

VISIÓN
DE CASO
Clasificación del artículo:
REFLEXIÓN

SOFTWARE PARA GESTIÓN Y ADMINISTRACIÓN DE IMÁGENES UTILIZANDO TECNOLOGÌA MULTIMEDIA GSM®

SOFTWARE FOR MANAGEMENT AND ADMINISTRATION OF IMAGES USING GSM® MULTIMEDIA TECHNOLOGY

Hermes Javier Eslava[1]
Edgar Javier Cruz López[2]
Juan Carlos Ramos Buitrago[3]

Resumen

Teniendo en cuenta los recursos e infraestructura en Telecomunicaciones de los operadores de telefonía móvil, el presente artículo muestra el desarrollo y la implementación de un sistema donde a través de mensajes multimedia integrado con una aplicación Web, en forma autorizada, se administran, clasifican y publican las imágenes enviadas desde un móvil. El enrutamiento se ha ejecutado por el operador de telefonía móvil (COMCEL®) hacia un servidor VAS[4]- Value Added Service, Servicios de valor agregado - donde se analizan los mensajes, se guardan en un servidor Web y luego se registran en una base de datos.

Palabras clave
Mensajería multimedia, mensajería corta, aplicación Web, metodologías de desarrollo de software, RUP (Rational Unified Process).

Summary
Having in mind the resources and infrastructure in telecommunications with which it counts the operators of mobile telephony, this article show the development and implemented of a system where trough of a multimedia messages, By means of the application Web different authorized users will be able to administer, to classify and to publish the images sent from its cellular one in a Web. The routed is for the operator of mobile telephony (COMCEL®) toward the VAS server, analyze the messages, the guard in a Web server and it registers them in a database.

Key Words
Messaging multimedia, short messaging, Web application, methodologies of software development, RUP (Rational Unified Process).

1 Licenciado en Electrónica de la Universidad Pedagógica Nacional. Msc. en Telecomunicaciones de la Universidad Nacional de Colombia. Docente de planta adscrito a la Facultad Tecnológica de la Universidad Distrital, hjeslavab@udistrital.edu.co

2 Tecnólogo en Electrónica de la Universidad Distrital Francisco José de Caldas

3 Tecnólogo en Electrónica de la Universidad Distrital Francisco José de Caldas

4 VAS, Value Added Service, Servicios de valor agregado

Introducción

La integración entre variadas tecnologías (MMS, SMS y Web) para la interacción del usuario con diferentes interfaces, por ejemplo las de teléfonos móviles y la Web, requiere de una cuidadosa investigación que parte del análisis entre las diferencias entre cada tecnología y cada interfaz, con el fin de establecer la forma más adecuada de presentar la información, en cada una de las entidades del sistema, de modo que se mantenga la integridad de dicha información, [14], [15].

Actualmente, los paradigmas de programación, [2],[4],[6], permiten que los datos que cada usuario envía, sean ordenados de tal manera que pueden presentarse a otros usuarios de una forma estructurada para facilitar la consulta de la información en una base de datos. Igualmente, los criterios de usabilidad del software, implican el diseño de interfaces amigables y atractivas, donde con elementos HTML como los de cualquier sitio Web[5], permitan al usuario un rápido acople con aplicaciones de distinta naturaleza. Esto es crítico en el caso que nos ocupa, pues deben ejecutarse procesos de edición y organización de material fotográfico y tiempos de entrega mínimos; asuntos que deben hacerse remotamente desde cualquier lugar donde se tenga acceso a Internet y cuya validación instantánea se realiza en base a confiables reportes que se generan automáticamente.

1. Descripcion del problema

Supongamos que un operador de telefonía móvil deba entregar reportes gráficos periódicamente acerca del estado de sus puntos de venta y de eventos realizados por sus distribuidores en todo el país. Normalmente, el envío de tales imágenes se realiza en CD, debido a que el tamaño de estas es demasiado grande para ser enviado por correo electrónico.

En primer lugar, cuando son sitios bastante alejados del centro del país el tiempo se vuelve un factor crítico para la edición y entrega del producto final, sin contar con que el material se expone a riesgos tales como pérdida o daños físicos.

En segundo lugar, el uso de aplicaciones[6] para envío de mensajes multimedia en general no se ajustan a las necesidades particulares de las compañías, y no se puede hacer ningún tipo de modificación o ajuste pues el código fuente es cerrado. Y en tercer lugar, la edición de las imágenes es un proceso largo y cuidadoso de clasificación -zona, ciudad, tipo de imagen, etc.- hecho por personal que trata las imágenes a tamaños estándar y optimizados para obtener una resolución aceptable.

