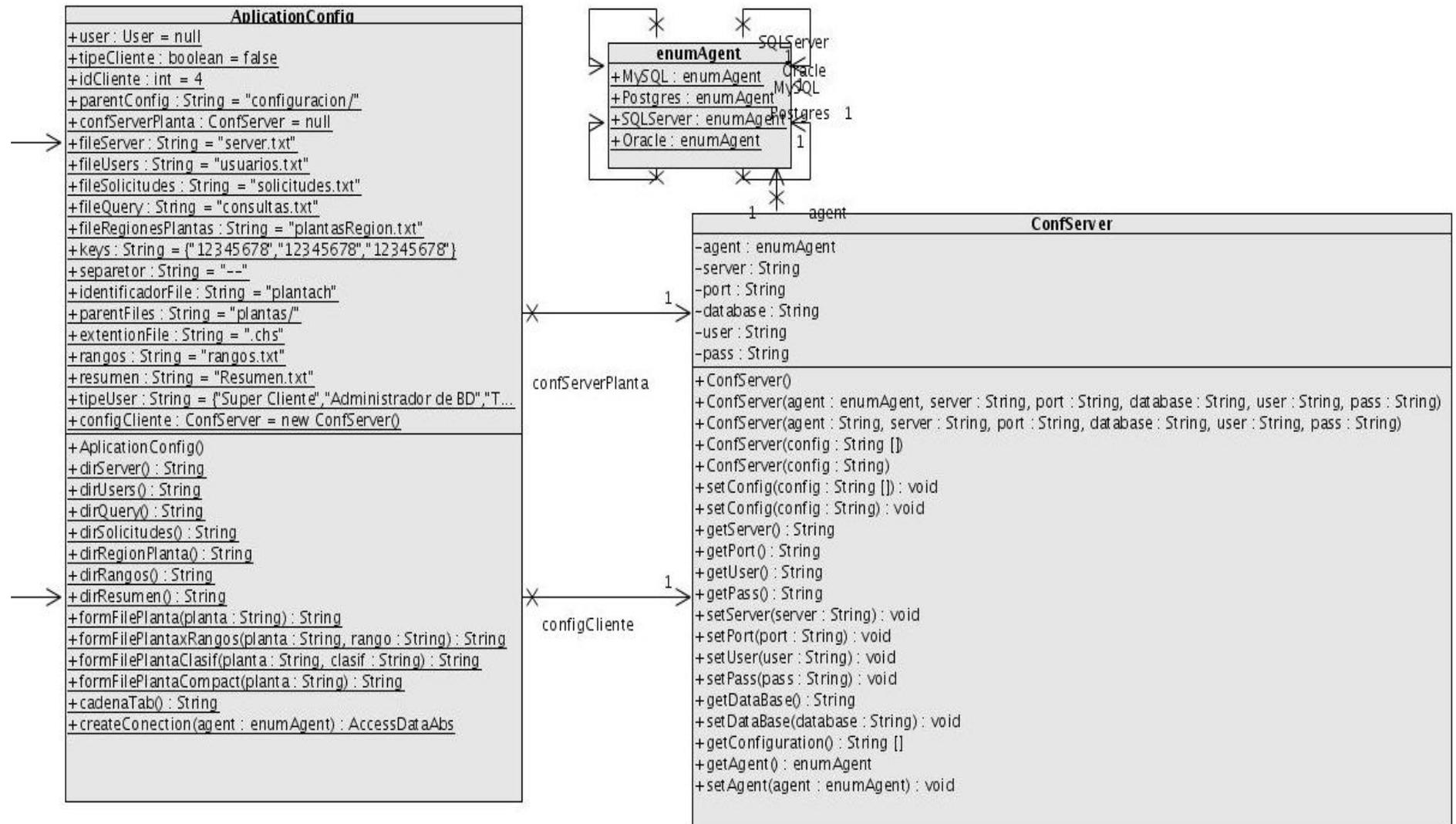


Figura 4.3. Diagrama de clases del diseño para Generar archivos con datos: central derecho.

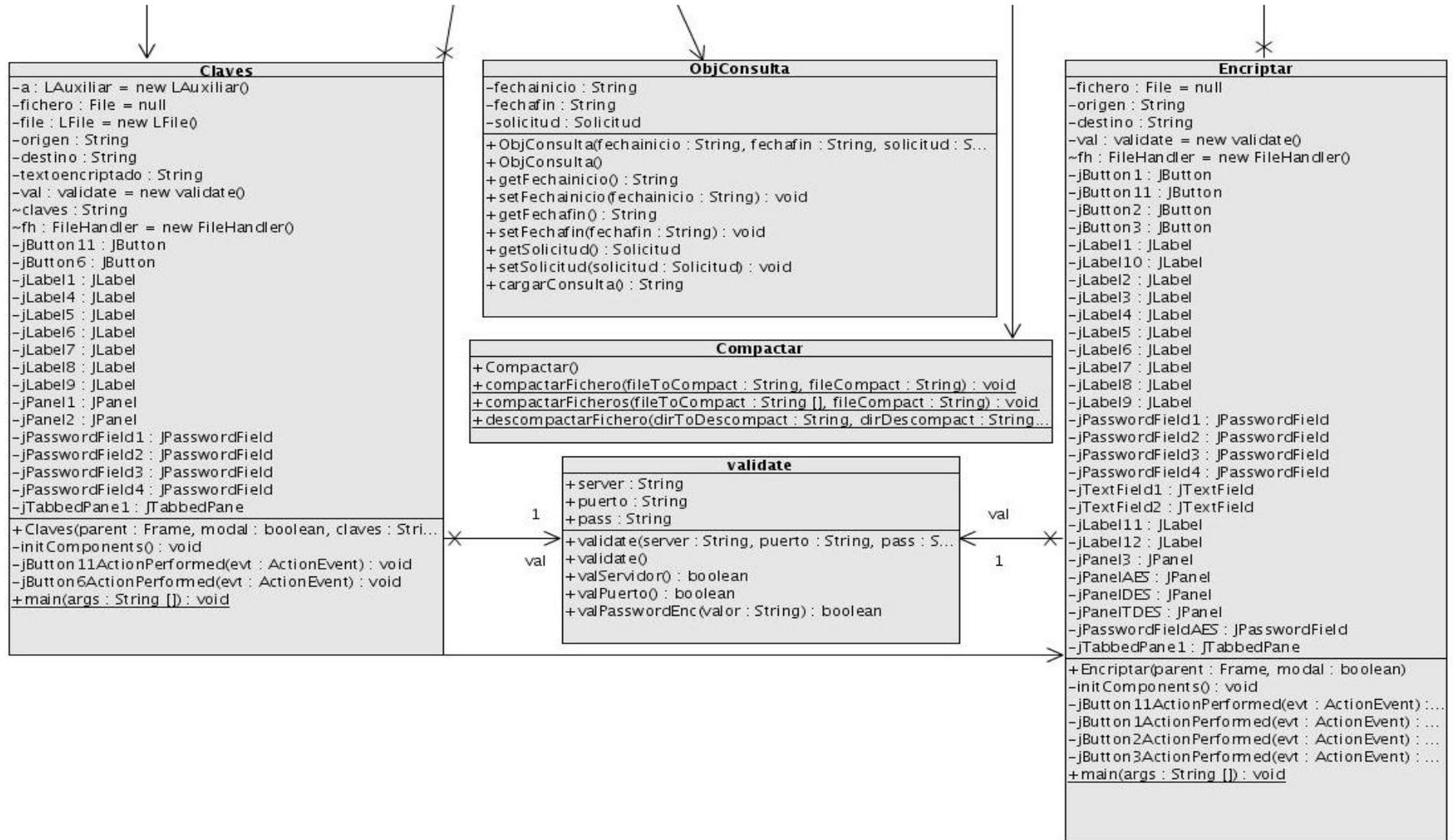


Figura 4.4. Diagrama de clases del diseño para Generar archivos con datos: inferior.

4.3.2 Algunas clases, ficheros y sus funciones

¿Cómo desarrollar una aplicación flexible, con conexión a varios gestores de BD?

Con la creación de varios ficheros .txt que se configuran de acuerdo a las características de cada cliente y la utilización de clases que se apoyan en estos ficheros para obtener la estructura de datos de cada cliente y los diferentes Gestores de BD que pueden ser tratados desde la aplicación. Se garantiza que la misma cumpla con el requisito de conectarse a varios gestores de BD como (PostgreSQL, MySQL, SQL Server, Oracle).

Tipo	Nombre	Descripción
Fichero	solicitudes.txt	En este fichero se almacena el nombre de todas las solicitudes y los campos de la base de datos que determine el cliente que serán procesados por la aplicación.

Tabla 4.1. Descripción del fichero solicitudes.txt.

Tipo	Nombre	Descripción
Fichero	usuarios.txt	En este fichero se almacenan los nombres, las contraseñas encriptadas con el método de encriptación MD5 y el rol de todos los usuarios con acceso a la herramienta.

Tabla 4.2. Descripción del fichero usuarios.txt.

Tipo	Nombre	Descripción
Fichero	consultas.txt	En este fichero se encuentra las distintas consultas que se emplearan en dependencia de los distintos gestores de BD.

Tabla 4.3. Descripción del fichero consultas.txt.

Tipo	Nombre	Descripción
Clase	validate	Valida los datos entrados por el cliente cuando se esta estableciendo una conexión a la BD como(server, puerto, pass)

Tabla 4.4. Descripción de la clase validate.

Capítulo 4: Análisis y diseño del sistema

Tipo	Nombre	Descripción
Clase	AccessDataAbs	Es la clase donde se encuentran todo los métodos, los driver y las startUrl para los distintos gestores de BD, necesarios para lograr la conexión. De la cual heredan: (MySQLAccess se utiliza para crear una conexión a una BD en MySQL), (OracleAccess se utiliza para crear una conexión a una BD en Oracle), (PostgresAccess se utiliza para crear una conexión a una BD en Postgres) y (SQLAccess para crear una conexión a una BD en SQLServer).

Tabla 4.5. Descripción de la clase AccessdataAbs.

Tipo	Nombre	Descripción
Clase	FileHandler	Es la clase donde se encuentran los métodos para realizar todas las operaciones con los ficheros que son creados por la herramienta cumpliendo con la clasificación, el encriptado y compactado de la información como por ejemplo: Para la clasificación (devolverPlantasxRegiones ()). Para crear los ficheros (crearFicheros ()). Para dividir los ficheros (DividirFichero ()). Para encriptar y compactar ficheros (encriptarCompactarFichero ()).

Tabla 4.6. Descripción de la clase Filehandler.

Tipo	Nombre	Descripción
Clase	AdministerConfig	Es la clase donde se implementan los métodos relacionados con la configuración de la herramienta como: cargarConfiguracioAplicacion () guardarConfiguracion () testConexion ()

Tabla 4.7. Descripción de la clase AdministerConfig.

Tipo	Nombre	Descripción
Clase	EncriptadorDES	Clase que se encarga de encriptar los datos con el estándar de encriptación DES.

Tabla 4.8. Descripción de la clase EncriptadorDES.

Tipo	Nombre	Descripción
Clase	EncriptadorTDES	Clase que se encarga de encriptar los datos con el estándar de encriptación TripleDES.

Tabla 4.9. Descripción de la clase EncriptadorTDES.

4.3.3 Diagramas de interacción

Un diagrama de interacción consiste en un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos. Un diagrama de secuencia es un diagrama de interacción que destaca la disposición temporal de los mensajes.

Por cada realización de caso de uso se ha realizado un diagrama de interacción (específicamente diagrama de secuencia), donde se expone el flujo principal de información entre los objetos del diseño, con sus métodos y parámetros.

A continuación se muestran los diagramas de secuencia del diseño de dos casos de uso fundamentales del sistema, el resto se encuentra en el **ANEXO 4**.

