



UNIVERSIDAD CENTRAL DE VENEZUELA  
FACULTAD DE CIENCIAS  
ESCUELA DE COMPUTACIÓN  
CENTRO DE ENSEÑANZA ASISTIDA POR COMPUTADOR - CENEAC

**Trabajo Especial de Grado:**  
**Plataforma de envío, recepción y gestión**  
**de mensajes multimedia sobre ambiente Web,**  
**Tecnologías de Código Abierto y Módems GSM.**

**Trabajo Especial de Grado presentado ante la ilustre**  
**Universidad Central de Venezuela por el**  
**Br. Carlos Alberto Izquierdo Sánchez.**  
**CI 18.032.508**

**Tutor:**  
**Profa. Yusneyi Carballo Barrera**

**Julio 2011**

# Acta

Quienes suscriben, miembros del jurado designado por el Consejo de Escuela de Computación, para examinar el Trabajo Especial de Grado titulado “Plataforma de envío, recepción y gestión de mensajes multimedia sobre ambiente Web, Tecnologías de Código Abierto y Módems GSM” y presentado por el Br. Carlos Alberto Izquierdo Sánchez, de Cédula de Identidad 18.032.508 a los fines de optar al título de **Licenciado en Computación**, dejamos constancia de lo siguiente:

Leído como fue dicho trabajo, por cada uno de los miembros del jurado, se fijó el día 04 de Agosto de 2011, a las 3:00 pm., para que el autor lo defendiera en forma pública, lo que este hizo en la Sala PA III de la Escuela de Computación, mediante una presentación oral de su contenido, luego de lo cual respondió a las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobar con la nota de        puntos.

En fe de lo cual se levanta la presente Acta, en Caracas el día 04 de Agosto de 2011.

Dejando también constancia de que actuó como coordinador del jurado en Profa. Yusneyi Carballo Barrera.

---

Profa. Yusneyi Carballo Barrera

Tutora

---

Prof. Daniel Villavicencio  
(Jurado)

---

Prof. Ana Morales  
(Jurado)

## **Resumen**

La finalidad del presente Trabajo Especial de Grado consiste en el desarrollo de una plataforma que contemple e integre estos dos componentes: una herramienta con la capacidad de enviar, recibir, recuperar e interpretar mensajes multimedia mediante la comunicación con un modem GSM que se conectará a la red de la operadora móvil y una aplicación Web que permita gestionar los contenidos multimedia y registros transaccionales asociados, además de proveerle una interfaz al usuario final. Para ello, se aplicó una adaptación del modelo de desarrollo en cascada, bajo una metodología Ad Hoc, la cual se realizó en cinco fases: descripción de requerimientos, análisis, diseño, implementación y, finalmente, la fase de pruebas de la plataforma desarrollada. Como resultado final se obtuvo un sistema de fácil manejo, con la capacidad de realizar envíos y lectura de mensajes multimedia de forma masiva para cualquier usuario vía Web.

## **Palabras clave**

Servicio de Mensajería Multimedia (MMS), plataforma de envío y recepción MMS, operadora móvil, MMS Center (MMSC), red GPRS, modem GSM, comando AT, Unidad de Protocolo de Datos (PDU).

# Índice de Contenido

Glosario.....	XI
Introducción .....	14
Capítulo I: Problema de Investigación.....	16
1.1    Título .....	16
1.2    Planteamiento del Problema .....	16
1.3    Objetivos .....	17
1.3.1    Objetivo General.....	17
1.3.2    Objetivos Específicos .....	17
1.4    Justificación e Importancia.....	18
1.5    Esquema General de la Solución .....	19
1.6    Alcance .....	21
1.7    Limitaciones .....	22
Capítulo II: Marco Conceptual.....	23
2.1    Sistema Global para las Comunicaciones Móviles (GSM, Global System for Mobile Communications) .....	23
2.1.1    Definición.....	23
2.1.2    Especificaciones.....	24
2.1.3    Tarjeta SIM .....	25
2.1.4    Sistema de Mensajes Cortos (SMS, Short Message Service) .....	26
2.2    Servicio General de Paquetes vía Radio (GPRS, General Packet Radio Service) .....	26
2.2.1    Definición.....	26
2.3    Sistema de Mensajería Multimedia (MMS, Multimedia Messaging System) .....	28
2.3.1    Definición.....	28
2.3.2    Parámetros a configurar para el envío de Mensajes Multimedia .....	29
2.3.2.1    Centro MMS (MMSC, Multimedia Messaging System Centre).....	29
2.3.2.2    Nombre de Punto de Acceso (APN, Access Point Name) .....	30
2.3.2.3    Pasarela WAP (WAP Gateway) .....	30
2.3.3    Estructura MMS.....	31
2.3.4    Proceso de Envío y Recepción .....	35
2.4    Modem .....	39

2.4.1	Definición.....	39
2.4.2	Modem GSM.....	39
2.4.3	Enfora GSM1218 SA-GL .....	39
2.5	Comandos AT .....	43
2.5.1	Definición.....	43
2.5.2	Notación .....	44
2.5.3	Lista de Comandos AT más utilizados.....	45
2.5.4	Ejemplos de secuencias de comandos AT .....	51
2.5.5	Comandos AT como herramienta para la programación avanzada .....	54
2.6	Aplicación Web.....	55
2.6.1	Definición.....	55
2.7	Lenguaje de Marcado de Hipertexto (HTML, HyperText Markup Language) .....	55
2.7.1	Definición.....	55
2.8	Hojas de Estilo en Cascada (CSS, Cascade Style Sheet).....	57
2.8.1	Definición.....	57
2.8.2	Sintaxis.....	57
2.8.3	Ventajas y Desventajas .....	59
2.9	JavaScript.....	60
2.9.1	Definición.....	60
2.9.2	Ventajas y Desventajas .....	60
2.10	Preprocesador de Hipertexto, PHP .....	62
2.10.1	Definición .....	62
2.10.2	Ventajas y Desventajas.....	63
2.11	MySQL .....	64
2.11.1	Definición .....	64
2.12	Java .....	65
2.12.1	Definición .....	65
2.12.2	Características fundamentales .....	66
Capítulo III: Marco Aplicativo .....		67
3.1	Modelo de desarrollo en cascada .....	68
3.1.1	Definición.....	68
3.1.2	Etapas .....	69
3.2	Envío y Recepción MMS .....	71
3.2.1	Requerimientos .....	71
3.2.2	Análisis.....	71

3.2.2.1	Diagrama de Casos de Uso .....	72
3.2.2.2	Modelo Objetos del Dominio .....	75
3.2.3	Diseño .....	76
3.2.3.1	Diagrama de Clases .....	76
3.2.3.2	Modelo Entidad Relación .....	81
3.2.3.3	Entidades de la Base de Datos del Sistema .....	83
3.2.3.4	Diagrama de Secuencia .....	88
3.2.4	Implementación.....	93
3.2.4.1	Descripción de la Ventana gráfica del motor.....	93
3.2.4.1	Justificación del uso de Java y MySQL .....	95
3.2.4.2	Archivo de configuración de la aplicación motor .....	96
3.2.4.3	Envío de mensajes.....	98
3.2.4.4	Comunicación con el modem GSM .....	100
3.2.4.5	Decodificación de Notificaciones MMS.....	101
3.2.4.6	Recuperación de contenido multimedia .....	102
3.2.5	Pruebas y Resultados.....	103
3.2.5.1	Envío de mensajes (Aplicación – Teléfono Móvil) .....	103
3.2.5.2	Recepción de mensajes (Teléfono Móvil - Aplicación) .....	106
3.2.5.3	Envío y Recepción de mensajes (Aplicación – Aplicación).....	109
3.3	Administración de Contenidos MMS y Registro Transaccional .....	110
3.3.1	Requerimientos .....	110
3.3.2	Análisis.....	111
3.3.2.1	Diagrama de Casos de Uso .....	111
3.3.2.2	Modelo Objetos del Dominio .....	114
3.3.3	Diseño .....	115
3.3.3.1	Diagrama de Secuencia .....	115
3.3.4	Implementación.....	120
3.3.4.1	Descripción del módulo de Envío Web.....	120
3.3.4.2	Descripción del módulo de Configuración Operadora .....	124
3.3.4.3	Descripción del módulo Historial de Mensajes Enviados .....	125
3.3.4.4	Descripción del módulo Historial de Mensajes Recibidos .....	127
3.3.4.5	Descripción del módulo Registro de Usuario .....	128
3.3.5	Pruebas de Integración y Resultados.....	129
3.3.5.1	Envío MMS .....	129
3.3.5.2	Recepción MMS .....	133
Conclusiones y Recomendaciones.....		136
Referencias Bibliográficas.....		138

# Índice de Figuras

Figura 1: Esquema estructural de la solución planteada .....	19
Figura 2: Estructura General de un Mensaje Multimedia .....	31
Figura 3: Flujos de transaccionales generales para el intercambio de un mensaje multimedia .....	36
Figura 4: Flujo de transacciones en el envío de un mensaje multimedia .....	37
Figura 5: Flujo de transacciones en la recepción de un nuevo mensaje multimedia .....	38
Figura 6: Modem Enfora GSM1218 SA-GL .....	42
Figura 7: Comando AT petición .....	44
Figura 8: Respuesta correcta a un comando AT.....	45
Figura 9: Respuesta de error a un comando AT.....	45
Figura 10: Esquema estructural de una aplicación Web en PHP .....	62
Figura 11: Etapas del modelo en cascada a ser implementado.....	69
Figura 12: Diagrama de casos de uso del proceso de envío y recepción MMS .....	72
Figura 13: Diagrama de objetos del dominio del proceso de envío y recepción MMS .....	75
Figura 14: Diagrama de Clases Proceso Envío y Recepción MMS (Superior Izquierdo).....	77
Figura 15: Diagrama de Clases Proceso Envío y Recepción MMS (Inferior Izquierdo) .....	78
Figura 16: Diagrama de Clases Proceso Envío y Recepción MMS (Superior Derecho) .....	79
Figura 17: Diagrama de Clases Proceso Envío y Recepción MMS (Inferior Derecho).....	80
Figura 18: Diagrama Entidad-Relación de la base de datos del sistema.....	82
Figura 19: Diagrama de Secuencia del proceso de envío MMS .....	89
Figura 20: Diagrama de Secuencia del proceso de recepción MMS (parte 1) .....	90
Figura 21: Diagrama de Secuencia del proceso de lectura MMS (parte 2).....	91
Figura 22: Diagrama de secuencia de proceso envío MMS de prueba .....	92
Figura 23: Ventana de la Aplicación Motor MMS .....	93
Figura 24: Archivo de configuración de la aplicación motor ( <i>sendmms.properties</i> ) .....	96
Figura 25: Código de envío de un mensaje multimedia (SendMms.java).....	99
Figura 26: Código de establecimiento de la comunicación con el modem GSM (Communicator32.java)....	100
Figura 27: Estructura de una notificación MMS .....	101
Figura 28: Recuperación de contenido multimedia del mensaje recibido .....	102
Figura 29: Mensaje de error en solicitud de recuperación del mensaje.....	108

Figura 30: Modelo de casos de uso de la aplicación Web .....	111
Figura 31: Diagrama de objetos del dominio de la Aplicación Web .....	114
Figura 32: Diagrama de secuencia del proceso Envío Web .....	115
Figura 33: Diagrama de secuencia de proceso Configurar Operadora .....	116
Figura 34: Diagrama de secuencia de proceso consultar Historial de Enviados .....	117
Figura 35: Diagrama de secuencia de proceso consultar Historial de Recibidos .....	118
Figura 36: Diagrama de secuencia del proceso Registro de Usuario .....	119
Figura 37: Vista del módulo de Envío Web .....	120
Figura 38: Alerta de error en Envío Web.....	121
Figura 39: Vista del módulo de Iniciar Sesión .....	121
Figura 40: Vista de mensaje de error en Iniciar Sesión .....	122
Figura 41: Vista del módulo de Envío Web después de la autenticación del usuario .....	123
Figura 42: Vista del detalle de un envío en el módulo Envío Web.....	124
Figura 43: Vista del módulo Configurar Operadora .....	124
Figura 44: Vista de módulo Configurar Operadora una vez seleccionada .....	125
Figura 45: Vista del módulo Historial de Mensajes Enviados .....	126
Figura 46: Vista del detalle de un envío en el módulo Historial de Mensajes Enviados.....	127
Figura 47: Vista del módulo Historial de Mensajes Recibidos .....	128
Figura 48: Ventana de Registro de Usuario.....	129



# Índice de Tablas y Gráficos

Tabla 1: Parámetros del encabezado MMS .....	33
Tabla 2: Valores del parámetro <i>content-type</i> en el encabezado MMS.....	34
Tabla 3: Parámetros del encabezado de un <i>body part</i> .....	35
Tabla 4: Comparación entre los modems GPRS Enfora Spider y Sierra Wireless Airlink .....	42
Tabla 5: Descripción de caso de uso Enviar MMS pendientes .....	73
Tabla 6: Descripción de caso de uso Leer MMS recibidos .....	74
Tabla 7: Enviar MMS de prueba .....	74
Tabla 8: Descripción de los campos de T_EnvioBloque .....	83
Tabla 9: Descripción de los campos de T_Transmision.....	84
Tabla 10: Descripción de los campos de T_Respuesta.....	85
Tabla 11: Descripción de los campos de T_Usuario.....	86
Tabla 12: Descripción de los campos de T_Mms .....	86
Tabla 13: Descripción de los campos de T_Parts .....	87
Tabla 14: Descripción de los campos de C_Operadora.....	88
Tabla 15: Pruebas de envío aplicación – teléfono móvil (tamaño imagen 10KB).....	104
Tabla 16: Pruebas de envío aplicación – teléfono móvil (tamaño imagen 17KB).....	104
Tabla 17: Pruebas de envío aplicación – teléfono móvil (tamaño imagen 23KB).....	105
Tabla 18: Pruebas de envío aplicación – teléfono móvil (tamaño imagen 30KB).....	105
Tabla 19: Pruebas de recepción teléfono móvil – aplicación (tamaño imagen 10KB).....	107
Tabla 20: Pruebas de recepción teléfono móvil – aplicación (tamaño imagen 30KB).....	107
Tabla 21: Pruebas de envío y recepción aplicación – aplicación (tamaño imagen 10KB).....	109
Tabla 22: Descripción de caso de uso Enviar mensaje multimedia .....	112
Tabla 23: Descripción de caso de uso Registrar Usuario.....	112
Tabla 24: Descripción de caso de uso Consultar Historial de Enviados .....	113
Tabla 25: Descripción de caso de uso Consultar Historial de Recibidos .....	113
Tabla 26: Descripción de caso de uso Configurar Operadora .....	113
Tabla 27: Prueba de envíos multimedia masivos.....	130
Gráfico 1: Porcentaje de envíos exitosos y fallidos (Tamaño de imagen 10 KB) .....	130
Gráfico 2: Porcentaje de envíos exitosos y fallidos (Tamaño de Imagen 15 KB) .....	131

Gráfico 3: Porcentaje de envíos exitosos y fallidos (Tamaño de imagen 20 KB) .....	131
Gráfico 4: Porcentaje de envíos exitosos y fallidos (Tamaño de imagen 25 KB) .....	131
Gráfico 5: Comportamiento de la tasa de mensajes enviados fallidos en base al tamaño de la imagen.....	133
Tabla 28: Prueba de recepción multimedia masiva .....	134
Gráfico 6: Comportamiento de los fallos de recuperación multimedia en base al tamaño de la imagen ....	135

## Glosario

A continuación se suministra un glosario de términos a fin de hacer más sencilla la lectura del documento.

**AMR (Adaptative Multi-Rate):** formato de compresión de audio optimizado para la codificación de voz. AMR ha sido adoptado como el estándar de codificación de audio por la 3GPP en octubre de 1998 y actualmente se utiliza ampliamente en GSM. Gestiona dinámicamente el ancho de banda seleccionando entre ocho diferentes tasas de bits.

**Banda de frecuencia:** intervalos de frecuencias del espectro electromagnético asignados a diferentes usos dentro de las radiocomunicaciones.

**CSD (Circuit Switched Data):** forma original de transmisión de datos desarrollada para los sistemas de telefonía móvil basados en el acceso múltiple por división de tiempo (TDMA) como el GSM. CSD usa un intervalo de tiempo (time slot) de radio individual para enviar 9.6 kbit/s al subsistema de conmutación de red de GSM, donde podría conectarse por medio del equivalente a un módem normal a la red telefónica conmutada.

**EDGE (Enhanced Data Rates for GSM Evolution):** tecnología de la telefonía móvil celular, que actúa como puente entre las redes 2G y 3G. EDGE se considera una evolución del GPRS (General Packet Radio Service). Esta tecnología funciona con redes GSM. EDGE puede alcanzar una velocidad de transmisión de datos de 384 Kbps.

**EFR (Enhanced Full Rate):** estándar de codificación de voz desarrollado para mejorar la pobre calidad del codec FR de GSM. Tiene una velocidad 12,2 Kbps.

**Espectro:** medida de la distribución de amplitudes de cada frecuencia.

**FR (Full Rate):** primer estándar de codificación de voz digital usado en sistemas de telefonía móvil de GSM. La calidad de este codec de voz es bastante pobre para los estándares modernos. Tiene una velocidad de 13 Kbps.

**Gateway:** dispositivo, con frecuencia un ordenador, que permite interconectar redes con protocolos y arquitecturas diferentes a todos los niveles de comunicación. Su propósito es traducir la información del protocolo utilizado en una red al protocolo usado en la red de destino.

**Half Rate:** sistema de codificación de voz para GSM que opera a 5,6 Kbps, requiere la mitad del ancho de banda del codec Full Rate, duplica la capacidad de la red para tráfico de voz, a expensas de la calidad de audio. Recomendado cuando la batería del dispositivo es baja, ya que ahorra 30% más de energía.

**IMT-2000:** estándar global para la tercera generación de redes de comunicaciones inalámbricas 3G, definidas por un sistema de estándares de la Unión Internacional de Telecomunicaciones. IMT-2000 proporciona un marco para el acceso inalámbrico mundial uniendo los diversos sistemas de redes terrestres y satelitales.

**JDK (Java Development Kit):** software que provee herramientas de desarrollo para la creación de programas en java. Puede instalarse en una computadora local o en una unidad de red.

**JRE (Java Runtime Environment):** conformado por una Máquina Virtual de Java o JVM, un conjunto de bibliotecas Java y otros componentes necesarios para que una aplicación escrita en lenguaje Java pueda ser ejecutada. El JRE actúa como un "intermediario" entre el sistema operativo y Java.

**JVM (Java Virtual Machine):** máquina virtual de proceso nativo, es decir, ejecutable en una plataforma específica, capaz de interpretar y ejecutar instrucciones expresadas en un código binario especial (el Java bytecode), el cual es generado por el compilador del lenguaje Java.

**MIME (Multipurpose Internet Mail Extensions):** estándar que clasifica recursos y provee información a programas acerca de cómo manejarlos. Serie de convenciones o especificaciones dirigidas al intercambio a través de Internet de todo tipo de archivos (texto, audio, vídeo, etc.) de forma transparente para el usuario.

**Multimedia:** cualquier objeto o sistema que utiliza múltiples medios de expresión (físicos o digitales) para presentar o comunicar información (video, texto, imágenes, etc).

**PDU (Protocol Data Unit):** Unidades de Datos de Protocolo. Se utiliza para el intercambio entre unidades parejas, dentro de una capa del modelo OSI.

**Proxy:** programa o dispositivo que realiza una acción en representación de otro. El servidor proxy, es un ordenador que intercepta las conexiones de red que un cliente hace a un servidor de destino.

**Push:** estilo de comunicaciones sobre Internet donde la petición de una transacción se origina en el servidor.

**RS232:** interfaz que designa una norma para el intercambio serie de datos binarios entre un equipo terminal de datos (DTE) y un equipo de comunicación de datos (DCE). Consiste en un conector de 25 pines, aunque es normal encontrar la versión de 9 pines, más barato e incluso más extendido para cierto tipo de periféricos, como el ratón serie del PC.

**Roaming:** capacidad de cambiar de un área de cobertura a otra sin interrupción en el servicio o pérdida en conectividad. Permite a los usuarios seguir utilizando sus servicios de red inalámbrica cuando viajan fuera de la zona geográfica en la que contrataron el servicio.

**Throughput:** volumen de trabajo o de información que fluye a través de un sistema. Así también se le llama al volumen de información que fluye en las redes de datos. Particularmente significativo en almacenamiento de información y sistemas de recuperación de información, en los cuales el rendimiento es medido en unidades como accesos por hora.

**WAP (Wireless Application Protocol):** estándar seguro que permite que los usuarios accedan a información de forma instantánea a través de dispositivos inalámbricos como PDAs, teléfonos móviles, etc.

# Introducción

Las necesidades de comunicación de las personas son muy diversas y existen situaciones en las que lo que requerimos es comunicarnos de una forma barata, rápida y precisa. Para satisfacer esta necesidad sobre telefonía móvil, se ha desarrollado el sistema de mensajes cortos o SMS, que se ha convertido sorprendentemente en el servicio móvil de datos que más beneficios genera para las operadoras de telefonía móvil GSM.

Con la base de un modelo de negocio exitoso en los SMS y como evolución tecnológica, el servicio de mensajería multimedia o MMS tiene un fuerte potencial para convertirse en un servicio de datos masivo, atrayendo a todos los actuales usuarios del SMS y ampliando los límites del servicio. La mensajería multimedia ofrecerá nuevas oportunidades de negocio y nuevas formas de ingresos a las operadoras y los proveedores de contenidos.

La empresa ALS TELECOM C.A. actualmente provee servicio de envío de mensajes de texto (SMS) de alto volumen a compañías terceras. Sin embargo, actualmente la empresa desea expandir su plataforma para incorporar el servicio de mensajes multimedia (MMS) para satisfacer las nuevas necesidades de los clientes, yendo de la mano con el crecimiento tecnológico de las telecomunicaciones, lo que se traducirá en una mayor productividad y satisfacción de sus clientes.

Por estas razones, se plantea como objetivo de la investigación, desarrollar una plataforma de fácil integración, para el envío y recepción de MMS y para la administración Web de los contenidos multimedia y registros transaccionales, utilizando tecnologías de telefonía móvil y tecnologías comunes disponibles para el desarrollo Web.

El documento resume esta investigación y presenta una solución, estando organizado de la siguiente forma:

En el Capítulo 1 se describe el problema de la investigación, tratando los puntos de: planteamiento del problema, objetivos de la investigación, esquema general de la solución, justificación e importancia, alcance de la investigación y limitaciones.

El Capítulo 2 presenta el marco conceptual donde se muestran las bases en las que se fundamenta el desarrollo de este trabajo. Se describen los protocolos de red, estándares y las tecnologías involucradas en

el desarrollo como GSM, GPRS, SMS, MMS, Comandos AT (*Attention*) y otros conceptos necesarios para la comprensión del funcionamiento de la mensajería multimedia. Adicionalmente, se detalla y explican las herramientas y lenguajes de programación a utilizar para la elaboración de la plataforma en cuestión (HTML, CSS, JavaScript, PHP, MySQL y Java).

El capítulo 3 está constituido por el marco aplicativo, en el cual se describe el conjunto de etapas realizadas para la construcción del sistema y la metodología utilizada para dicho desarrollo. En este capítulo, el desarrollo de la plataforma se divide en dos procesos, la herramienta motor para el envío y recepción MMS y la aplicación Web que provee la interfaz para el usuario final y la administración de los contenidos multimedia. Cada proceso posee sus fases de desarrollo comenzando con la recopilación de requerimientos; análisis del problema formalizado a través del modelo de: casos de uso y objetos del dominio; diseño y modelado a través del: diagrama de clases, diagrama de base de datos, y diagrama de secuencia; y, finalmente, se especifica cómo ha sido la implementación de la solución y las pruebas realizadas para comprobar su correcto funcionamiento.

Por último se presentan las conclusiones y recomendaciones a las cuales condujo el desarrollo de este trabajo.

# Capítulo I: Problema de Investigación

En este capítulo se puntualiza el contexto del problema a solucionar, haciendo énfasis en la descripción del Sistema de Mensajería Multimedia, su estructura, funcionamiento, elementos que intervienen en el proceso de transmisión de los mensajes multimedia y los protocolos involucrados en este sistema. Se especifica el problema particular a resolver y una solución adecuada para la implementación de una plataforma que pueda satisfacer las necesidades de enviar y recibir mensajes multimedia de forma masiva sobre un ambiente Web

Luego se define el objetivo general y los objetivos específicos del Trabajo Especial de Grado, así como su alcance, la importancia y justificación del mismo, enfocando también los beneficios que genera a los actores involucrados en el problema.

## 1.1 Título

Plataforma de envío, recepción y gestión de mensajes multimedia sobre ambiente Web, Tecnologías de Código Abierto y Módems GSM.

## 1.2 Planteamiento del Problema

Las empresas especializadas en el desarrollo de aplicaciones móviles, entre otras ALS Telecom, buscan innovar de acuerdo a la identificación de necesidades del mercado venezolano.

Hoy en día, para los usuarios de dispositivos móviles, surge la necesidad de intercambiar información más completa y enriquecida que el texto simple proporcionado por los SMS, información que incluya contenido multimedia como imágenes, audio, o video.

Según la investigación previa realizada, hasta la fecha en el mercado venezolano no se disponen de herramientas o aplicaciones que permitan enviar y recibir mensajes multimedia (MMS) de forma masiva,



motivo por el cual requiere investigar e identificar los elementos necesarios para gestionar de forma automatizada los mensajes multimedia para el desarrollo de una solución.

Actualmente ALS Telecom cuenta con una plataforma Web para el envío y recepción de mensajes cortos de texto. La Empresa desea incorporar el servicio de envío y recepción de mensajes multimedia, por lo que se desea desarrollar un nuevo sistema que provea dicho servicio y facilite las labores ejecutadas diariamente dentro de la organización, en el área de producción.

Una plataforma Web para el envío y recepción de los MMS permitirá a las empresas y consumidores disponer de un servicio más innovador que les permita compartir información más completa y detallada que los mensajes cortos de texto (SMS) con sus clientes y colaboradores.

## **1.3 Objetivos**

### **1.3.1 Objetivo General**

Desarrollar una Plataforma de Gestión para mensajes multimedia (MMS) sobre un ambiente Web, que combine Tecnologías de Código Abierto y Módems GSM.

### **1.3.2 Objetivos Específicos**

1. Recolectar los requerimientos por parte de ALS Telecom, que permitan establecer los lineamientos de la aplicación y su alcance.
2. Diseñar lógica y físicamente una base de datos relacional que permita la administración, envío y recepción de mensajes MMS.
3. Realizar una evaluación de los comandos AT necesarios para la configuración del módem GSM que enviará y recibirá mensajes MMS.

4. Realizar una evaluación de las librerías de código abierto que implementen MMS y seleccionar la(s) que mejor se adapte a los parámetros requeridos por una operadora móvil celular venezolana.
5. Diseñar e implementar una aplicación que sirva de interfaz para comunicar en tiempo real los módems para envío y recepción MMS y la base de datos relacional, utilizando librerías de código abierto.
6. Diseñar e implementar una aplicación Web que permita acceder a la administración, envío y recepción de mensajes MMS.
7. Realizar las pruebas pertinentes a las aplicaciones desarrolladas.
8. Documentar la implementación de las funcionalidades del sistema.
9. Elaborar una documentación que permita administrar y mantener la aplicación.

## **1.4 Justificación e Importancia**

ALS Telecom posee una plataforma que provee un servicio de mensajería SMS de forma masiva a un conjunto de Empresas que necesitan transmitir información de propósito específico a sus clientes de manera rápida y confiable. La evolución de la tecnología, en específico de las telecomunicaciones genera un mayor nivel de exigencia por parte de los usuarios de dispositivos móviles, los cuales desean optimizar su proceso de comunicación y enriquecer los contenidos de la información que comparten con otros, demandando así el poder transmitir más que una simple cadena de caracteres.

En la investigación previa realizada, hasta la fecha, en Venezuela no se cuenta con sistemas ni compañías que provean un servicio de mensajería multimedia, es por esto que surge la necesidad del desarrollo de una plataforma que gestione mensajes de contenido más completo y de mayor impacto a la percepción de los usuarios móviles, como lo es el mensaje multimedia.

Con la implementación de este sistema, ALS Telecom contará con una plataforma que permitirá ofrecer soluciones que contemplen el envío y recepción de MMS, preparándose para satisfacer necesidades emergentes del mercado venezolano y potenciando así nuevas maneras de hacer publicidad intercambiando información colaborativa o de disfrute personal.

Si bien el mensaje multimedia y el correo electrónico poseen una estructura muy similar, las ventajas que brindan los MMS se indican a continuación:

- Los elementos multimedia contenidos en el correo son incluidos como adjuntos de manera independiente cada uno, mientras que en los MMS se les puede describir su comportamiento especificando la sincronización y secuencia entre estos, además de su ubicación en la pantalla del dispositivo móvil.
- El envío y recepción de los MMS se realiza a través de una conexión siempre activa (*always-on*), es decir que no se requiere de una autenticación para establecer la conexión con el proveedor del servicio, como en el caso del correo electrónico.
- No todos los teléfonos móviles que soportan MMS tienen la capacidad para navegar en Internet a través de conexiones de alto ancho de banda (conexiones 3G) para acceder o sincronizar sus cuentas de correo electrónico con el dispositivo de forma rápida.

## 1.5 Esquema General de la Solución

A continuación, se ilustran los diferentes elementos que la plataforma debe estar en capacidad de controlar para garantizar el correcto monitoreo, administración, envío y recepción de mensajes MMS, a través de sus diferentes elementos:

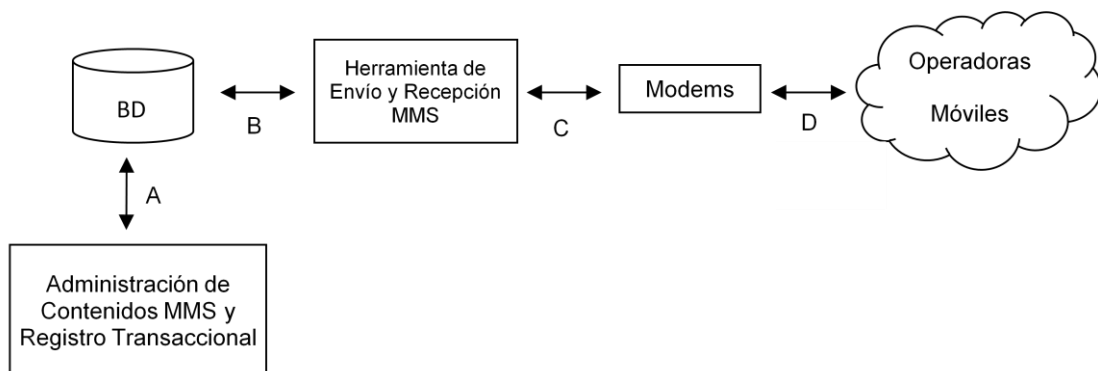


Figura 1: Esquema estructural de la solución planteada

Fuente: Elaboración propia

El proyecto de Trabajo Especial de Grado se orientará al análisis, diseño, implementación y verificación de la plataforma MMS incluyendo su Base de Datos, el Módulo de Envío y Recepción MMS y el Módulo Web de Administración de Contenidos MMS, así como los elementos de integración entre módulos desarrollos señalados con las letras A, B, C y D.

Entre la letra A y los módulos relacionados se contempla el desarrollo de una aplicación Web para la interacción con el usuario final, la cual incluirá las siguientes funcionalidades:

- Envío de mensajes multimedia y filtrado dinámico de los destinatarios.
- Historial de envíos realizados y respuestas recibidas.
- Autenticación de usuarios.

Cada una de estas funcionalidades implicará acceso a la base de datos, para la consulta de información transaccional a visualizar en los reportes o para la inserción de la data de configuración y contenido de los mensajes multimedia a enviar.

En la letra B se contempla la aplicación motor para la gestión de los MMS y su interacción con la base de datos. Entre las funcionalidades de esta herramienta se pueden destacar:

- Construcción de los MMS.
- Envío y Recepción de los MMS.
- Consulta en la base de datos del contenido de los MMS (destinatarios, cuerpo, elementos multimedia, etc.) y los parámetros de configuración de la conexión de datos.
- Actualización del estatus de envío de los MMS.

En la letra C define la interacción entre el motor MMS y el módem GSM, que incluye el envío de comandos AT por parte de aplicación al dispositivo módem para configurar su operatividad, además del envío de los parámetros de conexión WAP obtenidos de la base de datos.

La interacción contemplada entre los elementos en la letra D consiste en el establecimiento de la conexión GPRS/WAP entre el dispositivo módem y la operadora móvil basada en los parámetros obtenidos de la herramienta motor. Esta interacción se lleva a cabo de manera automática siempre y cuando se configure adecuadamente el modem y se encuentre en un entorno de buena cobertura móvil.