2. Criterios de diseño

Basado en requerimientos de usuarios finales, se optó por una alternativa que se ajusta a las exigencias de la infraestructura de la red y su funcionalidad. Tales requerimientos, [10],[14], pueden definirse así:

- Permitir la recepción de mensajes multimedia provenientes desde el MMSC del operador de telefonía móvil, solo desde números autorizados administrables desde una interfaz Web.
- Almacenar ordenadamente la información para su posterior clasificación, el sistema debe enviar una notificación al usuario vía SMS indicando la correcta recepción del mensaje.

5 Los elementos normalmente utilizados por los sitios Web son listas desplegables, casillas de verificación, botones de selección, cajas de texto, iconos gráficos, textos de ayuda.

6 desarrollado por compañías como Netwin Software, Compañía neozelandesa desa-rrolladora de software. http://www.netwinsite.com

- Tener Servicio FTP de IIS, para realizar la transferencia de archivos de los mensajes a las carpetas de destino.
- Comunicación con MMSC y SMSC a nivel de protocolos de datos.
- Documentar la instalación y arquitectura.

La aplicación diseñada en 3 capas[7] para facilitar la ejecución de cambios y optimizar la velocidad de comunicación entre la aplicación y la base de datos, que cumple con estos requisitos por menor tiempo de diseño, desarrollo e instalación, bajo costo y facilidad de implementación, se desarrolló en SQL SERVER 2000® en un servidor remoto Web IIS (Internet Information Services) es compatible con el sistema Operativo Windows 2003 Server Standard Edition®, permite el almacenamiento de más de 10000 registros y archivos de imágenes, carga imágenes digitalizadas a la aplicación vía Web, de tal forma que el procesamiento posterior es igual al de los mensajes multimedia. Se utilizaron Lenguajes de Script libres como PHP 4.3.11, compatibles con plataformas Microsoft®; lenguajes de script para aplicaciones a nivel de protocolo (MMS y SMS) JAVA J2EE®; Servidor Web Tomcat 4.0 para ejecutar aplicación Web desarrollada en lenguaje JAVA®.

3. Metodología, desarrollo e implementación de la aplicación

La metodología de desarrollo de software que se utilizó es la RUP (Rational Unified Process) que se caracteriza por fomentar las buenas prácticas pero en especial la de desarrollo iterativo que consiste en organizar una serie de mini-proyectos cortos de duración fija denominados iteraciones. El resultado de cada una de estas iteraciones es un sistema que puede ser probado, integrado y ejecutado. Cada iteración posee sus etapas de análisis de requisitos, diseño, implementación y pruebas.

El ciclo de vida iterativo se basa en el mejoramiento de los sistemas por medio de múltiples iteraciones. De esta manera el sistema crece incrementalmente a lo largo del desarrollo, por esta razón este enfoque es también llamado desarrollo iterativo e incremental.

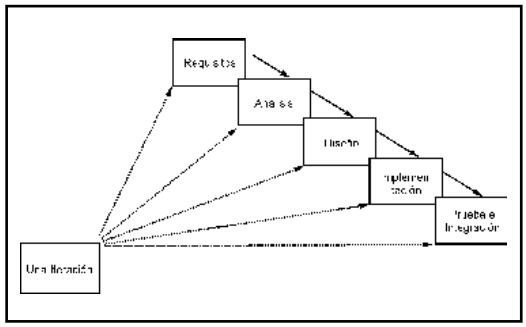
Basados en la metodología RUP y debido a las cuatro diferentes tecnologías involucradas en los requerimientos, se ha dividido el proyecto en los siguientes mini-proyectos o iteraciones:

- Iteración base de datos. Es la base de todo el proyecto ya que el buen funcionamiento de las demás Iteraciones depende del buen diseño y funcionamiento de esta.
- Iteración MMS. Comprende la etapa de comunicación, recepción y almacenamiento de mensajes enviados vía multimedia por los usuarios de la aplicación
- Iteración SMS. Es el proceso que envía los mensajes de respuesta vía SMS a los usuarios cuando estos envían un Mensaje MMS.
- Iteración Aplicación Web. Es la encargada de la administración de los mensajes y la información. Incluye el manejo para cada perfil de usuario y la carga de archivos vía Web.

