



**Universidad de las Ciencias
Informáticas**

Facultad 2

**Título: Desarrollo del Portal WAP para la plataforma de gestión de
contenidos Gina**

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

Autores: Harley Cruz Cosío.

Dianelys del Valle Roques.

Tutor: Ing. Darien J. Alvarez de la Cruz.

Ciudad de La Habana, 24 de Junio de 2009

“Año 50 del triunfo de la Revolución”



“No hay más que asomarse a las puertas de la tecnología y la ciencia contemporánea para preguntarnos si es posible vivir y conocer ese mundo futuro sin un enorme caudal de preparación y conocimientos”.

Fidel Castro Ruz

Declaración de autoría

Declaramos ser los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Dianelys del Valle Roques

Harley Cruz Cosio

Ing. Darien J. Alvarez de la Cruz

Firma del autor

Firma del autor

Firma del tutor

Datos de Contacto

Nombre: Darien Jesús Alvarez de la Cruz.

Correo: dalvarez@uci.cu

Título: Ing. en Ciencias Informáticas.

Centro de Estudios: Universidad de Ciencias Informáticas. Ciudad de la Habana. Cuba.

Graduado: Julio del 2007.

Honores: Título de Oro.

Experiencia profesional:

Comenzó en su etapa de estudiante, donde trabajaba como miembro de la Empresa ProcyonSoluciones <http://www.procyonsoluciones.com> especializada en el desarrollo de software para Dispositivos Móviles. En esta empresa trabajó 3 años consecutivos.

Al obtener el título de Ingeniero en Ciencias Informáticas <http://www.uci.cu> es solicitado por su Universidad para que haga el servicio social como profesor de esa institución, en esta labor ha adquirido 2 años de experiencia, adquiriendo el grado de Profesor Instructor.

Proyectos en los que ha participado:

- Portal Wap para la entidad Cubacel.
- Implementación de la especificación JSR-212 de Java.
- Plataforma Blue Eye de ProcyonSoluciones.
- Servicio de Horóscopo mediante MMS.
- Sistema de Envío Masivo de Mensajes SMS.
- Sistema de alertas mediante mensajes SMS.

Postgrados:

- Técnicas de Dirección.
- Metodología de la Investigación Científica.
- Organización de la Investigación Científica desde la Producción.
- Docencia e Innovación Universitaria.
- Economía y Política en la Construcción Socialista.

- Negociación.
- Ciencia Tecnología y Sociedad.

Eventos y Publicaciones:

- Universidad 2008.
- Cittel 2008.
- Fórum de Ciencia y Técnica 2007, 2008 y 2009.
- Informática 2009.
- Publicación en la Revista Cubana de Ciencias Informáticas.

Agradecimientos

A la Revolución Cubana, a Nuestro Comandante en Jefe. A mis maestros que a lo largo de mi vida han inculcado en mí, el deseo y necesidad de estudiar. A mis seres más queridos, a mi compañero de tesis, tutor, todos lo que me ayudaron a que fuera posible cumplir este sueño.

Dianelys

A mi familia, en especial mis abuelos y mi madre, quienes me comunicaron la ética y el rigor que guían mi transitar por la vida. A mis compañeros de estudio por confiar en mi, a mi tutor por su asesoramiento científico y estímulo para seguir creciendo intelectualmente. Gracias al apoyo que me han dado, he llegado a realizar uno de los anhelos más grandes de la vida, fruto del inmenso apoyo y confianza que en mí se depositó y con los cuales he logrado terminar mis estudios profesionales, que constituyen el legado más grande que pudiera recibir y por lo cual les viviré eternamente agradecido.

Harley

Dedicatoria

Le dedico este trabajo y todo mi esfuerzo a las personas que más amo y estimo mi mamá por complacerme en todo, a mi papá por ser tan igual a mi, a mi hermanito por ser mi orgullo, a mis abuelas que siempre me han apoyado en todo, a mis dos familias, amigos los cuales siempre te hacen ver la verdad, muy especial a mi novio por ser apoyo en todo momento y la persona más especial que he conocido.