## **1.6 Alcance**

El alcance propuesto de la presente investigación contempla el diseño e implementación de la plataforma para envío y recepción de los mensajes multimedia, así como la administración de los contenidos MMS a enviar y recibir.

Para lograr el alcance propuesto, esta investigación abarcará los siguientes elementos:

- Configuración de los dispositivos inalámbricos (módems) para el establecimiento de la comunicación con el MMS Center de la operadora por medio de una conexión WAP/GPRS.
- Diseño y desarrollo de una aplicación web para la administración de contenidos, realizar envíos MMS, autenticación de usuario y visualización de reportes transaccionales, tanto de envíos MMS realizados como de mensajes SMS y MMS recibidos.
- Diseño y desarrollo de una aplicación que permita comunicar la aplicación Web para la administración de contenidos y el dispositivo inalámbrico (módems) para el envío y recepción de mensajes multimedia con el MMS Center de la operadora. Esta aplicación coordinará el envío y recepción de los mensajes multimedia y la comunicación con MMS Center de la operadora.
- Pruebas de funcionalidad de la aplicación motor coordinadora del envío y la recepción MMS y pruebas de integración de todos los componentes de la plataforma (motor, aplicación Web, base de datos y módem GSM).

## 1.7 Limitaciones

Las limitaciones de la presente investigación son las siguientes:

- La operadora telefónica a utilizar para el envío de los MMS será DIGITEL, dado que fue la única que ofreció los parámetros de configuración necesarios para acceder a la conexión con su MMS Center. El tamaño de los mensajes multimedia no excederá los 25KB, ya que este es el límite de bytes permitidos para transmitir en cada sesión al establecer la conexión WAP/GPRS para MMS, límite establecido por la operadora DIGITEL.
- El contenido multimedia de los mensajes se limitará únicamente a texto y una imagen, en específico de los formatos JPEG, PNG y GIF.

## Capítulo II: Marco Conceptual

En este capítulo se presentan los fundamentos conceptuales del desarrollo realizado en el presente trabajo, y se encuentra dividido en dos secciones fundamentales. La primera parte corresponde a la definición, características, funcionamiento, protocolos fundamentales entre otros aspectos, de los estándares GSM (*Global System for Mobile Communications*), MMS (*Multimedia Messaging System*) y GPRS. Adicionalmente se especifica la definición y características de Módem GSM y el conjunto de Comandos AT, requeridos para interactuar con este dispositivo.

La segunda parte de este capítulo describe las características, ventajas y desventajas de las tecnologías Web como HTML (*HyperText Markup Language*), JavaScript, CSS (*Cascading Style Sheets*), PHP (*Hypertext Preprocessor*), así como las herramientas seleccionadas para la codificación de los módulos de la plataforma de mensajería. Además se mencionará la definición y características del sistema manejador de bases de datos MySQL.

### 2.1 Sistema Global para las Comunicaciones Móviles (GSM, Global System for Mobile Communications)

#### 2.1.1 Definición

GSM es un estándar mundial para la comunicación mediante teléfonos móviles que incorporan tecnología digital.

“*Groupe Special Mobile*” fue creada dentro de la Conferencia Europea de Administradores de Correos y Telecomunicaciones (CEPT, siglas de su nombre en francés “*Conférence Européenne des administrations des Postes et des Télécommunications*”) cuyo propósito era desarrollar un único sistema de radio comunicación digital para Europa [1].

GSM es un sistema de telefonía bien definido que permite el *roaming* internacional y la capacidad de ofrecer a alta velocidad servicios avanzados de transmisión de voz, datos y video, y otros servicios de valor agregado [7].

Este estándar opera en las bandas de frecuencia de 900 MHz y 1.8 Ghz en Europa y en Estados Unidos entre 850 MHz y 1.9 GHz. La banda de 850 MHz es también usada para GSM y 3G (Tercera Generación de transmisión de voz y datos a través de telefonía móvil) en Australia, Canadá y varios países de Suramérica [3].

Por haber armonizado el espectro de banda en casi todo el mundo, la capacidad de *roaming* de GSM les permite a los usuarios acceder al mismo servicio en cualquier parte del globo, lo cual provee conectividad en más de 218 países [3].

En el 2010, las redes terrestres GSM abarcan más del 80% de la población mundial con más de 4.700 millones de conexiones.

### 2.1.2 Especificaciones

- **Banda de Frecuencia:** Las banda de frecuencia especificadas por GSM son 900 MHz y 1800 MHz. Actualmente hay operadoras que poseen redes en ambas frecuencias, también existen redes de 1900 MHz y más reciente de 850 MHz [18].
- **Distancia del Dúplex:** La distancia del dúplex es de 80 MHz. Esta se define como la distancia entre las frecuencias del enlace de subida y de bajada. Un canal posee dos frecuencias, en GSM estas frecuencias son separadas por 80 MHz [4].
- **Separación del Canal:** Es la separación entre las frecuencias portadoras adyacentes. En GSM, esta es de 200 MHz [4].
- **Modulación:** la modulación es el proceso del envío de una señal al cambiar las características de la frecuencia portadora. Esto es realizado en GSM vía GMSK (*Gaussian Minimum Shift Keying*) [4].
- **Tasa de Transmisión:** GSM es un sistema digital con una tasa de transferencia por encima de 270 Kbps [4].



- **Método de Acceso:** GSM utiliza el concepto del Acceso Múltiple por División de Tiempo (TDMA, *Time Division Multiple Access*). TDMA es una técnica en la cual diferentes llamadas pueden compartir la misma portadora. A Cada llamada se le asigna una ranura de tiempo particular [4].
- **Codificador de Voz:** GSM utiliza Codificación Lineal Predictiva (LPC, *Linear Predictive Coding*). El propósito de LPC es el de reducir la tasa de bits. La LPC provee parámetros para un filtro que imita las cuerdas vocales. La señal pasa a través de este filtro, dejando atrás una señal residual. La conversación es codificada en 13 Kbps [4].

### 2.1.3 Tarjeta SIM

La Tarjeta SIM (SIM, *Subscriber Identity Module*) o tarjeta de Módulo de Identificación del Suscriptor, es una tarjeta que se utiliza en los teléfonos móviles en la que se almacena de forma segura la información del usuario del teléfono necesaria para identificarse en la red (clave de autenticación e identificación del área local). La tarjeta SIM también almacena datos del operador necesarios para el servicio de mensajes cortos y otros servicios [11].

La tecnología GSM está basada en el uso de la tarjeta SIM que almacena todos los datos del cliente. El servicio está asociado a la tarjeta SIM y no al equipo móvil [18].

La capacidad de almacenamiento de una tarjeta SIM va desde 2KB hasta 1GB, aunque las más comunes son las de 16, 32, 64 y recientemente de 128 KB.

Una tarjeta SIM contiene la siguiente información:

- El número telefónico del abonado (MSISDN).
- El estado de la tarjeta SIM.
- El código de servicio (operador).
- El PIN (Código de identificación personal).
- El PUK (Código personal de desbloqueo).

## 2.1.4 Sistema de Mensajes Cortos (SMS, Short Message Service)

El sistema de mensajes cortos de GSM provee de un medio único para enviar mensajes cortos alfanuméricos u otro tipo de información a un usuario de un teléfono móvil. GSM distingue dos tipos de mensajes cortos [2]:

Un mensaje SMS es una cadena alfanumérica de hasta 140 caracteres o de 160 caracteres de 7 bits, y cuyo encapsulado incluye una serie de parámetros. En principio, se emplean para enviar y recibir mensajes de texto normal, pero existen extensiones del protocolo básico que permiten incluir otros tipos de contenido, dar formato a los mensajes o encadenar varios mensajes de texto para permitir mayor longitud. [2]

Los beneficios de SMS para sus suscriptores se centran en la conveniencia, flexibilidad, e integración de los servicios de voz y datos. Desde esta perspectiva, el beneficio primario consiste en la habilidad para usar un dispositivo móvil como una extensión de la computadora personal. SMS elimina la necesidad de tener varios dispositivos, permitiendo la integración en uno único (terminal móvil) todos los servicios de mensajería [2].

## 2.2 Servicio General de Paquetes vía Radio (GPRS, General Packet Radio Service)

### 2.2.1 Definición

GPRS Es una extensión del Sistema Global para Comunicaciones Móviles ó GSM para la transmisión de datos no conmutada (o por paquetes). Es una solución para datos móviles que ofrece eficiencia espectral para nuevos y más veloces servicios de datos, así como para *roaming* internacional. Por tratarse de una tecnología de datos inalámbricos, GPRS ofrece velocidades de datos máximas de 115 Kbps y un *throughput* promedio de 30-40 Kbps. A GPRS a menudo se lo denomina tecnología de "2.5G" porque constituye el primer paso de un operador GSM hacia la tercera generación (3G) [9].

GPRS es una tecnología basada en paquetes, lo que significa que los datos están divididos en paquetes que se transmiten en breves ráfagas sobre una red IP. Este diseño es mucho más eficiente que las redes conmutadas por circuitos, dando lugar a una reducción de los costos operativos de la red. El diseño de paquetes beneficia a los usuarios en dos formas primordiales. Primero, GPRS provee una conexión "siempre activa" ("*always-on*") que no exige que el usuario deba conectarse cada vez que desea obtener acceso a datos. En segundo lugar, los usuarios sólo pagan por los datos en sí, en lugar de pagar por el tiempo de aire empleado en establecer una conexión y descargar los datos.

GPRS se desarrolla sobre la plataforma GSM y en IP, la norma universal utilizada en Internet, y no en una norma exclusivamente inalámbrica que exija equipos propietarios. El hecho de que GPRS utiliza una tecnología abierta y totalmente normalizada lo convierte en el ideal para la provisión de acceso inalámbrico a otras redes basadas en IP, tales como LAN's corporativas e ISP's. Otra ventaja de la base IP de GPRS es que los operadores y sus socios pueden desarrollar y lanzar avanzados servicios de datos mucho más rápidamente y de manera menos costosa, beneficio que se atribuye a la amplia disponibilidad de "*know-how IP*" y equipos ofrecidos en versiones estándar [9].

Una conexión GPRS está establecida por la referencia a su nombre del punto de acceso (APN). GPRS se puede utilizar para servicios tales como *Wireless Application Protocol* (WAP), Servicio de Mensajes Cortos (SMS), Servicio de Mensajería Multimedia (MMS), Internet y para los servicios de comunicación, como el correo electrónico y la *World Wide Web*. Para fijar una conexión de GPRS para un módem inalámbrico, un usuario debe especificar un APN, opcionalmente un nombre y contraseña de usuario, y muy raramente una dirección IP, todo proporcionado por el operador de red.

## 2.3 Sistema de Mensajería Multimedia (MMS, Multimedia Messaging System)

### 2.3.1 Definición

La mensajería multimedia móvil o MMS (*Multimedia Messaging Service*) es un estándar universalmente aceptado como parte de las especificaciones técnicas del 3GPP, que permite a los usuarios de teléfonos móviles con capacidades de MMS enviar y recibir en un mismo mensaje texto con formato, sonido, imágenes, animaciones y vídeo clips. Los vídeos, sonidos e imágenes de alta resolución pueden ser descargados al teléfono desde servidores de contenidos WAP, transferidos al teléfono mediante un accesorio adjunto, como por ejemplo una cámara digital, o recibidos en un mensaje MMS. Los mensajes MMS pueden ser enviados a otro teléfono móvil con capacidad de MMS o a una dirección de e-mail. MMS al igual que SMS, garantiza al usuario el envío de la información y utiliza la tecnología de almacenamiento y reenvío, es decir, no asegura el envío de los datos en tiempo real [12].

Al contrario de los servicios de correo electrónico actuales, donde las diferentes medias son recibidas como archivos adjuntos, en el MMS, los mensajes pueden venir acompañados de informaciones sobre la sincronización y presentación de las diversas medias enviadas [12].

El MMS es un estándar establecido por el *Third Generation Partnership Project* (3GPP), que abarca todas las entidades a nivel mundial que participan del desarrollo de la próxima generación. A pesar de previsiones de demora en la implantación del estándar 3GPP, casi todos los principales operadores mundiales de comunicaciones móviles ya tienen plataformas soportando prototipos de los servicios MMS [12].

El MMS posibilita un intercambio de uno o más elementos de los medios de comunicación entre usuarios o un servidor de contenido, sin la necesidad de que el servicio sea prestado en tiempo real. Este servicio debe soportar y aprovechar los avances ya realizados en el área de multimedia y de los servicios de mensajes actuales, añadiendo requisitos específicos para la movilidad, buscar siempre mantener la compatibilidad y reutilizar los estándares ya establecidos [12].

## 2.3.2 Parámetros a configurar para el envío de Mensajes Multimedia

### 2.3.2.1 Centro MMS (MMSC, Multimedia Messaging System Centre)

El MMSC ó Centro de Mensajes multimedia (*Multimedia Message Service Center*) es similar en función a un centro de mensajes cortos SMSC, si bien presenta mayor capacidad y funciones más avanzadas [32].

El MMSC, en un sistema móvil GSM, debe comunicarse con la red núcleo de GPRS para el transporte de los mensajes, con el subsistema de red (NSS – VLR – HLR) para la tarificación y gestión de permisos del usuario, con un SMSC para remitir los avisos de mensajes entrantes y con Internet para poder enviar mensajes de correo electrónico. Además, se contempla la posibilidad de que el MMSC reconozca las características del terminal del usuario (tipo y tamaño de pantalla, capacidad de visualización de ciertos tipos de archivo y no de otros, etc.) y adapte automáticamente el contenido recibido al terminal para que se vea correctamente [32].

En otro tipo de redes, el método de transporte será distinto, pero el MMSC conservará la capacidad de adaptación del contenido, si está habilitada esta capacidad [32].

Éste es un resumen de sus funciones [32]:

- Recibir a través de la red núcleo GPRS los mensajes multimedia entrantes.
- Almacenar los mensajes entrantes hasta que puedan ser transportados hasta el usuario.
- Enviar al usuario los mensajes por la red GPRS cuando el usuario los descargue (normalmente, de forma automática a petición de su terminal).
- Recibir y redirigir hasta el MMSC destino los MMS salientes.
- Decidir si el tipo de contenido de los mensajes entrantes es válido para el terminal del usuario y, en su caso, procesarlo para que pueda mostrarse.
- Comunicarse con el HLR y VLR para verificar si el usuario tiene permiso para enviar mensajes y poder tarificarlos.

### **2.3.2.2 Nombre de Punto de Acceso (APN, Access Point Name)**

APN o *Access Point Name* es el nombre de un punto de acceso para GPRS que debe configurarse en el teléfono móvil para que pueda acceder a Internet.

Un punto de acceso es:

- Una dirección IP a la cual un móvil se puede conectar.
- Un punto de configuración que es usado para esa conexión.
- Una opción particular que se configura en un teléfono móvil.

Los APN pueden ser variados. Son usados en redes tanto públicas como privadas. Por ejemplo:

- grancompania.mnc012.mcc345.gprs
- internet.compania.com
- tuwap.com

Una vez que el dispositivo se ha conectado, usa el servidor DNS para hacer el proceso llamado Resolución de APN, que finalmente da la IP real del APN, esto con la finalidad de conectarse al servicio asociado a dicha dirección (Internet, MMS, WAP, etc.).

### **2.3.2.3 Pasarela WAP (WAP Gateway)**

Con el fin de enviar mensajes multimedia el teléfono necesita conectarse a un *WAP Gateway*, el cual está situado entre el móvil y el *MMS Center*. El *WAP Gateway* usado por MMS puede ser diferente al utilizado por otras conexiones de datos (WAP o Internet), por lo tanto, el teléfono móvil debe poseer la configuración correcta del *WAP Gateway* para los mensajes multimedia [20].

El *WAP Gateway* es un servidor inalámbrico que establece la comunicación entre el dispositivo inalámbrico (teléfono móvil o modem GPRS) y el servidor de páginas, con la finalidad de verificar la validez del

documento WML enviado por este último para determinar si debe ser descartado o transformado a un formato comprensible por el teléfono móvil (binario) [19].

### 2.3.3 Estructura MMS

En un entorno MMS, un mensaje multimedia puede tomar múltiples formas con el fin de ser transmitido de manera eficiente a través de la red móvil. El enlace entre el cliente MMS y el *MMS Centre* es frecuentemente de ancho de banda limitado, por lo tanto, los mensajes multimedia son codificados de forma binaria para transferirlos de manera eficiente sobre este enlace. El rango de contenidos de los mensajes multimedia varía desde texto simple hasta objetos multimedia sofisticados con opciones de sincronización entre ellos. A los objetos multimedia se les incluye un encabezado o *header*, la cual permite incorporar varios elementos de red para direccionar el mensaje hacia distintos recipientes, así como elementos que caracterizan el contenido del mensaje (clase, prioridad, asunto, etc) [5].

Los objetos multimedia son encapsulados en “contenidos” individuales conocidos como *body parts*.

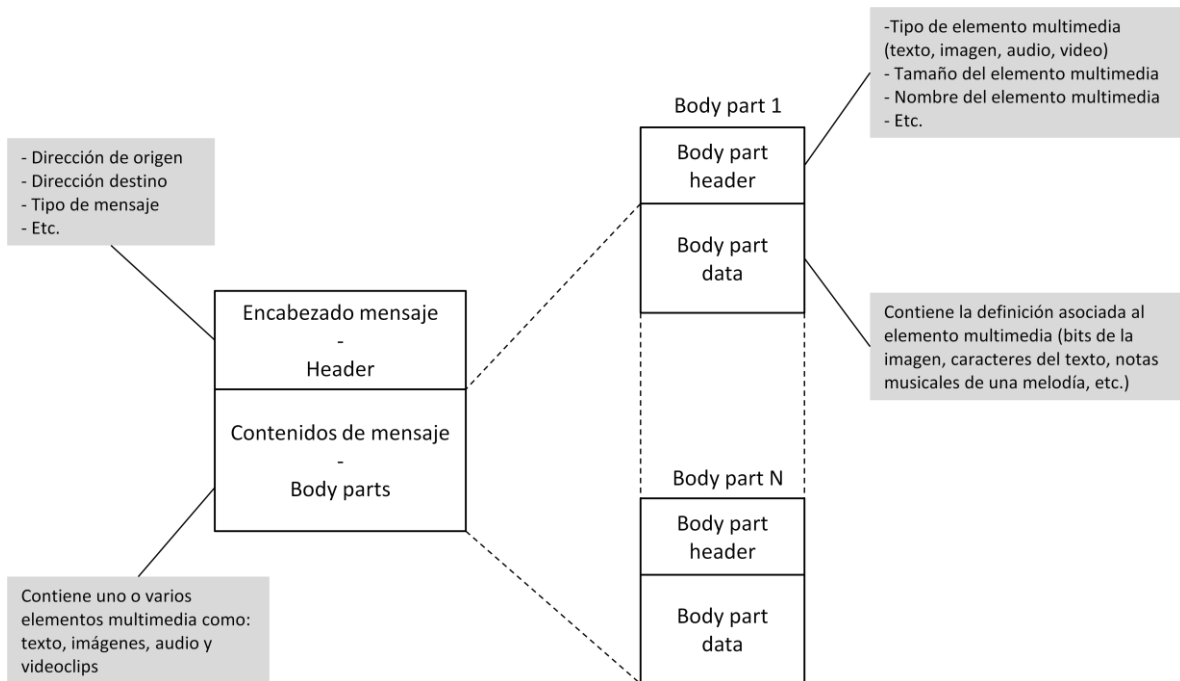


Figura 2: Estructura General de un Mensaje Multimedia [5]

Como es mostrado arriba, un mensaje multimedia consiste en un *header* (encabezado) y *body parts* (contenidos de mensaje encapsulados). El *header* contiene la siguiente información del mensaje:

- Dirección del emisor del mensaje.
- Direcciones destino del mensaje, organizada en recipientes primarios y secundarios (To, Cc y Bcc).
- Prioridad del mensaje (baja, media o alta).
- Clase del mensaje (auto, personal, informativo o notificación).
- Fecha y hora de envío.
- Período de validez.
- Asunto del mensaje.
- Visibilidad de la dirección del emisor.
- Otras características.

Cada una de estas características posee un conjunto de parámetros que a su vez poseen un conjunto de valores. Estos parámetros son definidos en el RFC-2822. El nombre formal de estos parámetros contiene el prefijo "X-MMS". En la siguiente tabla se especifican los parámetros más comunes que se pueden incluir en el *header* del mensaje [33]:



<b>Parámetro</b>	<b>Descripción</b>
To	Dirección del recipiente primario del mensaje.
Cc	Dirección del recipiente secundario del mensaje ( <i>Carbon Copy</i> ).
Bcc	Dirección del recipiente secundario del mensaje ( <i>Blind Carbon Copy</i> ).
From	Dirección del emisor del mensaje.
Date	Fecha y hora en la que fue enviado el mensaje.
Message-ID	Identificador único del mensaje.
Subject	Asunto del mensaje.
X-Mms-Expiry	Período de validez del mensaje. Pasado este tiempo, el MMS Centre descartará el mensaje si no se ha entregado a su destino.
X-Mms-Delivery-Time	El emisor puede especificar una fecha de entrega más anticipada para darle mayor prioridad al mensaje.
X-Mms-Distribution-Indicator	El proveedor de servicios de valor agregado puede usar esta bandera para indicar que el mensaje no puede ser redistribuido libremente.
X-Mms-Delivery-Report	Especifica si el emisor desea un reporte de entrega del mensaje para cada recipiente.
X-Mms-Read-Report	Especifica si el emisor desea un reporte de lectura del mensaje por parte de cada recipiente.
X-Mms-Message-Class	Indica el tipo de clase del mensaje (personal, informativo, anunciativo).
X-Mms-Priority	Prioridad de envío del mensaje (baja, media, alta).
X-Mms-Sender-Visibility	Indica si se desea colocar visible o no la dirección del emisor del mensaje.
Content-type	El tipo de contenido del mensaje.

**Tabla 1: Parámetros del encabezado MMS [5]**

El valor asignado al parámetro “*Content-Type*” indica como están organizados los *body part* dentro del mensaje. Los tres valores que puede tomar este campo son descritos en la siguiente tabla:

Valor del Content-type	Descripción
Multipart/Mixed (application/vnd.wap.multipart.mixed)	Una estructura mixta multipartes ó <i>multipart mixed</i> contiene uno o más <i>body parts</i> . El orden en el cual aparecen no tiene importancia. Esta estructura comúnmente es usada cuando los mensajes no contienen una descripción para la sincronización y presentación de sus elementos multimedia.
Multipart/Alternative ó (application/vnd.wap.multipart.alternative)	Una estructura multipartes alternativa ó <i>multipart alternative</i> contiene uno o más <i>body parts</i> , y cada uno representa a un sólo elemento independientemente de su contenido. Por ejemplo, la estructura puede contener una imagen JPEG y otra imagen GIF, ambas representan el mismo elemento y el cliente MMS receptor seleccionará la forma más apropiada para representar y renderizar el mensaje.
Multipart/Related ó (application/vnd.wap.multipart.related)	Una estructura multiparte relacionada ó <i>multipart related</i> es usada para agregar múltiples <i>body parts</i> en una estructura. Este tipo de <i>content-type</i> incluye un archivo XML que especifica el comportamiento (sincronización, presentación y temporización) de los elementos multimedia contenidos en la estructura.

Tabla 2: Valores del parámetro *content-type* en el encabezado MMS [5]

A cada *body part* se le asocian en su encabezado los parámetros especificados a continuación:

Parámetro	Descripción
Content-ID	Un único identificador para el <i>body part</i> en el mensaje. El identificador es típicamente insertado entre corchetes. Ejemplo: Content-ID: <ui bp 0006>
Content-Location	Un nombre alfanumérico con su extensión al final del mismo utilizado para identificar el objeto multimedia contenido en el <i>body part</i> . Este parámetro es usado comúnmente cuando el usuario desea extraer y almacenar el elemento multimedia. Ejemplo: Content-Location: house.jpg
Content-Disposition	Es un indicador que especifica cómo debe ser mostrado el objeto multimedia.
Content-Type	El tipo de contenido del correspondiente elemento multimedia. Ejemplos: Content-type: image/jpeg Content-type: audio/midi Opcionalmente, a ciertos tipos se le pueden asociar uno o más parámetros como la tabla de caracteres o nombre del archivo. Ejemplo: Content-type: text/plain; charset = US-ASCII Content-type: image/GIF; name = house.jpg

Tabla 3: Parámetros del encabezado de un *body part* [5]

### 2.3.4 Proceso de Envío y Recepción

El transporte de los MMS entre el usuario y el MMSC se realiza mediante una conexión WAP GPRS; para este fin, los terminales de usuario aceptan distintas definiciones de perfil de servicio WAP según lo que interese. Así, normalmente los operadores especifican un perfil WAP gratuito para transportar los MMS; la tarificación la realiza el MMSC y no se hace en la red núcleo de GPRS [5].

El tráfico GPRS tiene costo de uso, puesto que se emplea para conectarse a Internet o a la red WAP móvil. En cambio, el tráfico GPRS que generan los mensajes multimedia normalmente no se cobra, ni al enviarlos ni al recibirlos, puesto que resulta más sencillo para los usuarios y los operadores su tarificación en bloque como mensajes, a cargo del MMSC. Sin embargo, si el usuario se encuentra fuera de la cobertura de su operadora móvil y recibe o envía un MMS, es probable que sí tenga que abonar una cantidad por el tráfico de datos, al conectarse a una red ajena que deseará percibir un importe por el tráfico de datos realizado [5].

Las entidades MMS (Clientes MMS o MMSC) se comunican por medio de transacciones. Una transacción comúnmente se compone de una solicitud de un servicio (*request*) y su correspondiente respuesta (*Response/Confirmation*) que contiene el resultado de la transacción. Una Unidad de Datos de Protocolo (PDU, *Protocol Data Unit*) es asociada a cada *request* o *response* de un servicio, las cuales son transacciones que se generan en el intercambio de un MMS entre dos usuarios. Un PDU está compuesto por un conjunto de parámetros obligatorios, opcionales y/o condicionales [5].

El flujo de transacciones más simple para el intercambio de un mensaje es el que involucra a dos Clientes MMS pertenecientes a un mismo Entorno MMS (MMSE, *Multimedia Messaging System Environment*). El proceso de intercambio de un mensaje se define en cuatro pasos mostrados en la Figura 3 y descritos posteriormente:

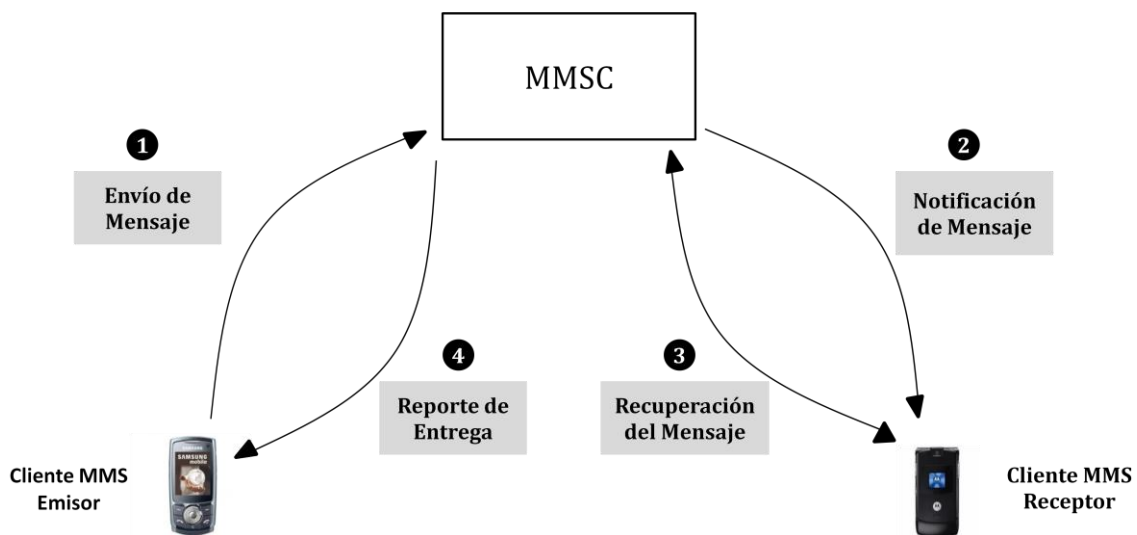


Figura 3: Flujos de transaccionales generales para el intercambio de un mensaje multimedia [5]

### 1. Envío del mensaje al MMSC

Esta transacción está compuesta por el envío de un *request* al MMSC, que contiene el mensaje multimedia a enviar (M-send.req PDU) y su correspondiente *response* por parte del MMSC para notificar al Cliente MMS emisor la recepción de su mensaje (M-send.conf PDU) [5].

Por medio de definiciones técnicas del protocolo WAP, el dispositivo móvil establece la conexión de datos con la red, la cual permite la comunicación con el MMSC. Esta conexión puede llevarse a cabo sobre una red de datos por conmutación de circuitos (Ej. GSM) o sobre una red por conmutación de paquetes (Ej. GPRS). La conexión es típicamente establecida a través de una pasarela WAP ó *WAP Gateway*. Es común para las operadoras móviles asignar un *WAP Gateway* dedicado para el manejo del tráfico MMS, además de la destinada originalmente para el tráfico común y navegación por la red Internet [5].

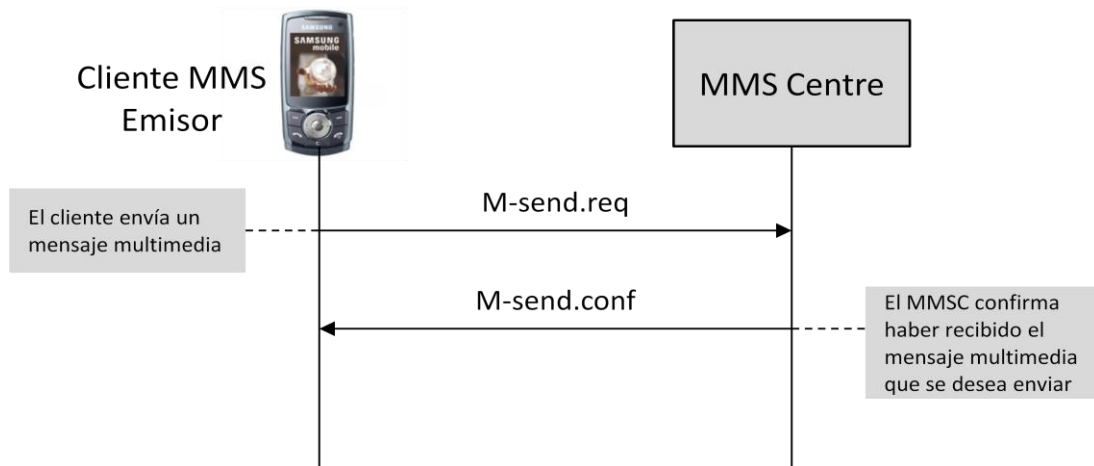


Figura 4: Flujo de transacciones en el envío de un mensaje multimedia [5]

## 2. Notificación de mensaje entrante

Esta transacción está compuesta por una notificación de un nuevo mensaje multimedia entrante para el Cliente MMS por parte del MMSC (M-notification.ind PDU) y un *response* que confirma la recepción de dicha notificación (M-notifyresp.ind PDU) [5].

Una vez que el mensaje enviado es aceptado por el MMSC, este analiza e identifica la dirección destino. El MMSC construye una notificación compacta que describe brevemente en su encabezado las características del mensaje (asunto, dirección del emisor, prioridad, tamaño en bytes, etc.) y lo envía a su correspondiente dirección destino [5].

### 3. Recuperación del contenido del mensaje multimedia

Esta transacción está compuesta por una solicitud o *request* enviada al MMSC por parte del Cliente MMS para obtener el contenido del mensaje (WSP/HTTP GET.req PDU), y de un *response* enviado del MMSC al Cliente MMS con el contenido solicitado (M-retrieve.conf PDU). Opcionalmente el Cliente MMS envía un mensaje de confirmación de recepción del contenido al MMSC (M-acknowledge.ind PDU) [5].

### 4. Reporte de entrega al emisor

El MMSC envía un reporte de entrega al Cliente MMS emisor para informarle que el destinatario recibió el mensaje multimedia (M-delivery.ind PDU). El reporte es enviado únicamente si el autor del mensaje lo solicita, esto por medio de un parámetro especificado en el encabezado del mensaje multimedia enviado en la primera transacción [5].