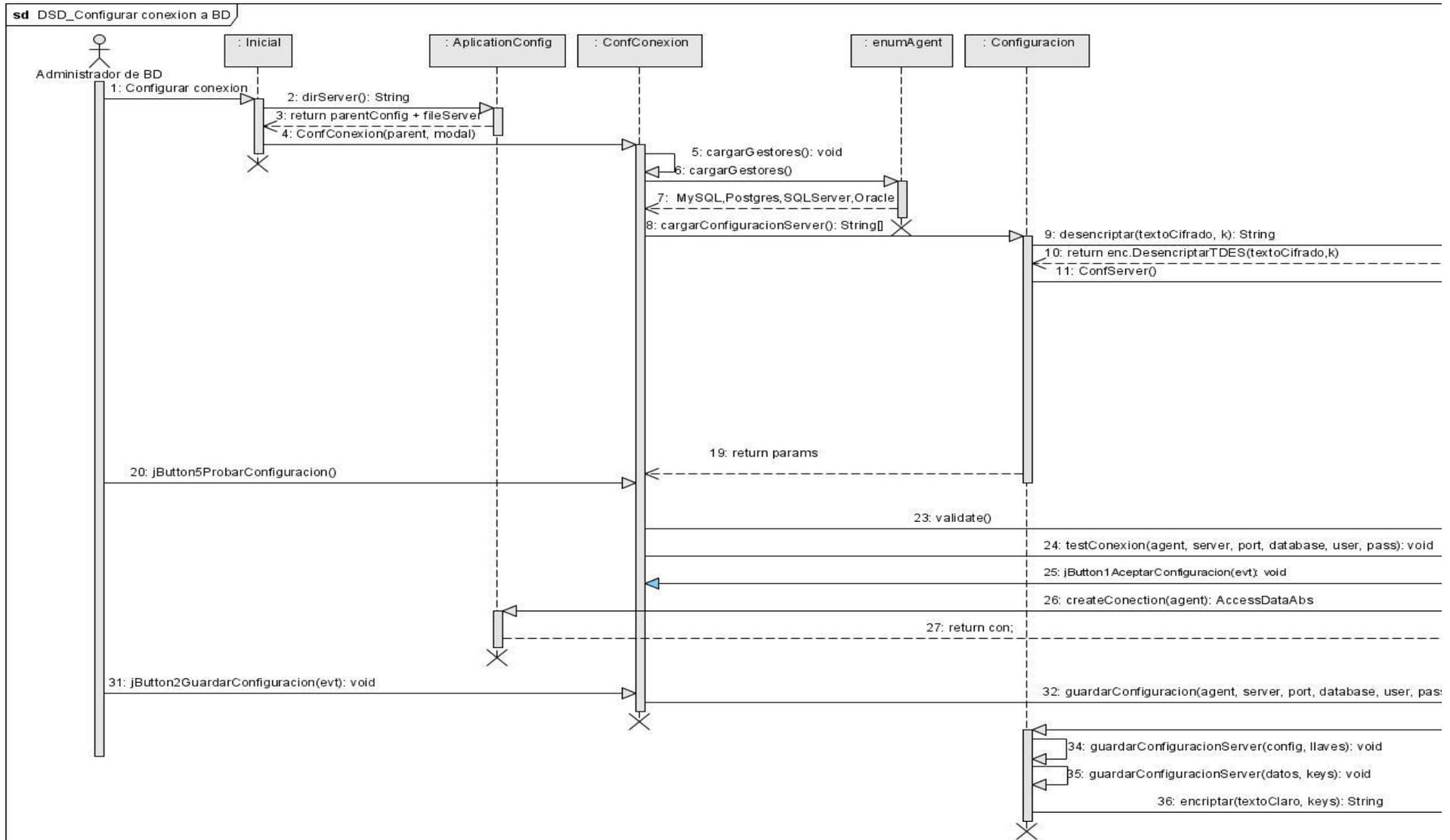


Figura 4.5. Diagrama de secuencia para Configurar conexión a base de datos: izquierdo.

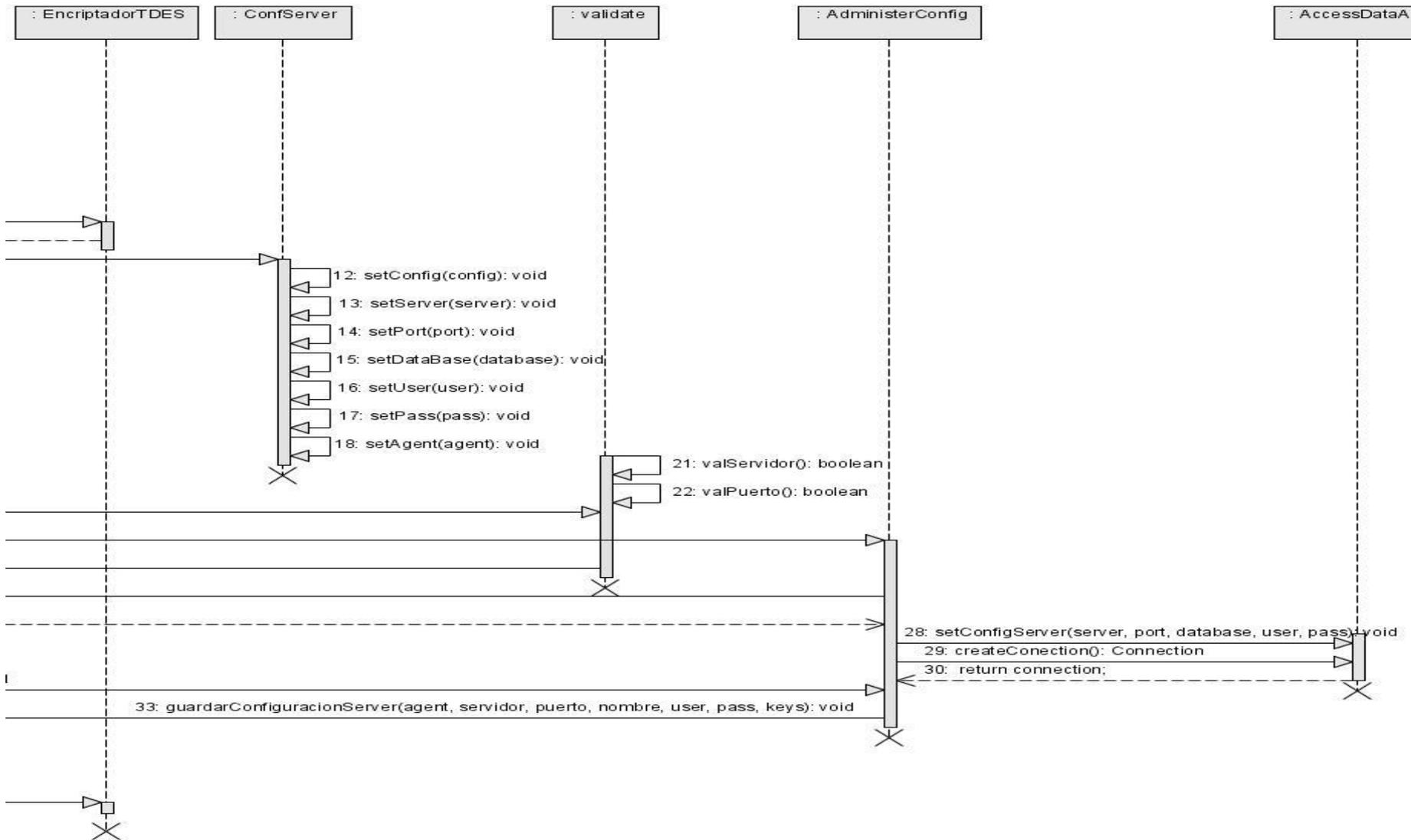


Figura 4.6. Diagrama de secuencia para Configurar conexión a base de datos: derecho.

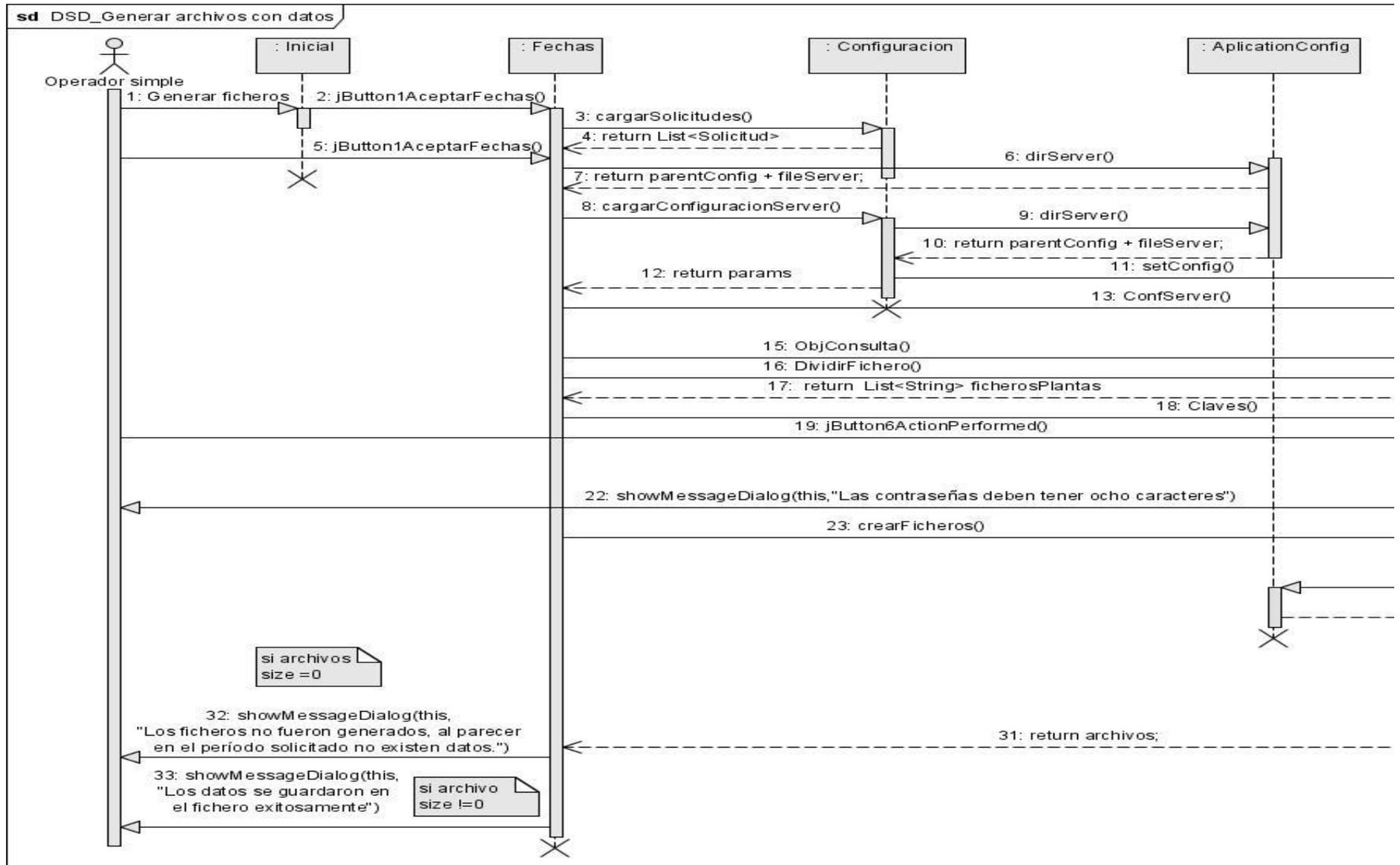


Figura 4.7. Diagrama de secuencia para Generar Archivos con datos: izquierdo.