Cada iteración se implementa por medio de una cascada realizando las etapas de Requisitos, Análisis, Diseño, Implemen-

7 Las tres capas son Interfaz de usuario, modelo de objetos y modelo de datos.

Figura 1. Iteración RUP [16]



tación y Pruebas, al finalizar se evalúan los resultados del proceso y se comparan con los de otras iteraciones para corregir posibles errores.

A. Iteración base de datos

La base de datos es el lugar en donde se almacenan todos los datos necesarios para atender a las necesidades de todos los usuarios de forma directa. Los datos pueden ser una combinación de voz, imágenes, texto y números. La base de datos se considera desde dos puntos de vista, el físico y el lógico. La base de datos física está compuesta de los medios de almacenamiento, como las cintas o discos. Y la Lógica atiende la búsqueda, asociación y recuperación de los datos almacenados para satisfacer necesidades específicas de información. También tiene que ver con el componente de software del sistema que incluye técnicas lógicas y asociativas de datos como índices, directorios, listas, llaves, apuntadores, redes, árboles y relaciones [1].

Siguiendo el modelo entidad-relación, la Base de datos tiene la facilidad de recuperar los datos sin importar donde se encuentren y evita la duplicidad de la información que aumenta el espacio de almacenamiento, factor crítico en bases de datos que contienen demasiada información. Este sistema relacional, conocido por ser un estándar en el mercado, solventa los problemas presentados por otros sistemas[8] en operaciones comerciales [12].

B. Iteración de mensajería multimedia mms

"La definición del servicio MMS esta siendo liderada por el proyecto de la asociación 3GGP, y las especificaciones elaboradas hasta la fecha[9] proponen un servicio que no tiene equivalencia directa con su predecesor en las redes de telecomunicaciones fijas, con el correo electrónico de Internet, ni con el servicio de mensajes cortos de texto. Las principales y más novedosas características que introduce la mensajería móvil multimedia son" [15]:

- Permite el envío a dispositivos móviles de presentaciones audiovisuales sincronizadas formadas por una o varias páginas con objetos multimedia que pueden ser Imágenes, sonidos, texto con formato (estilo, tamaño o color) y Video[10].
- Utiliza el lenguaje SMIL[11] para la integración de los objetos multimedia en la presentación sincronizada.
- Permite proporcionar un nivel de servicio personalizado a cada usuario dependiendo del terminal.
- Los servidores tienen la capacidad de almacenar los mensajes y enviar la notificación al teléfono móvil para que sea descargado.

1. Arquitectura Mensajería Multimedia

Los servidores de aplicaciones MMS que proporcionan servicios de valor agregado (VAS) a los usuarios se conectan al MMSC

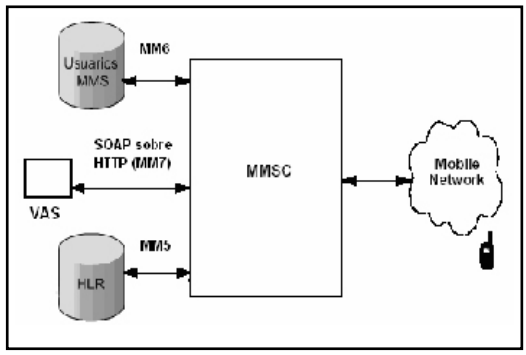
8 Otros sistemas de bases de datos son jerárquico y en red, los cuales utilizan punteros, pero presentan problemas para recuperar datos basados en otras interrelaciones que no estén definidas antes de que el sistema se ponga en marcha.

9 Consulta de información más detallada sobre el estándar en www.3gpp.org

10 MMS es compatible con formatos JPEG, GIF 87a, GIF 89a, WBMP y PNG, AMR, MPEG-4, AAC, SP-MIDI, y H.263

11 W3C: W3C Recommendation Synchronized Multimedia Integration Language (SMIL 1.0). 15 de junio 1998, http://www.w3.org/TR/SMIL/

Figura 3. Comunicación MM7



a través del punto de referencia (MM7) de la arquitectura MMS.

Debido a que este proyecto utiliza un servidor externo de servicios de valor agregado esta iteración consiste en implementar la interconexión con el MMSC utilizando el estándar MM7.