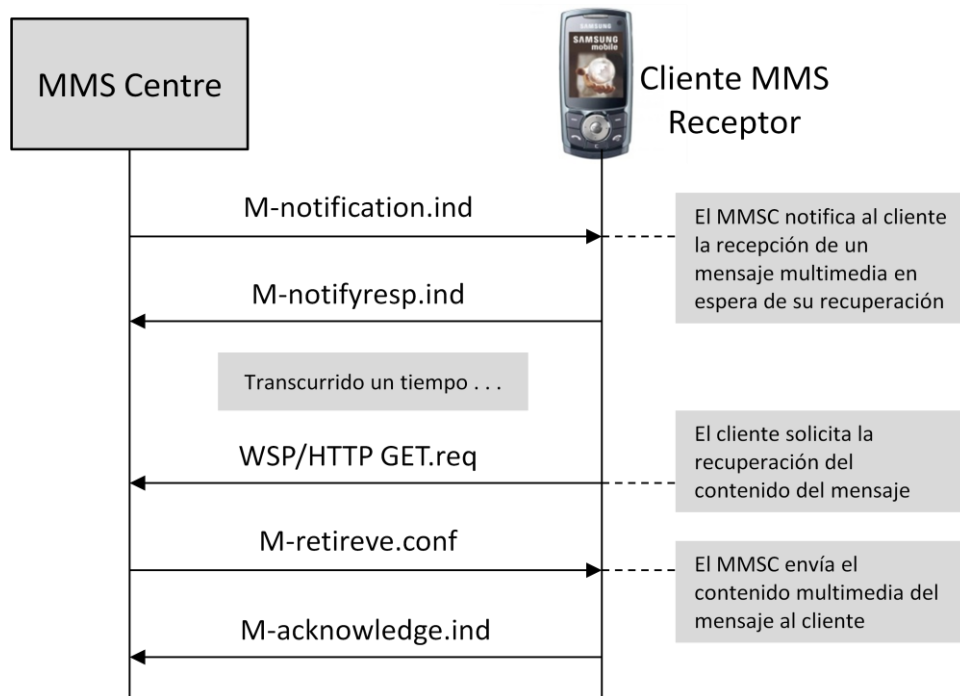


Figura 5: Flujo de transacciones en la recepción de un nuevo mensaje multimedia [5]

## 2.4 Modem

### 2.4.1 Definición

Modem ó Modulador-Demodulador es un dispositivo que permite la comunicación entre dos ordenadores usando una línea telefónica. Al ser el ordenador un sistema digital y las líneas telefónicas convencionales analógicas, en primer lugar el módem del ordenador que envía la información ha de convertir en analógicas las señales digitales (modulación), para que sean transmitidas por la línea telefónica hasta el ordenador de destino, cuyo módem deberá reconvertirlas en digitales (demodulación) [14].

### 2.4.2 Modem GSM

Los módems GPRS conectan uno o varios puertos de comunicación con formato USB, serial RS232 o RS485, ó Ethernet a la red de celular, es decir, es similar a un teléfono móvil pero adicionalmente posee un puerto de comunicación para la conexión y comunicación con el computador. Estos dispositivos al igual que un teléfono celular GSM, se le incorpora una tarjeta SIM [21].

El módem GPRS abre la comunicación hacia una dirección IP configurada en él, de modo que después de ello queda establecida la comunicación. Se requiere entonces que la IP del receptor sea fija, ya que de lo contrario tendríamos que reconfigurar el módem en cada conexión [21].

### 2.4.3 Enfora GSM1218 SA-GL

El Módem Enfora GSM1218 SA-GL es uno de los más económicos y compactos Modem *Stand Alone* vía IP (Módem GSM/GPRS) disponible actualmente. El Modem SA-GL es un dispositivo cuatribanda (GSM 850/900/1800/1900) y funciona con una línea integrada en una tarjeta SIM [15].

El modem soportan una completa lista de características que incluyen: Voz, SMS, llamadas conmutadas de datos o CSD (*Circuit Switched Data*), 10 protocolos TCP/IP integrados, conversión serial a GPRS y GPRS a

serial, un sistema de manejo de eventos configurables por el usuario así como soporte de un amplio set de comandos AT (*Attention*). Por estas características mencionadas, el SA-GL es el modem seleccionado para el desarrollo de la plataforma tecnológica en la presente investigación.

El modem se puede conectar a cualquier dispositivo que tenga un puerto serial o USB (Universal Serial Bus) hacia redes celulares e Internet. El Módem es compatible en cualquier plataforma (Mac, Unix, Windows, etc.), pero el software de gestión y configuración que incluye sólo es compatible con Windows 98, 2000 Profesional y XP [16].

La especificación técnica del modem Enfora GSM1218 SA-GL se muestra a continuación:

### ***Características físicas***

- Dimensión (largo x ancho x altura): 63.5 x 63.5 x 23.9 mm.
- Peso: 80 gramos.
- Carcasa: plástica.
- Antena: Conector SMA (*Subminiature Version A*) ó de cable coaxial.

### ***Radio performance***

- Frecuencia de Radio: 850/900/1800/1900 MHz.
- Sensibilidad: -106 dB (Decibeles).
- Poder de transmisión: Clase 4 (2W @850/ 900 MHz) Clase 1 (1W @1800/1900 MHz).

### ***Funcionalidad GSM***

- Voz: codecs FR (*Full Rate*), EFR (*Enhanced Full Rate*), HR (*Half Rate*) y AMR (*Adaptive Multi-Rate*).
- Data CS (Data Circuit Switched – Conmutación de Circuitos de datos): Asíncrona.
- GSM SMS: Texto, PDU (*Protocol Data Unit*), MO/MT, tecnología de difusión de celda (*Cell broadcast*).



### ***Interfaz de aplicación***

- Protocolos de host: Comandos AT, UDP/API, PPP
- Protocolos internos: PPP (*Point to Point Protocol*), UDP (Unit Data Protocol) API (*Application Programming Interface*), TCP (*Transmission Control Protocol*) API, UDP PAD (*Packet Assembler/Disassembler*), TCP PAD.
- Control/Estatus de API: comandos AT, UDP API, TCP API, comandos AT sobre SMS.
- Interfaz física: Serial RS-232.
- Interfaz periférica: 8-pin I/O (*In/Out – Entrada/Salida*): *Mic in, headset out*, retorno de audio, tierra, *input power*, dos I/O definidas por el usuario, y un output dedicado.

### ***Acceso a la SIM***

1.8 / 3V con mecanismo de seguro.

### ***Entorno***

- Operatividad: -30°C a 70°C.
- Almacenamiento: -40°C s 85°C.
- Humedad: por encima de 95% no condensada.

### ***Poder***

Voltaje de corriente directa: 6-40 V.

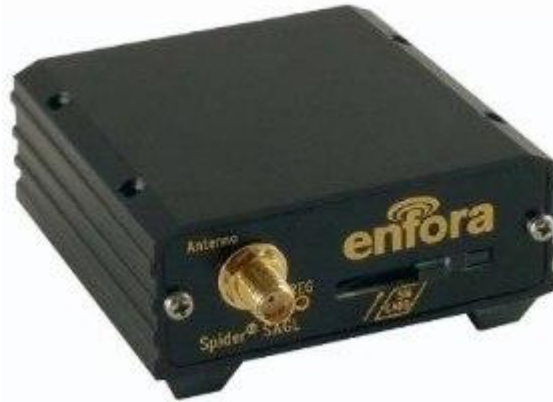


Figura 6: Modem Enfora GSM1218 SA-GL

Características	Enfora Spider SA-GL	Sierra Wireless Airlink GL6100
Bandas de Frecuencia (MHz)	850/900/1800/1900	850/900/1800/1900
Procesador	No posee	Procesador ARM946
Compatibilidad con Tarjeta SIM	Sí	Sí
Códex de voz	HR, FR, EFR, AMR	HR, FR, EFR, AMR
Compatibilidad con Sistemas Operativos	Todos	Windows, Linux
Temperatura Operacional	Entre -30° y 70°	Entre -30° y 75°
Disponibilidad en Venezuela	Sí	Poca disponibilidad
Precio	130 \$US	191 \$US

Tabla 4: Comparación entre los modems GPRS Enfora Spider y Sierra Wireless Airlink [15]

Las características más resaltantes para establecer un criterio de preferencia hacia el modem Enfora GSM1218 SA-GL con respecto al modem Sierra Wireless Airlink GL6100 son las siguientes:

- Precio más económico.
- Mayor disponibilidad en el país.
- Compatibilidad con cualquier sistema bajo cualquier plataforma, siempre que posea puerto USB o puerto Serial.

## 2.5 Comandos AT

### 2.5.1 Definición

Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación entre el usuario y un terminal modem, que es de carácter genérico en su mayoría, ya que un mismo comando funciona en modelos de distintas marcas haciendo que un programa basado en comandos AT sea inmensamente robusto y compatible con la mayor parte de los dispositivos disponibles en el mercado. La gran parte de los módems disponibles reconocen los comandos AT más utilizados, por lo mismo la tecnología GSM ha adaptado el uso de estos comandos. Los comandos AT se denominan así por la abreviatura de *Attention* en inglés [8].

El conjunto de comandos AT o comandos Hayes es un lenguaje de comandos específico desarrollado por *Hayes Microcomputer Company* en 1977 [17].

Un lenguaje de comandos es un lenguaje interpretado de dominio que por lo general tiene un fuerte acoplamiento con las partes fundamentales del sistema operativo. Los lenguajes de comandos frecuentemente proveen una gramática o sintaxis simple muy cercana al lenguaje natural para mayor facilidad de aprendizaje y memorización. Un ejemplo es el Shell de los sistemas operativos de las computadoras personales.

El conjunto de comandos AT consiste en una en una serie de cadenas de caracteres cortas las cuales combinadas producen comandos completos que permiten controlar las funcionalidades de un módem (operaciones de discado y finalización de llamadas, manipulación de mensajes cortos y configuración de parámetros de una conexión de datos) [17].

Los comandos AT son interpretados en una consola telnet (*Hyperterminal* en el caso de los sistemas operativos Windows) que se debe configurar previamente con los parámetros de conexión del modem tales como nombre del puerto serial, velocidad de la conexión en bits por segundo, control de flujo de errores, entre otros. Esta aplicación interpretadora recibe los comandos a través de un script en un archivo de texto plano, donde se puede interpretar más de una comando, o por la línea de comandos de la consola, donde se interpreta un comando a la vez.

## 2.5.2 Notación

El envío de comandos AT requiere la siguiente estructura [10]:

- Petición:



Figura 7: Comando AT petición

<CR>: *carriage return* (Presionar "Enter" o "Intro" para la ejecución de cualquier comando)

- Respuesta correcta:

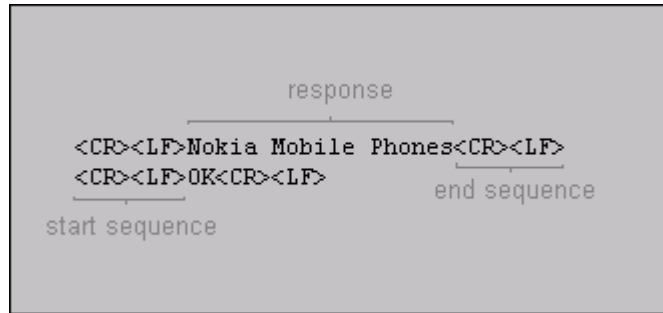


Figura 8: Respuesta correcta a un comando AT

<LF>: *line feed*

- Respuesta incorrecta:

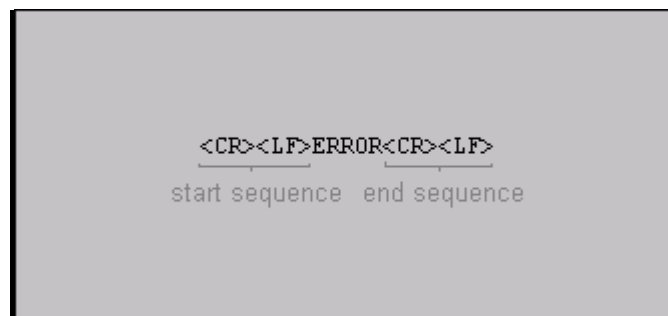


Figura 9: Respuesta de error a un comando AT

### 2.5.3 Lista de Comandos AT más utilizados

A continuación se detallan los comandos AT más importantes a la hora de configurar la tarjeta SIM y la conectividad del dispositivo modem, así como la conexión de datos (Internet) a utilizar para el envío de los mensajes multimedia (llamada GPRS) [8]:

- **AT:** comando que verifica que el módem se encuentra en línea con el PC. Debe retornar OK como respuesta.
- **AT+IPR=<ipr>:** Configura la velocidad a la que trabajará el módem con el puerto serial. El valor <ipr>, debe ser alguno de los dados por el comando AT+IPR=?. Para verificar la velocidad actual del modem se ejecuta AT+IPR=.
- **AT+CPIN=\*\*\*\*:** ingresa el código PIN. Para verificar el estado de la tarjeta SIM se ejecuta el comando “AT+CPIN?” y su formato de respuesta es:

*<code> READY: el modem no necesita código PIN*

*SIM PIN: el modem está esperando el código PIN*

*SIM PUK: el modem está esperando el código PUK*

*PH-SIM PIN: el modem está esperando el password de ingreso desde el teléfono a la SIM.*

- **AT+CREG=<n>:** Configura la forma en que se presentan los resultados de registro a la red, los cuales indican si la tarjeta SIM se encuentra registrada en la red o no. Para verificar el estatus de conexión de la tarjeta a la red se ejecuta “AT+CREG?”. Formato de salida +CREG: <n>,<stat> dónde:

*<n> 0: Deshabilita los resultados del registro de red*

*1: Habilita los resultados del registro de red*

*2: Habilita los resultados y además la información de ubicación.*

*<stat> 0: Tarjeta SIM no registrada.*

*1: Registrada.*

*2: no registrada, pero el módulo está buscando una red a la cual registrarse.*

*3: Registro de red denegado.*

*4: Desconocido.*

*5: Registrada, roaming.*

- **AT%BAND=<mode>,<band>**: Configura el modo en que se selecciona una banda. En caso de elegir modo manual, se debe especificar la banda. Para verificar la banda seleccionada actualmente se ejecuta "AT%BAND?".

<b>&lt;mode&gt;</b>	<i>0: Automático</i>
	<i>1: Manual.</i>
<b>&lt;band&gt;</b>	<i>1: GSM 900 MHz</i>
	<i>2: DCS 1800 MHz</i>
	<i>4: PCS 1900 MHz.</i>
	<i>31: GSM/EGSM/DCS/PCS/850</i>

- **AT+COPS=<mode>,<format>,<oper>**: Intenta seleccionar el operador de red especificado por <oper>, con el formato dado por <format>, ocupando el modo <mode>. Si el modo es automático, entonces el resto de los valores se omite. Para consultar el operador de red seleccionado se ejecuta "AT+COPS?".

<b>&lt;mode&gt;</b>	<i>0: Registro automático, 1: Registro manual, 2: Desregistrado de la red.</i>
<b>&lt;format&gt;</b>	<i>0: long format, 1: short format, 2: PLMN.</i>
<b>&lt;oper&gt;</b>	<i>Se configura el dato del operador en el formato dado por &lt;format&gt;.</i>

- **AT+CSQ**: Indica el nivel de potencia de la señal y la calidad de ésta. Su formato de salida es: +CSQ: <rssi>,<ber> dónde:

<b>&lt;rssi&gt;</b>	<i>Indicador de la potencia de la señal recibida.</i>
<i>0</i>	<i>-113 dBm o menos</i>
<i>1</i>	<i>-111 dBm</i>
<i>2-30</i>	<i>-109 -53 dBm</i>
<i>31</i>	<i>-51 dBm o más</i>
<i>99</i>	<i>desconocido o no detectable</i>

*<ber> Indica en porcentaje el bit error rate del canal*

*0-7 entre 0.1% y 15% o más.*

*99 desconocido o no detectable*

- **AT+CFUN=<fun>,<rst>**: selecciona el nivel de funcionalidad <fun> en el modem. El nivel "Máxima funcionalidad" es cuando el máximo nivel de poder es alcanzado. "Funcionalidad mínima" es cuando el mínimo nivel de poder es alcanzado. Para realizar llamadas GPRS el nivel de funcionalidad debe ser el máximo.

*<fun> 0: Mínima funcionalidad*

*1: Máxima funcionalidad*

*<rst> 0: No reiniciar el modem*

- **AT+FCLASS=<mode>**: Configura el módulo GSM en un cierto modo de operación dado por <mode> dónde:

*<mode> 0: Datos.*

*1: Voz.*

El comando AT+FCLASS? pregunta por el actual modo de operación en el que se encuentra el módulo. Entrega por respuesta <mode>.

- **AT+CMGF=<mode>**: Configura el tipo de formato de los mensajes de texto SMS. Donde:

*<mode> 0: formato PDU para mensajes SMS.*

*1: formato TEXTO para mensajes SMS.*

La diferencia entre ambos es que en PDU el dispositivo maneja los mensajes listos para enviarlos por la red, mientras que en el modo TEXTO, se deben convertir al recibirlos y reconvertirlos a PDU al enviarlos. "AT+CMGF?" pregunta por el formato actual de los mensajes.

- **AT+CMGW="<número>"**: Permite escribir un mensaje de texto y guardarlo en memoria, para ser enviado posteriormente. Donde <número> es el celular del destinatario. Al presionar ENTER, se debe esperar a que aparezca el símbolo '>' para luego comenzar a escribir el mensaje de texto. Luego se



debe presionar las teclas CTRL+Z y el mensaje será guardado. Como respuesta aparecerá el índice en el cual el mensaje fue guardado: +CMGW: <índice>. Este <índice> se debe ocupar para luego enviar el mensaje, ya que este comando sólo guarda en memoria el mensaje y no lo envía.

- **AT+CMSS=<índice>**: Envía un mensaje SMS guardado en la memoria <memws> (ver AT+CPMS), en la posición dada por <índice>. En caso de no haber un mensaje en esa posición, saldrá un error, del mismo modo si es que el número con el cual se guardó el mensaje no es válido. Si el envío del mensaje tiene éxito, la respuesta del comando será: +CMSS: <mr> Donde <mr> representa el número de referencia del mensaje.
- **AT+CMGL="<modo>"**: Comando que permite leer los mensajes guardados en la tarjeta SIM. Dependiendo del valor de <modo> (debe ir entre comillas y respetando las mayúsculas) se pueden ver los siguientes mensajes:

*<modo> REC READ: Muestra sólo los mensajes recibidos que ya se han leído.*

*REC UNREAD: Muestra sólo los mensajes recibidos aún sin leer.*

*STO UNSENT: Muestra sólo los mensajes guardados que no han sido enviados.*

*STO SENT: Muestra los mensajes guardados que han sido enviados.*

*ALL: Muestra todos los mensajes almacenados.*

- **AT+CGATT=<n>**: Permite iniciar la secuencia para el registro y conexión a la red GPRS. Donde:

*<n> 0: Detiene la secuencia para registro y conexión.*

*1: Comienza la secuencia para registro y conexión.*

Si por respuesta el MODEM entrega un OK, indica que la secuencia tuvo éxito.

- **AT+CGREG=<n>**: Si <n>=0, deshabilita la opción de aviso cada vez que se registra a una red GPRS (al mismo estilo que AT+CREG). Si <n>=1 habilita la opción de aviso. Para consultar si la tarjeta SIM está registrada al a red GPRS se ejecuta "AT+CGREG?". GPRS. La respuesta tiene el mismo formato que AT+CREG. La respuesta debe ser +CGREG: 0,1 o +CGREG: 0,5 para continuar. En vez de un cero, puede

aparecer un 1 o un 2 según la configuración, pero lo importante es que el segundo dígito debe ser un 1 o un 5, que indica que se ha conectado a la red GPRS.

- **AT%CGAATT=0,1:** Configura el módem para que automáticamente se registre y conecte a la red GPRS cuando el módem se enciende y además lo configura para que la desconexión sea manual.
- **AT+CGDCONT:** Para conectarse a una red de internet, es necesario dar cierta información sobre la cesión del suscriptor cuando éste tiene una cesión activa. Esta información es la que se conoce con el nombre de contexto PDP (Packet Data Protocol), la cual se entrega a la red y cuando un móvil desea ingresar a una red GPRS, debe primero registrarse y conectarse a ella (lo que se llama en inglés como "attach") y luego debe activar el contexto PDP. Para esto último se utiliza el siguiente comando:

*AT+CGDCONT=<n>,"IP", "<APN>", "",0,0 Configura el contexto PDP. Donde:*

*<n>                    número identificador del contexto. Puede haber más de uno y este número indica cual se activa.*

*"IP"                    Indica el tipo de protocolo a utilizar. En este módulo siempre se utiliza "IP".*

*<APN>                Nombre del punto de acceso (Access Point Name). Lo entrega el proveedor de servicio de red. En el caso de Movistar es: gprs.movistar.ve. "" Debe ir una dirección IP, pero si se dejan las comillas sin nada, la dirección es asignada dinámicamente por el ISP (Internet Service Provider) o proveedor de servicios de Internet. Se puede ver la dirección una vez conectado utilizando AT+CGPADDR.*

*0                        Compresión de información desactivada.*

*0                        Compresión de cabecera desactivada.*

- **AT+CGACT=<state>,<n>:** Activa el contexto dado por AT+CGDCONT, con el número identificador <n>. El estado viene dado por <state>=1 para activarlo y 0 para desactivarlo.

Algunos comandos, llevan al final un signo de interrogación (?). Esto quiere decir que se está pidiendo información. Mientras que un signo igual (=) quiere decir que se está configurando un parámetro, donde luego del signo igual se ingresa el valor o valores de los parámetros separados por coma que se desean

ajustar. La expresión igual interrogación (=?), se usa para obtener todo el rango de valores posibles que se pueden configurar.

## 2.5.4 Ejemplos de secuencias de comandos AT

Los ejemplos enunciados a continuación forman parte del conjunto de instrucciones para la configuración de la conexión GPRS, así como el envío y lectura de mensajes cortos de texto. El primer ejemplo proporciona un uso de los comandos AT para la configuración de la conexión de datos por medio de la cual se enviarán los mensajes multimedia, y el segundo ejemplo es el conjunto de necesarios para el envío y recepción de un mensaje corto de texto ó SMS.

### 2.5.4.1 Configuración para llamada GPRS

Verificación inicial de parámetros en la conexión y en el modem:

<i>AT+CSQ</i>	<i>Verificación de potencia y calidad de la señal.</i>
<i>+CSQ: 21, 99</i>	<i>Respuesta: potencia entre -109 y -53 dBm y tasa de error desconocida.</i>
<i>OK</i>	<i>(El comando se ejecutó exitosamente).</i>
<i>AT+CPIN</i>	<i>Verificación de la tarjeta SIM.</i>
<i>+CPIN: READY</i>	<i>Respuesta: La tarjeta SIM no requiere el código PIN.</i>
<i>OK</i>	
<i>AT+COPS?</i>	<i>Verificación de la selección de la operadora móvil.</i>
<i>+COPS: 0,0,"DIGITEL TIM"</i>	<i>Respuesta: Registro automático a la operadora, formato largo, operadora móvil "DIGITEL TIM".</i>
<i>OK</i>	
<i>AT+CREG?</i>	<i>Verificación registro de la tarjeta SIM en la red.</i>

*+CREG: 1,1* *Respuesta: activados resultados de registro a la red, tarjeta SIM registrada.*

*OK*

*AT+CGREG?* *Verificación de registro de la tarjeta SIM en la red GPRS.*

*+CGREG: 0,1* *Respuesta: Tarjeta SIM registrada en la red GPRS.*

*OK*

*AT%BAND?* *Verificar la banda de frecuencia seleccionada.*

*%BAND: 0,31* *Respuesta: selección automática, banda de frecuencia GSM/EGSM/DCS/PCS/850.*

*OK*

*AT+IPR?* *Verificación de la velocidad del hyperterminal (recomendable 115200).*

*+IPR: 115200* *Respuesta: velocidad de 115200.*

*OK*

*AT+CFUN=1* *Configuración modo de máxima funcionalidad del modem.*

*OK* *Respuesta: parámetro configurado exitosamente.*

*AT+FCLASS=0* *Configuración del modem en llamada de datos.*

*OK* *Respuesta: parámetro configurado exitosamente.*

Una vez verificado lo anterior, se procede a configurar el módem para la conexión de datos. Lo primero que se debe hacer es registrar y conectar la tarjeta SIM a la red GPRS. Para ello se utilizan los siguientes comandos:

<i>AT%CGAATT=0,1</i>	<i>Configuración para el que modem se registre y conecte automáticamente a la red GPRS al encenderse.</i>
<i>OK</i>	<i>Respuesta: parámetro configurado exitosamente.</i>
<i>AT+CGATT=1</i>	<i>Inicio de secuencia para el registro y conexión a la red GPRS.</i>
<i>OK</i>	<i>Respuesta: secuencia iniciada exitosamente.</i>
<i>AT+CGDCONT=1,"IP","expresate.digitel.ve",,,,0,0</i>	<i>Configuración del contexto PDP con el APN de la operadora Digitel.</i>
<i>OK</i>	<i>Respuesta: parámetro configurado exitosamente.</i>
<i>AT+CGACT=1,1</i>	<i>Activación del PDP configurado de ID = 1.</i>
<i>OK</i>	<i>Respuesta: parámetro configurado exitosamente.</i>

#### **2.5.4.2 Envío y lectura de un SMS**

A continuación se especifica la secuencia de comandos para enviar y leer un SMS.

<i>AT+CMGF=1</i>	<i>Configuración de formato del SMS a tipo texto.</i>
<i>OK</i>	<i>Respuesta: parámetro configurado exitosamente.</i>
<i>AT+CMGW="04125590152"</i>	<i>Se guarda un mensaje con número destinatario "04125590152".</i>
<i>&gt; Mensaje de prueba</i>	<i>Texto del mensaje: "Mensaje de prueba".</i>
<i>+CMGW: 7</i>	<i>Respuesta: mensaje almacenado en la posición de memoria de memoria 7.</i>
<i>OK</i>	<i>Mensaje creado y almacenado exitosamente.</i>
<i>AT+CMSS=7</i>	<i>Envío del mensaje almacenado en la posición de memoria 7.</i>
<i>+CMSS: 7</i>	<i>Respuesta: El mensaje de la posición 7 se ha enviado exitosamente.</i>
<i>OK</i>	
<i>AT+CMGL="STO SENT"</i>	<i>Lista los mensajes guardados que han sido enviados.</i>
<i>+CMGL: 7, "STO SENT", "04125590152", ,</i>	
<i>Mensaje de prueba</i>	<i>Respuesta: ID de memoria, tipo de mensaje, destinatario y abajo el mensaje.</i>
<i>OK</i>	

### 2.5.5 Comandos AT como herramienta para la programación avanzada

Actualmente, la mayoría de las aplicaciones de dominio específico como la que se desea elaborar en el presente trabajo, se desarrollan en lenguajes de alto nivel (Java, C#, C++, etc.) debido a su portabilidad y soporte de librerías para el manejo de datos y comunicación con los dispositivos periféricos del computador.

Los lenguajes de alto nivel proveen una herramienta llamada socket que permite el intercambio de un flujo de datos entre la aplicación y el dispositivo modem. Los *sockets* permiten a la aplicación enviar comandos AT para manipular las funcionalidades del dispositivo, en otras palabras, los *sockets* envían los comandos AT como cadenas de caracteres directamente al puerto donde se encuentra configurado el modem, que recibe la data y la interpreta como órdenes para desempeñar la funcionalidad asociada al comando.

Los comandos AT son un lenguaje de bajo nivel que solo provee instrucciones para la manipulación del modem que se ejecutan de forma secuencial, por esta razón al utilizar lenguajes de alto nivel para el envío de los comandos, se pueden aprovechar las metodologías y herramientas de programación para el desarrollo de aplicaciones de múltiples dominios con mayor grado de complejidad.

Con este punto se abarcan todas las definiciones, protocolos y herramientas involucradas en la investigación y desarrollo del proceso de envío y recepción de los mensajes multimedia. A continuación se definen las tecnologías Web y los lenguajes de programación seleccionados para la elaboración de la interfaz de gestión de reportes y transacciones, así como el desarrollo del motor de envío y recepción de los mensajes multimedia.

## 2.6 Aplicación Web

### 2.6.1 Definición

Se denomina aplicación Web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor Web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web (HTML, JavaScript, Java, ASP.net, PHP, etc.) en la que se confía la ejecución al navegador [23].

Una aplicación Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo [23].

## 2.7 Lenguaje de Marcado de Hipertexto (HTML, HyperText Markup Language)

### 2.7.1 Definición

HTML es el acrónimo de *HyperText Markup Language* (lenguaje de marcado de hipertexto). Es el lenguaje más utilizado para la presentación de textos estructurados en formato hipertexto, estándar de las páginas Web [22].

HTML es utilizado por prácticamente todos los navegadores Web del mercado, con el fin de presentar al visitante de una página Web el contenido de la misma tal como el diseñador quiere que se muestre a su público. HTML es el estándar usado en la *World Wide Web*, y se ha convertido en uno de los formatos más populares que existen para la construcción de documentos [22].

HTML es un lenguaje que basa su sintaxis en un elemento de base al que llamamos etiqueta, donde por lo general toda etiqueta representa una instrucción. La etiqueta presenta frecuentemente dos partes: un comienzo y un final, y todo lo incluido en el interior de esa etiqueta sufrirá las modificaciones que caracterizan a esta etiqueta, esto le indica al navegador Web como va a representar esa información, imagen, video, etc., en el browser para ser visualizada por el usuario desde su computador. Por ejemplo: `<b>` es una etiqueta que indica al browser que todo lo contenido dentro de esas etiquetas va a estar en negrita y su correspondiente etiqueta de cierre es `</b>` [23].

**Ejemplo:**

Código HTML: `<b>Hola Mundo</b>`

Visualización en el browser: Hola Mundo

La estructura básica de un documento HTML es la siguiente:

```
<html>

  <head>

    <title>Titulo de la página</title>

  </head>

  <body>

    Cuerpo de la página

  </body>

</html>
```

Las etiquetas `<html>` y `</html>` representan el documento HTML. Las etiquetas `<head>` y `</head>` representan el encabezado donde colocaremos etiquetas de índole informativo, como por ejemplo el titulo de la página. Las etiquetas `<body>` y `</body>` representan el cuerpo de la pagina, que será donde colocaremos nuestro texto, imágenes, etc. delimitados a su vez por otras etiquetas [23].



## 2.8 Hojas de Estilo en Cascada (CSS, Cascade Style Sheet)

### 2.8.1 Definición

Hojas de Estilo en Cascada, es un lenguaje que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a mostrar, o incluso cómo va a ser resaltada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. Se trata de una especificación sobre los estilos y formatos aplicables a un documento HTML, y trata de dar la separación definitiva de la lógica (estructura) y el físico (presentación) del documento [24].

### 2.8.2 Sintaxis

Una hoja de estilo se compone de reglas de visualización (reglas de estilo). Cada regla consta de un selecto, que es el que indica a que elemento o parte de la página se aplica el estilo; normalmente los selectores son etiquetas HTML. A cada selector debe seguir una declaración del estilo que ha de serle aplicado. Toda declaración tiene dos partes: Propiedad (por ejemplo, color) y valor (por ejemplo, blue) [25].

Los selectores disponibles son todas las etiquetas HTML. A cualquier etiqueta HTML se le puede asignar un estilo. Los estilos, entonces, se definen así:

*selector { propiedad:valor }*

*h1 {color : blue}*

Los selectores se escriben omitiendo las llaves < y >, es decir, simplemente h1, h2, etc.

La declaración {propiedad : valor} ha de ir encerrada en llaves { }.

Pueden asignarse varias propiedades a un mismo selector, separadas por punto y coma:

```
Selector { propiedad1: valor1; propiedad2: valor2 }
```

```
h1 { color: yellow; background-color: red }
```

También se pueden asignar propiedades de forma conjunta a varios selectores:

```
Selector1, Selector2, {propiedad1:valor1; propiedad2:valor2}
```

Como ejemplo, veamos una definición de un estilo para los elementos (Selectores) h1 y h2:

```
<head>
```

```
<title></title>
```

```
<style>
```

```
h1 { font-size: x-large; color: red; }
```

```
h2 { font-size: large; color: blue; }
```

```
</style>
```

```
</head>
```

## 2.8.3 Ventajas y Desventajas

### *Ventajas*

- Control centralizado de la presentación de un sitio Web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Los Navegadores permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio web, con lo que aumenta considerablemente la accesibilidad. Por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar más los enlaces.
- Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario. Por ejemplo, para ser impresa, mostrada en un dispositivo móvil, o ser "leída" por un sintetizador de voz.
- El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño (siempre y cuando no se utilice estilo en línea).

### *Desventajas*

La mayor desventaja de CCS es el soporte para algunos navegadores (por ejemplo Internet Explorer), los cuales tienen diferencias notables a navegadores tipo Firefox y muchos otros que siguen los estándares del W3C. Esto a veces puede causar errores inesperados de visualización en IE como el modelo de caja de errores, mientras que funciona bien en todos los otros navegadores.

## 2.9 JavaScript

### 2.9.1 Definición

JavaScript es un lenguaje de programación del lado del cliente, ya que es el navegador el que soporta la carga de procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores en la actualidad, es el lenguaje de programación del lado del cliente más utilizado [26].

JavaScript provee herramientas para la creación “efectos especiales” en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones JavaScript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador [26].