Dianelys

A mi abuela y mi madre, por darme la vida, por su cariño y ejemplo.

A mis maestros por compartir generosamente sus conocimientos.

A Kenia, por su apoyo y comprensión.

A la UCI por brindarme todas las facilidades a lo largo de estos años de formación académica.

A mis compañeros de estudio, por brindarme su amistad y apoyo moral.

Harley

Resumen

El avance en la telefonía móvil ha permitido que los teléfonos se utilicen no solo para comunicarse mediante llamadas telefónicas, o mensajes de texto sino que ha propiciado la navegación por internet utilizando el protocolo WAP (del inglés: Wireless Application Protocol). Esto permite a través de un browser la visualización de las páginas de los portales web para móviles.

En Cuba varias entidades desarrollan aplicaciones para celulares, pero no existe ningún sitio web que brinde servicios a través de la red de telefonía inalámbrica. Por lo que el presente trabajo está dirigido principalmente a la creación de un sitio web que sea accesible por la mayoría de los dispositivos móviles.

El mismo está pensado para brindar servicios de descarga de contenidos digitales, información referente al estado del tiempo, noticias, anuncios y realizar búsquedas y consultas sobre el directorio telefónico de páginas amarillas de la Empresa de Telecomunicaciones de Cuba S.A (ETECSA).

Este documento presenta las principales características de algunas de las herramientas para realizar sitios web para móviles. Además de poner de manifiesto las mejores prácticas y soluciones en este sentido.

Palabras Claves: celulares, WAP, portal WAP, MyMobileWeb, sitio web, dispositivos móviles, ETECSA.

Abstract

The development of wireless telecommunication has enabled cellular phones are used to communicate through phone calls, text messages and has internet browsing by using the Wireless Application Protocol. This allows using a browser to display pages of mobile websites.

In Cuba, some entities develop applications for mobile phones, but in there is no website that offers services through the wireless network. Therefore, this research is focused at creating a website that is accessible by most mobile devices.

It is designed to provide download services for digital content, information concerning the weather, news, announcements, search and consultations on the telephone directory of yellow pages from Empresa de Telecomunicaciones de Cuba S.A (ETECSA).

This document present the main characteristics of some tools for develop mobile websites. In addition, to highlighting best practices and solutions in this regard.

Keywords: cell phones, Wireless Application Protocol, WAP portal, MyMobileWeb, website, mobile devices, ETECSA.

Índice

Introducción	1
Capítulo 1 Fundamentación Teórica	4
1.1 Introducción	4
1.2 Telefonía Móvil.....	4
1.3 Arquitectura de la tecnología WAP	6
1.4 Tecnologías y tendencias en el desarrollo de portales WAP	10
1.4.1 Alternativas adoptadas.....	13
1.4.2 Repositorios de Descripción de Dispositivos	17
1.4.3 Soluciones en la adaptación de contenidos Web	20
1.4.4 Herramientas para el desarrollo de portales WAP	21
1.5 Metodología de desarrollo.....	26
1.6 Herramienta CASE	27
1.7 Plataforma de Desarrollo	29
1.8 Entorno de desarrollo.....	30
1.9 Herramienta para el Control de Versiones.....	30
1.10 Conclusiones del Capítulo	31
Capítulo 2 Características del Sistema	32
2.1 Introducción	32
2.2 Objetivos Estratégicos	32
2.3 Modelo de Dominio	34
2.3.1 Conceptos del Modelo de Dominio	34
2.4 Propuesta de Sistema.....	35
2.5 Requerimientos del Sistema	38
2.5.1 Requerimientos Funcionales.....	38
2.5.2 Requerimientos no Funcionales.....	39
2.6 Modelo del Sistema	42
2.6.1 Definición de los actores del sistema.....	42
2.6.2 Diagrama de Casos de Uso del sistema.....	43
2.6.3 Descripción Detallada de los Casos de Uso del Sistema	45
2.7 Conclusiones del Capítulo	53
Capítulo 3 Análisis y Diseño del Sistema	54
3.1 Introducción	54