2. Estructura de protocolo MM7

MM7 es un protocolo que establece la comunicación entre el centro de servicios multimedia MMSC y el servidor de servicios de valor agregado VAS. La comunicación entre el centro de mensajería multimedia y el servidor de servicios de valor agregado se hace utilizando el protocolo SOAP sobre HTTP. SOAP (Simple Object Access Protocol) es un protocolo que permite la comunicación entre aplicaciones programadas en cualquier lenguaje y que se ejecuten en cualquier plataforma. Dicha comunicación se establece a través de mensajes que están codificados en lenguaje XML para poder ser leídos fácilmente.

3. Protocolo SOAP

SOAP define la codificación XML de los mensajes MM7. La estructura básica del documento para los mensajes MM7 en formato SOAP contiene dos partes:

- SOAP envelope. Sección obligatoria del nivel superior que contiene datos XML relacionados con el mensaje. El SOAP envelope contiene: Header, body, fault.
- SOAP attachment. Es una sección opcional del nivel superior que por lo general contiene información codificado en base64[12] con estructura de mensaje Multipart MIME[13].

4. Protocolo de transporte http

El protocolo MM7 requiere que los mensajes SOAP de solicitud sean transferidos como HTTP POST[14] y los mensajes MM7 de respuesta, transferidos como mensajes HTTP de respuesta. Los headers para mensajes MM7 transferidos por medio de HTTP POST son:

- Content-Type. Especifica el tipo de contenido del mensaje http. Si el mensaje SOAP lleva datos adjuntos el valor es multipart/related, sino el valor es text/XML.
- SOAPAction. Pone una cadena vacía en el mensaje MM7 indicando que el HTTP Request-URI[15] (Servidor de origen) es quien proporciona el intento del mensaje SOAP.

Para los mensajes de respuesta HTTP el header generalmente solo contiene el campo Content-Type: text/XML. A continuación se muestran las estructuras para los mensajes MM7.

5. Recepción de Mensajes

Para realizar envíos de mensajes multimedia entre el MMSC y un VAS, se debe crear

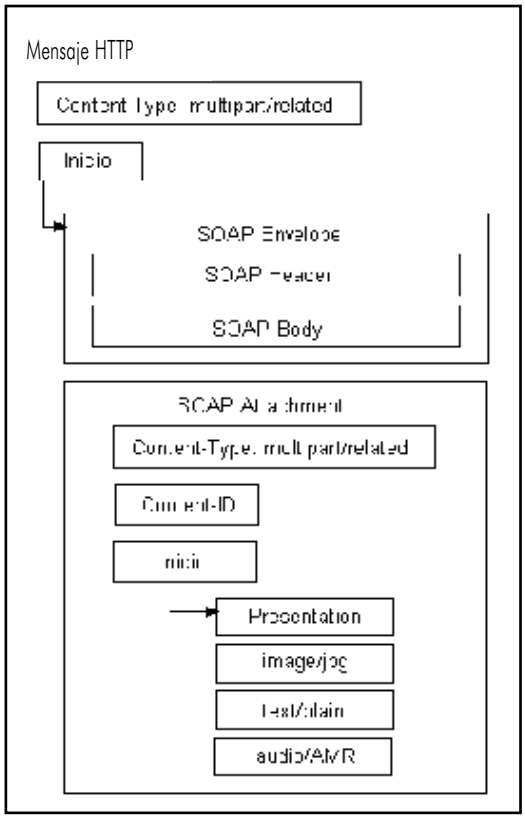
12 Base 64 es un método de codificación donde se convierten los datos binarios a caracteres ASCII

13 MIME (Multipurpose Internet Mail Extensions), es un estándar que permite enviar cualquier tipo de archivo adjunto en un mensaje de texto a través de Internet.

14 HTTP POST es un método del protocolo que se usa para hacer peticiones a un servidor de destino el cual acepta el contenido de la petición y lo procesa.

15 URI es un texto corto que identifica un recurso específico. Es diferente de URL.

Figura 4. Estructura de mensaje MM7 con adjuntos [10]



una cuenta y configurarla con la información del servidor VAS en la configuración del gateway MMS/SMS[16] del operador. Allí mismo debe configurarse el número corto[17] al cual deben ser enviados los mensajes para que el MMSC se encargue de enrutarlos hacia el VAS previamente configurado. En la aplicación VAS se configuran los datos de la cuenta (código corto, IP gateway, puerto, usuario y contraseña) para poder establecer la conexión a través de HTTP POST. Una vez esta terminada la configuración en las dos partes se debe generar un HTTP POST de contenido desde el VAS hacia el gateway MMS para que este reconozca que dicho VAS propone una solicitud en formato MM7. Una vez se establece la conexión entre los dos puntos el servidor VAS queda escuchando constantemente las solicitudes realizadas por el MMSC en la IP y puerto especificados en la cuenta. El MMSC reenviará los mensajes que lleguen al número corto hacia el servidor VAS. Cuando el VAS recibe un mensaje confirma la correcta recepción de este al MMSC.