JavaScript es un lenguaje de programación interpretado, lo que no hace necesaria la compilación de sus programas para ejecutarlos, es decir, los programas en JavaScript se pueden correr directamente en cualquier navegador [26].

### 2.9.2 Ventajas y Desventajas

#### *Ventajas*

- Es un código “interpretado” por el cliente.
- Es un código orientado a objetos.
- Es un código integrado a HTML.
- Reutilización de código de programación [27].

## ***Desventajas***

JavaScript tiene algunas limitaciones, las cuales derivan de su objetivo principal, que es adicionar interactividad a las páginas web [27].

- JavaScript no puede interactuar con el servidor. Una de sus principales desventajas es su principal fortaleza. A excepción de la tecnología de desarrollo AJAX (*Asynchronous JavaScript And XML*), JavaScript trabaja enteramente dentro del navegador del cliente y no se comunica con el servidor ni con otras máquinas.
- JavaScript no puede crear gráficos. JavaScript no pueden crear sus propios gráficos, solamente puede manipular gráficos hechos en otros programas.
- JavaScript trabaja de forma diferente en diferentes navegadores que no sigan el estándar W3C [27].

## 2.10 Preprocesador de Hipertexto, PHP

### 2.10.1 Definición

PHP es un acrónimo recursivo que significa *Hypertext Pre-Processor*. PHP es un lenguaje interpretado de propósito general ampliamente usado, diseñado especialmente para desarrollo Web y que puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas Web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. La mayor parte de su sintaxis es similar a la de C, Java y Perl con algunas diferencias específicas. La meta del lenguaje es facilitar a los desarrolladores la creación dinámica de páginas [28].

En el nivel más básico PHP es equiparable a un CGI (*Common Gateway Interface*) cualquiera. La mayor fortaleza de PHP es que está diseñado para soportar accesos a muchos tipos de bases de datos como: Adabas D, dBase, Empress, FiclePro, informix, Internase, Solid, Sybase, Velocis, Unix dbm, mSQL, MySQL, Oracle, PostgreSQL, SQL Server y Microsoft Access.

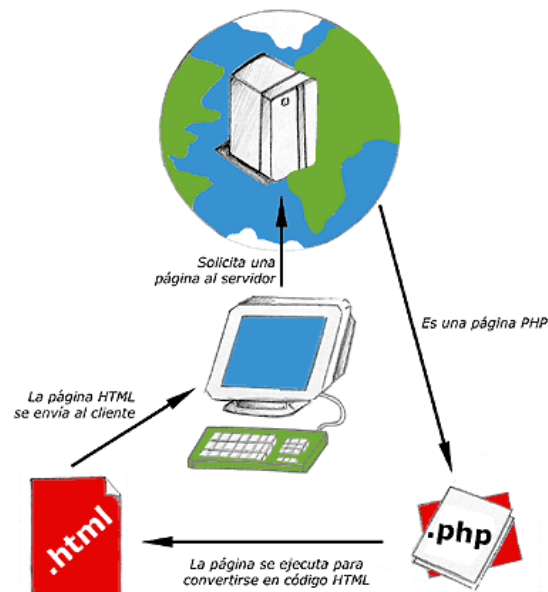


Figura 10: Esquema estructural de una aplicación Web en PHP [30]

Adicionalmente, PHP soporta la utilización protocolos como IMAP, SNMP, NNTP, POP3 o HTTP a nivel de *socket*.

Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparen rápidamente. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP.

## 2.10.2 Ventajas y Desventajas

### ***Ventajas***

- Es un lenguaje multiplataforma.
- Completamente orientado al desarrollo de aplicaciones Web dinámicas con acceso a información almacenada en una Base de Datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de expandirse y complementarse utilizando módulos llamados “/ext/” o extensiones.
- Es un lenguaje de código libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite el desarrollo de aplicaciones con enfoque orientado a objetos.
- Es un lenguaje débilmente tipeable (no es necesaria la declaración de variables ni fijarles un tipo de dato en específico), aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (a partir de la versión PHP5).

## ***Desventajas***

- La ofuscación de código es la única forma de ocultar los códigos fuentes.
- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias HTML, PHP y SQL.
- No posee manejo adecuado de la tabla Unicode.
- La programación orientada a objetos es aún muy deficiente para aplicaciones grandes.

## **2.11 MySQL**

### **2.11.1 Definición**

MySQL es un sistema de gestión de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones [29].

MySQL es muy utilizado en aplicaciones web, en plataformas: Linux/Windows-Apache-MySQL-PHP/Perl/Python.

El servidor de bases de datos MySQL es la base de datos de fuente abierta más popular en el mundo. Su arquitectura lo hace extremadamente rápido y fácil de adaptar.

En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones [29].

Las principales características de este gestor de bases de datos son las siguientes [29]:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.



- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
- Gran portabilidad entre sistemas operativos y arquitecturas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y contraseñas, manteniendo un nivel seguridad en los datos.

Una vez definidas las tecnologías Web y de manejo de la base de datos, se describirá a continuación los conceptos y características sobre Java, plataforma a utilizar para el desarrollo del motor de envío y recepción de los mensajes multimedia.

La correspondencia entre Java y PHP en la presente investigación se establece a través de los módulos a desarrollar en dichos lenguajes que acceden a una misma base de datos MySQL, donde el módulo Java proveerá las funcionalidades para el envío MMS, obteniendo la información necesaria para llevar a cabo dicho proceso y registrando las transacciones y el módulo PHP efectuará el acceso a la base de datos para la gestión y visualización de reportes y transacciones vía Web.

## **2.12 Java**

### **2.12.1 Definición**

Java es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principio de los años 90's.

Java es un lenguaje moderno, de alto nivel, que recoge los elementos de programación que típicamente se encuentran en todos los lenguajes de programación, permitiendo la realización de programas profesionales.

## 2.12.2 Características fundamentales

- Java es un lenguaje orientado a objetos, eso implica que su concepción es muy próxima a la forma de pensar humana.
- Es un lenguaje que es compilado, generando ficheros de clases compilados, pero estas clases compiladas, son en realidad interpretadas por la máquina virtual de java. Siendo la máquina virtual de java la que mantiene el control sobre las clases que se estén ejecutando.
- Es un lenguaje multiplataforma: El mismo código java que funciona en un sistema operativo, funcionará en cualquier otro sistema operativo que tenga instalada la máquina virtual java.
- Es un lenguaje seguro: La máquina virtual, al ejecutar el código java, realiza comprobaciones de seguridad, además el propio lenguaje carece de características inseguras, como por ejemplo los punteros.
- Gracias al API de java podemos ampliar el lenguaje para que sea capaz de, por ejemplo, comunicarse con equipos mediante red, acceder a bases de datos, crear páginas HTML dinámicas, crear aplicaciones visuales de ventanas, etc.

El Lenguaje de Programación Java proporciona un conjunto de librerías de propósito específico para el desarrollo de aplicaciones de escritorio en diversos dominios. Dentro del contexto de lo planteado en la presente investigación, Java provee librerías que permiten el desarrollo de componentes vitales de la plataforma de mensajería multimedia, entre las cuales podemos especificar:

- Librería de gestión de la conexión GPRS/WAP a través de la cual se envían los mensajes multimedia.
- Librerías que implementan los sockets y gestionan de la comunicación con el modem serial para el envío de los comandos AT.
- Conexión con la base de datos MySQL para el registro transaccional de los mensajes.

## Capítulo III: Marco Aplicativo

En este capítulo se presenta una adaptación del proceso de desarrollo de software al caso particular de estudio, basado en el modelo en cascada, dentro del contexto de una metodología *Ad-hoc*, para la construcción de la plataforma de envío y recepción de mensajes multimedia (MMS) sobre un ambiente Web, tecnologías de código abierto y módems GPRS. En este sentido, se describen cada una de las fases del método utilizado para el desarrollo la herramienta.

El desarrollo de la solución se dividió en dos procesos:

- Envío y Recepción MMS (Aplicación Motor).
- Administración de Contenidos MMS y Registro Transaccional (Aplicación Web).

Los procesos son independientes entre sí, es por esto que cada uno posee su propia cascada como modelo de desarrollo, la cual se amoldó a los requerimientos y necesidades de estos.

Fue aplicado un modelo de desarrollo en cascada ya que es sencillo de implementar y resulta apropiado para sistemas donde los requerimientos iniciales son claros y no poseen tendencia de cambiar radicalmente a lo largo del tiempo, caso similar al de la herramienta propuesta.

Se utilizó una metodología *Ad-hoc* ya que permite adaptar y personalizar el modelo de desarrollo aplicado de acuerdo con las características específicas de la aplicación a construir.

De igual manera será utilizado *Unified Modeling Language* (UML) como formalismo para modelar y documentar las distintas vistas de la aplicación.

## **3.1 Modelo de desarrollo en cascada**

### **3.1.1 Definición**

Es un ciclo de vida que admite iteraciones. Después de cada etapa se realiza una o varias revisiones para comprobar si se puede pasar a la siguiente. Es un modelo rígido, poco flexible, y con muchas restricciones. Aunque fue uno de los primeros, y sirvió de base para el resto de los modelos de ciclo de vida.

Una de sus ventajas, además de su planificación sencilla, es la de proveer un producto con un elevado grado de calidad sin necesidad de un personal altamente calificado. Se pueden considerar como inconvenientes: la necesidad de contar con todos los requerimientos o la mayoría al comienzo del proyecto, y, si se han cometido errores y no se detectan en la etapa inmediata siguiente, es costoso y difícil volver atrás para realizar la corrección posterior.

Además, los resultados no los veremos hasta que no estemos en las etapas finales del ciclo, por lo que, cualquier error detectado nos trae retraso y aumenta el costo del desarrollo en función del tiempo que insume la corrección de éstos.

Es un ciclo adecuado para los proyectos en los que se dispone de todos los requerimientos al comienzo, para el desarrollo de un producto con funcionalidades conocidas o para proyectos, que aun siendo muy complejos, se entienden perfectamente desde el principio.

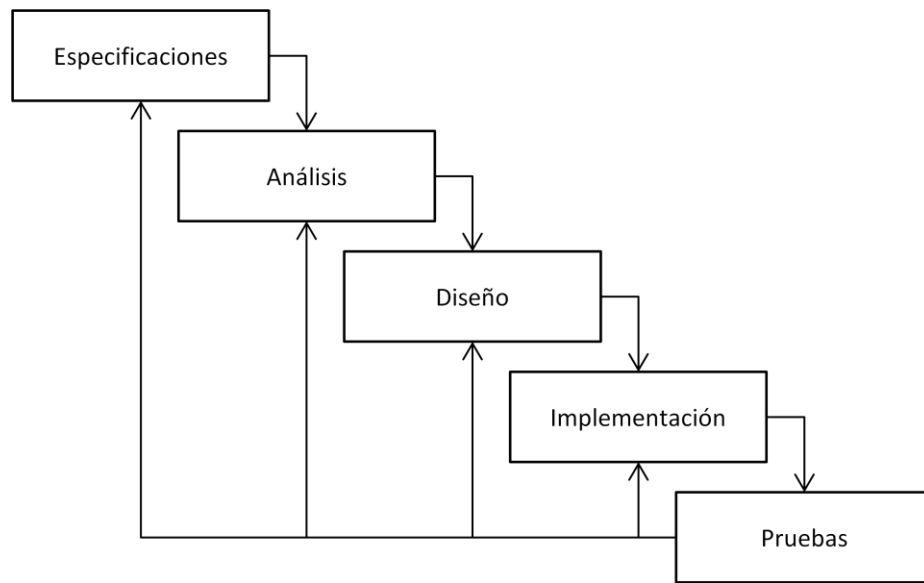


Figura 11: Etapas del modelo en cascada a ser implementado

*Elaboración: Fuente propia*

### 3.1.2 Etapas

- **Especificación de requerimientos:** Esta etapa tiene como objetivo la identificación de los requerimientos y funcionalidades que ofrecerá al usuario del sistema a desarrollar (qué, y no cómo, se va a desarrollar).

Dado que normalmente se trata de necesidades del cliente para el que se creará la aplicación, el documento resultante suele tener como origen una serie de entrevistas cliente-proveedor situadas en el contexto de una relación comercial.

Por medio de esta etapa se obtendrá un nuevo documento que definirá con más precisión el sistema requerido por el cliente (el empleo de los casos de uso es una muy buena elección para llevar a cabo la especificación del sistema).

- **Análisis:** En esta etapa se debe entender y comprender de forma detallada cual es la problemática a resolver, verificando el entorno en el cual se encuentra dicho problema, de tal manera que se obtenga

la información necesaria y suficiente para afrontar su respectiva solución. Esta etapa es conocida como la del “qué” se va a solucionar [31].

Se determinan los elementos que intervienen en el sistema a desarrollar, su estructura, relaciones, evolución temporal, funcionalidades, para así obtener una descripción clara de qué producto se va a construir, qué funcionalidades aportará y qué comportamiento tendrá [6].

- **Diseño:** Una vez que se tiene la suficiente información del problema a solucionar, es importante determinar la estrategia que se va a utilizar para resolver el problema. Esta etapa es conocida bajo el “cómo” se va a solucionar [31].

En esta etapa se debe determinar cómo debe ser construido el sistema en cuestión; se define en detalle entidades y relaciones de las bases de datos, se selecciona(n) el(los) lenguaje(s) a utilizar, el Sistema Manejador de Bases de Datos, Patrones de Diseño, *Frameworks*, IDE's, etc [6].

- **Implementación:** partiendo del análisis y diseño de la solución, en esta etapa se procede a desarrollar el correspondiente programa que solucione el problema mediante el uso de una herramienta computacional determinada [31].

En esta fase se codifican los algoritmos y estructuras de datos, definidos en las etapas anteriores, en el correspondiente lenguaje de programación o para un determinado sistema gestor de bases de datos. En muchos proyectos se pasa directamente a esta etapa; son proyectos muy arriesgados que adoptan un modelo de ciclo de vida de *code & fix* (codificar y corregir) donde se eliminan las etapas de especificaciones, análisis y diseño con la consiguiente pérdida de control sobre la gestión del proyecto [6].

- **Pruebas:** Los errores humanos dentro de la programación de los computadores son muchos y aumentan considerablemente con la complejidad del problema. Cuando se termina de escribir un programa de computador, es necesario realizar las debidas pruebas que garanticen el correcto funcionamiento de dicho programa bajo el mayor número de situaciones posibles a las que se pueda enfrentar [31].

El objetivo de esta etapa es garantizar que el programa no contiene errores de diseño o codificación y la verificación de que el sistema desarrollado cumple con los requerimientos expresados inicialmente por el cliente y que han dado lugar al presente proyecto [6].

## **3.2 Envío y Recepción MMS**

Este proceso consta de las actividades necesarias para el desarrollo de una aplicación motor de escritorio *stand alone* que permita la codificación, decodificación e intercambio (envío y recepción) de mensajes multimedia con uno o más clientes móviles.

### **3.2.1 Requerimientos**

Los requerimientos identificados para este proceso son los siguientes:

- Configuración de los parámetros necesarios en el modem GSM y establecimiento de la conexión a la red GPRS de la operadora móvil.
- Construcción y codificación de mensajes multimedia a nivel de binario.
- Envío de mensajes multimedia a través de la conexión GPRS establecida anteriormente.
- Recepción, decodificación y descarga del contenido multimedia de los MMS recibidos.
- Administración de contenidos e información en una base de datos, necesaria para llevar a cabo los procesos de envío y recepción MMS.
- Elaboración de una interfaz gráfica para la interacción del usuario con la aplicación motora.

### **3.2.2 Análisis**

Se enfoca en el análisis desde la perspectiva del problema, con la finalidad de modelar el proceso que se desea automatizar de manera objetiva, permitiendo posteriormente realizar el diseño de la solución. En esta etapa se define una especificación conceptual del motor de envío y recepción MMS y se elaboran los modelos de casos de uso y objetos del dominio.

### 3.2.2.1 Diagrama de Casos de Uso

Especifica la funcionalidad y el comportamiento del sistema mediante la interacción de otros usuarios y/o sistemas.

La Figura 12 presenta el modelo de casos de uso elaborado para modelar las funcionalidades y el comportamiento del motor de envío y recepción MMS.

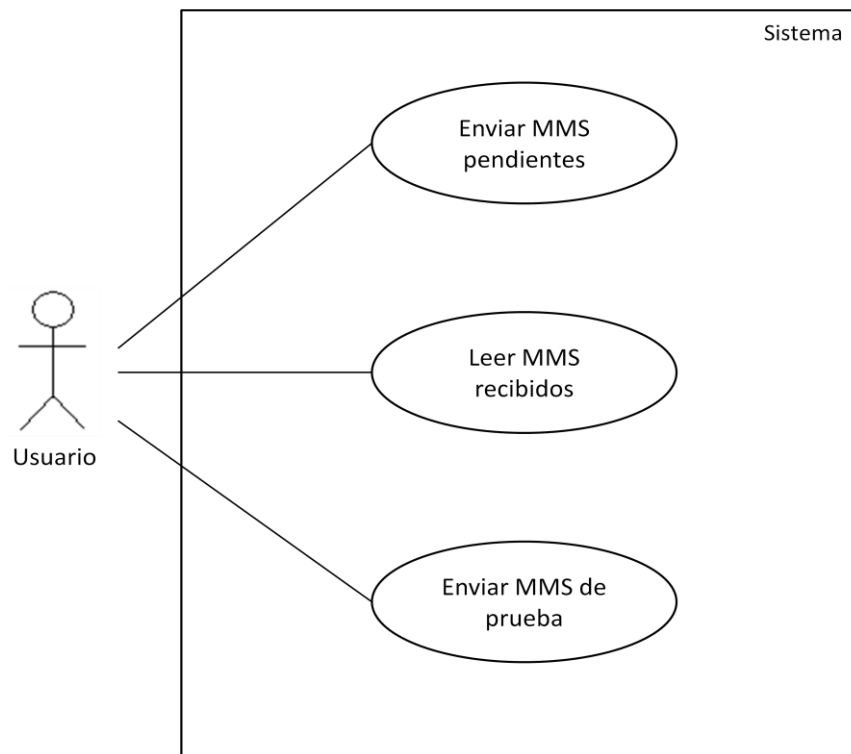


Figura 12: Diagrama de casos de uso del proceso de envío y recepción MMS



## Descripción de los actores

**Usuario:** persona que puede acceder a la herramienta para acceder a las funcionalidades que provee la interfaz gráfica del motor.

## Descripción de los casos de uso

A continuación se describen los casos de uso presentados en el modelo:

<b>Nombre:</b>	Enviar MMS pendientes
<b>Nivel:</b>	0
<b>Resumen:</b>	El usuario inicia/activa el envío de los mensajes multimedia pendientes, para que el motor los ejecute.
<b>Dependencias:</b>	Este caso de uso no posee dependencias.
<b>Actor:</b>	Usuario.
<b>Precondición:</b>	Para que el motor pueda realizar el procesamiento de los mensajes pendientes, se debe editar correctamente en el archivo de configuración los parámetros necesarios para el procesamiento de los MMS como la operadora móvil, conexión de red a utilizar y la base de datos a conectarse.
<b>Descripción:</b>	<ul style="list-style-type: none"><li>✓ El usuario activa el envío de mensajes.</li><li>✓ El motor verifica en la base de datos si existen mensajes pendientes por enviar.</li><li>✓ El motor construye los mensajes multimedia a nivel de bytes (PDU's MMS).</li><li>✓ Se establece la conexión con el MMSC de la operadora y se envían los mensajes.</li><li>✓ Se actualiza en la base de datos el estatus de los mensajes pendientes por enviar a enviados.</li></ul>
<b>Postcondición:</b>	Se actualiza el estatus y la fecha de envío de los mensajes.

**Tabla 5: Descripción de caso de uso Enviar MMS pendientes**

<b>Nombre:</b>	Leer MMS recibidos
<b>Nivel:</b>	0
<b>Resumen:</b>	El usuario inicia/activa la lectura de los mensajes recibidos por el modem GSM almacenados en la tarjeta SIM.
<b>Dependencias:</b>	Este caso de uso no posee dependencias.
<b>Actor:</b>	Usuario.
<b>Precondición:</b>	Se debe editar correctamente en el archivo de configuración los parámetros necesarios para el procesamiento de los MMS como la operadora móvil, conexión de red a utilizar y la base de datos a conectarse.
<b>Descripción:</b>	<ul style="list-style-type: none"> <li>✓ El usuario activa la lectura de los mensajes recibidos.</li> <li>✓ El motor verifica los mensajes almacenados en la tarjeta SIM del modem, si el mensaje es un SMS, extrae su información y la almacena en la base de datos, si es un MMS, lo decodifica y obtiene el URL proporcionada por el MMSC donde se encuentra alojado el contenido multimedia.</li> <li>✓ Se establece la conexión con la operadora móvil y se procede a realizar la descarga del contenido del mensaje desde la URL obtenida.</li> <li>✓ Se almacena en la base de datos el paquete binario obtenido que contiene los objetos multimedia.</li> </ul>
<b>Postcondición:</b>	Se eliminan de la tarjeta SIM los mensajes recibidos procesados.

**Tabla 6: Descripción de caso de uso Leer MMS recibidos**

<b>Nombre:</b>	Enviar MMS de prueba
<b>Nivel:</b>	0
<b>Resumen:</b>	El usuario envía un MMS de prueba desde la interfaz del motor.
<b>Dependencias:</b>	Este caso de uso no posee dependencias.
<b>Actor:</b>	Usuario.
<b>Precondición:</b>	Se debe editar correctamente en el archivo de configuración los parámetros necesarios para el procesamiento de los MMS como la operadora móvil y conexión de red a utilizar (no es necesario conectarse a la base de datos).
<b>Descripción:</b>	<ul style="list-style-type: none"> <li>✓ El usuario coloca el asunto, texto e imagen del mensaje que desea enviar.</li> <li>✓ El motor construye el mensaje multimedia a nivel de bytes (PDU MMS).</li> <li>✓ Se establece conexión con el MMSC de la operadora y se envía el mensaje.</li> </ul>
<b>Postcondición:</b>	Este caso de uso no tiene postcondición.

**Tabla 7: Enviar MMS de prueba**

### 3.2.2.2 Modelo Objetos del Dominio

Contiene los objetos pertenecientes al dominio de la aplicación, con sus respectivas descripciones y relaciones. La Figura 13 presenta el modelo objeto del dominio elaborado para este proceso.

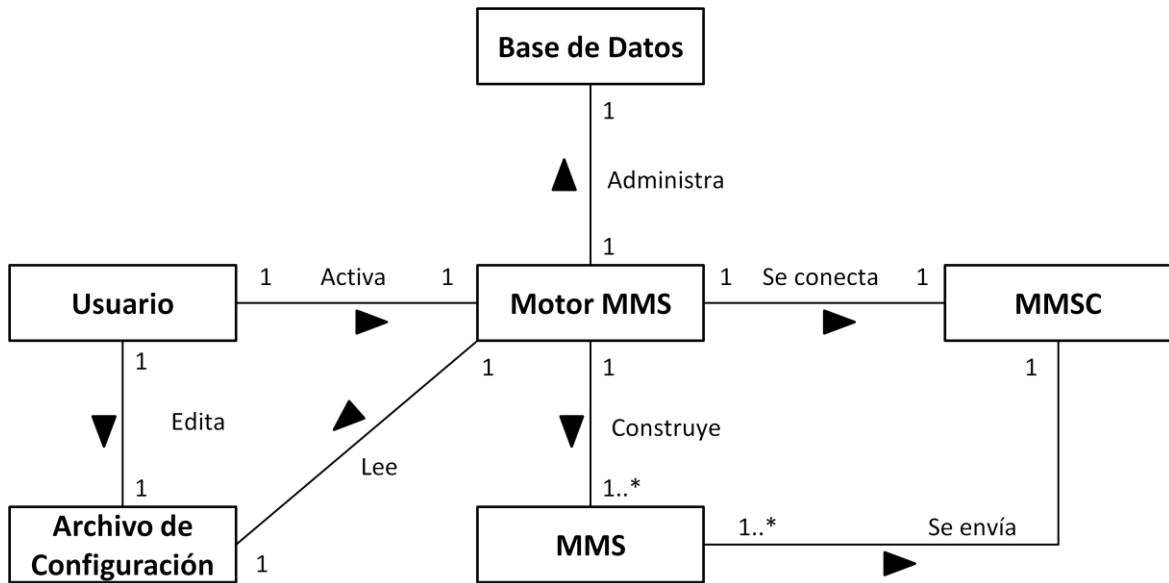


Figura 13: Diagrama de objetos del dominio del proceso de envío y recepción MMS

Los objetos identificados en el dominio son:

- **Motor MMS:** representa a la aplicación que se encarga de crear, enviar y recibir los mensajes multimedia estableciendo conexión con el MMSC y gestionando la información asociada en la base de datos.
- **Usuario:** representa a la persona que interactuará con la aplicación para activar el envío y recepción de los mensajes.
- **MMS:** objeto que representa al mensaje multimedia construido, enviado y/o recibido por el motor a través de la red móvil.

- **MMSC:** representación del MMS Centre de la operadora móvil, encargado de recibir, enrutar, almacenar y enviar mensajes multimedia.
- **Base de Datos:** objeto que representa el repositorio de datos donde se almacena información transaccional relacionada a los mensajes multimedia.
- **Archivo de Configuración:** representa el documento de texto que contiene los parámetros de configuración necesarios para el funcionamiento del motor (base de datos, datos de conexión, restricciones de los MMS, etc.).

### 3.2.3 Diseño

La fase de diseño se encuentra orientada al dominio de la solución y tiene como entrada los artefactos productos de la fase de análisis y la arquitectura donde se va a desarrollar la aplicación. Su objetivo consiste en diseñar lo relacionado con la implementación refinando los artefactos de análisis y definiendo los elementos de software que conforman la solución. En esta fase se realizan el diagrama de clases, modelo entidad relación y un diagrama de secuencia describiendo el comportamiento de la aplicación motor, todos estos diagramas haciendo referencia al proceso de envío y recepción MMS.

#### 3.2.3.1 Diagrama de Clases

A continuación se presenta en las Figuras 17, 18, 19 y 20 el diagrama de clases persistentes elaborado para este proceso.

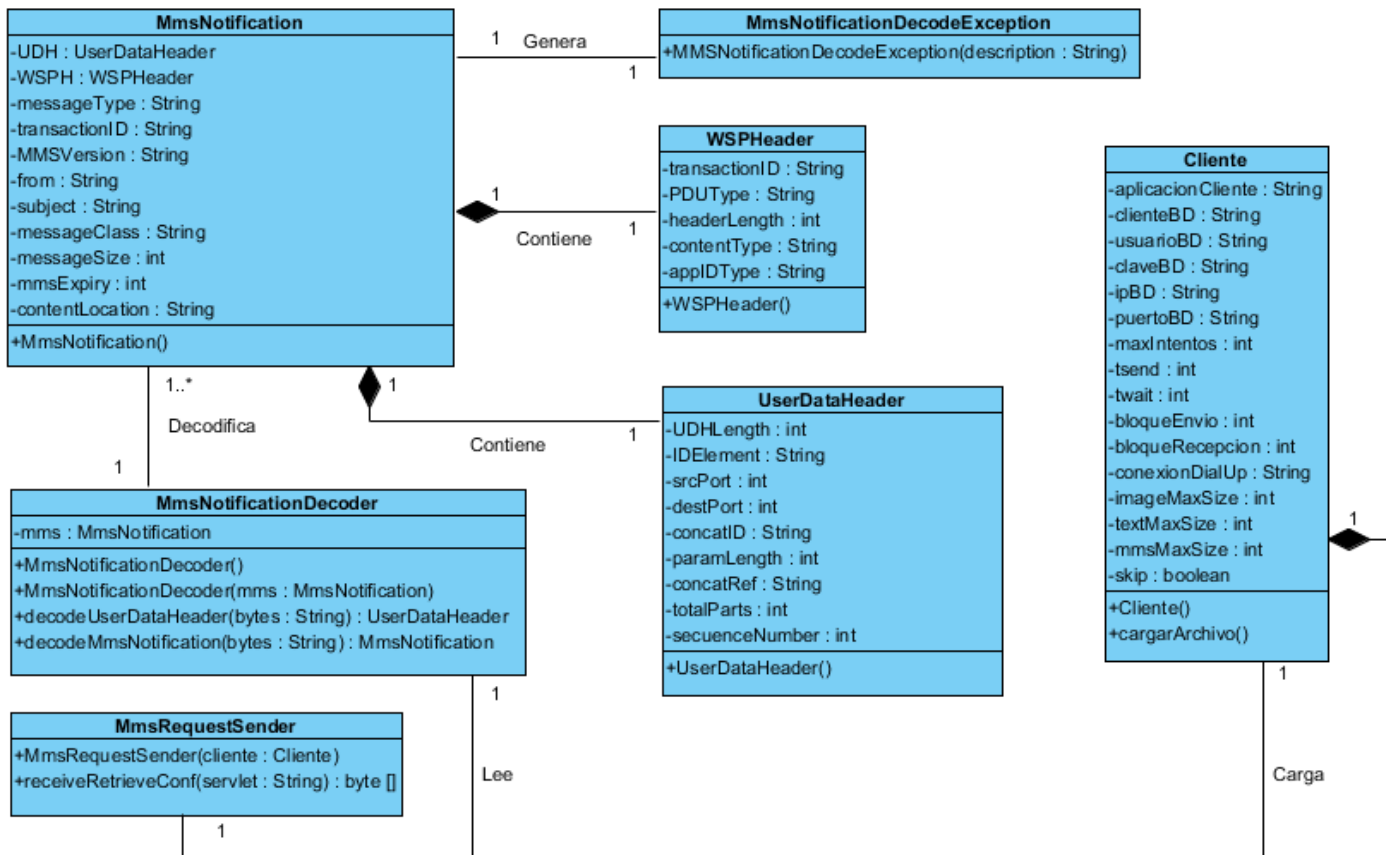


Figura 14: Diagrama de Clases Proceso Envío y Recepción MMS (Superior Izquierdo)

A continuación una descripción breve sobre la funcionalidad de cada una de las clases:

- **MmsNotification:** clase que implementa la estructura de una notificación MMS con sus parámetros (Clase *Bean*).
- **MmsNotificationDecoder:** implementa el proceso de decodificación de un notificación MMS y crea un objeto de tipo *MmsNotification* con toda la información asociada.
- **MmsNotificationDecodeException:** clase de tipo excepción, la cual instanciada cuando ocurre un error en el proceso de decodificación de una notificación MMS.

- **MmsRequestSender:** clase encargada de implementar el proceso de recuperación del contenido multimedia de los mensajes.
- **WSPHeader:** implementa la estructura del encabezado WSP (*Wireless Session Protocol*) que forma parte de una notificación MMS (Clase *Bean*).
- **UserDataHeader:** implementa la estructura del encabezado UDH (*User Data Header*) que forma parte de una notificación MMS (Clase *Bean*).
- **Cliente:** clase que contiene toda la información cargada del archivo de configuración *sendmms.properties*.