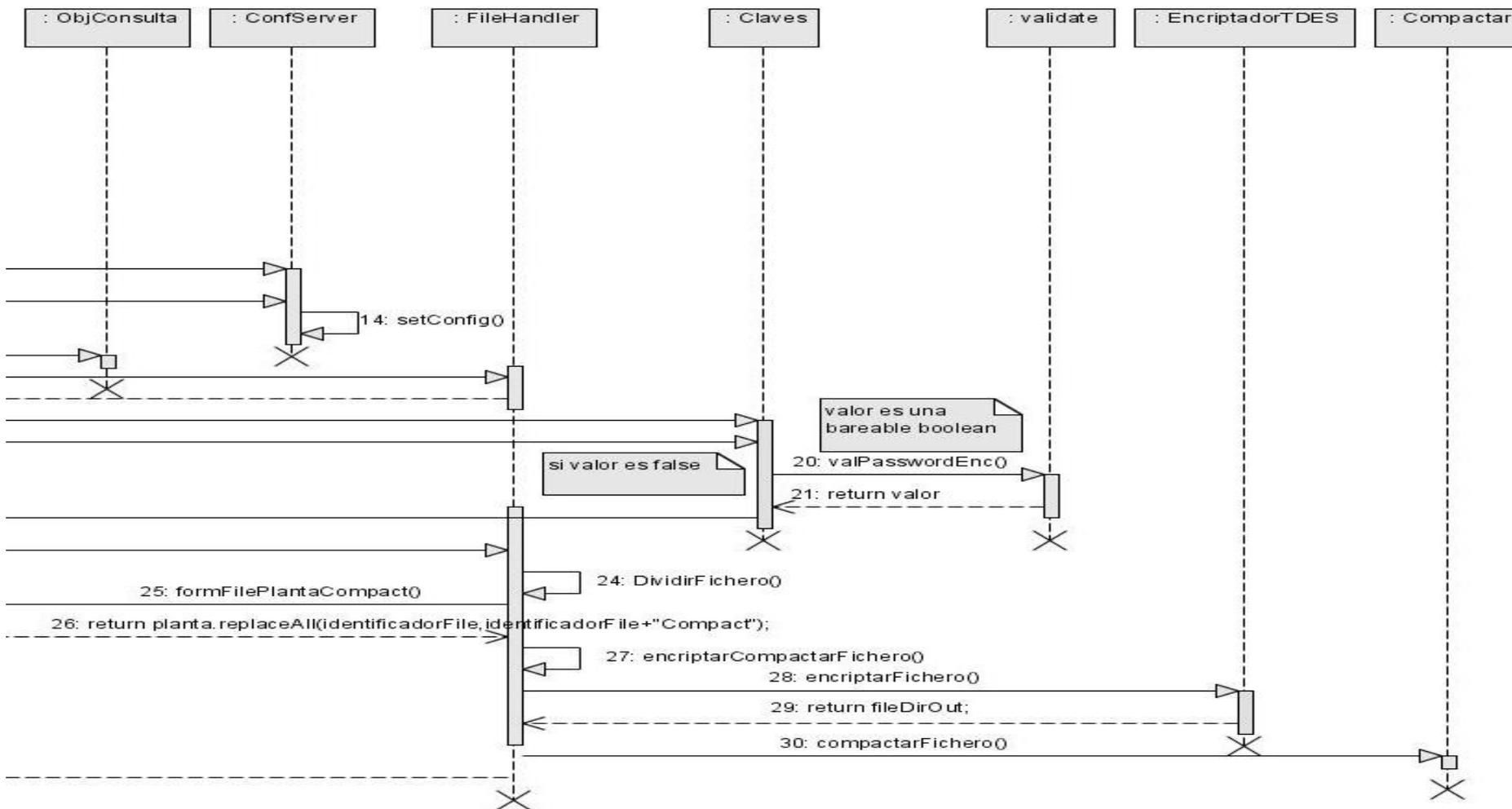


Figura 4.8. Diagrama de secuencia para Generar Archivos con datos: derecho.

4.3.4 Prototipo de interfaz



Figura 4.9. Interfaz de usuario del sistema.

4.4 Conclusiones

En este capítulo se describió la solución propuesta a partir de los diagramas de clases del análisis y diseño por casos de uso, junto con los de secuencia. Se mostró el prototipo de interfaz de usuario.

CAPÍTULO 5: IMPLEMENTACIÓN Y PRUEBA

5.1 Introducción

En el presente capítulo se describe la implementación del software a través de los diagramas de componentes. Además con el diagrama de despliegue se muestra cómo se realiza la distribución de los nodos necesarios para el despliegue de la aplicación. También se realiza pruebas de “Caja blanca” y “Caja negra” para obtener una mayor seguridad del sistema.

5.2 Diagrama de despliegue

El diagrama de despliegue representa la arquitectura en tiempo de ejecución de un sistema. Muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se relacionan en esos nodos.

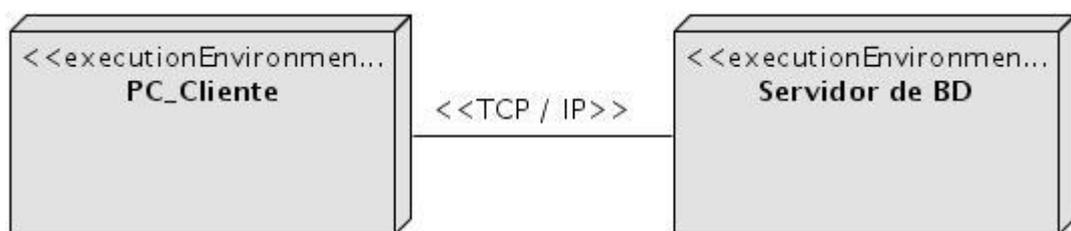


Figura 5.1. Diagrama de Despliegue.

5.3 Diagrama de componentes

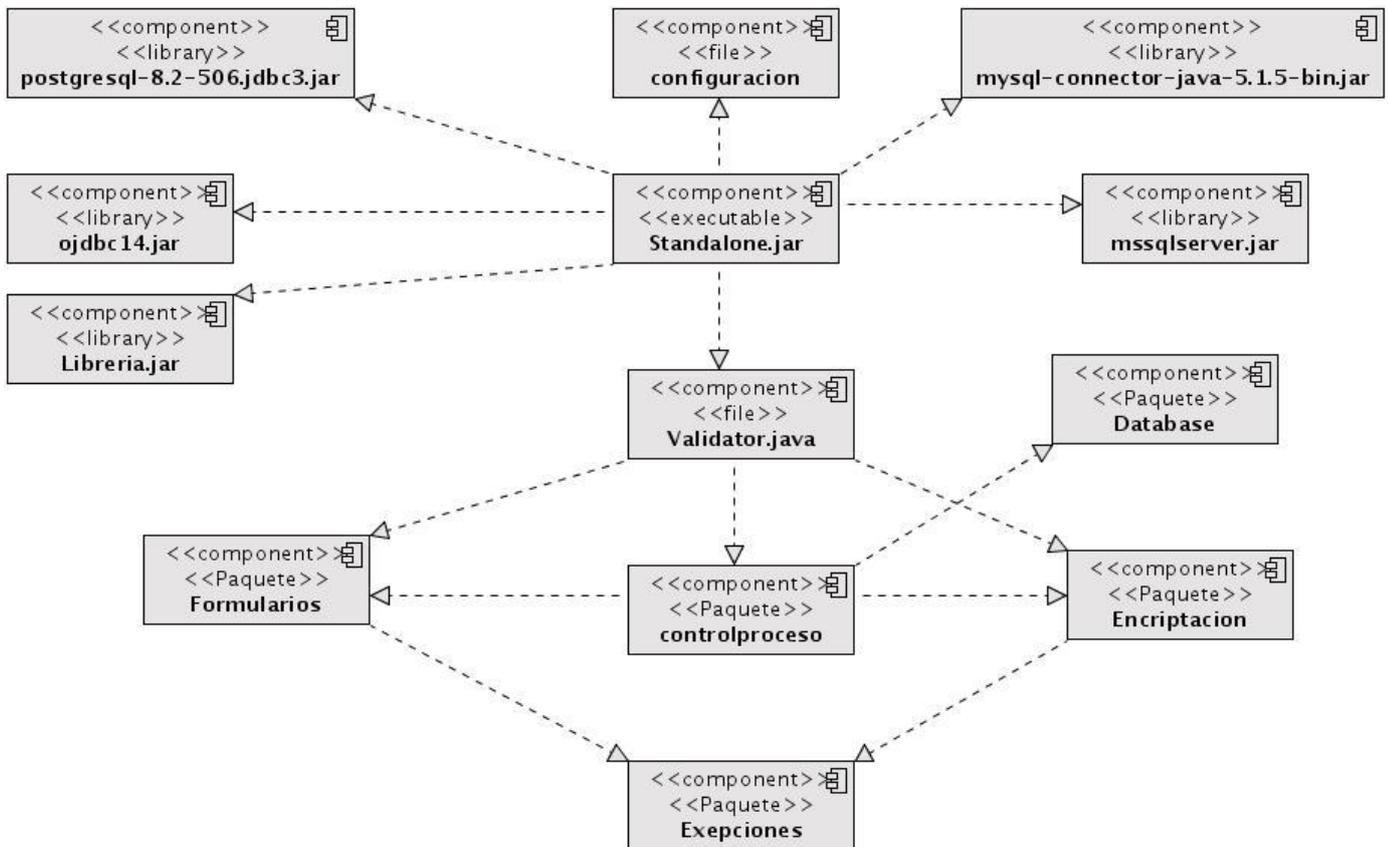


Figura 5.2. Diagrama de Componentes.

5.4 Modelo de prueba

Al sistema implementado se le hicieron varias pruebas, para ello se utilizaron los métodos de pruebas de caja blanca y de caja negra.

La prueba 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 un resultado correcto, así como que la integridad de la información externa se mantiene.

La prueba de caja blanca del software se 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.

5.4.1 Modelos de prueba de caja negra

Entrada	Resultados	Condiciones
<p>Una vez que el usuario escoge crear un nuevo usuario de la aplicación, el sistema muestra un formulario para la captura de los datos necesarios para crear un nuevo usuario:</p> <p>Nuevo Usuario: El usuario teclea el nombre y la contraseña del usuario que desea crear, repite la contraseña para la confirmación, además selecciona el Tipo de usuario que puede ser:</p> <ul style="list-style-type: none"> ➔ Súper Cliente. ➔ Administrador de BD. ➔ Técnico. 	<p>El sistema muestra un mensaje de confirmación “Realmente desea crear el usuario “X””.</p> <p>El sistema adiciona el nuevo usuario al fichero de configuración usuarios.txt y muestra el mensaje “El usuario “X” se ha creado satisfactoriamente”.</p>	<p>El usuario debe llenar todos los campos de información.</p> <p>El usuario no debe crear un usuario que ya existe.</p>

Tabla 5.1. Prueba de caja negra al caso de uso Administrar usuarios clientes.

Entrada	Resultados	Condiciones
<p>Una vez que el usuario escoge la opción descriptar fichero, el sistema muestra un formulario para la captura de los datos necesarios para descriptar el fichero deseado:</p> <p>Descriptar Fichero: El usuario busca el origen donde está guardado el fichero encriptado y selecciona el destino de los datos, además selecciona el Algoritmo de encriptación deseado, que puede ser:</p> <ul style="list-style-type: none"> ➔ DES. ➔ Triple DES. <p>Luego el usuario teclea la contraseña</p>	<p>El sistema descripta el fichero en la dirección indicada.</p> <p>El sistema muestra un mensaje de confirmación “El fichero se ha creado satisfactoriamente”.</p>	<p>El usuario debe llenar todos los campos de información.</p> <p>La contraseña que teclee el usuario debe contener 8 caracteres.</p>

para desenscriptar el fichero.		
--------------------------------	--	--

Tabla 5.2. Prueba de caja negra al caso de uso Desenscriptar.

Entrada	Resultados	Condiciones
<p>Una vez que el usuario selecciona la opción ver las propiedades del fichero, el sistema muestra un formulario para mostrar los datos del fichero deseado:</p> <p>Propiedades de fichero: El usuario busca el fichero deseado y selecciona la opción ver datos del fichero:</p> <ul style="list-style-type: none"> ➔ Nombre. ➔ Tamaño. ➔ Cantidad de líneas. ➔ Tipo de archivo. <p>El usuario selecciona guardar el fichero con los datos.</p>	<p>El sistema muestra los datos del fichero buscado y lo guarda en la dirección previamente especificada.</p> <p>El sistema muestra un mensaje de confirmación “Se han guardado los datos correctamente”.</p>	<p>El usuario debe llenar todos los campos de información.</p>

Tabla 5.3. Prueba de caja negra al CU Generar resumen de propiedades de fichero.

5.4.2 Modelos de prueba de caja blanca

De acuerdo al segmento de código correspondiente al caso de uso Generar ficheros con datos, se le ha hecho la prueba de caja blanca.