6. Almacenamiento de mensajes

Una vez se recibe el mensaje correctamente la aplicación analiza el mensaje y lo descompone en número de origen, asunto, texto e imagen. Si el mensaje no trae ningún archivo adjunto o si no contiene una imagen adjunta se le notificará al usuario vía SMS que el mensaje no se ha guardado correctamente. Si el formato del mensaje es correcto se busca el número de origen en la base de datos de usuarios para validar que el número del remitente tenga permiso para guardar mensajes. Si el número no es válido se notificará al usuario que no esta registrado para enviar mensajes. Si el número es válido se procede a generar el registro en la base de datos para el mensaje y los archivos adjuntos.

Para copiar el archivo en la ruta final donde va a ser consultado por la aplicación Web se renombra agregando el identificador registrado en la base de datos para este y de esta manera asegurar que no haya dos archivos con el mismo nombre. El archivo también se redimensiona para ajustarlo al tamaño estándar para el sitio Web en caso de superar las dimensiones (si las dimensiones son menores no se redimensiona) se crea una copia de la imagen en tamaño más pequeño para mostrarla como vista preliminar y se envían vía FTP a las carpetas de destino previamente configuradas en la aplicación.

16 El gateway MMS/SMS es la puerta de enlace del MMSC.

17 Es el número al que los usuarios deben enviar los mensajes.

C . Iteración de mensajería corta SMS

El servicio de mensajería corta SMS es utilizado para enviar mensajes de texto de máximo 160 caracteres a través de una red de telefonía celular GSM. Estos mensajes pueden ser enviados de un teléfono celular a otro o desde aplicaciones en servidores externos a la red de telefonía móvil llamadas ESME (External Short Messaging Entity). Los usuarios de la red celular pueden recibir mensajes desde varios ESMEs, La infraestructura de red de la mensajería corta se basa en un SMSC (Short Message Service Center) el cual almacena los mensajes y los reenvía a los usuarios móviles. La comunicación entre el SMSC y el ESME se realiza a través del protocolo SMPP (Short Message Peer to Peer Protocol).

1. Protocolo SMPP

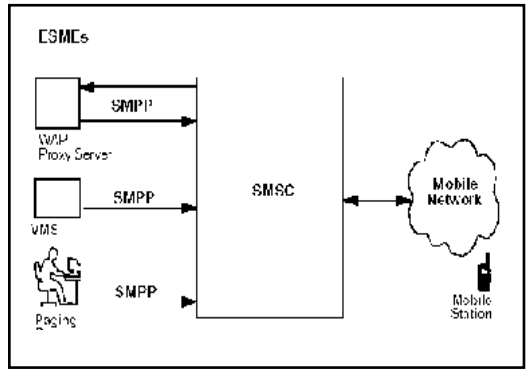
SMPP es un protocolo abierto diseñado para proporcionar una inferfase de comunicación de datos flexible para la transferencia de mensajes cortos. La conexión entre las dos entidades se puede establecer en redes TCP/IP o X.25 permitiendo el envío o recepción de mensajes hacia o desde el SMSC. El ESME también puede consultar, cancelar o reemplazar mensajes cortos utilizando SMPP. Este protocolo soporta una buena cantidad de características de funciones de mensajería en dos vías tales como [14]:

- Enviar mensajes a uno o varios destinatarios a través del SMSC.
- Un ESME puede recibir mensajes desde otros ESME a través el SMSC.
- Consultar el estado de los mensajes almacenados en el SMSC.
- Cancelar o reemplazar mensajes almacenados en el SMSC.
- Programar el envío de mensajes con fecha y hora.
- Seleccionar el modo del mensaje
- Establecer prioridades para el envío de mensajes.
- Definir el tipo de codificación de los mensajes.
- Ponerle un periodo de validez al mensaje.
- Asociar un tipo de servicio a cada mensaje.