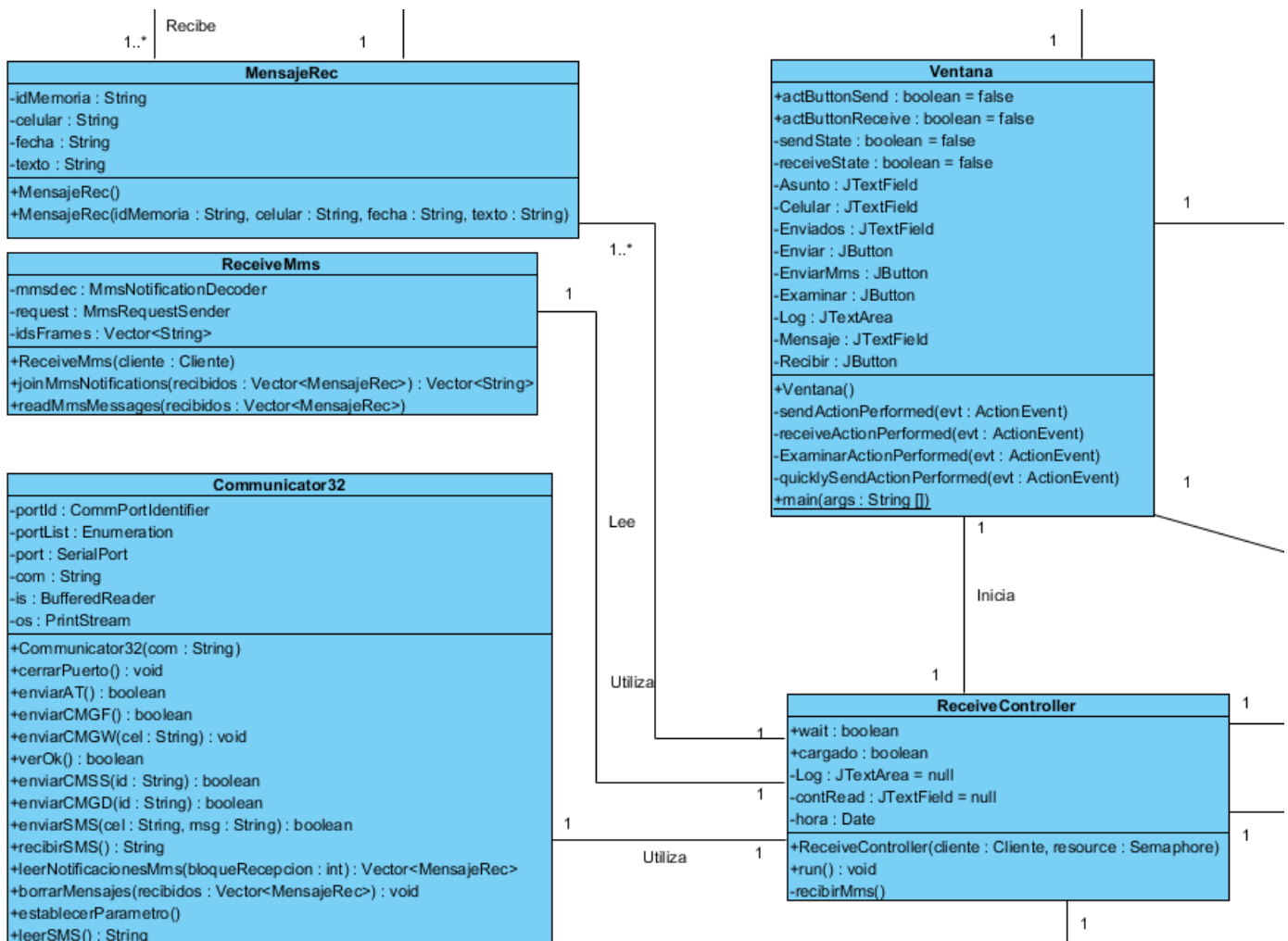


Figura 15: Diagrama de Clases Proceso Envío y Recepción MMS (Inferior Izquierdo)

- **MensajeRec:** almacena la información básica de un mensaje recibido por el modem GSM (Clase *Bean*).
- **ReceiveMms:** clase intermediaria que instancia al *MmsNotificationDecoder* y al *MmsRequestSender*, además de implementar métodos de procesamiento de las notificaciones MMS.
- **Communicator32:** implementa los métodos para abrir y cerrar conexión con los puertos seriales, creación de sockets y envío de comandos AT al modem GSM.
- **Ventana:** clase que hereda de *Swing* e implementa la interfaz de usuario de la aplicación motor.
- **ReceiveController:** clase controlador de la lectura y recuperación de mensajes multimedia, la cual hereda de *Thread*.

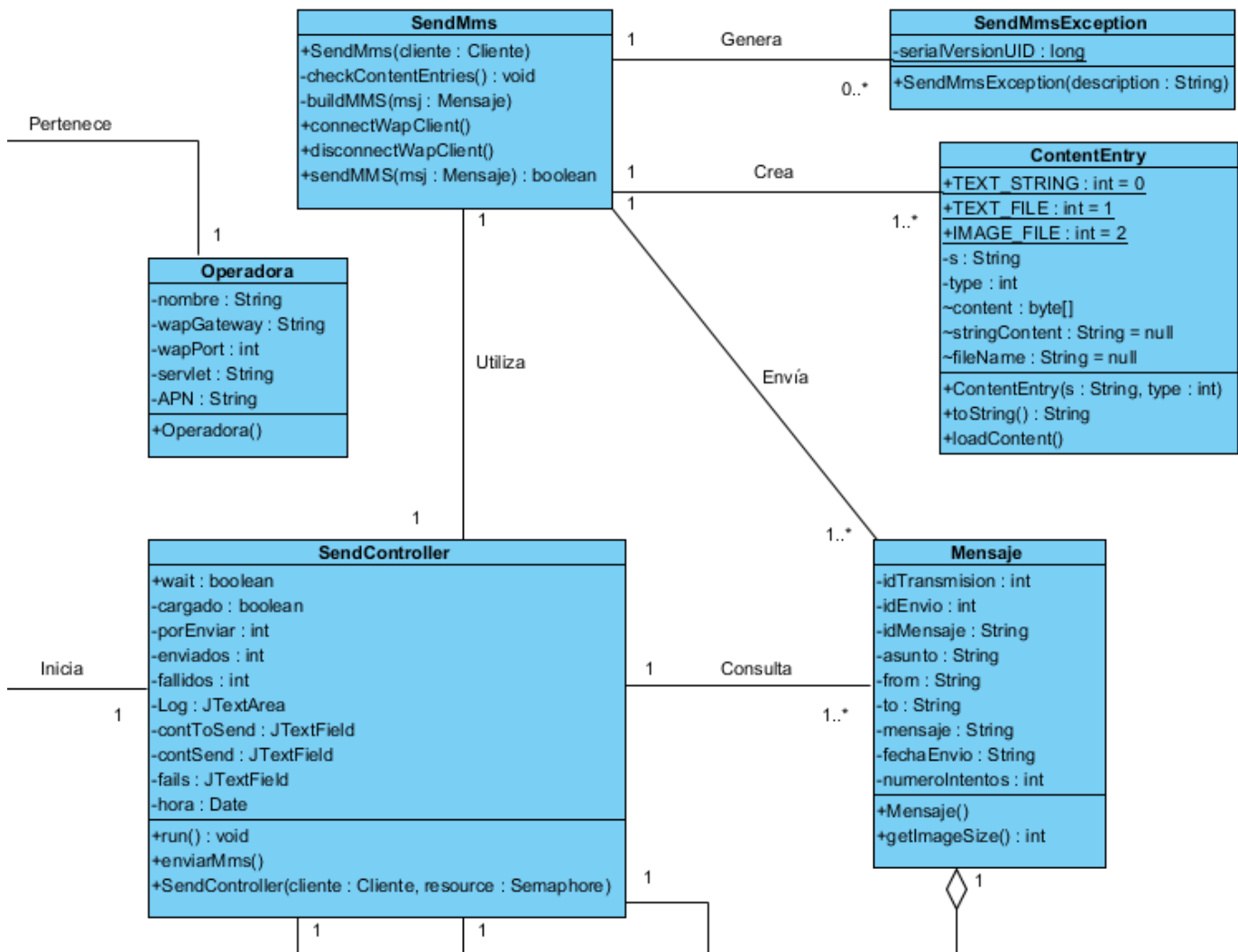


Figura 16: Diagrama de Clases Proceso Envío y Recepción MMS (Superior Derecho)

- **SendController:** clase controlador que hereda de *Thread* y coordina el proceso de envío de mensajes multimedia.
- **SendMms:** clase que implementa los métodos para construir y enviar mensajes multimedia.
- **SendMmsException:** clase de tipo excepción que se instancia cuando ocurre un error en el proceso de envío MMS.
- **ContentEntry:** implementa y gestiona la estructura de los elementos multimedia del mensaje.
- **Operadora:** contiene la información de configuración de la operadora para conectarse a su MMSC (Clase *Bean*).
- **Mensaje:** define la estructura básica de un mensaje multimedia a enviar (Clase *Bean*).

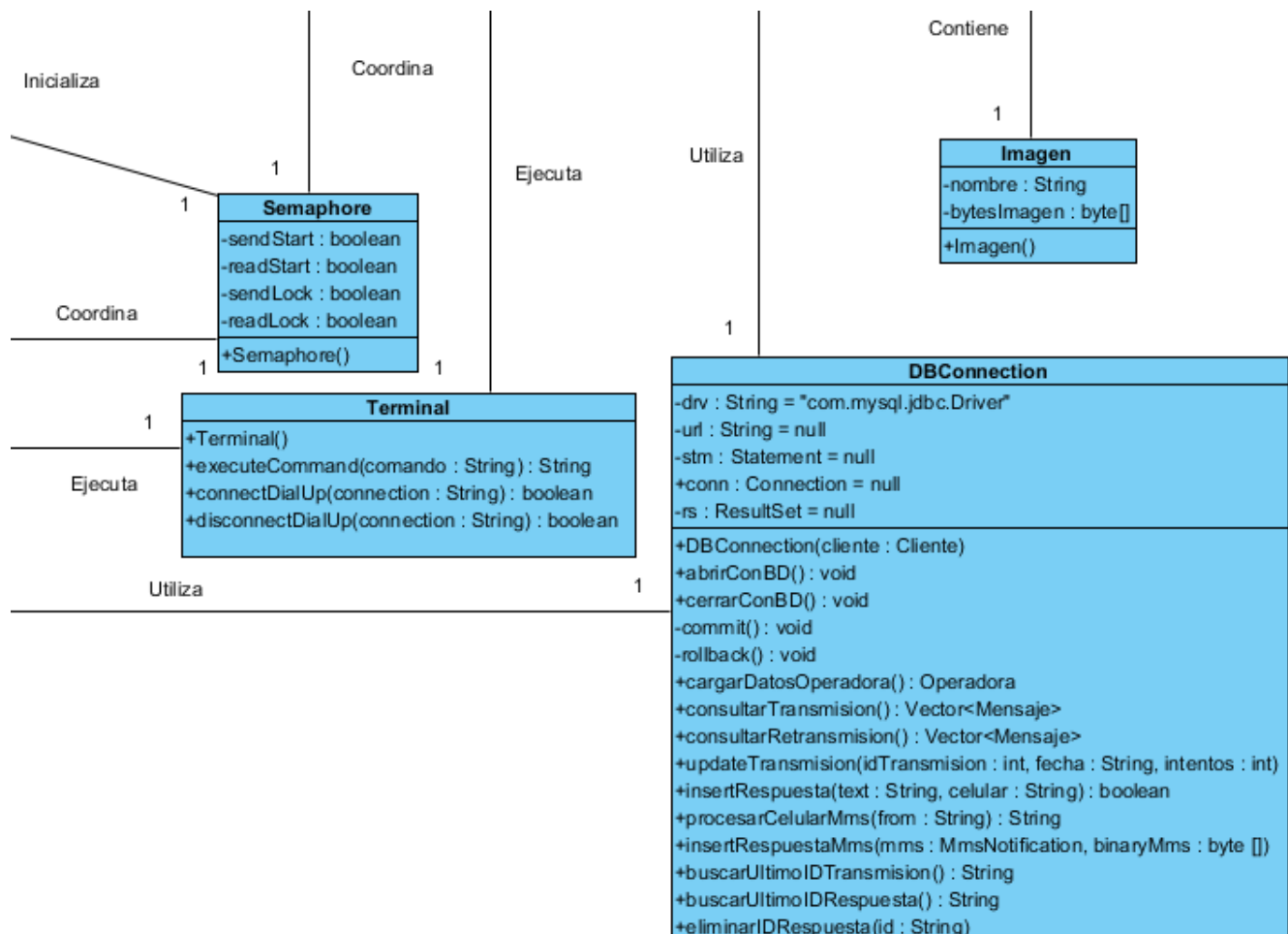


Figura 17: Diagrama de Clases Proceso Envío y Recepción MMS (Inferior Derecho)



- **Imagen:** estructura básica de una imagen de un mensaje multimedia (nombre y contenido en bytes).
- **Semaphore:** implementa las variables y mecanismo para establecer un control de concurrencia entre los hilos de *SendController* y *ReceiveController* para coordinar el acceso al modem GSM.
- **Terminal:** pequeña clase que permita la ejecución de comandos en la terminal de DOS.
- **DBConnection:** clase que contiene todos los métodos que realizan operaciones sobre la base de datos de la aplicación (abrir, cerrar, consultar, insertar, modificar y eliminar).

### 3.2.3.2 Modelo Entidad Relación

Permite visualizar la estructura de la base de datos que va a ser utilizada por la aplicación desarrollada. Este diagrama es elaborado tomando en cuenta las clases, objetos y relaciones definidos previamente así como también procesos como la normalización.

El modelo entidad relación de la base de datos a ser utilizada tanto en el Proceso de Envío y Recepción MMS como en el Proceso de Administración de Contenidos MMS y Registro Transaccional es el siguiente:

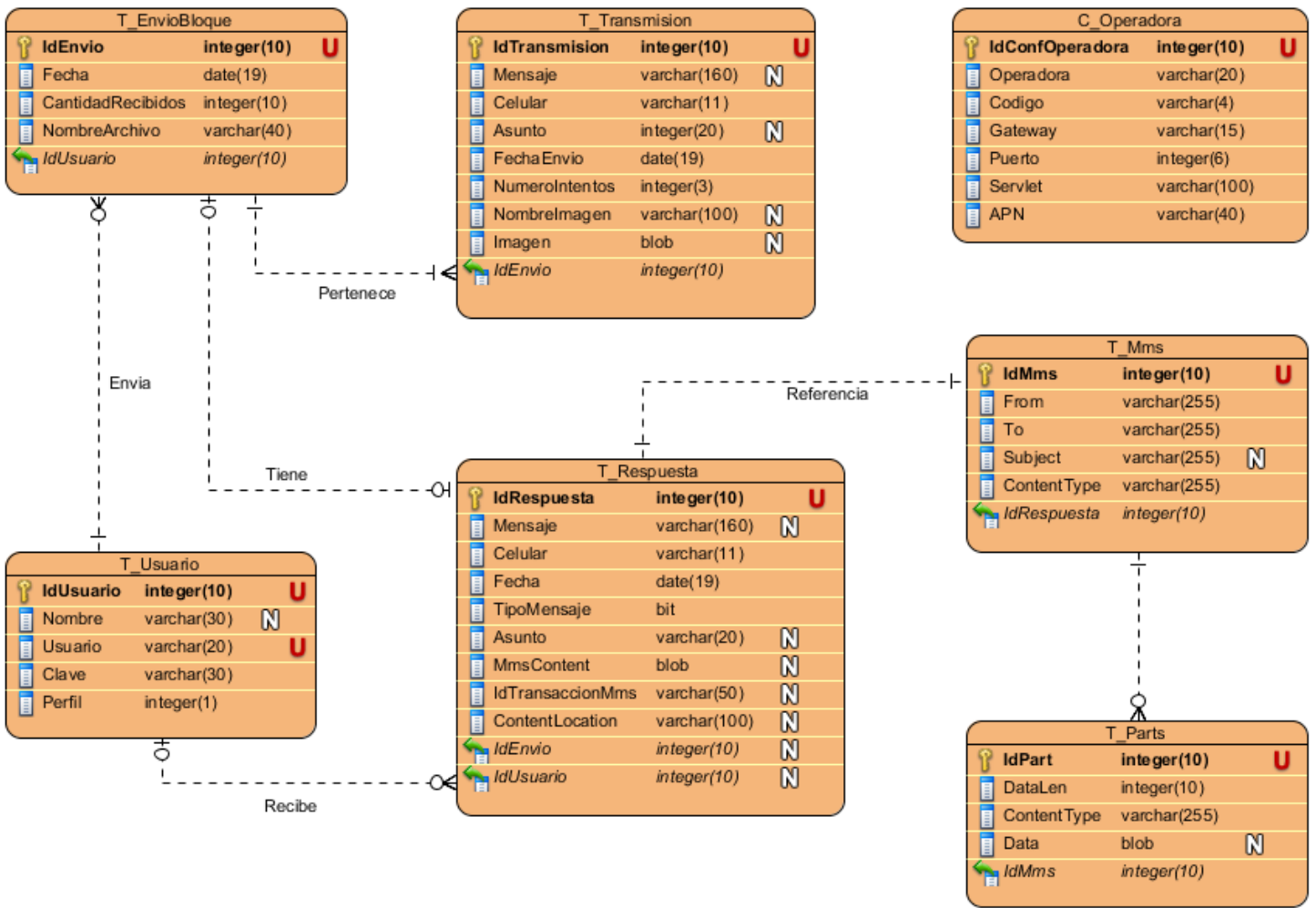


Figura 18: Diagrama Entidad-Relación de la base de datos del sistema

### 3.2.3.3 Entidades de la Base de Datos del Sistema

**T\_EnvioBloque:** Registra los envíos solicitados y posteriormente ejecutados por el Sistema. Cada envío tiene asociado una o más transmisiones.

Columna	Tipo	Longitud	Nulo	Descripción
IdEnvio	integer	10	No	Identificador único de cada envío registrado.
Fecha	datetime	19	No	Fecha y hora de creación del envío (horas en formato de 24). Ej: "2011-06-10 14:23:02".
CantidadRecibidos	integer	10	No	Numero de transmisiones asociadas al envío.
NombreArchivo	varchar	40	No	Nombre identificador del envío. Su formato es el siguiente: "EnvioWebMms <año>-<mes>-<dia> <hora>:<minutos>:<segundos>". Ej: "EnvioWebMms 2011-06-10 14:23:02"
IdUsuario	integer	10	No	Identificador del usuario Web que realizó el envío. Clave foránea de la tabla T_Usuario.

Tabla 8: Descripción de los campos de T\_EnvioBloque

**T\_Transmision:** Contiene las transmisiones asociadas a un envío bloque en particular. Cada una especifica un destinatario particular (Celular) al que se le enviará el mensaje.

Columna	Tipo	Longitud	Nulo	Descripción
IdTransmision	integer	10	No	Identificador único de cada transmisión.
Mensaje	varchar	160	Sí	Texto del mensaje a enviar.
Celular	varchar	11	No	Número telefónico del destinatario. Ej: "04122685440"
Asunto	varchar	20	Sí	Asunto del mensaje.
FechaEnvio	datetime	19	No	Fecha y hora exacta en la cual se realizó el envío del mensaje al destinatario correspondiente (horas en formato de 24). Ej: "2011-06-10 14:23:02"
NumeroIntentos	integer	3	No	Numero de intentos realizados para enviar la transmisión al destinatario. Después de un número preestablecido de intentos, el envío se coloca en suspenso modificando la fecha de envío con el valor "9999-12-31 23:59:59".
NombreImagen	varchar	100	Sí	Nombre de la imagen asociada al mensaje.
Imagen	blob		Sí	Conjunto de bytes de la imagen.
IdEnvio	Integer	10	No	Identificador del envío al que pertenece la transmisión. Clave foránea de la tabla T_EnvioBloque.

**Tabla 9: Descripción de los campos de T\_Transmision**

**T\_Respuesta:** Almacena los mensajes recibidos por el modem GSM y los clasifica en SMS o MMS según su contenido. En caso de recibir MMS, se le asocia un registro en la tabla T\_MMS que contiene el detalle del mismo.

Columna	Tipo	Longitud	Nulo	Descripción
IdRespuesta	integer	10	No	Identificador único de cada mensaje recibido.
Mensaje	varchar	160	Sí	Texto del mensaje recibido. En caso de ser un MMS, este dato se extrae de los bytes de MmsContent.
Celular	varchar	11	No	Número telefónico del destinatario. Ej: "04122685440"
Fecha	varchar	19	No	Asunto del mensaje recibido en caso de ser MMS.
TipoMensaje	integer	1	No	Indica el tipo de mensaje recibido, 0 = SMS y 1 = MMS
Asunto	varchar	20	Sí	Asunto del mensaje recibido en caso de ser MMS, si es SMS el campo se coloca vacío.
MmsContent	blob		Sí	Contenido codificado del MMS (en caso de serlo) en bytes. Este campo se coloca vacío si el mensaje es un SMS. Este contenido es interpretado en el registro de la tabla T_Mms asociado a este mensaje recibido.
IdTransaccionMms	varchar	50	Sí	Conjunto de bytes de la imagen.
ContentLocation	varchar	100	Sí	URL obtenida de la decodificación de la notificación MMS recibida, para la descarga del contenido multimedia del mensaje. Se coloca vacío en caso de ser un SMS.
IdEnvio	integer	10	Sí	Id del envío asociado al mensaje recibido en caso de ser una respuesta. Clave foránea
IdUsuario	integer	10	Sí	Id del usuario que realizó el envío al que se le asocia el mensaje como una respuesta.

**Tabla 10: Descripción de los campos de T\_Respuesta**

**T\_Usuario:** Contiene los datos de los usuarios registrados que interactúan con la aplicación Web (envían, reciben y consultan historiales MMS).

Columna	Tipo	Longitud	Nulo	Descripción
IdUsuario	integer	10	No	Identificador único del usuario.
Nombre	varchar	30	Sí	Nombre completo del usuario.
Usuario	varchar	20	No	Identificador alfanumérico del usuario.
Clave	Varchar	20	No	Contraseña del usuario para acceder a la aplicación Web
Perfil	integer	1	No	Indica la permisología del usuario para visualizar los distintos módulos de la aplicación, 1 = Administrador, 2 = Usuario común.

Tabla 11: Descripción de los campos de T\_Usuario

**T\_Mms:** Registra los Mms recibidos por el modem GSM junto con los campos obtenidos de la decodificación del Header del MmsContent, extraído del registro asociado en la tabla T\_Respuesta.

Columna	Tipo	Longitud	Nulo	Descripción
IdMms	integer	10	No	Identificador único del MMS.
From	varchar	255	Sí	La dirección del cliente MMS que envió el mensaje multimedia y el tipo (celular, correo, etc). Ej: "584122810644/TYPE=PLMN"
To	varchar	255	No	La dirección del cliente MMS al que va dirigido el mensaje multimedia y el tipo (celular, correo, etc). Ej: "584122810644/TYPE=PLMN"
Subject	Varchar	255	No	Asunto del mensaje.
ContentType	integer	255	No	Valor MIME del ContentType que indica el tipo de estructura de los <i>body parts</i> del mensaje multimedia (contenedor de los elementos multimedia). Ej: "application/vnd.wap.multipart.mixed"
IdRespuesta	integer	10		Id del registro al que se le asocia en la tabla T_Respuesta (clave foránea).

Tabla 12: Descripción de los campos de T\_Mms

**T\_Parts:** Para los mensajes de la tabla T\_MMS, se genera un registro en la tabla T\_Parts por cada *body part*, el cual contiene uno o más elementos multimedia. Para fines prácticos, la estructura manejada para los *body parts* en la presente investigación contiene un solo elemento multimedia cada uno.

Columna	Tipo	Longitud	Nulo	Descripción
IdPart	integer	10	No	Identificador único de un <i>body part</i> .
DataLen	varchar	255	Sí	Longitud en bytes del <i>body part</i> ( <i>header + body</i> ).
ContentType	varchar	255	No	Valor MIME del ContentType, el cual indica el tipo de elemento multimedia contenido en el <i>body part</i> .
Data	varchar	255	No	Paquete de datos asociado al <i>body part</i> .
IdMms	integer	255	No	Id del Mms al cual pertenece el <i>body part</i> .

**Tabla 13:** Descripción de los campos de T\_Parts

**C\_Operadora:** Tabla que almacena los parámetros de configuración MMS para cada operadora móvil registrada.

Columna	Tipo	Longitud	Nulo	Descripción
IdConfOperadora	integer	10	No	Identificador único la operadora móvil.
Operadora	varchar	20	Sí	Nombre de la operadora.
Codigo	varchar	4	Sí	Código de la operadora. Ej: "412"
Gateway	varchar	15	Sí	Dirección IP a la que se debe conectar la aplicación motor a través del modem para el envío y recepción MMS.
Puerto	integer	4	Sí	Puerto al que se debe conectar la aplicación motor a través del modem para el envío y recepción MMS.
Servlet	varchar	10	Sí	URL al que se debe conectar la aplicación motor a través del modem para el envío de mensajes multimedia.
APN	varchar	40	Sí	Nombre de punto de acceso para la conexión MMS de la operadora.

Tabla 14: Descripción de los campos de C\_Operadora

### 3.2.3.4 Diagrama de Secuencia

Permite generar una perspectiva cronológica de las interacciones que se presentan en el sistema entre los objetos presentes en un escenario y el conjunto de mensajes intercambiados entre ellos para llevar a cabo la funcionalidad descrita por el escenario. A continuación se presentan los diagramas de secuencia tanto para el envío como para la lectura de un mensaje multimedia.

La secuencia de acciones para enviar mensajes multimedia se inicia desde la clase Ventana, la cual implementa la interfaz de usuario de la aplicación motor.



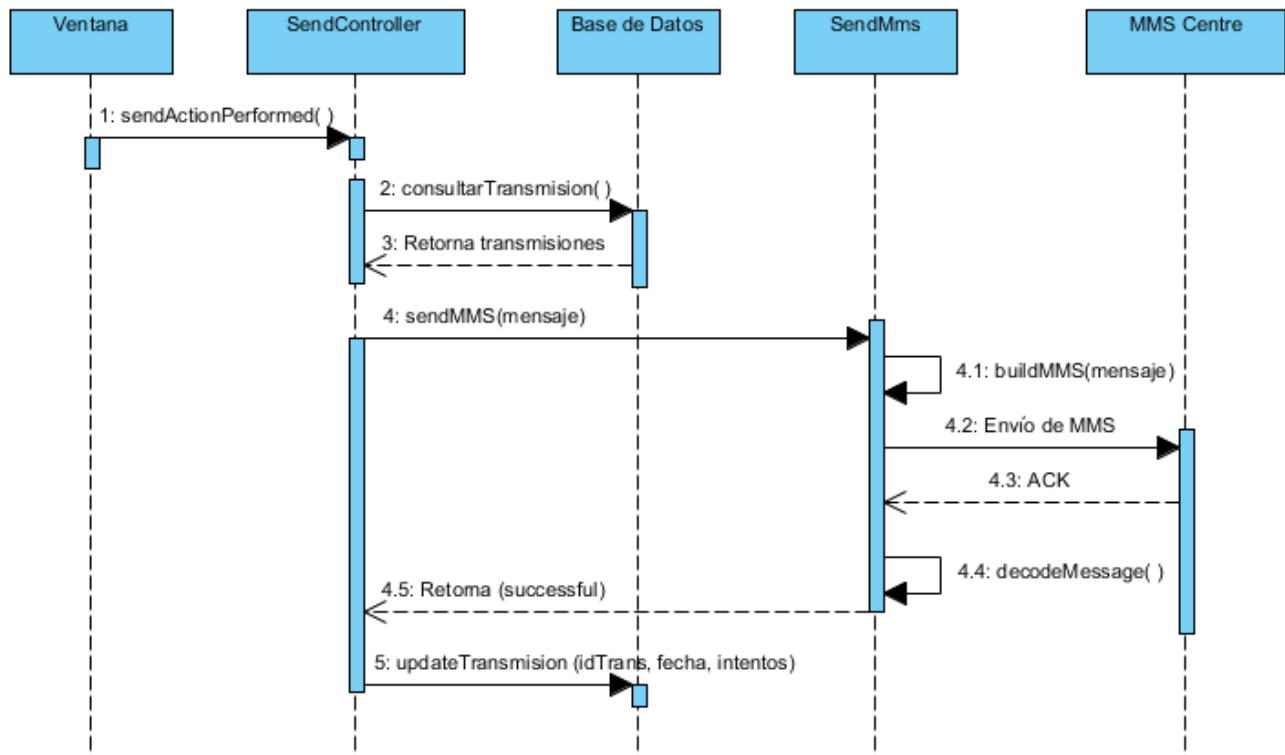


Figura 19: Diagrama de Secuencia del proceso de envío MMS

- El usuario activa el proceso de envío de mensajes haciendo click sobre el botón “Enviar” de la ventana, y por medio del método *sendActionPerformed()* el evento es capturado y se instancia a la clase *SendController*, controlador encargado de gestionar todas las acciones necesarias para llevar a cabo este proceso.
- El controlador de envío consulta en la base de datos a través del método *consultarTransmision()* en busca de transmisiones con estatus “pendiente por enviar”, posteriormente se selecciona uno de estos mensajes y se instancia a la clase *SendMms* enviándole como parámetro la información del mensaje a enviar.
- Una vez dentro de la clase *SendMms*, se construye el PDU a nivel binario del mensaje multimedia con la información recibida, luego se establece conexión con el *MMS Centre* de la operadora móvil y se le envía este paquete de datos.

- La aplicación se queda a la espera de una respuesta por parte de este ente, una vez recibida se decodifica el paquete para obtener el resultado de la operación y este se retorna al controlador para realizar la actualización del estatus del mensaje en la base de datos a “enviado” o “fallido” según sea el caso.

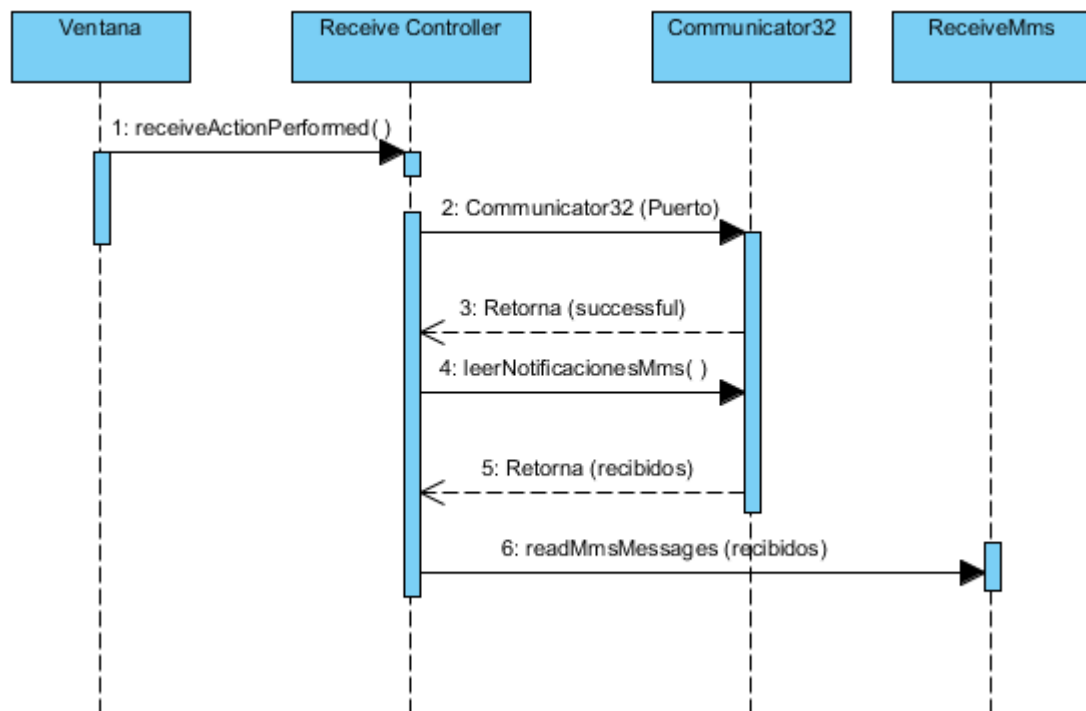


Figura 20: Diagrama de Secuencia del proceso de recepción MMS (parte 1)

La secuencia de acciones para leer mensajes multimedia recibidos se inicia desde la interfaz de usuario de la aplicación motor.

- El usuario activa el proceso de lectura de mensajes haciendo click sobre el botón “Recibir” de la ventana, y por medio del método *receiveActionPerformed()* el evento es capturado y se instancia a la clase *ReceiveController*, encargada de gestionar todas las acciones necesarias para llevar a cabo este proceso.

- Se instancia a la clase Communicator32 invocando su constructor, el cual realiza la apertura del puerto Serial COM (recibido por parámetro) de la computadora para establecer la comunicación con el modem GSM a través de sockets y así poder enviarle comandos AT.
- Una vez abierto el puerto, se invoca al método *leerNotificacionesMms* el cual consulta enviando el comando AT+CMGL a través del socket, los mensajes recibidos por el modem y así verificar cuales son notificaciones MMS.

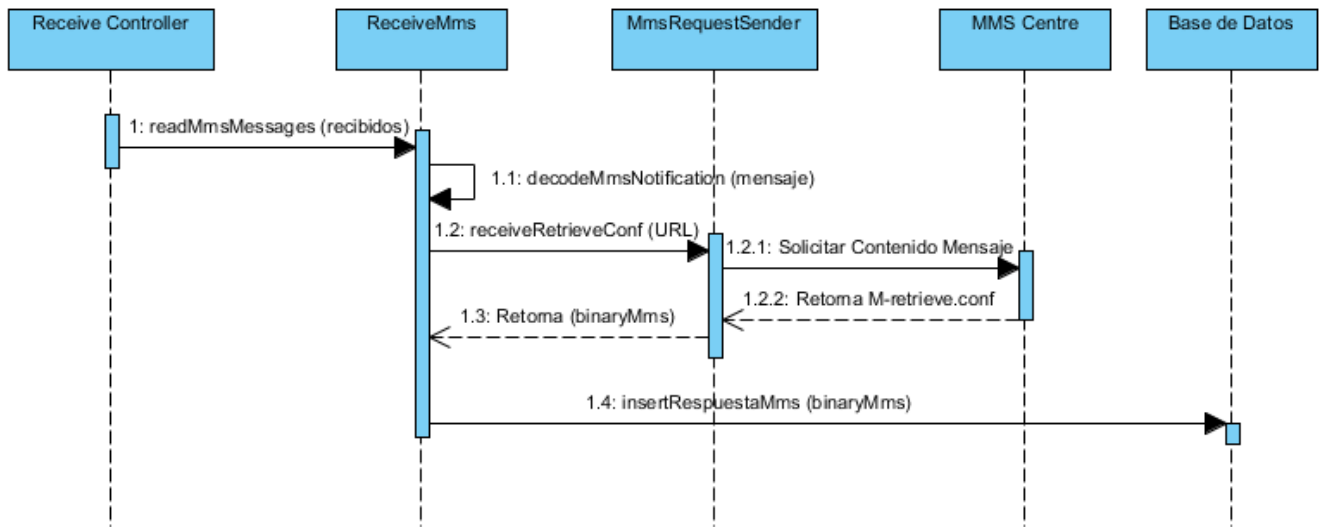


Figura 21: Diagrama de Secuencia del proceso de lectura MMS (parte 2)

- Una vez identificados estos mensajes, se invoca al método *readMmsMessages* de la clase *ReceiveMms*, la cual se va a encargar de decodificar las notificaciones MMS y solicitar la descarga del contenido multimedia. El método *decodeMmsNotification* realiza la interpretación a nivel de bytes de la notificación para obtener información primordial del mensaje, incluyendo el URL del cual se descargará los elementos multimedia.
- Por medio de la clase *MmsRequestSender* se invoca al método *receiveRetrieveConf* para conectarse al MMS Center en la red móvil (por medio del modem GSM) y acceder URL obtenida y recibir el PDU esperado.