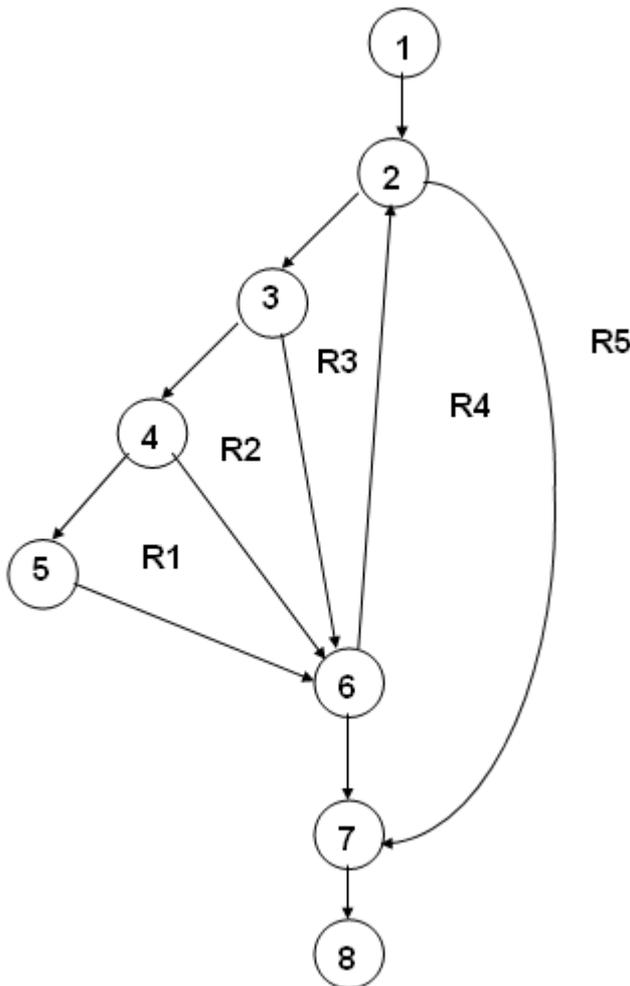
```

public static List<String> devolverPlantasxRegiones(List<String> regiones)
{
    String[] lista = LFile.readFileByLine(AplicationConfig.dirRegionPlanta());    1
    List<String> texto = new ArrayList();                                         1
    for (int i = 0; i < lista.length; i++)                                       2
    {
        String[] aux = lista[i].split(AplicationConfig.separetor);             2
        if (regiones.contains(aux[1]))                                           3
        {
            if (!texto.contains(aux[0]))                                         4
            {
                texto.add(aux[0]);                                              5
            }
        }
    }
}

```

```
}  
return texto;  
}
```

7
8



Complejidad Ciclomática:

V (G): Número de regiones del grafo

$$V(G) = A - N + 2$$

$$V(G) = P + 1$$

A: Número de aristas del grafo.

N: Número de nodos.

P: Número de nodos predicados.

$$V(G) = 5$$

Caminos: 1-2-3-4-5-6-2...7-8,
1-2-3-4-6-2...7-8, 1-2-3-6-2...7-
8, 1-2-7-8.

Camino: 1-2-7-8.

Caso de prueba: No devuelve ninguna lista de plantas por región.

Entrada: Seleccionó la opción generar archivos.

Resultados: No se obtiene ninguna lista de plantas por región.

Condiciones: Que no exista al menos una planta en alguna región.

Capítulo 5: Implementación y prueba

Camino: 1-2-3-4-5-6-2...7-8.

Caso de prueba: Devuelve una lista de las plantas de cada región.

Entrada: Seleccionó la opción generar archivos.

Resultados: Se obtiene una lista de las plantas de cada región que tenga al menos una planta.

Condiciones: Que la lista de las direcciones de las plantas no este vacía.

Camino: 1-2-3-4-6-2...7-8.

Caso de prueba: Devuelve una lista de las plantas de cada región.

Entrada: Seleccionó la opción generar archivos.

Resultados: Se obtiene una lista de las plantas de cada región.

Condiciones: Si la planta ya esta en la lista resultante no se adiciona.

Camino: 1-2-3-6-2...7-8.

Caso de prueba: Seleccionó la opción generar archivos.

Entrada: Seleccionó la opción generar archivos.

Resultados: Se obtiene una lista de las plantas de cada región.

Condiciones: Si una región entrada no está en la lista de regiones con plantas, no se adiciona al listado resultante y se verifica la próxima región del listado.

5.5 Conclusiones

En este capítulo se abordó la forma en que se desarrolló la aplicación haciendo uso para esto de los diagramas de componentes, también se hizo una descripción de cómo se realiza el despliegue de la aplicación mediante el diagrama de despliegue. Además se mostraron los resultados de las pruebas de caja blanca y de caja negra realizadas.

CONCLUSIONES

Con el desarrollo del presente trabajo y con la implementación de la herramienta auxiliar para el procesamiento de datos de clientes para el sistema de Correo Híbrido Masivo de IPOSTEL se concluye afirmando que:

- Se capturaron los requisitos funcionales y no funcionales necesarios, para mediante una herramienta auxiliar brindar una solución rápida al problema planteado, logrando la capacidad de conectarse a cualquier estructura de bases de datos y permitiendo que dentro de un sistema de PCHM se impriman los mensajes en la planta más cercana al destino y así satisfacer las necesidades del cliente.
- Se buscaron en Internet herramientas de procesamiento de datos, y con su estudio se llegó a la conclusión de desarrollar una aplicación de escritorio.
- La herramienta auxiliar fue modelada con Visual Paradigm y UML e implementada con lenguaje Java empleando el IDE NetBeans.
- Se estudiaron las formas de cifrado simétrico y asimétrico y se determinó escoger dos métodos de cifrado simétrico, DES y Triple DES por su rapidez respecto a los de cifrado asimétrico y se deja abierta la posibilidad de implementar otros métodos de encriptación.
- Se logró diseñar e implementar basados en la metodología RUP, una aplicación de escritorio que unida al Sistema de Correo Híbrido Masivo garantiza la integridad de la información mediante el empleo de diversas técnicas de encriptación y otras funcionalidades de compactado y clasificado de la información.
- Se logró una herramienta capaz de conectarse a cuatro sistemas gestores de bases de datos. Oracle, SQL Server, My SQL, y Posgre SQL y con capacidad para en una segunda fase implementar conexiones a otros gestores de bases de datos.

RECOMENDACIONES

Se recomienda al proyecto Correo Híbrido perteneciente al polo de servicios postales de la Universidad de las Ciencias Informáticas (UCI), que:

- Continúe el desarrollo de esta herramienta en una segunda fase para mejorar e incrementar las funcionalidades de la herramienta auxiliar.
- Que se le incorporen otros métodos de encriptación como el AES o el RSA que incrementen las opciones para garantizar la seguridad e integridad de la información.

BIBLIOGRAFÍA

Bibliografía Citada

- BARRIENTOS, A. M. *El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML*, 2007]. Disponible en: <http://www.monografias.com/trabajos16/lenguaje-modelado-unificado/lenguaje-modelado-unificado.shtml#PROCESO>
- CAMACHO, L. C. y G. FUENTES. *Análisis Comparativo entre CMMI & SPICE*. Facultad de Ciencias Puras y Naturales, UNIVERSIDAD MAYOR DE SAN ANDRÉS, 2007. p.
- CASTELLANO, M. y E. LLANO. *Módulo para la gestión de los procesos de administración económica de las Oficinas Postales Telegráficas de Venezuela*. La Habana, Universidad de las Ciencias Informáticas, 2007. 130. p.
- CIENCIA, S. *Origen y evolución histórica*, 2007. [Disponible en: <http://www.solociencia.com/informatica/influencia-internet-sociedad-actual-origen-evolucion-historica.htm>
- FOUNDATION, F. S. *La Definición de Software Libre*, 2008. [Disponible en: <http://www.gnu.org/philosophy/free-sw.es.html>
- GUMPERT, P. *Como encriptar texto y variables usando algoritmos estandar como DES o TRIPLEDES*, 2007. [Disponible en: <http://www.devjoker.com/contenidos/Articulos/276/-Como-encriptar-texto-y-variables-usando-algoritmos-estandar-como-DES-o-TRIPLEDES.aspx>
- IPOSTEL. 2007. [Disponible en: <http://www.ipostel.gov.ve>
- LARMAN, C. *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. Primera Edición. México, Prentice Hall, 1999. p.
- MARQUEZ, G. *Ingeniería de Software UML*, 2007]. Disponible en: <http://www.monografias.com/trabajos5/insof/insof.shtml>
- MARRERO, Y. *La Criptografía como elemento de la seguridad informática* 2003. [Disponible en: http://scielo.sld.cu/scielo.php?pid=S1024-94352003000600012&script=sci_arttext&tlng=es
- MORENO, A. *Acerca de la Criptografía*, 2008. [Disponible en: <http://microasist.com.mx/noticias/en/ampen1402.shtml>
- PROGRAMACIÓN, L. D. *Programación Java*, 2006. [Disponible en: <http://www.lenguajes-de-programacion.com/programacion-java.shtml>

Bibliografía

- REPORT, A. *Aplicaciones Empresariales: Información sobre Software y Soluciones Tecnológicas Open Source para la PYME y la mediana empresa*, 2007. [Disponible en: <http://www.aplicacionesempresariales.com/general/agata-report-tu-base-de-datos-al-descubierto.html>]
- RODRÍGUEZ, J. *Análisis y diseño de un sistema para la gestión del negocio de una planta de correo híbrido masivo* La Habana, Universidad de las Ciencias Informáticas, 2007. p.
- ROMÁN, C. A. *La Tecnología Java*, 2007. [Disponible en: http://profesores.fi-b.unam.mx/carlos/java/java_basico1_1.html]
- SAVARIS, C. B. *Beneficios que brinda Internet a un negocio convencional o tradicional*, 2008. [Disponible en: <http://www.continentalmarket.com/boletin/negocios-en-internet.htm>]
- SEBARES, M. *MSD Caja Fuerte, gestor de documentos y ficheros privados*, 2008. [Disponible en: <http://www.mssoft.com/esp/products/cajafuerte/features.htm>]
- VIRUEGA, D. *Primer encuentro Regional de dinamización de la red de cibercentros de Castilla y León.*, 2006.
- WIKIPEDIA. *Centro de proceso de datos*, 2008a.
- . *NetBeans*, 2008b. [Disponible en: <http://es.wikipedia.org/wiki/NetBeans>]

Bibliografía Consultada

- AMENGUAL, E. and A. MAS. *Calidad e Ingeniería del Software*,. *Revista Española de Innovación*, 2005. Volumen 1.
- CHILE, C. D., 2008. [Disponible en: <https://www.correos.cl>
- ESPAÑA, C. D. *Sociedad Estatal Correos y Telégrafos*, 2005. [Disponible en: <http://www.correos.es/dinamic/plantillas/home1.asp>
- FERNANDEZ, A. *Seguridad en sistemas operativos*, 2006. [Disponible en: <http://exa.unne.edu.ar/depar/areas/informatica/SistemasOperativos/MonogSO/SEGSO00.htm>
- HERRAMIENTAS. *Herramientas de Productividad de Software*, 2008. [Disponible en: http://www-142.ibm.com/software/dre/ecatalog/list.wss?locale=es_ES&category=L107029V38585S93
- INFORMACIÓN, S. D. L., 2008. [Disponible en: http://cibernarium.tamk.fi/seguridad_2/salausmenetelmat.htm
- MÉTODOS. *Métodos de encriptación*, 2008. [Disponible en: <http://www.soygik.com/metodos-de-encriptacion-explicados-graficamente>
- PERÚ, C. D. *Servicios Postales del Perú*, 2008. [Disponible en: <http://www.serpost.com.pe>
- RICA, C. D. C. *Servicios de correo*, 2008. [Disponible en: <http://www.correos.go.cr>
- SEPOMEX. *Servicio Postal Mexicano*, 2008. [Disponible en: <http://www.sepomex.gob.mx/Sepomex>
- TIPOS. *Tipos de encriptación*, 2008. [Disponible en: <http://jibe.freeshell.org/textes/cpc/tipos-encriptacion.html>

GLOSARIO

A continuación, en orden alfabético, se muestra el significado de algunos términos usados en este documento cuyo uso no es común y que pueden dificultar la comprensión del mismo:

- **BD (Base de Datos):** Conjunto de datos organizados de manera tal que pueden ser modificados y eliminados fácilmente y se puede realizar búsquedas en ellos.
- **CH:** Correo Híbrido.
- **CHM:** Correo Híbrido Masivo.
- **CPD:** Centros de Procesamiento de Datos.
- **Dirección Postal:** Son todos los campos que describen la ubicación en el espacio de un objeto.
- **IPOSTEL:** Instituto Postal y Telegráfico de Venezuela.
- **Online:** Actividad informática que se realiza con conectividad permanente, en línea.
- **Offline:** Actividad informática que se realiza sin conectividad.
- **PCHM:** Planta de Correo Híbrido Masivo.
- **RUP:** Proceso Unificado de Desarrollo.
Es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.
- **TICs:** Tecnologías de la Información y la Comunicación.
Son un conjunto de servicios, redes, software, aparatos que tienen como fin el mejoramiento de la calidad de vida de las personas dentro de un entorno, y que se integran a un sistema de información interconectado y complementario.
- **UML:** Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modelling Language).
Es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad.
- **UCI:** Universidad de las Ciencias Informáticas.

ANEXOS

ANEXO 1: Descripción textual de los casos de uso del sistema.