La estructura de comunicación entre un SMSC y un ESME se muestra a continuación:

El protocolo SMPP esta basado en el intercambio de unidades de datos de protocolo (PDU) de solicitudes y respuestas entre el SMSC y el ESME. Siempre una operación SMPP debe constar de una solicitud PDU asociada a una respuesta PDU. La única excepción es el PDU alert_notification, el cual no se responde. El intercambio de mensajes puede dividirse en tres grupos de transacciones de la siguiente manera [14]:

Figura 4. Estructura de mensaje MM7 con adjuntos [10]



- Mensajes enviados desde el ESME (Transmitter) hacia el SMSC.
- Mensajes enviados desde el SMSC hacia el ESME (Receiver)
- Mensajes enviados desde el ESME (Transceiver) hacia el SMSC y viceversa.

2. Sesión SMPP

Una sesión entre el ESME y el SMSC es iniciada por el ESME la primera vez que establece una conexión de red con el SMSC emitiendo un SMPP Bind Request para abrir la sesión SMPP. Durante una sesión SMPP el ESME puede emitir una serie de solicitudes al SMSC y debe recibir las respectivas respuestas para cada solicitud desde el SMSC. De la misma manera el SMSC puede realizar solicitudes al ESME y este debe responder [14].

3. Envío de mensajes cortos

Para realizar envíos de mensajes cortos SMS entre un ESME y el SMSC, se debe crear una cuenta y configurarla con la información del ESME en la configuración del gateway MMS/SMS del operador. Allí mismo debe configurarse el número corto a través el cual se enviarán las notificaciones. Esta configuración puede permitir envío de SMS, recepción de SMS o ambas. Para este caso la configuración solo permite el envío de mensajes desde el ESME hacia el SMSC debido a que solo se utiliza para informar a los usuarios de la correcta o fallida recepción de los mensajes multimedia.

En el ESME se configuran los datos de la cuenta (código corto, IP gateway, puerto, usuario y contraseña) para poder iniciar una sesión de tipo ESME transmitter con el SMSC. Adicional a esta configuración se debe configurar los parámetros de conexión de la base de datos donde se van a colocar los mensajes a enviar.

El ESME ejecuta un proceso que abre la sesión con el SMSC. Dicho proceso esta monitoreando constantemente la base de datos para verificar si hay mensajes nuevos para ser enviados. Cuando hay un mensaje nuevo este toma el número del destinatario y el mensaje a ser enviado enviándolo a través de la sesión en modo transmitter establecida con el SMSC.

D. Iteración aplicación web

En el diseño y desarrollo de una aplicación se manejan tres etapas: conceptual, lógica y física. La implementación de estas tres etapas garantiza la calidad, flexibilidad, adaptabilidad, mantenimiento y desempeño de la aplicación.

1. Diseño conceptual

En la etapa de diseño conceptual se analizan las actividades que realizan los usuarios por medio de diagramas de casos de uso, que describen detalladamente las actividades de cada usuario.

Una vez se tienen los caso de uso se procede a pasar a la etapa de diseño lógico, traduciendo los casos de uso del diseño conceptual en el conjunto de objetos que conforman el sistema.

2. Diseño Lógico

El diseño lógico comprende las siguientes tareas [17]:

- Identificar y definir los objetos de la aplicación y sus características.

- Definir las interfases.
- Identificar las dependencias entre objetos.
- Validar contra los escenarios de uso.
- Comparar con la arquitectura de la empresa.
- Revisar y refinar tanto como sea necesario.

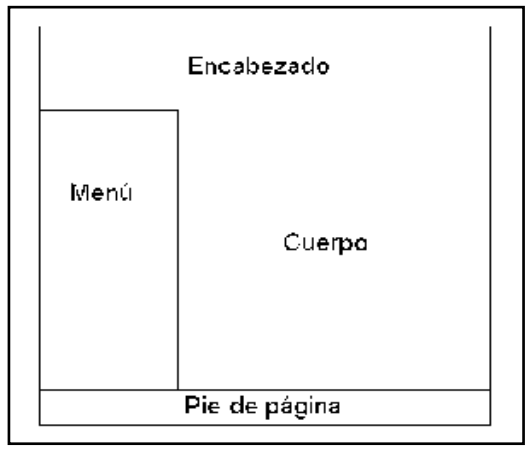
Para definir los objetos del sistema y sus características se puede emplear la técnica sujeto-verbo-objeto directo [17], y, luego se definen las interfases encargadas de permitir la comunicación con los objetos.
El diseño lógico se divide en tres niveles o capas de servicios:

- Nivel de servicios de usuario o capa de interfaz de usuario. Son interfases graficas o no visuales para facilitar el manejo de la información.
- Nivel de servicios de negocio o capa de objetos. Se encarga de Transformar y validar la información desde y hacia la capa de datos.
- Nivel de servicio de datos o capa de acceso a datos. Son los servicios de bajo nivel que apoyan los servicios de negocio y administra directamente los datos.