- El contenido binario obtenido es almacenado en la base de datos, junto con la información básica de identificación del MMS por medio del método *insertRespuestaMms*.

El procedimiento para enviar un mensaje de prueba es muy similar al del envío de MMS pendientes, con la diferencia de que no accede a la base de datos, ya que no es necesario consultar transmisiones pendientes por enviar ni actualizar el estatus final del envío.

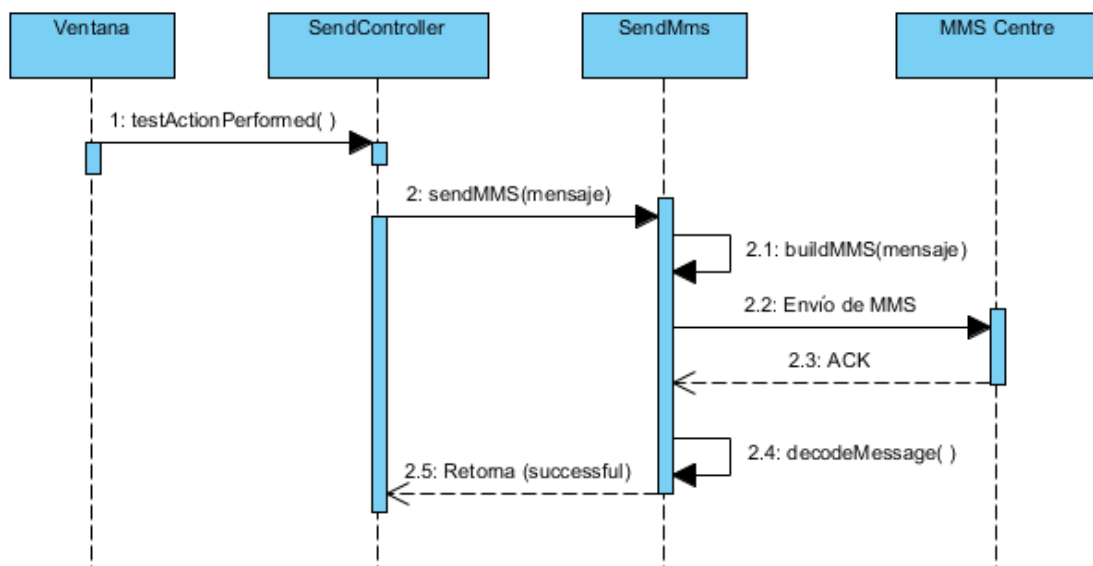


Figura 22: Diagrama de secuencia de proceso envío MMS de prueba

## 3.2.4 Implementación

### 3.2.4.1 Descripción de la Ventana gráfica del motor

La siguiente figura muestra un *screenshot* de la ventana gráfica del motor con una identificación de cada componente.

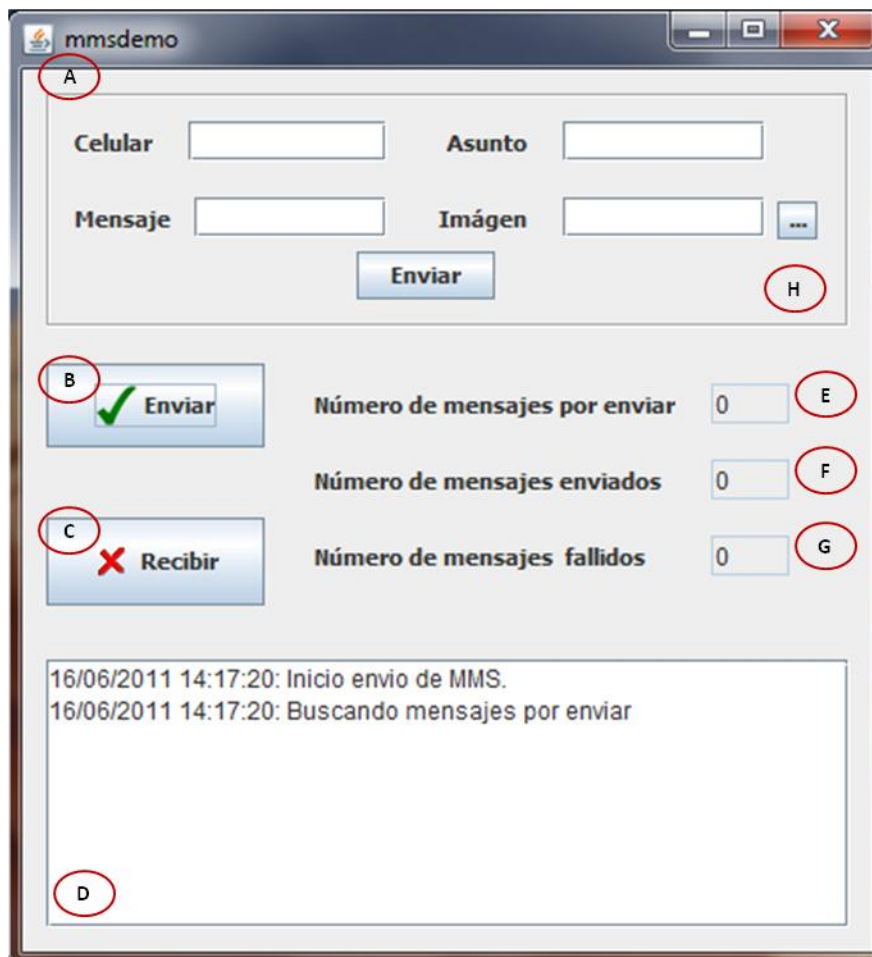


Figura 23: Ventana de la Aplicación Motor MMS

- A. Título de la ventana del motor, el cual corresponde al valor del parámetro *CarpetaCliente* del archivo de configuración *sendmms.properties*.

- B. Botón para iniciar el proceso de envío de mensajes pendientes.
- C. Botón para iniciar la lectura de mensajes recibidos por el modem, almacenados en la tarjeta SIM.
- D. *Log* donde se muestran las acciones realizadas por el motor en los procesos de envío y lectura.
- E. Indicador de mensajes por enviar desde que se levantó la ventana de la aplicación.
- F. Indicador de mensajes enviados desde que se levantó la ventana de la aplicación.
- G. Indicador del número de mensajes fallidos en intento de envío, desde que se levantó la aplicación.
- H. Panel de componentes para el envío de un mensaje multimedia de prueba.

La aplicación motor implementa dos hilos de ejecución, uno para el envío de mensajes (instancia de la clase *SendController* que hereda de la clase *Thread* de Java) y un hilo de ejecución para la lectura y recuperación de los mensajes multimedia (instancia de la clase *ReceiveController* que hereda de la clase *Thread* de Java). La ejecución de estos hilos se inicia cuando se da click al botón de la ventana correspondiente (Enviar y/o Recibir). La implementación de estos hilos viene dada por la necesidad de gestionar dos procesos independientes que acceden a un recurso crítico y compartido, el modem GSM, el cual solo puede establecer una sesión WAP en la conexión GPRS a la vez (envío o recuperación de contenido MMS), y adicionalmente el hilo de lectura de mensajes envía órdenes al modem a través de comandos AT y estos no se pueden enviar si actualmente si el dispositivo se encuentra conectado en la red GPRS para el intercambio MMS.

Para mantener un control de concurrencia se implementó la clase *Semaphore* (Semáforo en español). Un semáforo constituye el método clásico para restringir o permitir el acceso a recursos compartidos (por ejemplo, un recurso de almacenamiento del sistema o variables del código fuente) en un entorno de multiprocesamiento (en el que se ejecutarán varios procesos concurrentemente).

Si por ejemplo al hilo de *SendController* se le otorga el acceso al recurso crítico (modem), el conjunto de acciones ejecutarse en el sistema serían las siguientes:

- El hilo *ReceiveController* se coloca en estatus de espera (*wait*) y se queda verificando cíclicamente su estatus hasta que sea cambiado nuevamente.

- El hilo *SendController* verifica en la base de datos si hay mensajes pendientes por enviar.
- En caso de no haber mensajes por enviar, el hilo se coloca en estatus de espera (*wait*) y se reanuda la ejecución del hilo *ReceiveController* para otorgarle el acceso al recurso crítico.
- En caso de haber mensajes pendientes por enviar, se enviará como máximo el número de mensajes que indique el parámetro *bloqueEnvio* del archivo de configuración, luego el hilo cede el recurso crítico al *ReceiveController* y se coloca en estatus de espera (*wait*).
- El hilo *ReceiveController* al reanudarse verifica si existen nuevos mensajes recibidos por el modem, de no ser así, se sede nuevamente el control del recurso crítico al *SendController* y se coloca en estado de espera (*wait*).
- Si hay mensajes nuevos en el modem, se decodifican y se recupera su contenido multimedia, pero únicamente se procesarán como máximo el número de mensajes que indique el parámetro *bloqueRecepcion* del archivo de configuración, luego se coloca en estatus *wait* y se cede el control nuevamente al hilo *SendController*.

#### **3.2.4.1 Justificación del uso de Java y MySQL**

La elección de Java como lenguaje de programación y plataforma para el desarrollo de este módulo es apoyada por la variedad de librerías de propósito específico que provee, particularmente en este contexto para la gestión de la comunicación con el módem y la conexión GPRS por medio del protocolo WAP.

Las bases de datos de la plataforma SMS de ALS Telecom están implementadas en MySQL, además de ser el manejador que provee actualmente su servidor.

MySQL, en la actualidad, es un sistema manejador de bases de datos muy popular, con documentación extensa y de fácil integración con PHP y otros lenguajes. También es compatible con múltiples plataformas y sistemas.

MySQL es software libre, por consecuencia, no es necesario el pago de licencias para su utilización.

### 3.2.4.2 Archivo de configuración de la aplicación motor

La siguiente figura especifica la estructura del archivo *sendmms.properties* el cual contiene los parámetros de configuración necesarios para el funcionamiento de la aplicación.

```
1 aplicacionCliente=rmsdemo
2
3 carpetaCliente=rmsdemo
4
5 clienteBD=rmsdemo
6
7 usuarioBD=root
8
9 claveBD=
10
11 ipBD=127.0.0.1
12
13 puertoBD=3306
14
15 maxIntentos=2
16
17 tsend=32000
18
19 twait=3000
20
21 bloqueEnvio=2
22
23 bloqueRecepcion=2
24
25 conexionDialUp=Enfora GPRS
26
27 operadora=Digitel
28
29 #sender=
30
31 #subject=
32
33 #imageMaxSize=
34
35 #textMaxSize=
36
37 #mmsMaxSize=
38
39 #skip=false
```

Figura 24: Archivo de configuración de la aplicación motor (*sendmms.properties*)



Los parámetros del archivo se describen a continuación:

- **AplicacionCliente:** nombre identificador de la aplicación motor que aparecerá en el título de la ventana.
- **CarpetaCliente:** ruta y donde de la carpeta donde se encuentra la aplicación Web (no el motor) en el servidor.
- **ClienteBD:** nombre de la base de datos de la aplicación.
- **UsuarioBD:** usuario de la base de datos de la aplicación.
- **ClaveBD:** contraseña de la base de datos de la aplicación.
- **IpBD:** dirección IP o host donde se encuentra alojada la base de datos.
- **PuertoBD:** puerto para acceder a la base de datos.
- **MaxIntentos:** número máximo de intentos permitidos para enviar un MMS antes de colocarlo como fallido.
- **Tsend:** tiempo máximo en milisegundos para enviar un MMS, si se excede de este tiempo, se vuelve a repetir el envío del mensaje, siempre y cuando no se haya excedido el número máximo de intentos.
- **Twait:** tiempo de espera en milisegundos después de realizar la consulta de mensajes pendientes por enviar, y así no saturar el acceso a la base de datos.
- **BloqueEnvio:** número de mensajes a enviar consecutivamente por el hilo *SendController*, para luego ceder el control del recurso compartido (modem GSM) al hilo de lectura de mensajes *ReceiveController*.
- **BloqueRecepcion:** número de mensajes a leer consecutivamente por el hilo *ReceiveController* antes de ceder el control del recurso compartido (modem GSM) al hilo *SendController*.
- **Operadora:** nombre identificador de la operadora móvil a la cual se conectará la aplicación. Los parámetros de conexión asociados se obtienen del registro en la tabla *C\_Operadora* con dicho nombre.

- **Sender:** especifica una dirección (número celular, correo o nombre) por defecto del emisor de los mensajes multimedia (parámetro opcional).
- **Subject:** especifica un asunto por defecto para el mensaje multimedia en caso de no ser especificado por el cliente emisor (parámetro opcional).
- **imageMaxSize:** límite máximo para el tamaño de una imagen a adjuntar en un mensaje, especificado en bytes (parámetro opcional).
- **textMaxSize:** límite máximo de bytes para el texto del mensaje (parámetro opcional).
- **MmsMaxSize:** tamaño máximo permitido para un mensaje multimedia en bytes (parámetro opcional).
- **Skip:** si este parámetro tiene como valor “true”, se omiten cualquier error o defecto estructural del mensaje y se envía de todos modos (parámetro opcional).

### 3.2.4.3 Envío de mensajes

La siguiente figura captura el segmento de código que realiza el envío de un mensaje multimedia. Antes de ejecutar estas líneas se establece la conexión GPRS para la comunicación MMS por medio de una sesión WAP.

```

237     try {
238
239         /***** SEND MMS *****/
240
241         PostRequest request = new PostRequest(getCliente().getOperadora().getServlet());
242         request.setContentType("application/vnd.wap.mms-message");
243         request.setRequestBody(getEncodedMms());
244
245         System.out.println("Sending Message...");
246         Response response = getWapClient().execute(request, cliente.getTsend());
247
248         byte[] binaryMms = response.getResponseBody();
249         MmsDecoder dec = new MmsDecoder(binaryMms);
250         MmsMessage mms = dec.decodeMessage();
251
252         if (!mms.getResponseStatus().equals(MmsMessage.MMS_RESPONSE_STATUS_OK)) {
253             throw new MmsMessageException("Message not sent: error="+mms.getResponseStatus());
254         }
255
256         /***** END SEND MMS *****/
257
258     } catch (IllegalStateException e) {
259         e.printStackTrace();
260         return false;
261     } catch (SocketException e) {
262         e.printStackTrace();
263         return false;
264     } catch (IOException e) {
265         e.printStackTrace();
266         return false;
267     } catch (MmsDecoderException e) {
268         System.out.println(e.getMessage());
269     } catch (MmsMessageException e) {
270         System.out.println(e.getMessage());
271         return false;
272     }

```

Figura 25: Código de envío de un mensaje multimedia (SendMms.java)

En la línea 241 se crea una petición de tipo *POST Request* para acceder al *Servlet* (URL) a través del cual se realizará el envío del PDU que contiene el MMS. En las dos siguientes líneas se editan los parámetros de la solicitud (el tipo de contenido a enviar y los bytes del PDU). Luego se solicita el la respuesta a la transacción y el paquete de datos recibido que contiene el resultado de la operación (enviado por el MMSC) es decodificado y en caso de haber fallado transacción se genera una excepción.

### 3.2.4.4 Comunicación con el modem GSM

En la siguiente figura se muestra parte del código del constructor de la clase *Communicator32*, donde se establece la comunicación con el modem GSM utilizando la librería “*comm.jar*”, la cual implementa las rutinas necesarias para conectarse a los puertos seriales del computador.

En el código se verifican todos los puertos del computador y se busca el puerto serial del nombre que corresponde con la variable *com* de tipo *String*, y con la función *open* se abre el puerto, dando un tiempo máximo de 2 segundos para esperar respuesta, si no se logra establecer comunicación se dispara una excepción para indicar que el puerto no está configurado o se encuentra ocupado, en caso de éxito se configuran los parámetros necesarios tales como velocidad en kilobits, mecanismo de control de errores, etc, luego se procede a crear un flujo de entrada y salida de datos para gestionar el socket para así enviar comandos AT y recibir la respuesta por parte del modem del resultado de la ejecución de los mismos. Posteriormente se consultan los mensajes recibidos por el modem enviando el comando “*AT+CMGL=ALL*” a través del socket.

```
61 portList = CommPortIdentifier.getPortIdentifiers();
62
63 while(portList.hasMoreElements()) {
64
65     portId = (CommPortIdentifier) portList.nextElement();
66     if(portId.getPortType() == CommPortIdentifier.PORT_SERIAL) {
67         try {
68
69             if(portId.getName().equalsIgnoreCase(this.com)) {
70
71                 port = (SerialPort) portId.open("Communicator",2000);
72                 port.setSerialPortParams(115200, SerialPort.DATABITS_8,SerialPort.STOPBITS_1,SerialPort.PARITY_NONE);
73                 System.out.println("Communication established");
74                 is = new BufferedReader(new InputStreamReader(port.getInputStream()));
75                 os = new PrintStream(port.getOutputStream(),true);
76
77             }
78         } catch (PortInUseException e) {
79             System.out.println("PortInUseException");
80             e.printStackTrace();
81             successful = false;
82         } catch (UnsupportedCommOperationException e) {
83             System.out.println("UnsupportedCommOperationException");
84             e.printStackTrace();
85             this.successful = false;
86         } catch (IOException e) {
87             System.out.println("IOException");
88             e.printStackTrace();
89             this.successful = false;
90         }
91     }
92 }
```

Figura 26: Código de establecimiento de la comunicación con el modem GSM (Communicator32.java)

### 3.2.4.5 Decodificación de Notificaciones MMS

Una notificación MMS es una indicación enviada por el MMSC y recibida por el teléfono móvil o modem GSM que indica que se ha recibido un nuevo mensaje multimedia. Dependiendo de la configuración del teléfono, el contenido del mensaje se descarga automáticamente del MMSC, siempre y cuando se encuentre registrado en una red GPRS o UMTS.

La notificación es un PDU encapsulado en uno o más SMS bajo las especificaciones WAP Push y MMS, estructurados con una secuencia de parámetros especificados por el protocolo de transferencia (TP).

En la siguiente figura se especifica la estructura general de una notificación MMS y los parámetros que contiene.

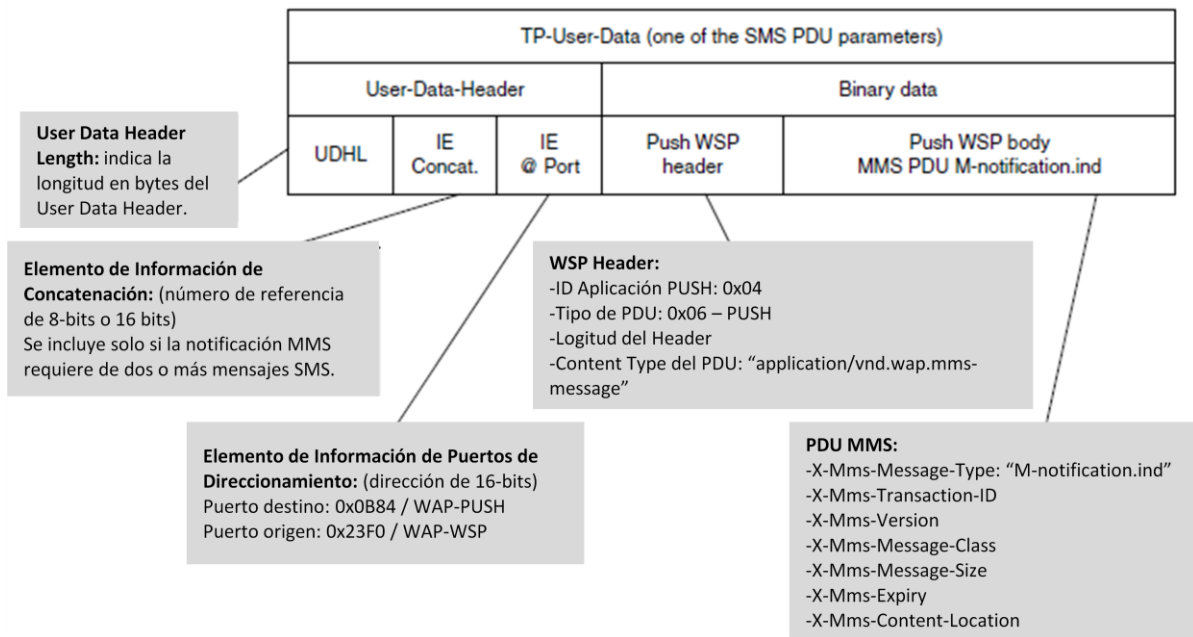


Figura 27: Estructura de una notificación MMS [7]

Los mensajes recibidos por el modem GSM (excepto los SMS corrientes) poseen esta estructura. Una vez leídos y almacenado su contenido, la aplicación motor procede a interpretarlos según la estructura WAP Push definida, para unir todas las notificaciones relacionadas a un MMS en específico (por lo general compuesto por dos SMS) y así extraer el contenido relevante, es específico la URL correspondiente al campo *X-Mms-Content-Location*, servlet al cual se conectará el modem para recuperar el contenido

multimedia del mensaje. La clase que implementa los métodos para la decodificación secuencial y ordenada de las notificaciones es *MmsNotificationDecoder*.

### 3.2.4.6 Recuperación de contenido multimedia

El método *receiveRetrieveConf* de la clase *MmsRequestSender* tiene como finalidad la recuperación del contenido multimedia de los mensajes. Se establece una sesión como cliente WAP (una vez que el modem esté conectado a la red GPRS) con los parámetros extraídos de la tabla *C\_Operadora* de la base de datos, correspondientes a la operadora especificada en el archivo de configuración. Se realiza una petición de tipo *GET Request* al servlet obtenido del *content location* de la notificación MMS, y se espera por el response por parte del MMSC, el cual contendrá el PDU del contenido del mensaje. Se coloca un tiempo límite de 20000 milisegundos para recibir respuesta por parte del MMSC, transcurrido el tiempo se reinicia la conexión WAP y se intenta nuevamente la recuperación del contenido.

```
23 public byte[] receiveRetrieveConf(String servlet) throws UnknownHostException, SocketException, IllegalStateException{
24
25     byte[] binaryMms = null;
26     WAPClient client = new WAPClient(cliente.getOperadora().getwapGateway(), cliente.getOperadora().getWapPort());
27     client.connect();
28     System.out.println("Connection Estabilished");
29     try {
30
31         Request req = new GetRequest(servlet);
32         Response resp = client.execute(req, 20000);
33         System.out.println("Successful Transmission");
34         binaryMms = resp.getResponseBody();
35
36     } catch (IOException e) {
37         e.printStackTrace();
38         binaryMms = null;
39     } catch (Exception e) {
40         e.printStackTrace();
41         binaryMms = null;
42     } finally {
43         // Disconnect from the WAP gateway
44         client.disconnect();
45         return binaryMms;
46     }
47 }
```

Figura 28: Recuperación de contenido multimedia del mensaje recibido

### 3.2.5 Pruebas y Resultados

En esta fase la herramienta o aplicación desarrollada es sometida a una serie de pruebas, con la finalidad comprobar su correcto funcionamiento, detectando posibles comportamientos inesperados o errores que puedan presentarse. A continuación se presentan las pruebas realizadas para la aplicación motor de envío y recepción MMS.

Para este proceso se realizaron tres tipos de pruebas:

- Envío de mensajes (de la aplicación hacia el teléfono móvil).
- Recepción de mensajes (de teléfono móvil hacia la aplicación).
- Envío y Recepción de mensajes (de la aplicación hacia la aplicación).

La configuración de los parámetros del archivo *sendmms.properties* a utilizar para estas pruebas es la indicada en la figura 26.

Los parámetros variables a considerar en los mensajes multimedia para realizar las pruebas son:

- Formato de las imágenes (JPG, PNG y GIF ó BMP).
- Tamaño de las imágenes en KB (Kilobytes).

#### 3.2.5.1 Envío de mensajes (Aplicación – Teléfono Móvil)

Para verificar esta funcionalidad se insertaron manualmente en la base de datos registros de envíos a realizar (registros en las tablas T\_EnvioBloque y T\_Transmision). El resultado de la construcción y envío del mensaje multimedia se verifica en el teléfono celular destinatario, chequeando que el contenido multimedia del mensaje recibido corresponda al contenido enviado por la aplicación.

A continuación se presenta cuatro tablas con las de pruebas realizadas para este tipo y los resultados obtenidos, considerando un entorno con cobertura GSM estable.

Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Número de envíos realizados	Número de envíos exitosos	Número de envíos fallidos
10 KB	JPEG		5	5	0
10 KB	PNG		5	5	0
10 KB	GIF		5	5	0

Tabla 15: Pruebas de envío aplicación – teléfono móvil (tamaño imagen 10KB)

Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Número de envíos realizados	Número de envíos exitosos	Número de envíos fallidos
17 KB	JPEG		5	3	2
17 KB	PNG		5	4	1
17 KB	GIF		5	4	1

Tabla 16: Pruebas de envío aplicación – teléfono móvil (tamaño imagen 17KB)



Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Número de envíos realizados	Número de envíos exitosos	Número de envíos fallidos
23 KB	JPEG		5	2	3
23 KB	PNG		5	2	3
23 KB	GIF		5	1	4

Tabla 17: Pruebas de envío aplicación – teléfono móvil (tamaño imagen 23KB)

Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Número de envíos realizados	Número de envíos exitosos	Número de envíos fallidos
30 KB	JPEG		5	0	5
30 KB	PNG		5	0	5
30 KB	GIF		5	0	5

Tabla 18: Pruebas de envío aplicación – teléfono móvil (tamaño imagen 30KB)

Según los resultados reflejados en las tablas 15, 16, 17 y 18, en base a las distintas pruebas realizadas, se puede observar que a medida que se adjuntan imágenes de mayor peso (en Kilobytes) en el mensaje multimedia, mayor es la cantidad de fallos de envío ocurridos. Los mensajes con imágenes de tamaño 30 KB en adelante (aproximadamente) son casi imposibles de enviar, esto se debe al escaso de ancho de banda en la red GPRS y la limitación de datos a enviar en cada sesión WAP, establecida por la operadora móvil Digitel (25 KB); a medida que se acerca a este límite, se pierde garantía la estabilidad de la conexión. El formato de la imagen no influye en el éxito o fallo del envío del mensaje.

La verificación de éxito de cada envío incluyó el chequeo de recepción del mensaje en cada teléfono celular y de que el contenido multimedia enviado correspondiera al contenido recibido en el dispositivo móvil.

### **3.2.5.2 Recepción de mensajes (Teléfono Móvil - Aplicación)**

Para la verificación de la recepción se realizaron envíos desde un teléfono móvil hacia el número celular de la línea del modem GSM. Los mensajes recibidos son almacenados en la tabla T\_Respuesta y para la verificación del contenido del mensaje, se comparó con un editor hexadecimal para cada elemento multimedia (imagen, texto del mensaje, etc.), los bytes enviados por el teléfono con respecto los bytes recibidos por el modem.

En las siguientes dos tablas se refleja las pruebas realizadas de recepción de mensajes multimedia desde la aplicación motor y los resultados obtenidos.

Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Número de envíos realizados	Número de mensajes recibidos	Número de mensajes erróneos
10 KB	JPEG		5	5	0
10 KB	PNG		5	5	0
10 KB	GIF		5	4	1

Tabla 19: Pruebas de recepción teléfono móvil – aplicación (tamaño imagen 10KB)

Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Número de envíos realizados	Número de mensajes recibidos	Número de mensajes erróneos
30 KB	JPEG		5	4	1
30 KB	PNG		5	5	0
30 KB	GIF		5	5	0

Tabla 20: Pruebas de recepción teléfono móvil – aplicación (tamaño imagen 30KB)

A diferencia de los resultados de las pruebas de envío desde la aplicación, todos los envíos realizados desde los teléfonos móviles pudieron realizarse de manera exitosa, esto se debe a que estos dispositivos se encuentran conectados de manera ininterrumpida a la red GPRS, por lo que el límite de datos a transferir a través de la conexión es mayor, además de que el cliente WAP del celular realiza la cantidad de intentos necesarios para finalizar el envío exitosamente.

Cabe destacar que para la recepción de los mensajes es suficiente tener la cobertura mínima GSM, ya que la notificación de recepción de un mensaje multimedia es un SMS, luego para descargar su contenido sí es necesaria la cobertura con la red GPRS.

La recuperación del contenido del mensaje se realizó de manera exitosa (el contenido recibido en la aplicación por medio del modem correspondió con el contenido enviado desde el teléfono móvil) en casi todos los casos sin distinción de formato ni del tamaño de la imagen. Los dos casos en los que no se pudo recuperar la información del mensaje fueron por problemas con el MMSC, el cual en vez de enviar el contenido del MMS, remitió el siguiente paquete de datos:

```
<?xml version='1.0'?>
<!DOCTYPE html PUBLIC '-//WAPFORUM//DTD XHTML Mobile 1.0//EN'
'http://www.wapforum.org/DTD/xhtml-mobile10.dtd'>
<html xmlns='http://www.w3.org/1999/xhtml'>
  <head>
    <title>The request failed</title>
  </head>
  <body>
    <p><big>Service Unavailable.</big></p>
    <p>
      <i>Technical description:</i><br/>503 Service Unavailable – Service
        temporary down, please try again later.</p>
  </body>
</html>
```

**Figura 29: Mensaje de error en solicitud de recuperación del mensaje**

En estos casos bastaba con solicitar una o dos veces más la recuperación del contenido del mensaje para recibirlo de manera exitosa.

### 3.2.5.3 Envío y Recepción de mensajes (Aplicación – Aplicación)

Para realizar estas pruebas se generaron envíos de la misma forma que en las primeras pruebas (insertando manualmente en la base de datos) colocando como destinatario el número celular del modem y posteriormente se verificó el contenido recibido al igual que en las segundas pruebas (comparando byte por byte la data enviada con respecto a la data recibida).

Las pruebas de este tipo se limitaron a un solo formato de imagen, debido a que no es influyente en el resultado final de las operaciones de envío y recepción.



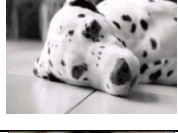

Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Envíos realizados	Mensajes recibidos exitosamente	Envíos fallidos	Mensajes Recibidos erróneos
10 KB	JPEG		5	5	0	0
17 KB	JPEG		5	3	1	1
23 KB	JPEG		5	2	3	0
30 KB	JPEG		5	0	5	0

Tabla 21: Pruebas de envío y recepción aplicación – aplicación (tamaño imagen 10KB)

En la tabla 19 se observa un comportamiento muy similar en los fallos de los envíos con respecto a las primeras pruebas, a la medida que aumenta el tamaño de la imagen. Casi todos los mensajes enviados se

recibieron de manera exitosa, incluyendo el contenido multimedia de cada uno, a excepción de uno solo, caso en el cual se recibió el mensaje de error indicado en la figura 31. Este error se pudo apreciar que ocurría mientras más solicitudes consecutivas en poco tiempo se le enviaban al MMSC para la recuperación de los mensajes, sin distinción alguna en el formato o tamaño de la imagen.

### **3.3 Administración de Contenidos MMS y Registro Transaccional**

Este proceso consta de las actividades necesarias para el desarrollo de la aplicación Web que provea al usuario final de las funcionalidades de envío de mensajes multimedia a uno más destinatarios, consulta de historiales de mensajes enviados, mensajes recibidos y monitoreo de mensajes pendientes por enviar.

#### **3.3.1 Requerimientos**

Los requerimientos identificados para la aplicación Web son los siguientes:

- Envío de un mensaje multimedia a varios destinatarios con el siguiente contenido: texto, asunto y una imagen (JPEG, PNG, GIF o BMP).
- Historial de mensajes enviados y pendientes por enviar filtrado por mes.
- Historial de mensajes recibidos filtrado por mes.
- Autenticación y registro de usuarios.
- Configuración de parámetros de conexión de la operadora móvil.
- Acceso a la base de datos común de la plataforma para consultar información transaccional o insertar nuevos registros para el envío MMS o configuración de parámetros del sistema.