Caso de uso:	Autenticar usuario (CU-0)	
Actor:	Usuario (inicia)	
Propósito:	Permitir al usuario entrar al sistema.	
Resumen:	El caso de uso se inicia cuando el usuario introduce su Usuario y Contraseña para intentar autenticarse si no ocurre ningún error en la acción logra autenticarse y se le asignan sus privilegios.	
Referencias:	RF 1	
Precondiciones:	El usuario desee acceder a la aplicación.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
	1. Muestra el formulario para la autenticación. Con los campos: usuario y contraseña y las siguientes opciones. a. Entrar.	
2. Llena los campos y presiona entrar.	3. Verifica que el usuario y la contraseña están bien, si no ir al Flujo Alterno 1.	
	4. Muestra una página con los vínculos a las opciones que le competen al usuario de acuerdo a su rol, finalizando el caso de uso.	
Flujo Alterno 1: “Datos incorrectos”		
Acción del Actor	Respuesta del Sistema	
	3. Verifica el tipo de error. a. Si es “Campos vacíos” muestra un mensaje de error “Campo vacío. Debe indicar el nombre de usuario y la contraseña.” b. Si es “Usuario y (o) Contraseña Incorrecta” muestra un mensaje de error “El nombre de usuario o la contraseña no son válidos.”	
Poscondiciones de Éxito:	Usuario autenticado.	
Poscondiciones de Fallo:	No se autentica el usuario.	

Tabla 3.4. Descripción textual del caso de uso del sistema Autenticar Usuario.

Caso de uso:	Administrar usuarios clientes (CU-1)
Actor:	Operador avanzado (inicia)
Propósito:	Permitir al operador administrar un usuario cliente.
Resumen:	El caso de uso se inicia cuando el usuario escoge la opción “Nuevo usuario”, introduce el nombre del usuario a crear, la contraseña y selecciona el tipo de usuario a crear y presiona aceptar terminando de esta forma el caso de uso.
Referencias:	RF 3
Precondiciones:	El operador avanzado debe estar registrado en el sistema.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción “Nuevo usuario”.	2. Muestra el formulario para la creación de los usuarios. Con los campos: usuario, contraseña, repetir contraseña y las opción de escoger el tipo de usuario (Súper Cliente, Administrador de BD, Técnico), seguido de las opciones: <ul style="list-style-type: none"> a. Aceptar. b. Cerrar.
3. Llena los campos y selecciona la opción deseada.	4. Gestiona la selección: <ul style="list-style-type: none"> a. Si selecciona la opción “Aceptar” sigue con el flujo básico. b. Si selecciona la opción “Cerrar”. Ir al paso 9 del flujo básico.
	5. Verifica la validez de los datos, si no ir al Flujo Alternativo1. 6. Muestra un mensaje en un Form (“Realmente desea crear el usuario (nombre del usuario a crear)”) con los botones:”Aceptar” y “Cancelar”.
7. Selecciona la opción deseada.	8. Gestiona la selección : <ul style="list-style-type: none"> a. Si selecciona “Aceptar” crea el usuario, ir al paso 8 del flujo básico. b. Si selecciona “Cancelar” ir al paso 9 del flujo básico.
	9. Termina el caso de uso.
Flujo Alterno 1: “Verificar Datos”	
Acción del Actor	Respuesta del Sistema
	5. Verifica que las contraseñas escritas estén correctas, si no muestra un mensaje en una Form (“Las contraseñas escritas no coinciden”) ir al paso 9 del flujo básico.

Poscondiciones de Éxito:	Se crea el usuario con la contraseña y los permisos correspondientes a su rol.
Poscondiciones de Fallo:	No se crea ningún usuario.

Tabla 3.5. Descripción textual del CU del sistema Administrar Usuarios Clientes.

Caso de uso:	Desencriptar (CU-3)
Actor:	Operador simple(inicia)
Propósito:	Permite desencriptar ficheros que se hayan encriptado.
Resumen:	El caso de uso se inicia cuando el usuario escoge la opción desencriptar fichero selecciona el fichero a desencriptar y el método de encriptación, desencripta dicho fichero terminando de esta forma el caso de uso.
Referencias:	RF5
Precondiciones:	El operador simple debe estar registrado en el sistema.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción “Desencriptar fichero”.	2. Muestra el formulario para desencriptar el fichero con las opciones: a. Origen del fichero encriptado. b. Destino de los datos planos. c. Algoritmo de encriptación. Y el botón “Cancelar” que finaliza el caso de uso en el momento que el usuario desee.
3. Selecciona la opción origen del fichero encriptado	4. El sistema brinda la posibilidad de buscar el fichero encriptado.
5. Selecciona el fichero a desencriptar.	
6. Selecciona la opción Destino de los datos planos.	7. El sistema brinda la posibilidad de seleccionar la dirección destino de los datos planos del fichero, además de nombrarlo y designar la extensión correspondiente.
8. Selecciona la dirección donde desea guardar el fichero. 9. Introduce el nombre del fichero.	
10. Selecciona el algoritmo de encriptación.	11. Si selecciona el “DES” ver flujo alternativo 1, si selecciona el “Triple DES”, el sistema muestra tres combobox para entrar las contraseñas de encriptación Seguido de los botones “Desencriptar” y “Cancelar”.
12. Introduce los datos solicitados.	14. Si selecciona “Cancelar” ver flujo alternativo 2, si selecciona

13. Selecciona una de las dos opciones “Desencriptar” o “Cancelar”.	<p>“Desencriptar” continúa en el paso 15 del flujo básico.</p> <p>15. Verifica la/las contraseña/s, si el número de caracteres entrados es distinto de 8 ver flujo alterno 3; si no el sistema muestra un mensaje informando el éxito de la acción.</p> <p>16. Finaliza la ejecución del caso de uso.</p>
Flujo Alterno 1: “Método DES”	
Acción del Actor	Respuesta del Sistema
	<p>11. El sistema muestra un combobox para entrar la contraseña de encriptación.</p> <p>12. Continúa en el paso 12 del flujo básico.</p>
Flujo Alterno 2: “Cancelar”	
Acción del Actor	Respuesta del Sistema
	14. Regresa al paso 10 del flujo básico.
Flujo Alterno 3: “Contraseña Incorrecta”	
Acción del Actor	Respuesta del Sistema
	14. Muestra un mensaje informando que el número de caracteres de la/las contraseña/s debe ser de 8 caracteres y la opción “Aceptar”.
15. Selecciona “Aceptar”. 16. Regresa al paso 12 del flujo básico.	
Poscondiciones de Éxito:	Se desencripta el fichero seleccionado.
Poscondiciones de Fallo:	No se desencripta el fichero seleccionado.

Tabla 3.6. Descripción textual del caso de uso del sistema Desencriptar.

Caso de uso:	Generar resumen de propiedades de ficheros (CU-5)
Actor:	Operador simple (inicia)
Propósito:	Permitir al operador generar un resumen de propiedades de ficheros.
Resumen:	El CU inicia cuando el operador simple desea conocer las propiedades de un determinado fichero.
Referencias:	RF 6
Precondiciones:	El operador simple debe estar registrado en el sistema.
Flujo Normal de Eventos	

Acción del Actor	Respuesta del Sistema
1. El operador simple selecciona la opción “procesar datos”.	2. El sistema muestra la lista de opciones: <ul style="list-style-type: none"> a. Generar ficheros con datos b. Desencriptar fichero. c. Propiedades de fichero.
3. Selecciona la opción “propiedades de fichero”.	4. El sistema muestra una ventana con el botón “Buscar” para seleccionar el fichero deseado y una sección de la ventana con el botón “Ver” y la lista de las propiedades a mostrar del fichero: <ul style="list-style-type: none"> a. Nombre b. Tamaño c. Cantidad de líneas d. Tipo de archivo. Además los botones “Cerrar” y “Guardar”.
5. Presiona “Buscar”.	6. El sistema brinda la posibilidad de buscar cualquier fichero existente en la PC.
7. Selecciona el fichero del cual desea conocer las propiedades. 8. Selecciona la opción “Ver”.	9. Muestra las propiedades del fichero.
10. Si selecciona “Guardar” Ver Sección Guardar, si no continúa con el flujo básico. 11. Selecciona la opción cerrar.	12. Finaliza la ejecución del caso de uso.
Sección: “Guardar”	
Acción del Actor	Respuesta del Sistema
10. Selecciona la opción guardar.	11. Verifica que exista un fichero seleccionado, si no hay fichero seleccionado ver flujo alterno 1, si existe un archivo seleccionado, muestra una ventana para seleccionar la dirección donde se desea guardar el fichero, nombrarlo y asignarle la extensión correspondiente con el tipo de fichero que se desee.
12. Selecciona la dirección donde desea guardar el fichero. 13. Introduce el nombre y la extensión del fichero.	14. El sistema guarda el fichero con el nombre y la extensión en la dirección especificada.

15. Regresa a la acción 9 del flujo básico.	
Flujo Alternativo 1: “No hay fichero seleccionado”	
Acción del Actor	Respuesta del Sistema
	11. Muestra un cartel informando que debe seleccionar un fichero o introducir manualmente su dirección y el botón “Aceptar”.
12. Selecciona el botón aceptar del mensaje, continúa con la acción 4 del flujo básico.	
Poscondiciones de Éxito:	Selección correcta del fichero.
Poscondiciones de Fallo:	Que no exista el fichero seleccionado o que la dirección sea incorrecta.

Tabla 3.7. Descripción textual del caso de uso del sistema Generar Resumen.

ANEXO 2: Diagramas de clases del análisis.

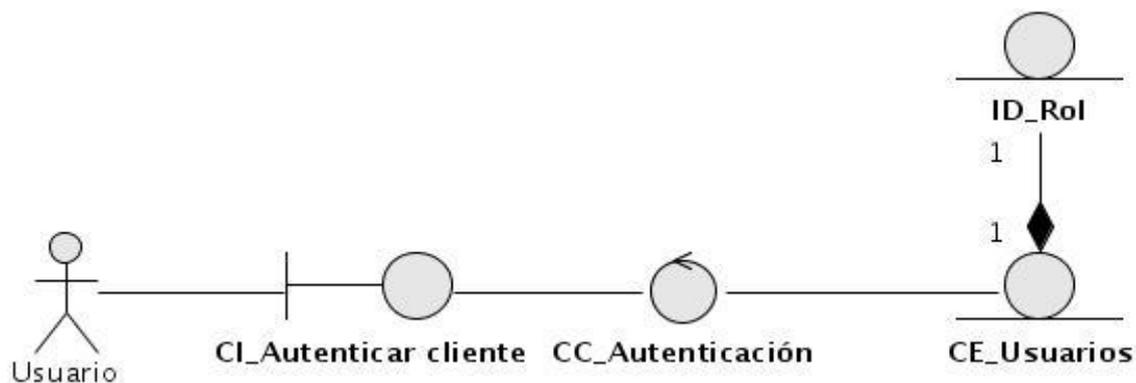


Figura 4.10. Diagrama de clases del análisis para Autenticar usuario.

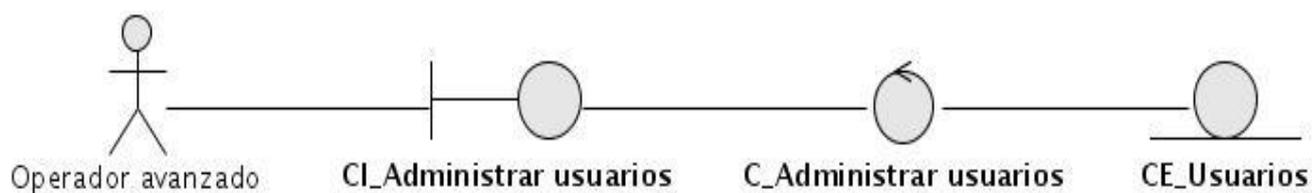


Figura 4.11. Diagrama de clases del análisis para Administrar usuarios clientes.