3.2 Modelo de Análisis	54
3.2.1 Diagrama de Clases de Análisis	55
3.3 Modelo de Diseño	55
3.3.1 Diagrama de Clases del Diseño.....	56
3.4 Patrones de Diseño	57
3.6 Prototipos de Interfaz	60
3.7 Conclusiones del Capítulo	62
Capítulo 4 Implementación y Prueba del Sistema	63
4.1 Introducción	63
4.2 Modelo de Implementación	63
4.2.1 Diagrama de Componentes	63
4.2.2 Diagrama de Despliegue.....	68
4.3 Modelo de Pruebas	68
4.3.1 Métodos de Prueba	68
4.3.2 Diseño de Casos de Prueba	69
4.4 Conclusiones del Capítulo	75
Capítulo 5 Estudio de Factibilidad	76
5.1 Introducción	76
5.2 Estimación basada en Puntos de Casos de Uso	76
5.2.1 Cálculo de los Puntos de casos de uso sin ajustar	77
5.2.2 Cálculo de los puntos de Casos de Uso ajustados	78
5.2.3 Estimación de esfuerzo	79
Estimación del esfuerzo total del proyecto.....	79
5.3 Beneficios tangibles e intangibles	80
5.4 Análisis de costos y beneficios	81
5.5 Conclusiones del Capítulo	82
Conclusiones Generales.....	83
Recomendaciones	84
Bibliografía	85
Referencias Bibliográficas	87
Anexos	88
Glosario de Términos.....	118

Índice de Figuras

Figura 1 Arquitectura del Protocolo WAP.	6
Figura 2 Funcionamiento de WAP.	8
Figura 3 WAP 2.0 soporta dos pilas de protocolos.	9
Figura 4 Cantidad de Usuarios por Tecnología.	10
Figura 5 Comparativa de Software en ambientes móviles.	16
Figura 6 Proceso Unificado de Desarrollo de Software.	27
Figura 7 Modelo de Dominio.	34
Figura 8 Arquitectura del portal WAP.	36
Tabla 1 Definición de los Actores del Sistema.	42
Figura 9 Diagrama de Casos de Uso del Sistema.	44
Figura 10 Diagrama de Clases de Análisis CU_MostrarContenidoParaDescargar.	55
Figura 11 Diagrama de Clases de DiseñoWeb CU_MostrarContenidoParaDescargar.	56
Figura 12 Diagrama de Clases de DiseñoWeb CU_MostrarDirectorioTelefonico.	57
Figura 13 Presentación con el Logo.	61
Figura 14 Presentación de Búsqueda.	61
Figura 16 Menú de Descargas.	61
Figura 17 Directorio Telefónico.	61
Figura 18 Descargar Contenido.	61
Figura 19 Mapa de Navegación del Portal WAP.	62
Figura 20 Diagrama de Componentes.	64
Figura 21 DiagramaComponentes CU_MostrarDirectorioTelefonico.	64
Figura 22 DiagramaComponentes CU_MostrarPromociones.	65
Figura 23 DiagramaComponentes CU_MostrarContenidoParaDescargar.	65
Figura 24 DiagramaComponentes CU_MostrarNoticias.	66
Figura 25 DiagramaComponentes CU_MostrarPronosticoTiempo.	66
Figura 26 DiagramaComponentes PaqueteComponenteWS_Client.	67
Figura 27 Diagrama de Despliegue.	68
Figura 31 Resultado de la prueba con ready.mobi.	74
Figura 32 Pruebas realizadas.	75
Figura 33 Mapa de Cobertura GSM en Cuba 2008.	88
Figura 34 Cobertura Celular en Cuba 2007.	88
Figura 35 Solución brindada por Volantis Systems.	89