3. Diseño físico

Esta basado en componentes, en general reutilizables, scripts o librerías, con funciones muy específicas. Por ejemplo se puede tener una función que abra una nueva ventana del navegador. Dicha función se puede reutilizar si se maneja un parámetro donde pueda definir que URL puede abrir en la nueva ventana.

Figura 1. Estructura por bloques el sitio



Los componentes de clases, plantillas e includes (archivos que realizan una funcionalidad específica dentro de una plantilla general) son ejecutados por el servidor y devueltos al usuario a través de la interfaz gráfica. Los scripts de navegador (Javascript) y hojas de estilos son ejecutados en el lado del cliente y son utilizados por todas las secciones que conforman el sitio. La interfaz de usuario interpreta estos componentes y muestra al usuario la información de tal forma que lo que es procesado en el servidor se devuelve en lenguaje HTML, siendo la ejecución de estas tareas transparente para él.

La estructura por bloques de plantillas del sitio esta conformada por cuatro partes: Encabezado, Menú, Cuerpo y Pie de página.

El acceso al sitio esta restringido por un usuario y una contraseña. Una vez el usuario ingresa sus datos la aplicación los compara con la información existente en la base de datos. Si los datos son correctos la aplicación abre una sesión para el usuario donde se crean tres variables identificador de usuario, nombre y perfil.

En la capa de modelo de objetos de negocio se encuentran los scripts con cada una de las clases diseñadas en la fase de elaboración. Cada método de las clases es invocado desde las plantillas para devolver un resultado específico. Los atributos de las clases son los parámetros requeridos para que estas devuelvan la información solicitada correctamente.

Las clases de la aplicación se comunican con las tablas de la base de datos a través procedimientos almacenados o store procedures. Dichos procedimientos son scripts realizados en lenguaje SQL y se encuentran guardados directamente en la base de datos. Los procedimientos almacenados optimizan las consultas con las bases de datos ya que la aplicación solo debe enviar los parámetros necesarios para su ejecución, evitando enviar consultas muy largas que podrían tardar más tiempo en ser recibidas cuando el servidor se encuentra en otra red.

Los procedimientos almacenados pueden contener varias consultas. Cada consulta es solicitada enviando un parámetro de acción (p.e. Consultar Zonas). Estas consultas son llamadas funciones y pueden ser escritas con códigos condicionales para ser ejecutadas. Los procedimientos almacenados pueden crear y definir sus propios parámetros sin necesidad de ser enviados desde la aplicación.

4. RESULTADOS

A continuación se muestra el resultado final de las principales pantallas de la aplicación.

Bandeja de entrada. Cuando los usuarios representantes o gerentes de ventas ingresan a la aplicación lo primero que ven es la bandeja de entrada done se muestran las imágenes que han cargado vía MMS o vía Web.



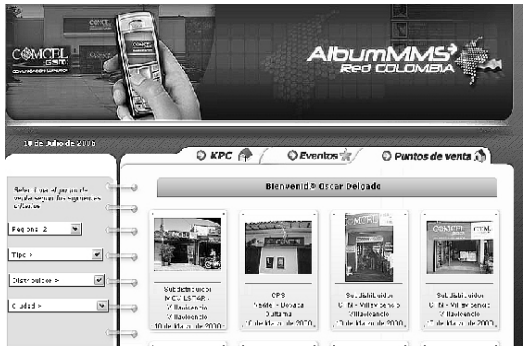
Menú. Muestra al usuario las opciones a las cuales tiene acceso dependiendo del perfil. Es mostrado en la parte izquierda de la pantalla.



Ventana emergente de información de la fotografía. Esta venta se muestra una vez se hace clic en la fotografía que se muestra en la bandeja e entrada.



Álbum final. Esta es la información que se muestra al usuario final una vez entra al álbum.