### 3.3.2 Análisis

En esta etapa se define una especificación conceptual de la aplicación Web y se elaboran los modelos de casos de uso y objetos del dominio.

#### 3.3.2.1 Diagrama de Casos de Uso

La Figura 31 presenta el modelo de casos de uso elaborado para modelar las funcionalidades y comportamiento de la aplicación Web.

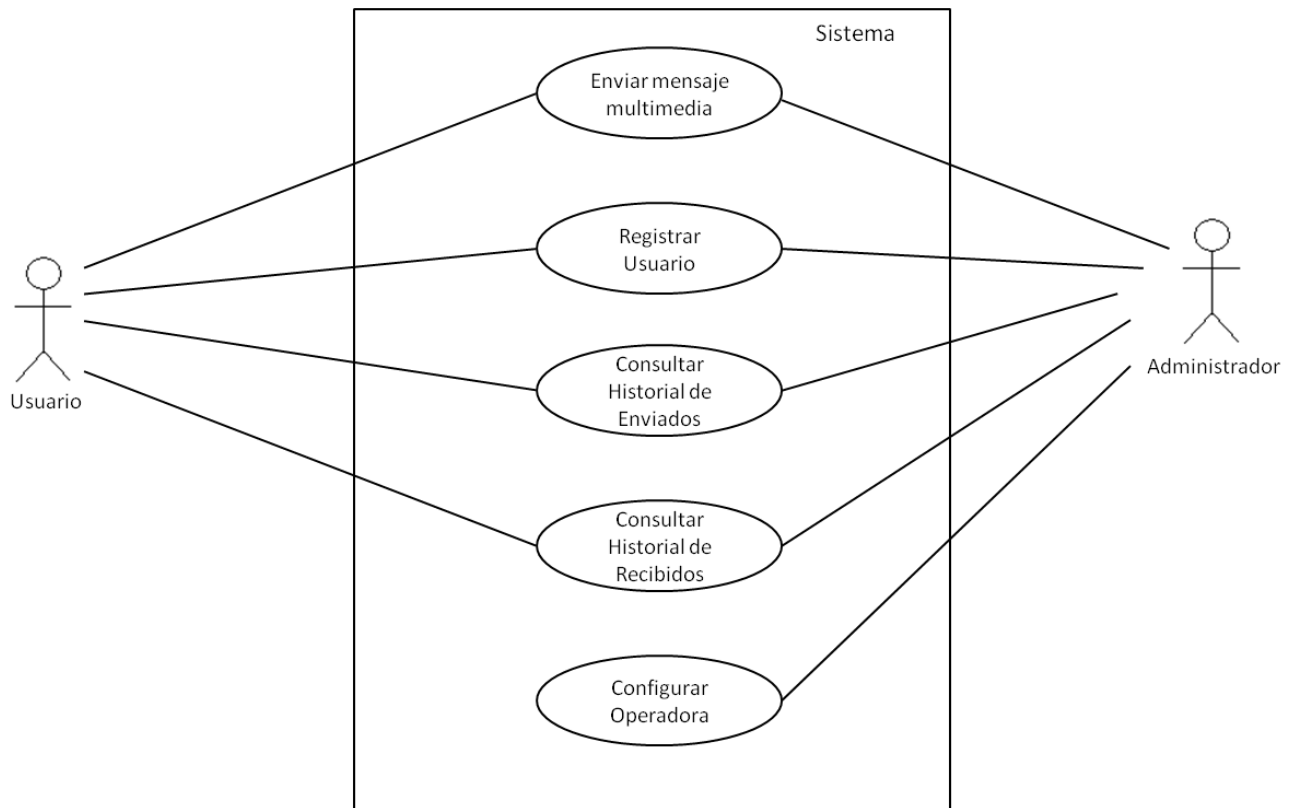


Figura 30: Modelo de casos de uso de la aplicación Web

## Descripción de los casos de uso

A continuación se describen los casos de uso presentados en el modelo:

<b>Nombre:</b>	Enviar mensaje multimedia
<b>Nivel:</b>	0
<b>Resumen:</b>	El usuario realiza un envío de un mensaje multimedia a uno o más destinatarios.
<b>Actor:</b>	Usuario.
<b>Precondición:</b>	Este caso de uso no posee precondición.
<b>Descripción:</b>	<ul style="list-style-type: none"> <li>✓ El usuario accede al módulo "Envío Web" del menú.</li> <li>✓ Se ingresan los destinatarios a los cuales va dirigido el mensaje.</li> <li>✓ Se coloca el asunto del mensaje.</li> <li>✓ Se ingresa el texto del mensaje.</li> <li>✓ Se selecciona la imagen a adjuntar (opcional).</li> <li>✓ Se hace click en el botón "Enviar".</li> </ul>
<b>Postcondición:</b>	Se genera el envío en la base de datos y queda pendiente por enviar hasta que el motor lo procese.

**Tabla 22: Descripción de caso de uso Enviar mensaje multimedia**

<b>Nombre:</b>	Registrar Usuario
<b>Nivel:</b>	0
<b>Resumen:</b>	Se ingresan los datos del nuevo usuario y se registra en la base de datos.
<b>Actor:</b>	Usuario.
<b>Precondición:</b>	Este caso de uso no posee precondición.
<b>Descripción:</b>	<ul style="list-style-type: none"> <li>✓ El usuario accede al módulo de registro de usuario.</li> <li>✓ Se ingresan los datos del usuario a registrar: nombre del usuario, login, contraseña y confirmación de contraseña.</li> <li>✓ Se le da click al botón "Guardar".</li> <li>✓ En caso de no ser iguales las contraseñas o de dejar un campo vacío, se imprime un mensaje de alerta de error.</li> </ul>
<b>Postcondición:</b>	Se registra el usuario en la base de datos.

**Tabla 23: Descripción de caso de uso Registrar Usuario**



<b>Nombre:</b>	Consultar Historial de Enviados
<b>Nivel:</b>	0
<b>Resumen:</b>	El usuario visualiza los mensajes enviados por él y los pendientes por enviar.
<b>Actor:</b>	Usuario.
<b>Precondición:</b>	El usuario debe estar autenticado.
<b>Descripción:</b>	El usuario accede al módulo "Historial->Enviados" del menú en el cual puede visualizar los envíos realizados y pendientes por enviar y acceder a la información detallada de cada uno, además de filtrarlos por mes.
<b>Postcondición:</b>	Este caso de uso no tiene postcondición.

**Tabla 24: Descripción de caso de uso Consultar Historial de Enviados**

<b>Nombre:</b>	Consultar Historial de Recibidos
<b>Nivel:</b>	0
<b>Resumen:</b>	El usuario visualiza los mensajes recibidos (MMS y SMS).
<b>Actor:</b>	Usuario.
<b>Precondición:</b>	El usuario debe estar autenticado.
<b>Descripción:</b>	El usuario accede al módulo "Historial->Recibidos" del menú en el cual puede visualizar los SMS y MMS recibidos y acceder a la información detallada de cada uno, además de filtrarlos por mes.
<b>Postcondición:</b>	Este caso de uso no tiene postcondición.

**Tabla 25: Descripción de caso de uso Consultar Historial de Recibidos**

<b>Nombre:</b>	Configurar Operadora
<b>Nivel:</b>	0
<b>Resumen:</b>	El Administrador configura los datos de una operadora móvil en específico.
<b>Actor:</b>	Administrador.
<b>Precondición:</b>	El Administrador debe estar autenticado.
<b>Descripción:</b>	<ul style="list-style-type: none"> <li>✓ El Administrador accede al módulo "Configurar Operadora" del menú.</li> <li>✓ Se selecciona la operadora a configurar.</li> <li>✓ Se cargan los datos actuales de la operadora, si estos se encuentran registrados en la base de datos.</li> <li>✓ Se ingresan o modifican los siguientes datos: Host, Puerto, Servlet y APN.</li> <li>✓ Se hace click sobre el botón "Guardar".</li> </ul>
<b>Postcondición:</b>	Se actualizan los datos de la operadora en la base de datos.

**Tabla 26: Descripción de caso de uso Configurar Operadora**

### 3.3.2.2 Modelo Objetos del Dominio

La Figura 13 presenta el modelo objeto del dominio elaborado para la aplicación Web.

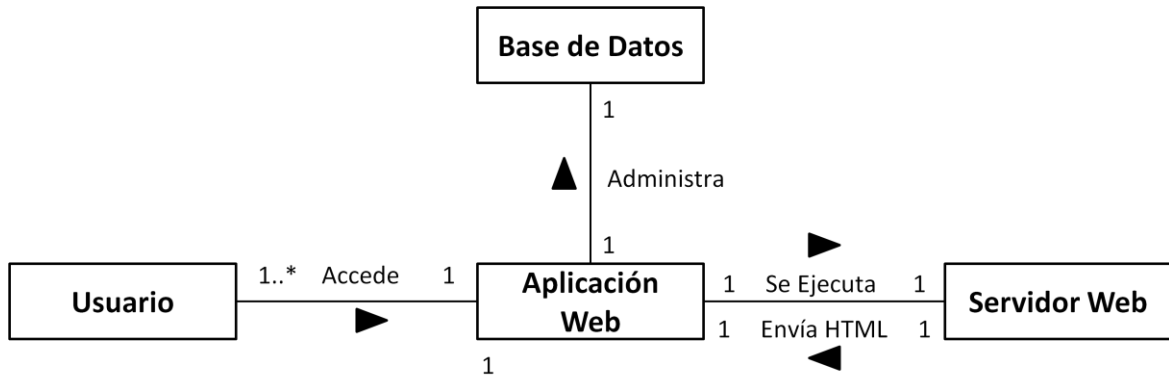


Figura 31: Diagrama de objetos del dominio de la Aplicación Web

Los objetos identificados en el dominio son:

- **Usuario:** representa al ente o persona que opera la aplicación Web. El usuario envía mensajes multimedia y puede monitorear los envíos y los mensajes recibidos.
- **Aplicación Web:** objeto que representa a la aplicación que provee las interfaces de usuario e implementa la lógica interna para la creación y monitoreo de envíos por parte del usuario.
- **Base de Datos:** objeto que representa al repositorio de almacenamiento utilizado por la aplicación Web para registrar los envíos generados por el usuario y del cual se consulta la información transaccional relacionada a los MMS. Esta base de datos es la misma utilizada por la aplicación motor.
- **Servidor Web:** Servidor que ejecuta los códigos en lenguaje PHP de la aplicación Web.

### 3.3.3 Diseño

En esta fase se realizan los diagramas de secuencia de la aplicación Web para describir la suceción de procesos.

#### 3.3.3.1 Diagrama de Secuencia

A continuación se presentan los diagramas de secuencia de los procesos en la aplicación Web, el cual muestra la interacción del conjunto de objetos a través del tiempo.

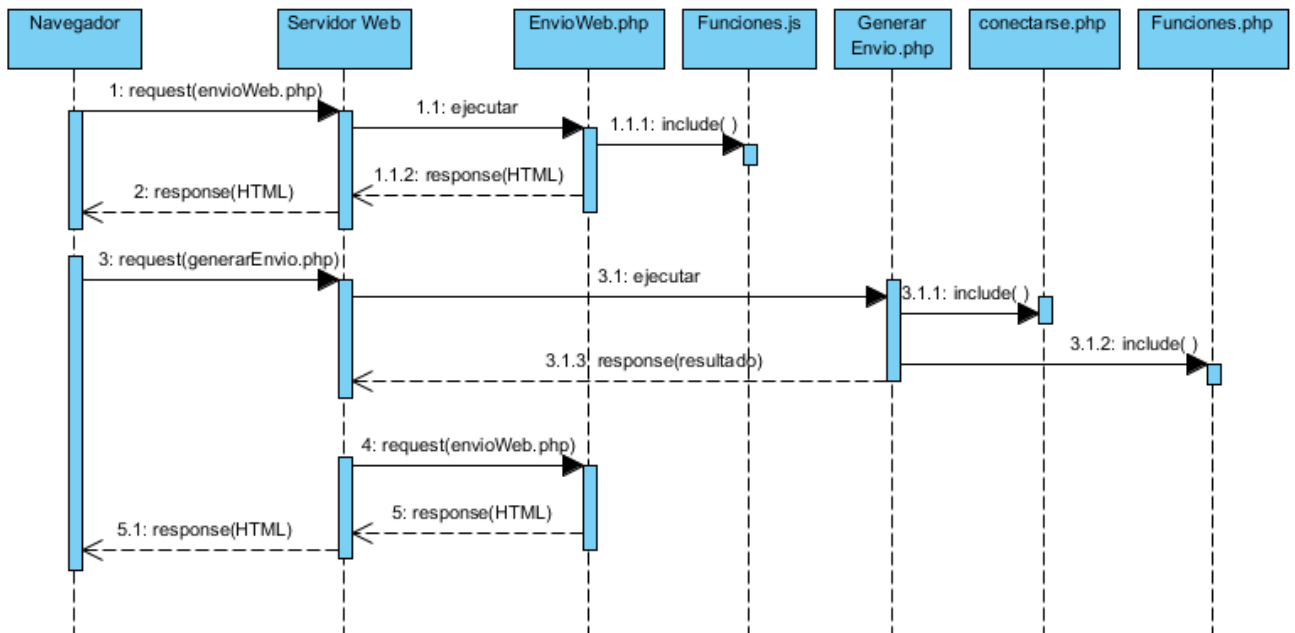


Figura 32: Diagrama de secuencia del proceso Envío Web

En la figura 32 se muestra la secuencia de acciones ejecutadas en el proceso de envío de un mensaje multimedia a través de la aplicación Web.

- El módulo de la aplicación *Envío Web*, el cual invoca a *EnvioWeb.php* provee la vista HTML con el formulario para especificar los datos del mensaje a enviar.
- Una vez pulsado el botón *Enviar*, se invoca a *GenerarEnvio.php*, el cual recibe los datos del formulario y genera los registros en las tablas de la base de datos *T\_EnvioBloque* y *T\_Transmision* con la información del envío solicitado. Este PHP incluye los archivos *Conectarse.php* y *Funciones.php*, que contienen las funciones que operan sobre la base de datos.
- Por último se redirecciona nuevamente a la interfaz del Envío Web.

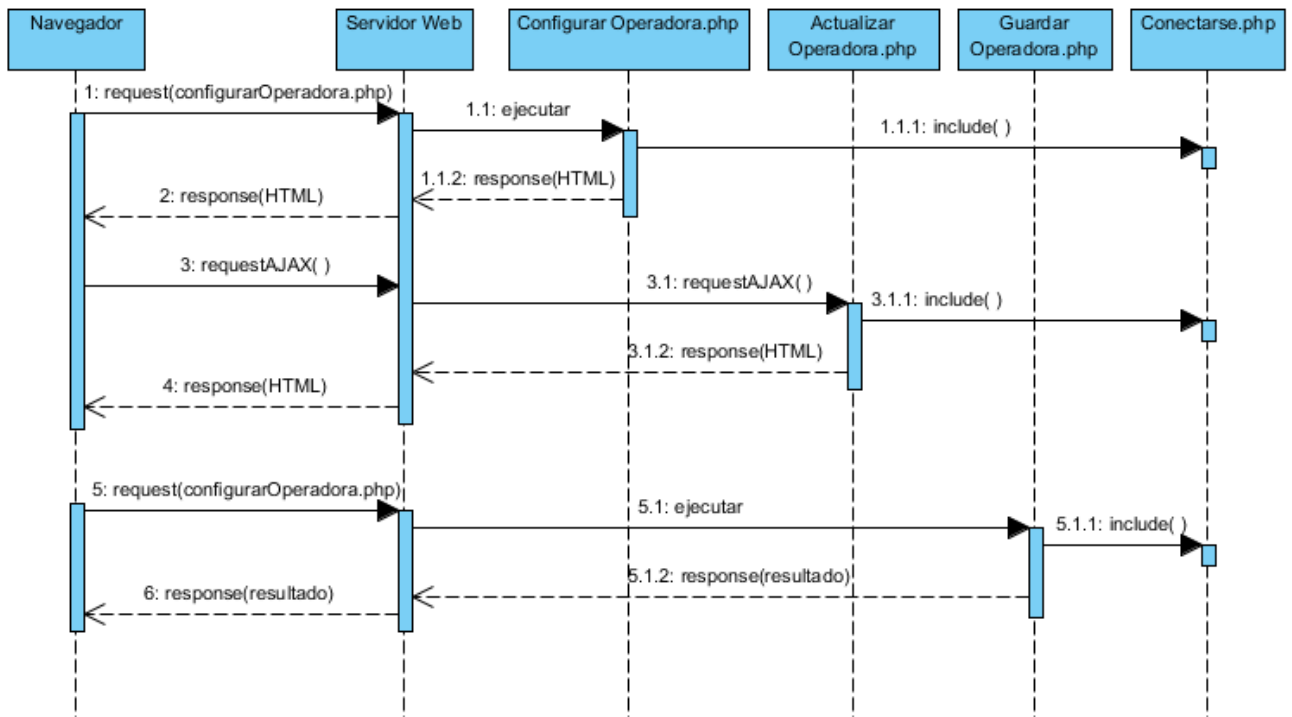


Figura 33: Diagrama de secuencia de proceso Configurar Operadora

La figura 33 muestra la secuencia de acciones llevadas a cabo para acceder al reporte historial de mensajes enviados.

- *ConfigurarOperadora.php* provee la vista Web (Pop-Up) para la configuración de los datos de conexión MMS para el modem GSM de una operadora móvil en particular. Inicialmente se selecciona la operadora a modificar, y por una petición asíncrona (AJAX, Asynchronous JavaScript And XML) a PHP *ActualizarOperadora.php* se cargan de la tabla C\_Operadora de la base de datos, los parámetros asociados a la operadora en el formulario de la vista.
- Los datos cargados se pueden editar, y una vez realizado los cambios se clickea el botón *Guardar*, lo cual generará una petición a *GuardarOperadora.php*.
- Este último PHP actualiza en la base de datos la información recibida del formulario para el registro de la operadora móvil objeto de modificación.

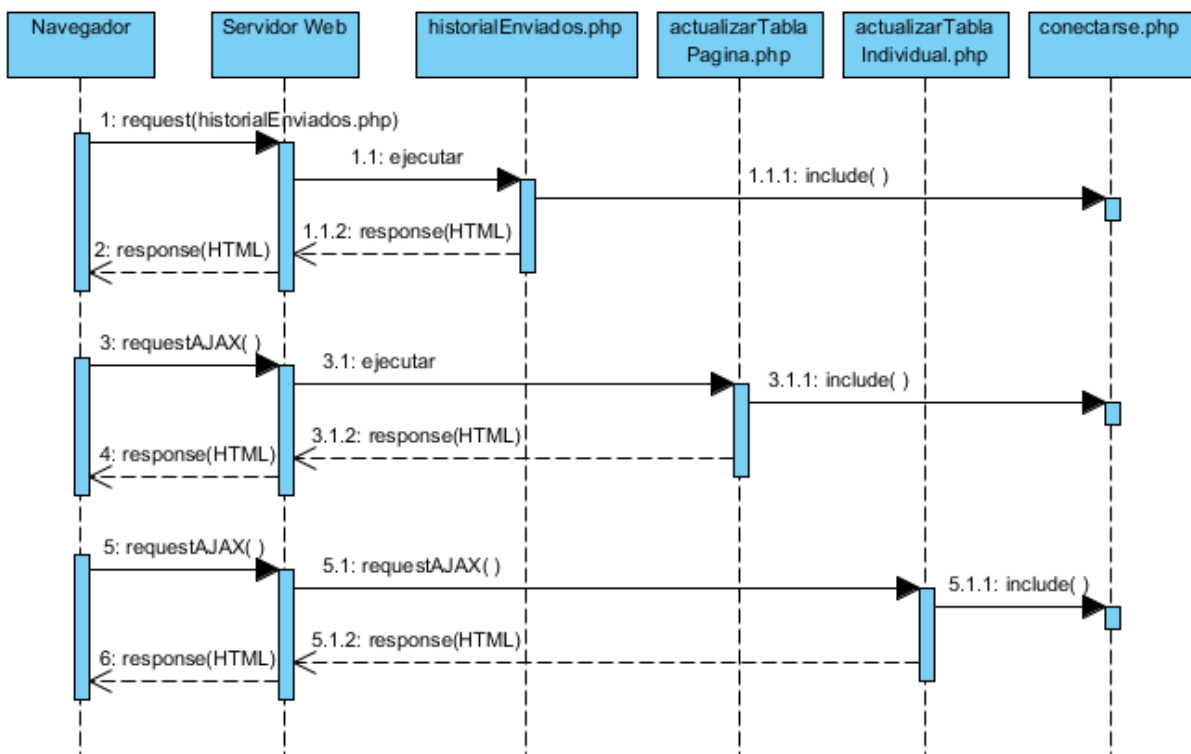


Figura 34: Diagrama de secuencia de proceso consultar Historial de Enviados

La figura 34 muestra la secuencia de acciones para la visualización del historial de mensajes enviados en la aplicación Web.

- El navegador al solicitar la ejecución del PHP *HistorialEnviados.php* obtiene como resultado el reporte paginado de los envíos realizados en el mes actual. Si se desea filtrar los envíos por otro mes o cambiar la página, se realiza una petición asíncrona para la ejecución de *ActualizarTablaPagina.php*, lo cual actualiza el reporte de acuerdo al criterio especificado anteriormente, sin tener que cargarse la vista por completo nuevamente. La inclusión de *conectarse.php* al igual que en los diagramas anteriores, provee de las funciones de acceso a la base de datos.
- Al darle click a un envío en particular, se actualiza vía AJAX el reporte al invocar *ActualizarTablaIndividual.php* para mostrar el detalle de las transmisiones asociadas al envío, las cuales especifican los celulares destinatarios y la fecha de envío del mensaje a cada uno.

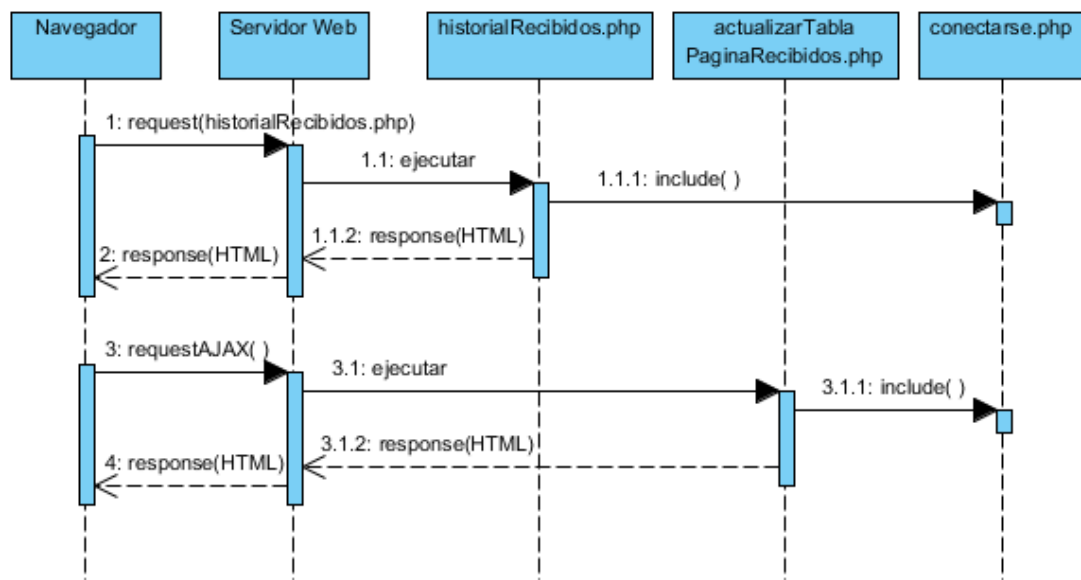


Figura 35: Diagrama de secuencia de proceso consultar Historial de Recibidos

La figura 35 describe la secuencia de acciones que lleva a cabo el proceso de visualización del historial de mensajes recibidos.

El procedimiento de llamadas es el mismo que para el historial de mensajes enviados, con la única diferencia de que el PHP invocado para la actualización del reporte vía AJAX al paginar o filtrar por mes es *ActualizarTablaPaginaRecibidos.php* y además, este reporte no provee detalle para cada mensaje recibido.

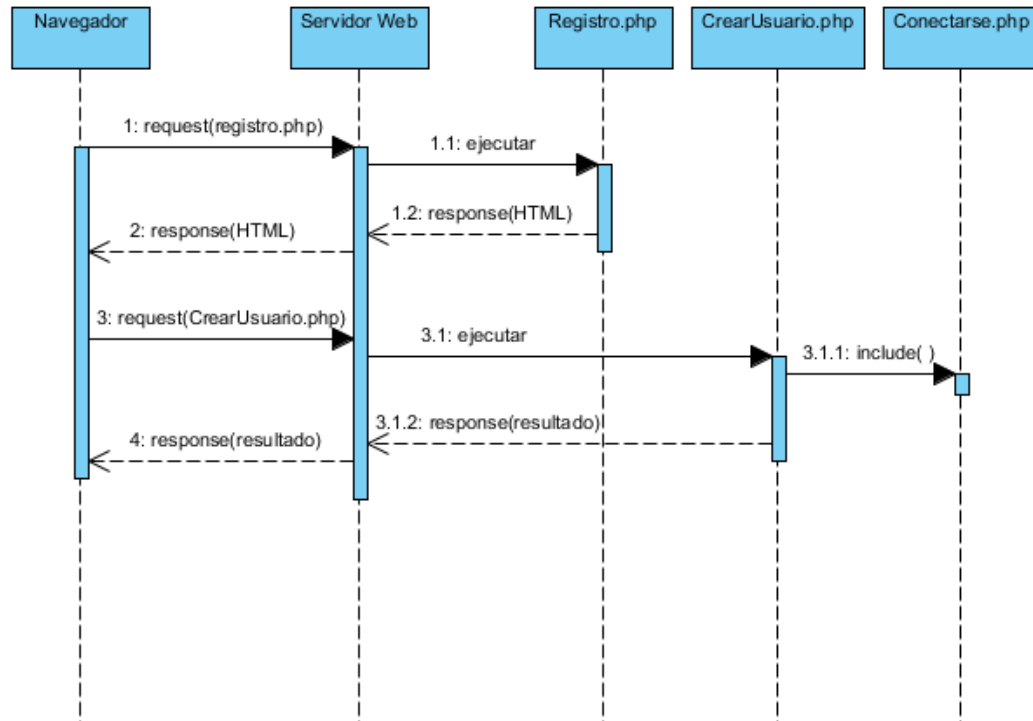


Figura 36: Diagrama de secuencia del proceso Registro de Usuario

En la figura 36 se muestran la secuencia de llamadas en el registro de un usuario en la aplicación Web.

- *Registro.php* provee la vista Web (Pop-Up) con el formulario de datos del usuario (nombre, login y clave).
- Una vez oprimido el botón *Guardar*, se generará una petición a *CrearUsuario.php*, el cual es el PHP que recibe la información del formulario y con esta crea un registro en la tabla *T\_Usuario* de la base de datos.

### 3.3.4 Implementación

#### 3.3.4.1 Descripción del módulo de Envío Web

La siguiente figura la vista del módulo de Envío Web de la aplicación.

**ALS TELECOM**

**NUEVOMUNDO**

Iniciar Sesión | Registrarse

**Envío Web**

**Ingrese los celulares destinatarios:**

Celulares 0 (Max 100)

**Ingrese el Asunto:**

**Ingrese el contenido del mensaje:**

Número de Caracteres 0 (Max 160)

**Seleccione la Imágen a adjuntar:**

Examinar...

Enviar

ALS Telecom © 2011  
contacto@alstelecom.com | (0212) 2424805

Figura 37: Vista del módulo de Envío Web



Para enviar un mensaje multimedia a través del módulo, se deben especificar los números celulares de los destinatarios separados por “;” y el texto del mensaje. El asunto y la imagen son campos opcionales. En caso de no colocar alguno de los dos primeros aparecerá la siguiente alerta:

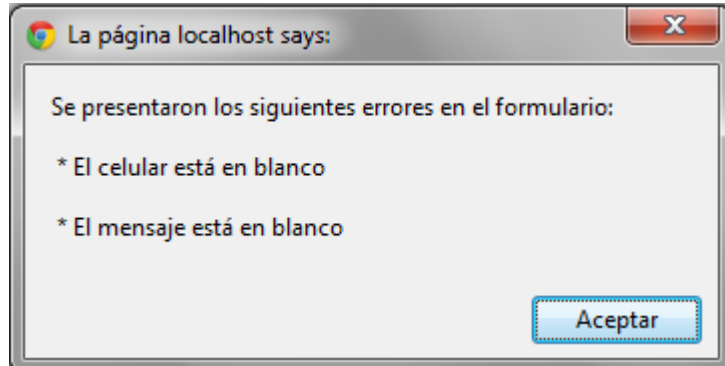


Figura 38: Alerta de error en Envío Web

Un usuario registrado en el sistema puede autenticarse accediendo a *Iniciar Sesión* en la parte superior derecha en la pantalla de *Envío Web*.



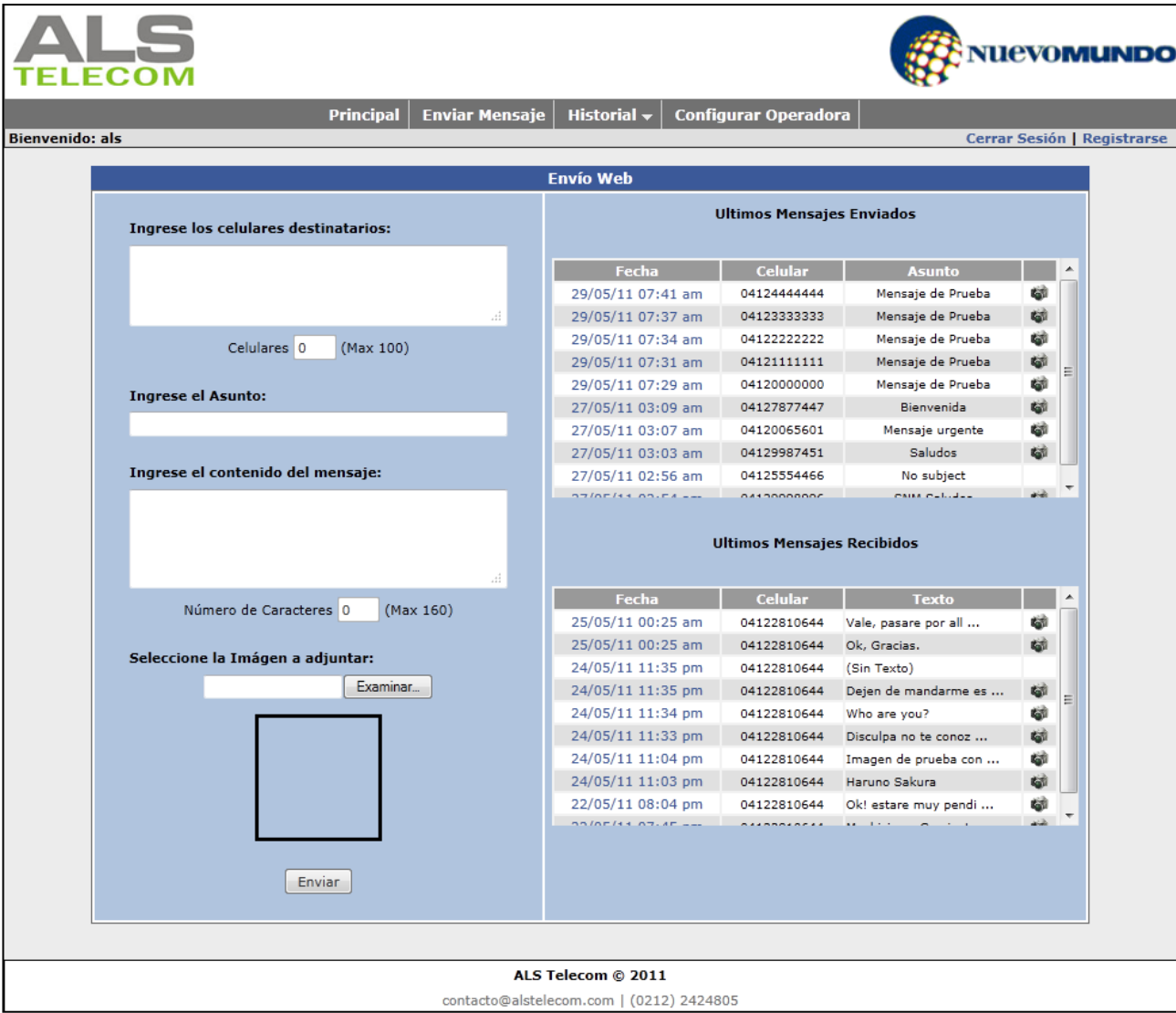
Figura 39: Vista del módulo de Iniciar Sesión

En caso de no existir el usuario especificado o de ingresar una contraseña inválida aparecerá el siguiente mensaje:

The screenshot displays the login interface for ALS Telecom. At the top left is the ALS TELECOM logo, and at the top right is the NUEVOMUNDO logo. The main content area features a central login box with a blue header titled 'Iniciar Sesión'. Below the header, a red error message reads 'Usuario o clave inválido'. The login form includes a 'Usuario:' field with the text 'als', a 'Contraseña:' field with masked characters '\*\*\*\*\*', and an 'Entrar' button. The footer of the page contains the text 'ALS Telecom © 2011' and the contact information 'contacto@alstelecom.com | (0212) 2424805'.