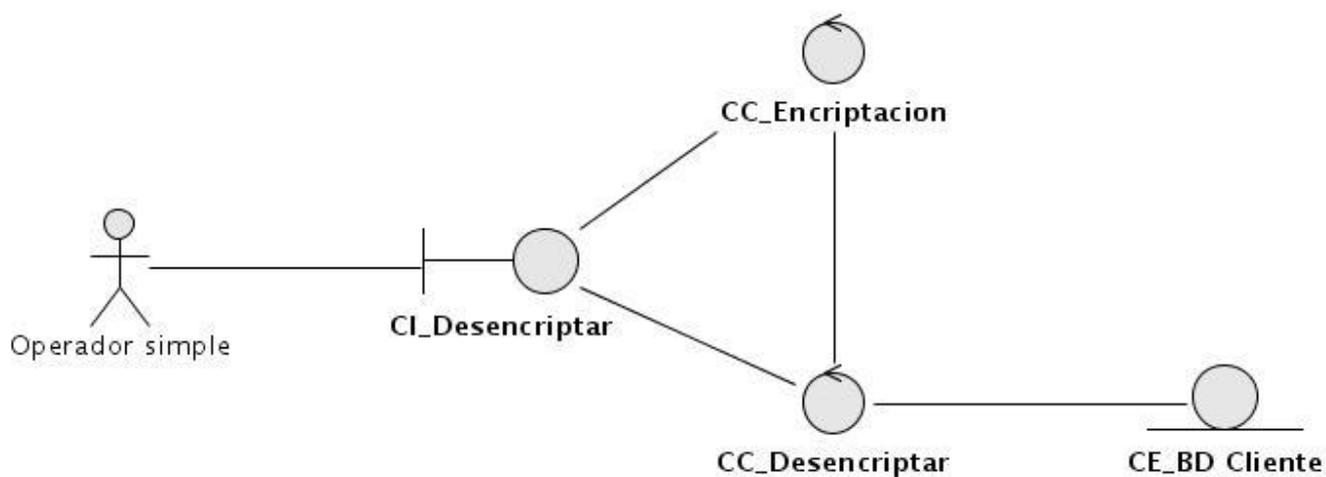


Figura 4.12. Diagrama de clases del análisis para Desencriptar.

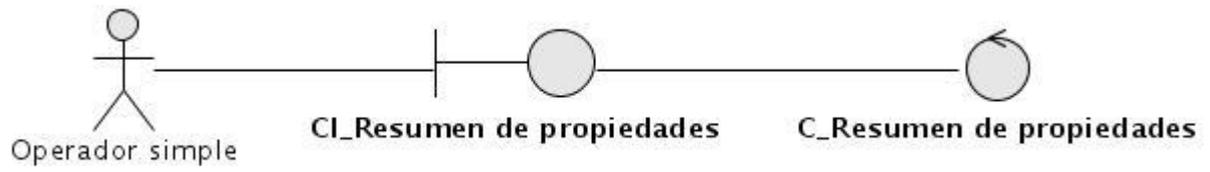


Figura 4.13. Diagrama de clases del análisis para Generar resumen.

ANEXO 3: Diagramas de clases del diseño.



Figura 4.14. Diagrama de clases del diseño para Autenticar usuario: superior.

Capítulo 4: Análisis y diseño del sistema

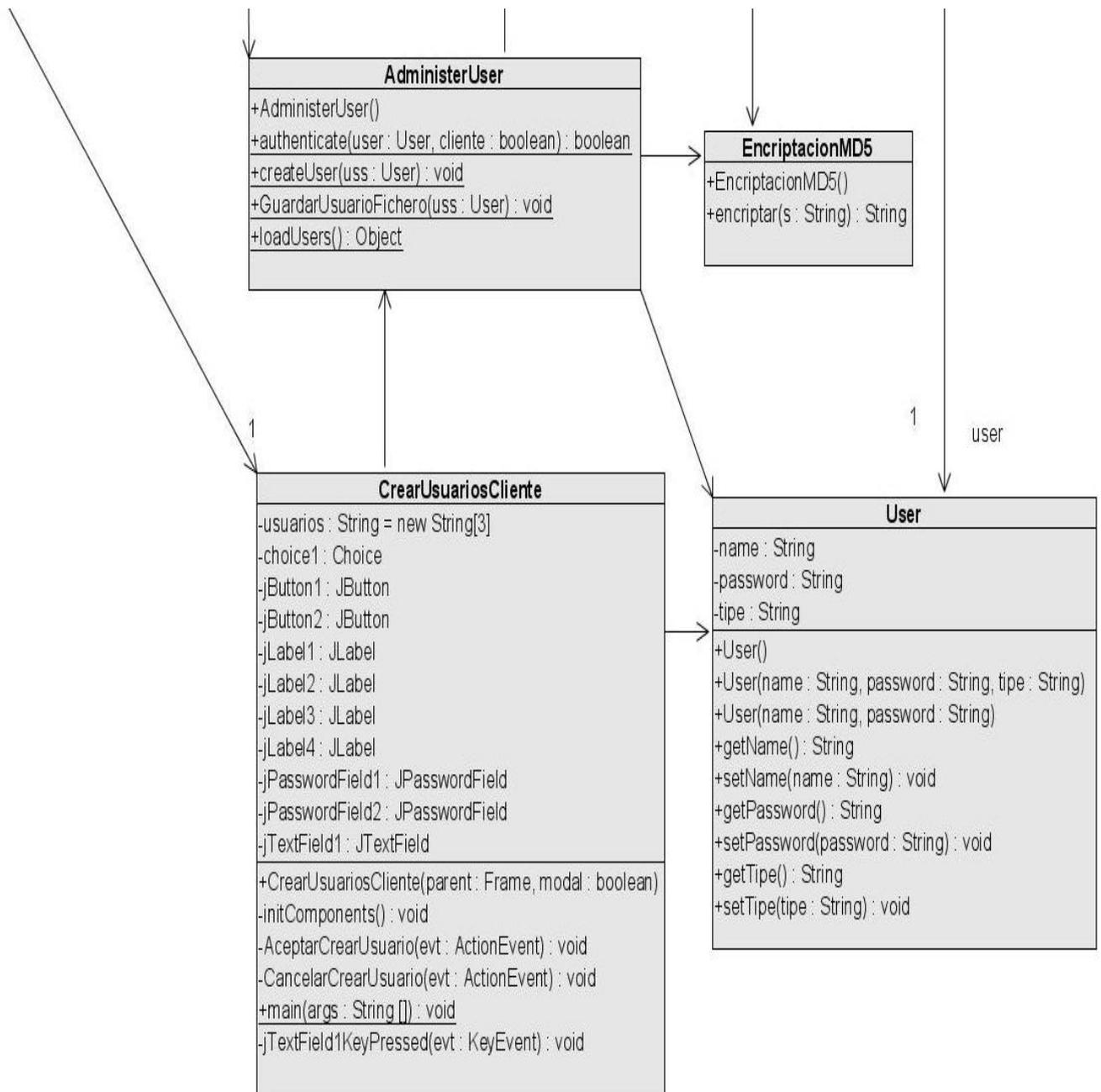


Figura 4.15. Diagrama de clases del diseño para Autenticar usuario: inferior.

Capítulo 4: Análisis y diseño del sistema



Figura 4.16. Diagrama de clases del diseño para Administrar usuarios clientes: superior.

Capítulo 4: Análisis y diseño del sistema

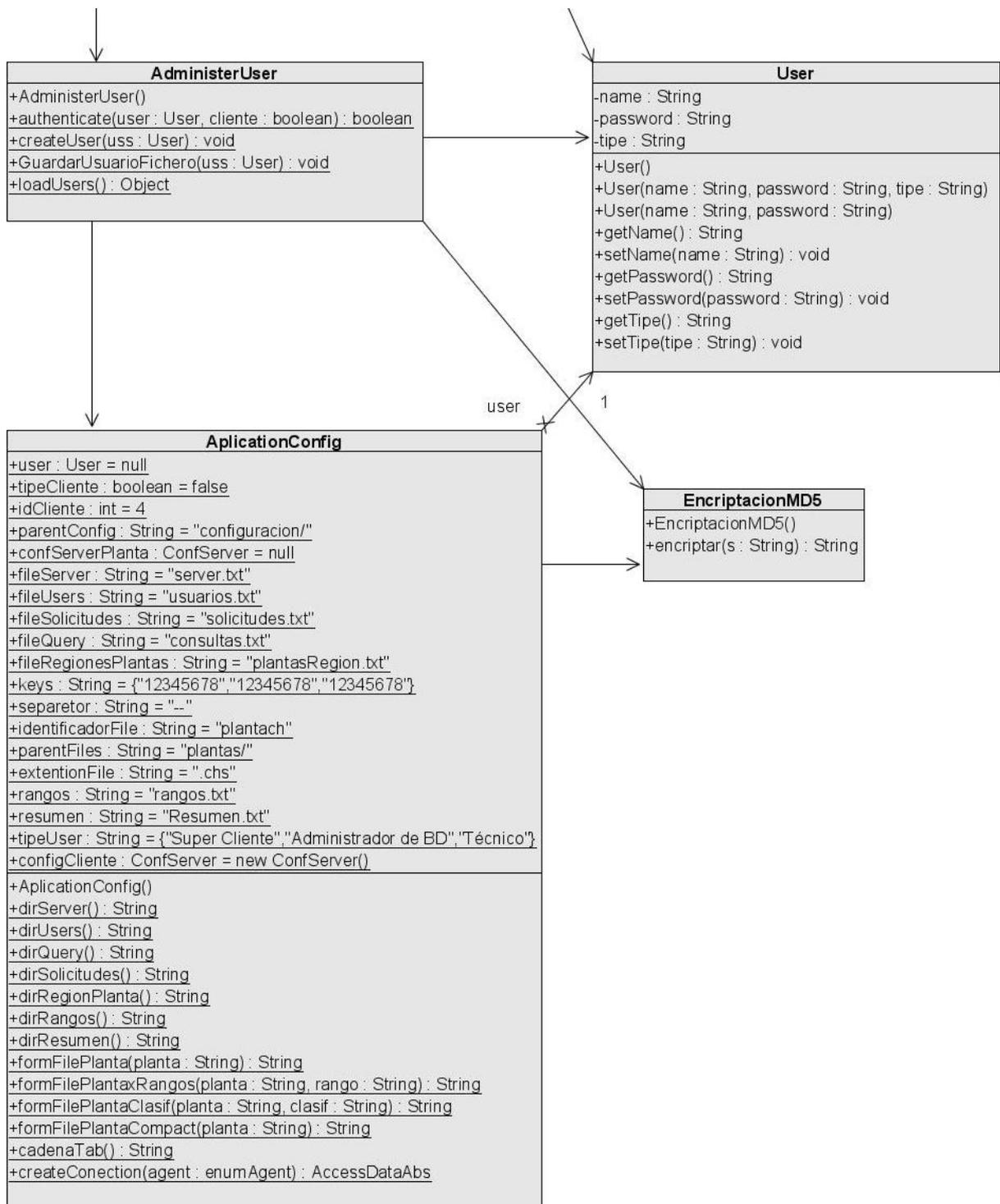


Figura 4.17. Diagrama de clases del diseño para Administrar usuarios clientes: inferior.

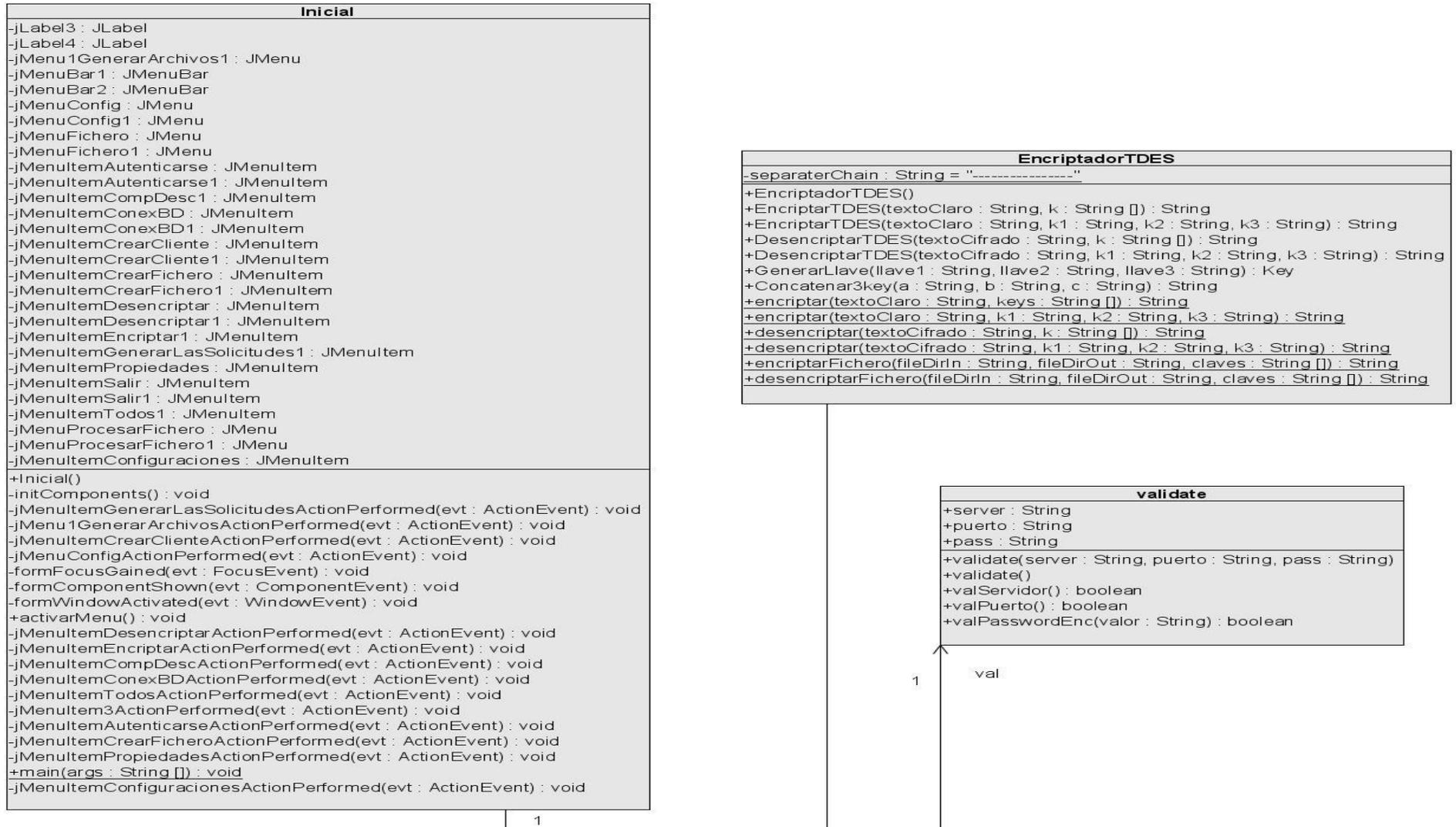


Figura 4.18. Diagrama de clases del diseño para Desencriptar: superior.