Conclusiones

- Las metodologías de desarrollo de software brindan una gran base para la construcción de aplicaciones de software de pequeño, mediano y gran tamaño ya que brindan las herramientas necesarias para la correcta planeación, desarrollo e implementación.
- Debido a que la mensajería multimedia permite no solo enviar imágenes sino también archivos de sonido, texto y presentaciones, el desarrollo de esta nueva aplicación utilizando el centro de mensajería multimedia abre el camino para realizar más aplicaciones similares que permiten la interacción del usuario con un sitio Web y su teléfono móvil. Entre estas se pueden destacar concursos de fotografía, registro fotográfico de usuarios en eventos para su posterior identificación, almacenamiento de archivos en un servidor remoto, transferencia de archivos entre móviles pasándolos a través de un servidor conectado a Internet, administración de archivos enviados vía MMS a servidor Web por medio de una aplicación Web, entre otros.

Referencias bibliográficas

[1] Burch, J & Grudnitski, G. (1992). Diseño y sistemas de información. Balderas – México. GRUPO NORIEGA EDITORES.

[2] Canós, J, Letelier, P & Penadés, M. (2005) Metodologías Ágiles en el Desarrollo de Software. Disponibilidad en Internet: < http://www.willydev.net/descargas/prev/TodoAgil.Pdf >

[3] Fernández Vilas, A. (2001). Introducción a UML. Disponibilidad en Internet: < http://www-gris.det.uvigo.es/~avilas/UML/node7.html >

[4] Fernández, E. & Medina, P. (2006). Metodologías de desarrollo de software. Disponibilidad en Internet: < http://alarcos.inf-cr.uclm.es/doc/ISOFTWAREI/Tema04.pdf>.

[5] Ferré Grau, X & Sánchez Segura, M. (2004). Desarrollo Orientado a Objetos con UML. Madrid. Disponibilidad en Internet: < http://www.clikear.com/manuales/uml/ >

[6] Giraldo, J. Ingeniería de software. Una aproximación a la medición de la calidad. Disponibilidad en Internet: < http://www.monografias.com/trabajos15/ingenieria-software/ingenieria-software.shtml >

[7] Larman, C. (2003). UML Y PATRONES. Una introducción al análisis y diseño orientado a objetos y al proceso unificado (2ª ED, pp. 13 -26). Madrid. PRENTICE HALL.

[8] Microsoft. (2005). Funding and Development of Internal Microsoft Business Applications. . Disponibilidad en Internet: < http://www.microsoft.com/technet/itshowcase/content/msbusappstsb.mspx >

[9] Marqués, M. (2001). Apuntes de ficheros y Bases de datos. Disponibilidad en Internet: < http://www3.uji.es/~mmarques/f47/apun/apun.pdf >

[10] Open Wave Systems. (2003). Openwave MMS Library. Disponibilidad en Internet < http://developer.openwave.com/omdtdocs/mms_sdk/mmsc_mm7_dev.pdf >

[11] Reynoso, C. (2004). Métodos Heterodoxos en Desarrollo de Software. Versión 1.0. Disponibilidad en Internet: <http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/heterodox.asp >

[12] Norick & Ro. (2003). aprenda bases de datos con MS SQL SERVER 2000. Disponibilidad en Internet: < http://usuarios.lycos.es/cursosgbd/ >

[13] Romero, P. (2004). Análisis y diseño orientado a objetos. Disponibilidad en Internet: < http://www.itlalaguna.edu.mx/academico/carreras/sistemas/Analisis%20y%20dise%F1o%20orientado%20a%20objetos/Resumen3.pdf >

[14] SMPP Developers Forum. (1999). Short Message Peer to Peer Protocol Specification v3.4. Disponibilidad en Internet < http://smsforum.net/doc/download.php?id=smppv50 >

[15] Telefónica. (2003). Las Telecomunicaciones Multimedia. (pp. 77 - 241). Disponibilidad en Internet < http://www.telefonica.es/sociedaddelainformacion/pdf/publicaciones/telecommultimedia/teleco_mm.pdf >

[16] Universidad Politécnica de Valencia, Departamento de Sistemas Informáticos y Computación. (2006). Rational Unified Process (RUP). Disponibilidad en Internet: < https://pid.dsic.upv.es/C1/Material/Documentos%20Disponibles/Introducción%20a%20RUP.doc >

[17] Zavala R. 2000. Diseño de un Sistema de Información Geográfica sobre Internet. Tesis de Maestría en Ciencias de la Computación. Universidad Autónoma Metropolitana-Azcapotzalco. México, D.F. En prensa. Disponibilidad en Internet < http://www.angelfire.com/scifi/jzavalar/apuntes/IngSoftware.htm >