**Figura 40: Vista de mensaje de error en Iniciar Sesión**

Cuando el usuario se autentica en el sistema, el módulo de *Envío Web* se complementa con un panel a la derecha, el cual permite visualizar un historial de los últimos diez envíos realizados por el usuario y los últimos diez mensajes recibidos por la aplicación.



**Figura 41: Vista del módulo de Envío Web después de la autenticación del usuario**

Al hacer click sobre la fecha de un mensaje enviado o recibido, se despliega un *Pop-Up* con el detalle del envío (celular, fecha de envío o recepción, mensaje, asunto e imagen en caso de que posea).

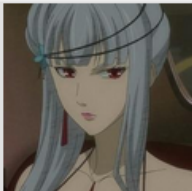
Celular	Fecha	Asunto	Texto	Imagen
04129987451	27/05/11 03:03 am	Kurikaesu	En 1833, le pidieron a Zhukovski que escribiera una nueva letra para una composición musical de Alekséi Fiódorovich Lvov llamada La plegaria del pueblo ruso. .	

Figura 42: Vista del detalle de un envío en el módulo Envío Web

### 3.3.4.2 Descripción del módulo de Configuración Operadora

La siguiente figura corresponde a la vista del módulo Configurar Operadora.

cerrar

### Ingrese datos de configuración de la Operadora

---

Operadora:

WAP Host:

WAP Port:

MMS Servlet:

APN:

Figura 43: Vista del módulo Configurar Operadora

Una vez que se selecciona la operadora en el componente *Select* correspondiente, se cargan sus datos almacenados en la base de datos, es ahí cuando el usuario puede editar la información existente y actualizarla al darle click en el botón *Guardar*.



The screenshot shows a web form titled "Ingrese datos de configuración de la Operadora" with a "cerrar" button in the top right corner. The form contains the following fields:

- Operadora:** A dropdown menu with "Digitel" selected.
- WAP Host:** A text input field containing "10.99.0.10".
- WAP Port:** A text input field containing "9201".
- MMS Servlet:** A text input field containing "http://mms.412.com.ve/ser".
- APN:** A text input field containing "expresate.digitel.ve".

At the bottom of the form is a "Guardar" button.

Figura 44: Vista de módulo Configurar Operadora una vez seleccionada

### 3.3.4.3 Descripción del módulo Historial de Mensajes Enviados

La siguiente figura corresponde a la vista del módulo de la aplicación Historial de Mensajes Enviados. Este reporte lista los registros de la tabla T\_EnvioBloque, los cuales corresponden a cada uno de los envíos que ha registrado el sistema con la fecha y hora en la que se realizó, el usuario que lo llevó a cabo, el asunto, mensaje literal, imagen (estos últimos tres datos son extraídos de cualquiera de las transmisiones asociadas a este envío bloque), la cantidad de transmisiones o destinatarios asociados a ese envío y por último a cuantos se les ha enviado. El historial permite filtrar por los últimos seis meses transcurridos e implementa paginación (10 envíos por página).

Historial de Mensajes Enviados						
Filtro por meses: mayo						
Fecha y Hora	Usuario	MMS			Registros	
		Asunto	Mensaje literal	Imagen	Cantidad Recibidos	Cantidad Enviados
2011-05-27 20:34:13		(Sin Asunto)	Hola como estas? tanto tiempo sin saber de ti		2	0
2011-05-27 15:49:35	ALS	Mensaje de Prueba	Mensaje de prueba con catorce destinatarios		14	5
2011-05-27 03:07:22	Maria Risi	Bienvenida	Estimado asegurado, SNM le da las gracias por elegirnos como su compañía de Seguros. Poliza contratada nro .		1	1
2011-05-27 02:56:05	Maria Risi	Mensaje urgente	Seguros Nuevo Mundo le da un mensaje de paz y amor, orgullosos de poder ofrecerle un servicio oportuno, cercano y de calidad como respaldo a su bienestar		1	1
2011-05-27 02:51:53	Maria Risi	Saludos	SNM innovando soluciones, le informa que a partir del dia de hoy usted dispondrá de mensajería de texto gratuita con información amplia y útil.		1	1
2011-05-27 02:50:28	Maria Risi	No subject	Esto es un mensaje de prueba		1	1
2011-05-27 02:48:58	Maria Risi	SNM Saludos	Nuevo Mundo, con el fin de dar cumplimiento a las providencias del SENIAT N° 0590 y 0257, solicita la consignación de sus facturas a la brevedad posible		1	1
2011-05-27 02:47:29	Maria Risi	Urgente	Nuevo Mundo Seguros e Inversora Primaban les dan la bienvenida y les agradece habernos seleccionado como empresa de seguro		1	1
2011-05-27 02:44:12	Maria Risi	Pendiente	Falling to Pieces, easily, somebody put me in together, midlife crisis		1	1
<b>Total de Mensajes Enviados: 24</b>						
Página 1/2						

Figura 45: Vista del módulo Historial de Mensajes Enviados

Al hacer click sobre la fecha y hora de algún envío en particular, se muestra el detalle del envío con los celulares de los destinatarios y el estatus de cada uno (*En Proceso* en caso de que no se le haya enviado el mensaje ó la fecha y hora del envío en caso afirmativo).

Historial de Mensajes Enviados	
Reporte Detallado	
Reporte de mensajes enviados el 2011-06-17 19:42:30	
Mensaje: Mensaje de prueba con catorce destinatarios	
Celular	Fecha
0412000000	2011/05/29 07:29:34
0412111111	2011/05/29 07:31:20
0412222222	2011/05/29 07:34:07
0412333333	2011/05/29 07:37:51
0412444444	2011/05/29 07:41:26
0412555555	En Proceso
0412666666	En Proceso
0412000000	En Proceso
0412111111	En Proceso
0412222222	En Proceso
0412333333	En Proceso
0412888888	En Proceso
Página <input type="text" value="1"/> << Página 1/2 >>	

Figura 46: Vista del detalle de un envío en el módulo Historial de Mensajes Enviados

### 3.3.4.4 Descripción del módulo Historial de Mensajes Recibidos

La siguiente figura corresponde a la vista del módulo de la aplicación Historial de Mensajes Recibidos. Este reporte lista los registros de la tabla T\_Respuesta, los cuales son los mensajes enviados por otros teléfonos celulares, recibidos por el modem y almacenados en la tarjeta SIM, tanto mensajes cortos de texto (SMS) como mensajes multimedia (MMS). El historial al igual que el de mensajes enviados permite filtrar por los últimos seis meses transcurridos e implementa paginación (10 envíos por página).

Historial de Mensajes Recibidos			
Filtro por meses: mayo ▼			
Fecha y Hora de Recepción	Asunto	Mensaje literal	Imagen
2011-05-25 00:25:38	Sin Asunto	Vale, pasare por alla mas tarde, tipo 2 de la tarde	
2011-05-25 00:25:15	ultima prueba	Ok, Gracias.	
2011-05-24 23:35:38	Sin Asunto	(Sin Texto)	
2011-05-24 23:35:13	Sin Asunto	Dejen de mandarme estos mensajes!	
2011-05-24 23:34:29	Sin Asunto	Who are you?	
2011-05-24 23:33:18	Sin Asunto	Disculpa no te conozco quien eres?	
2011-05-24 23:04:28	Sin Asunto	Imagen de prueba con GD	
2011-05-24 23:03:51	Responder ya	Haruno Sakura	
2011-05-22 20:04:41	Hola	Ok! estare muy pendiente!	
<b>Total de Mensajes Recibidos: 13</b>			
Página 1 ▼    << Página 1/2 >>			

Figura 47: Vista del módulo Historial de Mensajes Recibidos

### 3.3.4.5 Descripción del módulo Registro de Usuario

La siguiente figura corresponde a la ventana de registro de un usuario. Los datos a especificar son: nombre del usuario, nombre de la cuenta o *login* y la contraseña. El resultado final es el la inserción de un nuevo registro en la tabla T\_Usuario con los datos especificados.



The image shows a web form for user registration. At the top right, there is a link labeled "cerrar". Below it, the title "Ingrese datos del nuevo usuario" is displayed. The form contains four input fields, each with a label to its left: "Nombre:", "Nombre de usuario:", "Contraseña:", and "Confirmar Contraseña:". Below these fields is a button labeled "Guardar".

Figura 48: Ventana de Registro de Usuario

### 3.3.5 Pruebas de Integración y Resultados

En esta fase se realizan las pruebas finales de manera masiva de la plataforma de mensajería multimedia como un todo, es decir, el motor de envío y recepción integrado con la aplicación Web.

Para estas pruebas, los envíos son generados a través del módulo de *Envío Web* y tanto el contenido de los mensajes enviados como de los mensajes recibidos se verifica en los módulos de Historiales.

#### 3.3.5.1 Envío MMS

Para las pruebas se seleccionó un solo formato de imagen (JPEG), ya que en las pruebas anteriores se determinó que no influye en el proceso de envío y recepción. Para cada tamaño de imagen se realizaron 30 envíos.

Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Envíos Realizados	Envíos Exitosos	Envíos Fallidos
10 KB	JPEG		30	29	1
15 KB	JPEG		30	24	6
20 KB	JPEG		30	18	12
25 KB	JPEG		30	11	19

Tabla 27: Prueba de envíos multimedia masivos

Los resultados especificados en la tabla, se pueden expresar en un gráfico de torta por cada fila:

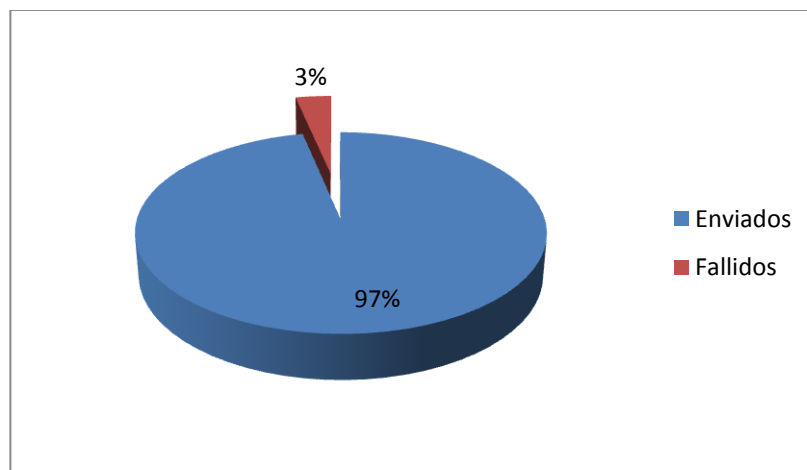


Gráfico 1: Porcentaje de envíos exitosos y fallidos (Tamaño de imagen 10 KB)

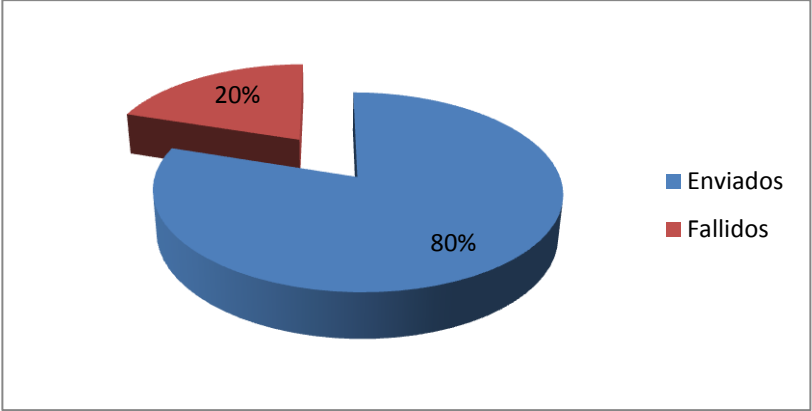


Gráfico 2: Porcentaje de envíos exitosos y fallidos (Tamaño de Imagen 15 KB)

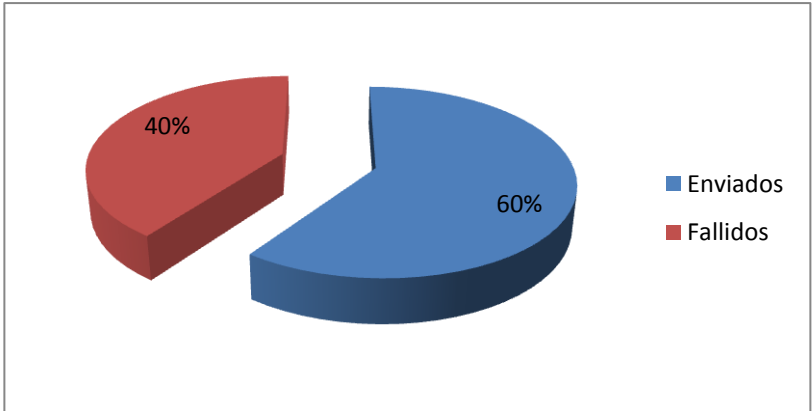


Gráfico 3: Porcentaje de envíos exitosos y fallidos (Tamaño de imagen 20 KB)

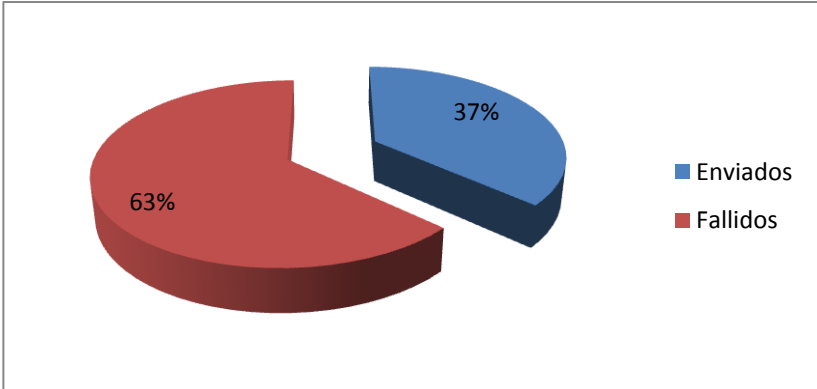


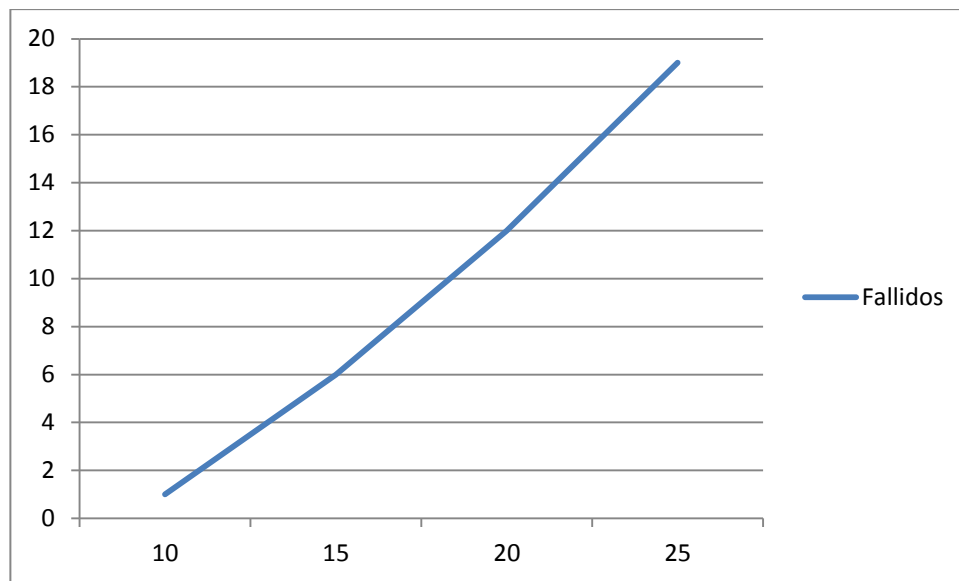
Gráfico 4: Porcentaje de envíos exitosos y fallidos (Tamaño de imagen 25 KB)

El análisis de las proporciones indicadas en cada uno de los gráficos para cada tamaño de imagen en base a las muestras tomadas indica lo siguiente:

- El porcentaje aproximado de éxito de un envío de un mensaje multimedia con una imagen de tamaño 10 KB es del 97%.
- El porcentaje aproximado de éxito de un envío de un mensaje multimedia con una imagen de tamaño 15 KB es del 80%.
- El porcentaje aproximado de éxito de un envío de un mensaje multimedia con una imagen de tamaño 20 KB es del 60%.
- El porcentaje aproximado de éxito de un envío de un mensaje multimedia con una imagen de tamaño 25 KB es del 37%.

Después de los 20 KB de imagen, no se puede garantizar ni el 50% del éxito de los envíos realizados. Estos fallos se deben al limitado ancho de banda y velocidades de transmisión de datos que proporciona la red GPRS. Adicionalmente las operadoras móviles establecen un límite de bytes a transmitir en cada conexión WAP/GPRS para MMS, en el caso particular de DIGITEL, se permite enviar 25 KB de datos en cada sesión, luego de excedida esta cantidad no se garantiza la estabilidad de la conexión.

El tamaño de las imágenes de los mensajes multimedia a enviar a través de la plataforma es limitado en el módulo de Envío Web por medio de una validación JavaScript, la cual no permite cargar imágenes de tamaño superior a los 20 KB.



**Gráfico 5: Comportamiento de la tasa de mensajes enviados fallidos en base al tamaño de la imagen**

En el gráfico anterior se puede apreciar que el incremento de los mensajes enviados fallidos no se comporta de forma lineal, sino que ligeramente se va incrementando de manera exponencial.

### **3.3.5.2 Recepción MMS**

Para las pruebas de recepción se seleccionó un solo formato de imagen (JPEG), al igual que en las pruebas de envío. El tamaño de las imágenes contenidas en los mensajes a enviar fue aumentado, debido a que por lo demostrado en las pruebas individuales del motor, la recuperación de contenidos multimedia es más confiable, esto se debe a que se transfiere menor cantidad de bytes a la hora de solicitar la descarga del contenido en comparación a su proceso de envío.

Los envíos se realizaron desde teléfonos celulares para así garantizar mayor fiabilidad en el resultado de la transmisión de los contenidos a través de la red GPRS.

Tamaño Imagen	Formato Imagen	Vista Previa Imagen	Envíos Realizados	Recibidos Exitosos	Recibidos Erróneos
10 KB	JPEG		30	28	2
20 KB	JPEG		30	29	1
30 KB	JPEG		30	27	3
40 KB	JPEG		30	24	6

**Tabla 28: Prueba de recepción multimedia masiva**

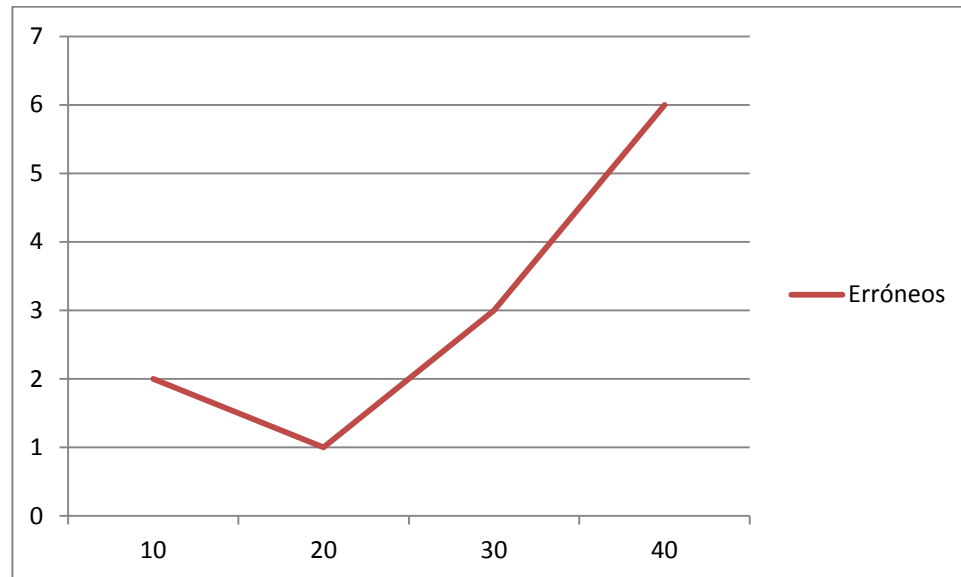
El error en la recepción de mensajes ocurrió en la fase de recuperación del contenido multimedia y nunca en la recepción de la notificación MMS. Se identificó que la causa de estos fallos lo generó uno de estos dos eventos:

- Se excedió el tiempo máximo de espera de una respuesta por parte del MMSC conforme a la solicitud de recuperación enviada.
- El contenido recibido contenía un mensaje de error, el cual notificaba que el servicio se encontraba temporalmente fuera de servicio (Figura 31).

No se notó una diferencia drástica en el fallo de la recuperación multimedia en base al aumento del tamaño de las imágenes contenidas en los mensajes, es decir, que la cantidad de bytes del mensaje ó imagen no fue completamente determinante para la ocurrencia de un fallo en dicho proceso, basándose en las características de la muestra seleccionada. A partir de imágenes de tamaño 40 KB se empieza a notar el incremento en la tasa de fallos a la hora de descargar el contenido del mensaje. Por esta razón se

recomienda realizar envíos de mensajes multimedia desde el teléfono celular hacia la aplicación con imágenes no mayores a 40 KB de tamaño.

En estos dos gráficos se puede apreciar el comportamiento de los fallos ocurridos en el proceso de recuperación del MMS, en base al incremento del tamaño de la imagen del mensaje.



**Gráfico 6: Comportamiento de los fallos de recuperación multimedia en base al tamaño de la imagen**

En base al gráfico anterior se puede apreciar como variable el comportamiento de la tasa de fallos en la recuperación multimedia para las muestras seleccionadas, hasta que se utilizan imágenes mayores a 30 KB, en ese punto se aprecia un comportamiento exponencial en el incremento de los fallos y no lineal.

## Conclusiones y Recomendaciones

El objetivo del presente Trabajo Especial de Grado consistió en el desarrollo de una plataforma que permite enviar y recibir MMS y administrar los contenidos e información relacionada. El objetivo fue cumplido ya que se logró obtener un producto que satisface los requerimientos planteados al comienzo de este trabajo y que satisface las necesidades de comunicación de los usuarios a través de la telefonía móvil.

Con el desarrollo de este sistema, se obtuvieron conocimientos sobre los protocolos y estándares involucrados en el intercambio de mensajes multimedia, tales como WAP, SMS, MMS, GSM, GPRS, así como sobre el conjunto de elementos que intervienen en dicho proceso y los parámetros a tomar en cuenta para lograr la conexión a las redes móviles.

La metodología aplicada e investigación realizada, permitió desarrollar una solución para envío y recepción MMS, que cumple con los siguientes requerimientos:

- Codificación y decodificación de mensajes MMS.
- Manejador de envío y recepción MMS.
- Acceso Web para el envío y recepción MMS.
- Almacenamiento y registro de estatus para las transacciones MMS.
- Integración de todos los elementos requeridos por la plataforma.

El seguimiento de las pautas establecidas en la metodología de desarrollo propuesta contribuyó, en gran medida, con el cumplimiento de los objetivos de la investigación. Se debe tomar en cuenta que dicha metodología posee etapas, con alcances bien delimitados, los cuales permitieron organizar el proceso de desarrollo de la herramienta, pudiendo llevar así un completo control de las tareas realizadas, rigiendo el conjunto de pasos que deben ser llevados a cabo para realizar una aplicación de este tipo.

Al ser el sistema configurable para cualquier operadora, permite además de abarcar un mercado local, extenderse a nivel mundial.



El desarrollo de este Trabajo Especial de Grado ofrece un nuevo aporte académico y profesional para las aplicaciones que implementen los MMS, al haber desarrollado un conjunto de utilitarios en lenguaje Java y PHP que permiten decodificar e interpretar mensajes multimedia y desplegar los contenidos asociados en una interfaz Web.

Actualmente en Venezuela no existe compatibilidad de intercambio MMS entre clientes de diferentes operadoras como es en el caso de los SMS, por ello solo se realizaron pruebas de envío y recepción MMS entre suscriptores de una misma operadora móvil.

Algunas recomendaciones a considerar para desarrollos futuros son:

Investigar los parámetros de configuración MMS para otras operadoras móviles del mercado venezolano como por ejemplo Movistar y Movilnet.

Implementar la capacidad de soporte en los mensajes para incluir diversos tipos de contenidos multimedia tales como formatos de audio o video.

Permitir la inclusión de más de dos elementos multimedia en el mensaje, adicional a la imagen y el texto que actualmente soportan, así como la inclusión e interpretación de un documento SMIL para dar sincronización, temporización y formato a todos los elementos, similar a una presentación de diapositivas.

Desarrollar en la aplicación Web un módulo de gestión de destinatarios que permita almacenarlos, modificarlos o eliminarlos sobre la base de datos, así como la implementación de un filtrado dinámico de destinatarios en el módulo *Envío Web*.

Contar con módems de acceso a la red 3G para mejorar las velocidades de transmisión.

## Referencias Bibliográficas

[1] Halonen, Timo; Romero, Javier; Melero, Juan. (2003). GSM, GPRS and EDGE Performance – Evolution Towards 3G/UMTS. 615 Págs.

[2] Redl, Siegmund; Weber, Matthias; Oliphant, Malcolm. (1998). GSM and Personal Communications Handbook. 526 Págs.

[3] GSMA (GSM Association). (2010). GSM Technology. Consultado el día 18 de Octubre del 2010 de la World Wide Web:

<http://gsmworld.com/technology/gsm/index.htm>

[4] The International Engineering Consortium. (2007). Web Proforum Tutorials. Consultado el día 21 de Octubre del 2010 de la World Wide Web:

<http://burnsidetelecom.com/whitepapers/gsm.pdf>

[5] Le Bodic, Gwenaël. (2005). Mobile Messaging Technologies and Services SMS, EMS and MMS. 432 Págs.

[6] Cantone, Dante. (2006). Implementación y Debugging – Ciclo de Vida del Software. Revista USERS.code. Editorial MP EDICIONES. 320 Págs.

[7] Master Magazine. (2005). Definición de GSM. Consultado el día 7 de Abril del 2010 de la World Wide Web:

<http://www.mastermagazine.info/termino/5188.php>

[8] Oyarce, Andrés; Aguayo, Paul. (2008). Guía del Usuario Enfora GSM-1218 Quad-Band SA-GL. 110 Págs.

[9] Tecnología Hecha Palabra. (2007). Glosario Técnico. Consultado el día 2 de Abril del 2010 de la World Wide Web:

[http://www.tecnologiahechapalabra.com/tecnologia/glosario\\_tecnico/articulo.asp?i=786](http://www.tecnologiahechapalabra.com/tecnologia/glosario_tecnico/articulo.asp?i=786)

[10] Sitio Web de “The Spanish Bluetooth Security Group – BlueHack”. (2005). Comandos AT. Consultado el día 28 de Marzo del 2010 de la World Wide Web

<http://bluehack.elhacker.net/proyectos/comandosat/comandosat.html>

[11] Masadelante.com. ¿Qué es una Tarjeta SIM? Consultado el 3 de Abril del 2010 de la World Wide Web:

<http://www.masadelante.com/faqs/tarjeta-sim>

[12] Millán, Ramón. (2002). Mensajes Móviles Multimedia MMS. Consultado el 2 de Abril de la World Wide Web:

<http://www.ramonmillan.com/tutoriales/mms.php>

[13] World Wide Web Consortium. (1998). The World Wide Web Consortium Issues SMIL 1.0 as a W3C Recommendation. Consultado el día 29 de Marzo del 2010 de la World Wide Web:\*

<http://www.w3.org/Press/1998/SMIL-REC>

[14] ALEGSA.com.ar. (2009). Diccionario de Informática – Definición de Modem. Consultado el día 30 de Marzo del 2010 de la World Wide Web:

<http://www.alegsa.com.ar/Dic/modem.php>

[15] Enfora ©. (2010). Enfora GSM1308 and GSM1318 Quad-Band Spider SA-G+ - User Guide. Consultado el 14 de Noviembre del 2010 de la World Wide Web:

[http://www.enfora.com/344/GSM1308UG-\\_Enfora\\_SA-G+\\_User\\_Guide.pdf](http://www.enfora.com/344/GSM1308UG-_Enfora_SA-G+_User_Guide.pdf)

[16] TECNOSEGUR Consultoría de Soluciones. (2008). Enfora SA-GL GSM/GPRS. Consultado el día 1 de Abril del 2010 de la World Wide Web:

<http://www.tecnosegur.com/productos/sagl.pdf>

[17] Martínez, Evelio. (2007). Comandos AT. Consultado el día 1 de Abril del 2010 de la World Wide Web:

[http://technet.microsoft.com/es-es/library/cc784492\(WS.10\).aspx](http://technet.microsoft.com/es-es/library/cc784492(WS.10).aspx)

[18] Digitel GSM. (2007) ¿Por qué GSM? Consultado el 27 de Octubre del 2010 de la World Wide Web:

<http://www.digitel.com.ve/Secciones/Corporativo.aspx?level=172&Seccion=176&Menu=A1>

[19] Osmosis Latina. (2005). Aplicaciones Inalámbricas (WAP/WML). Consultado el 10 de Noviembre de la World Wide Web:

<http://www.osmosislatina.com/aplicaciones/wap.htm>

[20] WinWAP Technologies. (2005). WinWAP Mobile Client SDK - How to send and receive MMS messages. 9 Págs.

[21] Asociación de la Industria Eléctrica – Electrónica Chile. GPRS. Consultado el 11 de Noviembre del 2010 de la World Wide Web:

<http://www.aie.cl/files/file/comites/ca/articulos/octubre-06.pdf>

[22] World Wide Web Consortium. (2001). Especificaciones de HTML. Consultado el día 14 de Noviembre del 2010 de la World Wide Web:

<http://html.conclase.net/w3c/html401-es/cover.html>

[23] Ortiz, Manuel. (2004). Aplicaciones de Bases de Datos Cliente Servidor. 55 Págs.

[24] World Wide Web Consortium. (2008). Guía Breve de CSS. Consultado el día 2 de Abril del 2010 de la World Wide Web:

<http://www.w3c.es/divulgacion/guiasbreves/hojasestilo>

[25] Ighside.net. (2007). CSS Sintaxis. Consultado el día 2 de Abril del 2010 de la World Wide Web:

<http://www.ighside.net/man/css/sintaxis.php>

[26] Alvarez, Miguel. Qué es JavaScript. Consultado el día 3 de Abril del 2010 de la World Wide Web:

<http://www.desarrolloweb.com/articulos/25.php>

[27] Instituto Tecnológico de Costa Rica. Introducción a JavaScript. Consultado el día 3 de Abril del 2010 de la World Wide Web:

<http://www.cidse.itcr.ac.cr/revistamate/HERRAmlnternet/GuionesEnJavascript/manualjs1.html>

[28] Santos, Herminio. (2001) ¿Qué es PHP? Consultado el día 4 de Abril del 2010 de la World Wide Web:

<http://www.maestrosdelweb.com/editorial/phpintro/>

[29] Enciclopedia Wikipedia. (2010). MySQL. Consultado el día 4 de Abril del 2010 de la World Wide Web:

<http://es.wikipedia.org/wiki/MySQL>

[30] Miguel Angel Alvarez. (2001). ¿Qué es PHP? Consultado el 16 de Noviembre del 2010 de la World Wide Web:

<http://www.desarrolloweb.com/articulos/392.php>

[31] Sitio Web de la Universidad Nacional de Colombia. Análisis y Diseño de Algoritmos. Consultado el 31 de Enero del 2011 de la World Wide Web:

<http://www.virtual.unal.edu.co/cursos/sedes/manizales/4060024/Lecciones/Capitulo%20I/problemas.htm>

[32] Sitio Web de Wikipedia. Sistema de Mensajería Multimedia. Consultado el 2 de Febrero del 2011 de la World Wide Web:

[http://es.wikipedia.org/wiki/Sistema\\_de\\_mensajer%C3%ADa\\_multimedia#El\\_MMSC\\_o\\_centro\\_de\\_mensajes\\_multimedia](http://es.wikipedia.org/wiki/Sistema_de_mensajer%C3%ADa_multimedia#El_MMSC_o_centro_de_mensajes_multimedia)

[33] RFC 2822. Internet Message Format. Consultado el 6 de Febrero del 2001 de la World Wide Web:

<http://www.ietf.org/rfc/rfc2822.txt>