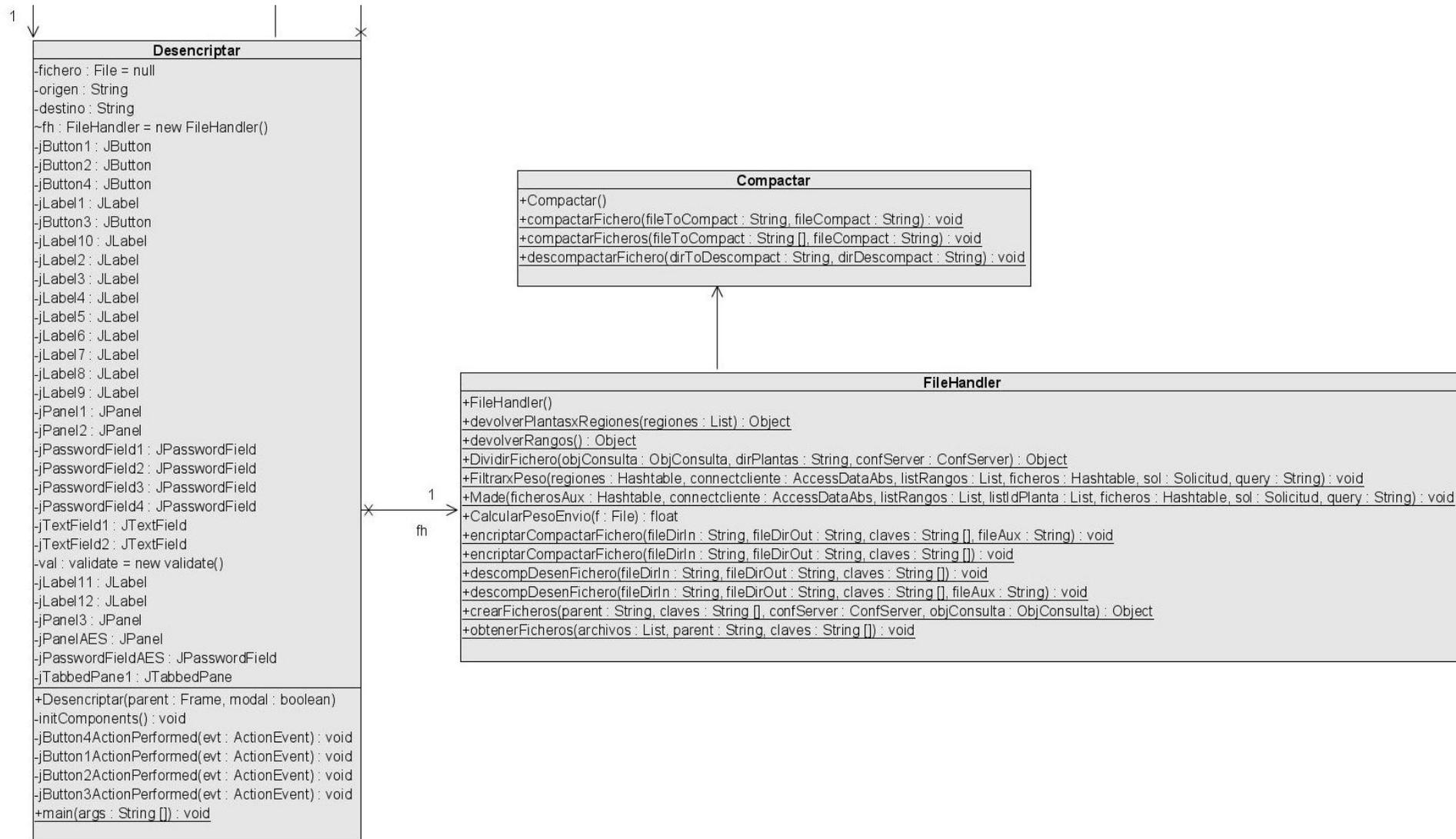


Figura 4.19. Diagrama de clases del diseño para Desencriptar: inferior.

Capítulo 4: Análisis y diseño del sistema

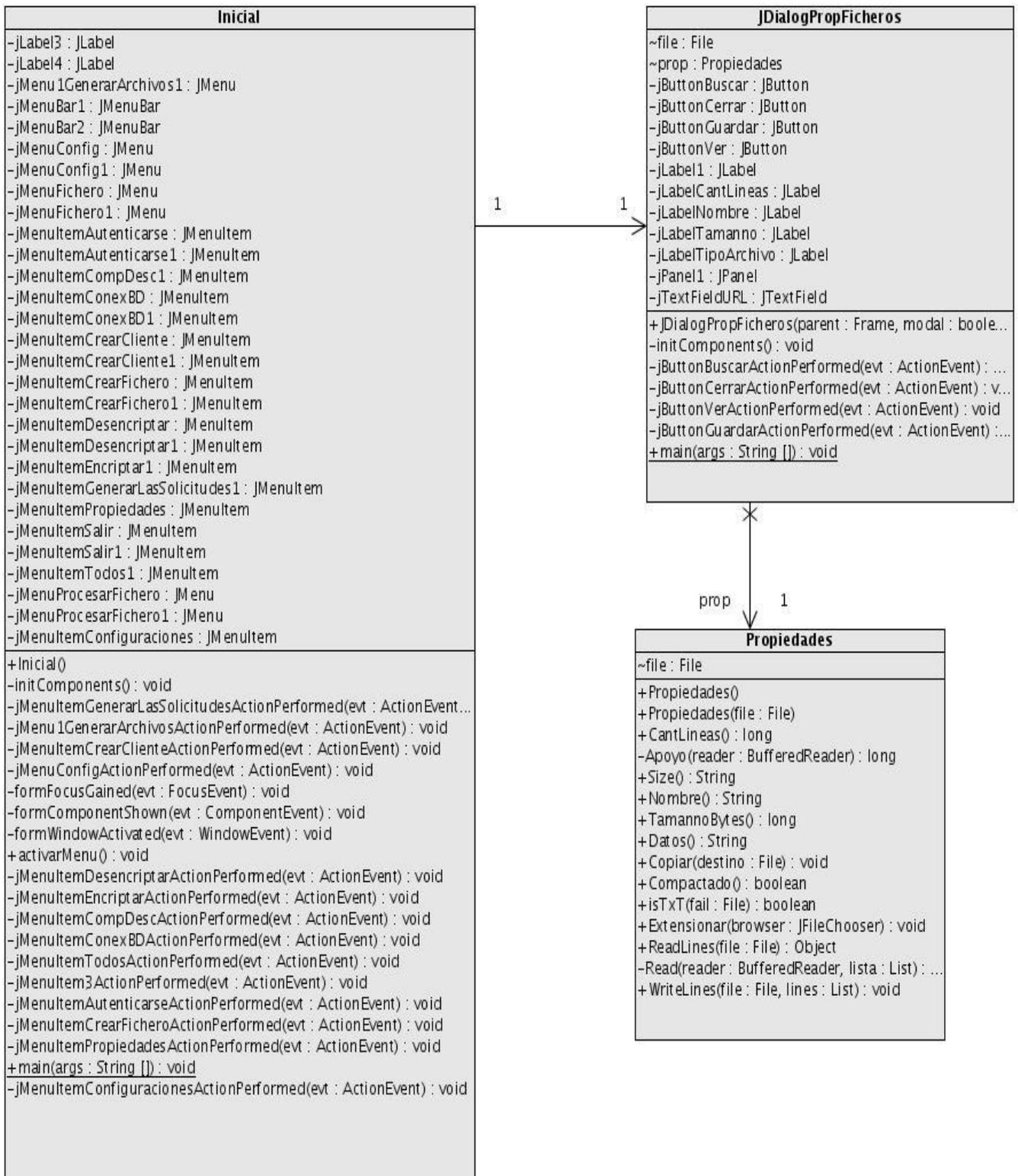


Figura 4.20. Diagrama de clases del diseño para Generar resumen de propiedades.

ANEXO 4: Diagramas de secuencia.

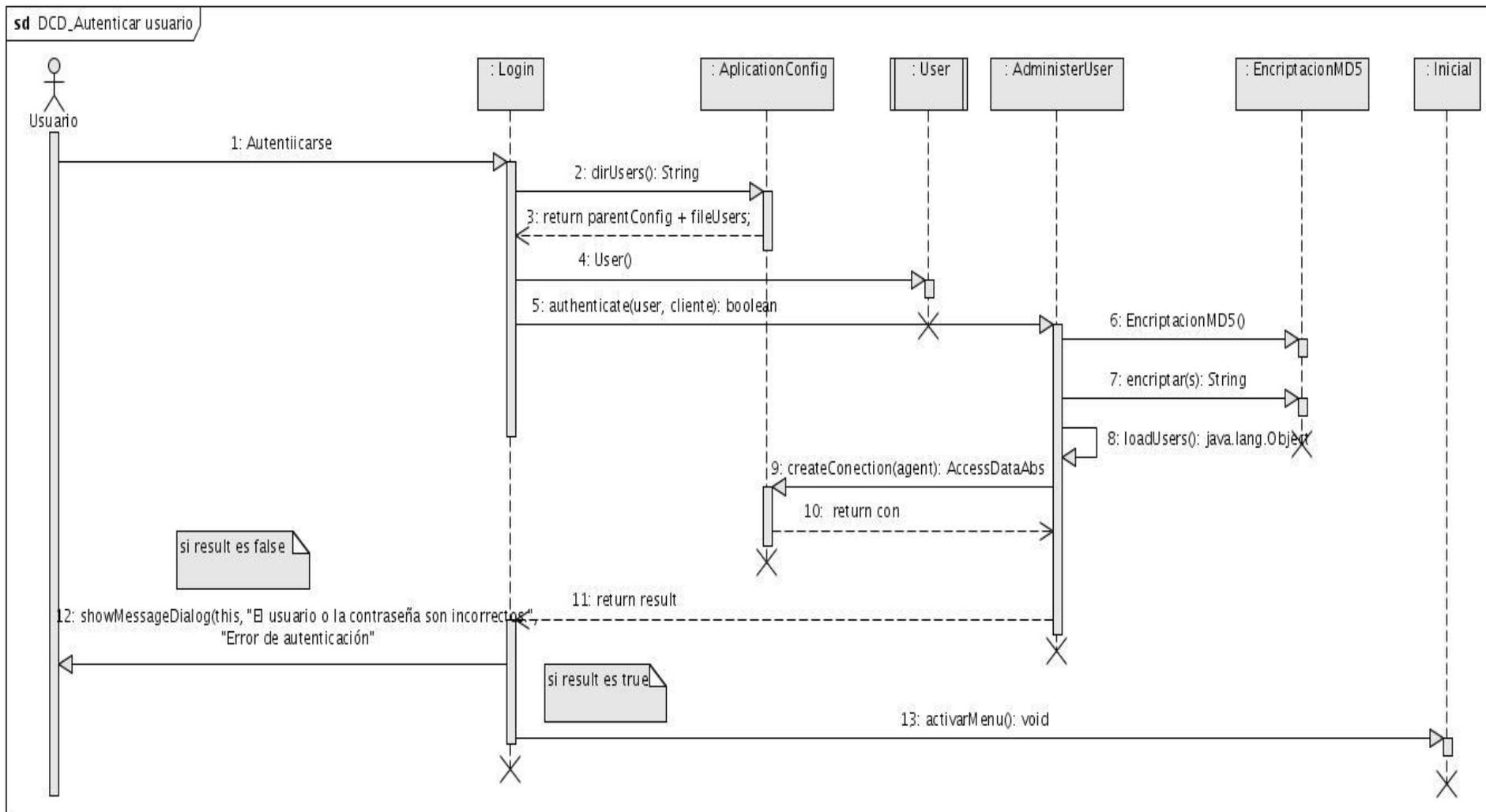


Figura 4.21. Diagrama de secuencia para Autenticar usuario.

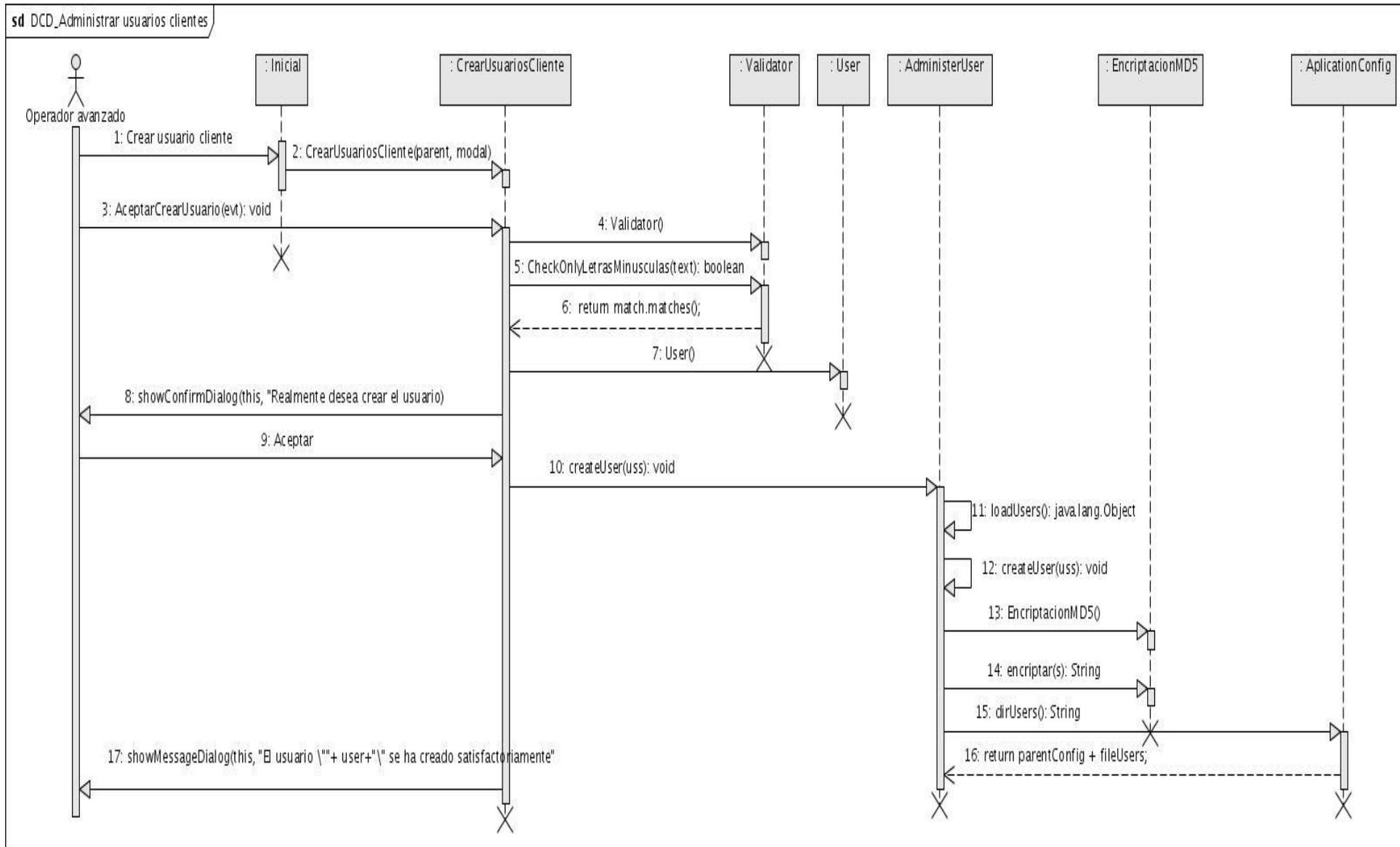


Figura 4.22. Diagrama de secuencia para Administrar usuarios clientes.

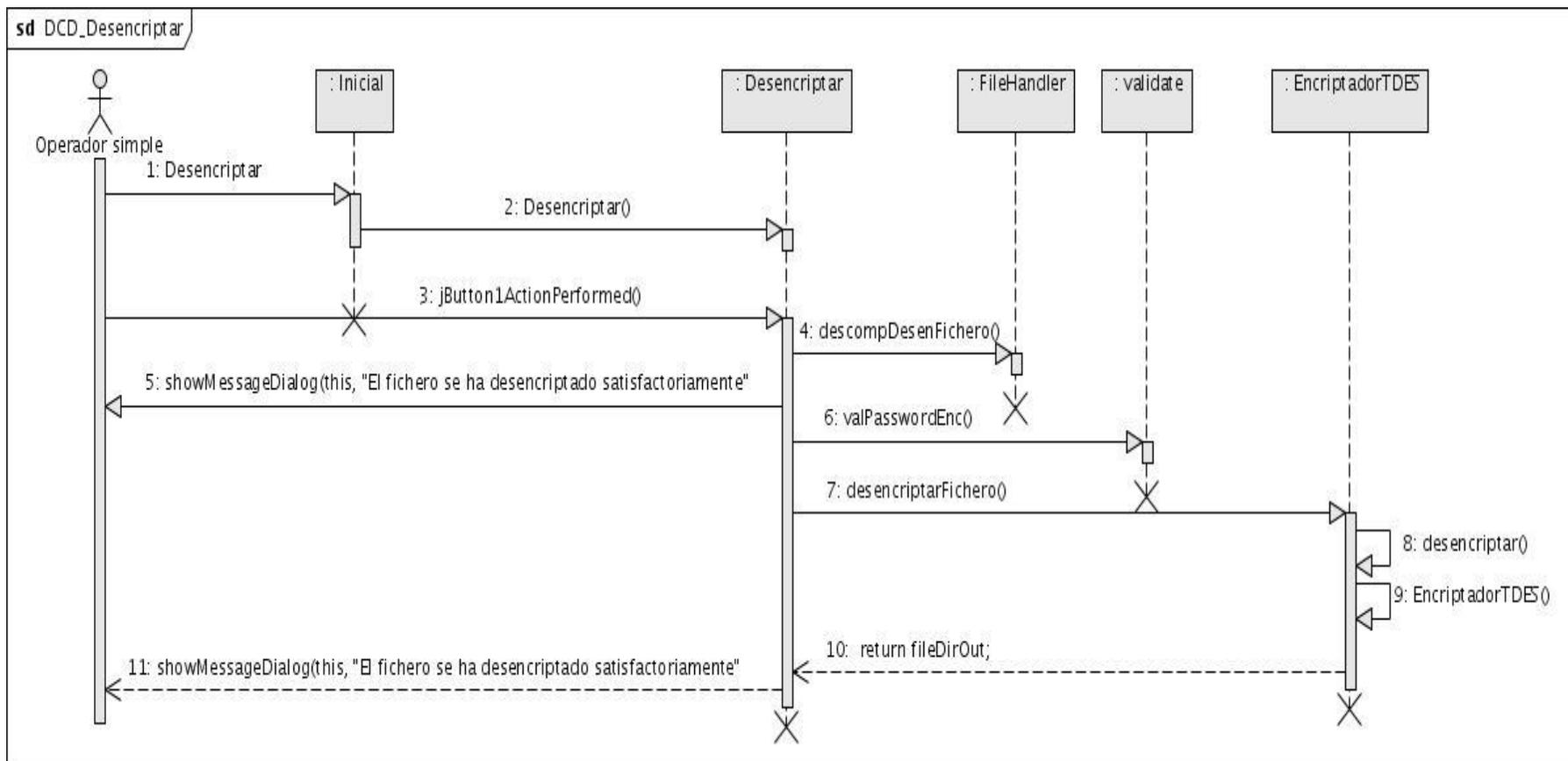


Figura 4.23. Diagrama de secuencia para Desencriptar.

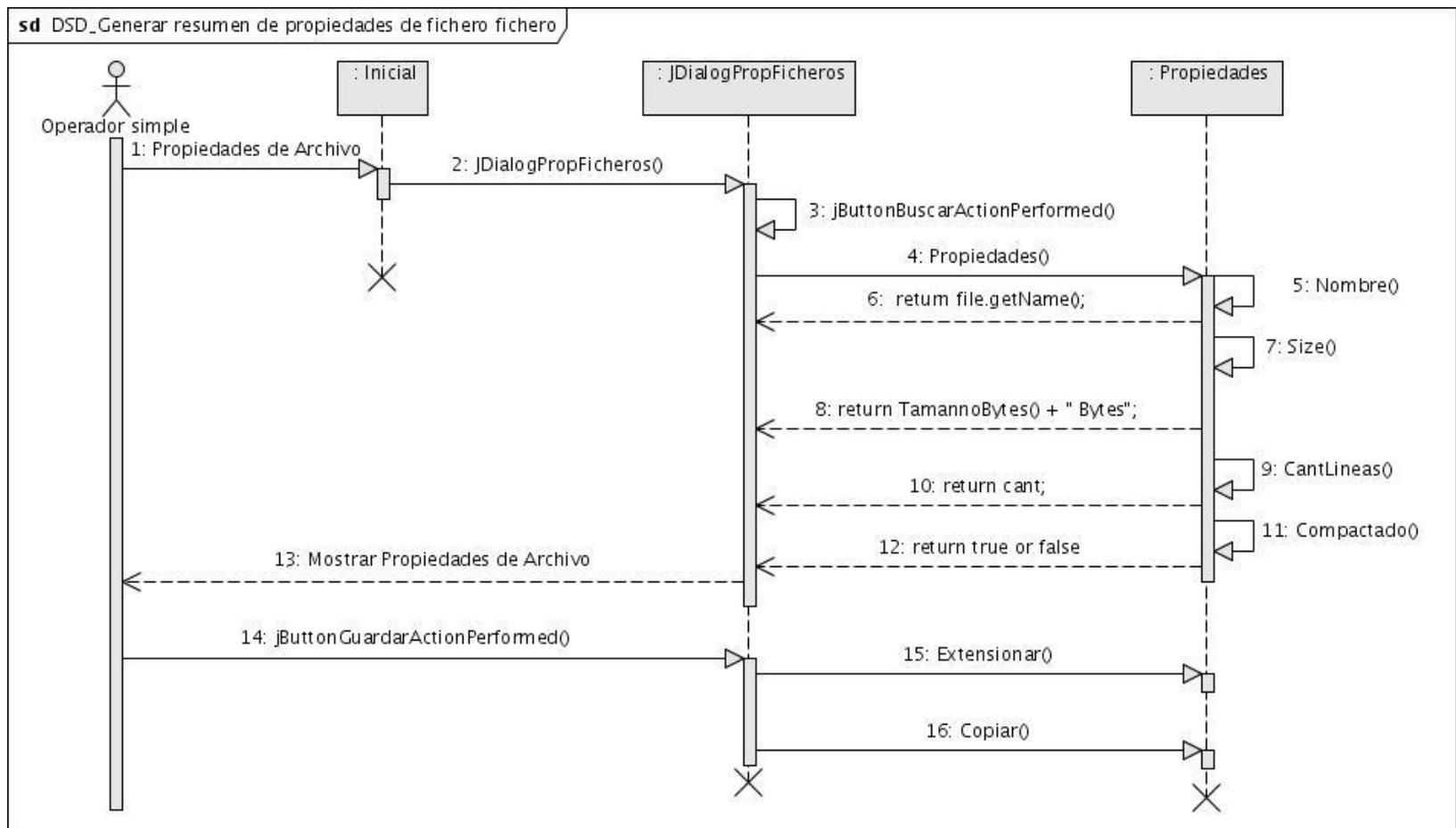


Figura 4.24. Diagrama de secuencia para Generar resumen de propiedades de fichero.