Figura 36 Estructura utilizada por WURFL.	90
Figura 37 Diagrama de clases de análisis CU_MostrarDirectorioTelefonico.....	98
Figura 38 Diagrama de clases de análisis CU_MostrarPromociones.....	99
Figura 39 Diagrama de clases de análisis CU_MostrarNoticias.....	100
Figura 40 Diagrama de clases de análisis CU_MostrarPronosticoT.	101
Figura 41 Diagrama de secuencia de análisis CU_MostrarContenidoDescarga.....	102
Figura 43 Diagrama de secuencia de análisis CU_MostrarDirectorioTelefonico-Alternativa1..	104
Figura 44 Diagrama de secuencia de análisis CU_MostrarDirectorioTelefonico-Alternativa2..	105
Figura 45 Diagrama de secuencia de análisis CU_MostrarNoticias.....	106
Figura 46 Diagrama de secuencia de análisis CU_MostrarPromociones.....	107
Figura 47 Diagrama de secuencia de análisis CU_MostrarPronosticoTiempo.....	108
Figura 48 Diagrama de Clases de Diseño Web CU_MostrarNoticias.	109
Figura 49 Diagrama de Clases de Diseño Web CU_MostrarPromociones.	110
Figura 50 Diagrama de Clases de Diseño Web CU_MostrarPronosticoTiempo.	111
Figura 51 Diagrama de Secuencia de Diseño Web CU_MostrarContenidoDescarga.....	113
Figura 52 Diagrama de Secuencia de Diseño Web CU_MostrarDirectorioTelefonico.....	114
Figura 53 Diagrama de Secuencia de Diseño Web CU_MostrarNoticias.....	115
Figura 54 Diagrama de Secuencia de Diseño Web CU_MostrarPronosticoTiempo.....	116
Figura 55 Diagrama de Secuencia de Diseño Web CU_MostrarMostrarPromociones.	117

Índice de Tablas

Tabla 1 Definición de los Actores del Sistema.....	42
Tabla 2 Descripción CU_MostrarPronósticoTiempo.....	45
Tabla 3 Descripción CU_MostrarContenidoParaDescargar.	46
Tabla 4 Descripción CU_MostrarNoticias.	49
Tabla 5 Descripción CU_MostrarPromociones.....	50
Tabla 6 Descripción CU_MostrarDirectorioTelefonico.....	51
Tabla 7 Caso de Prueba de CU_MostrarDirectorioTelefonico.....	69
Tabla 8 Descripción de tipos de Actores.	77
Tabla 9 Factor de peso de los actores sin ajustar.	77
Tabla 10 Descripción de tipos de casos de uso.	77
Tabla 11 Factor de Complejidad Técnica.	78
Tabla 12 Factor de Ambiente.....	78
Tabla 13 Esfuerzo total del Proyecto.....	80

INTRODUCCIÓN

Internet permite a los usuarios que navegan a través de la Web, disponer de un gran número de información y servicios en cualquier momento y lugar donde se encuentren. Esto unido al acelerado desarrollo de las telecomunicaciones inalámbricas, ha propiciado que las empresas proveedoras de servicios de valor agregado VASP (del inglés: Value Added Service Provider) desarrollen diversos sistemas para satisfacer las necesidades de clientes con dispositivos móviles. Presentan sus sitios con nuevas tecnologías con el objetivo de obtener mayores funcionalidades, que permiten la mayoría de las ocasiones consultar directorios telefónicos, pronósticos del tiempo, lectura de noticias y descargas de contenidos como música e imágenes.

Estos sitios en su mayoría, como los conocemos hoy, están pensados para equipos como los computadores personales (PCs) o laptops, no así para la amplia gama de dispositivos móviles que se encuentran activos actualmente, los cuales no pueden visualizar de forma correcta el contenido presente en la mayoría de las páginas web, pues los sitios convencionales no están adaptados a una serie de restricciones particulares que presentan estos equipos, como el despliegue de la información muy limitado, poseen métodos de entrada de información deficientes (teclados muy pequeños) y los patrones de navegación adoptados no fueron pensados para permitir un mejor acceso a la información desde un terminal móvil. Esto constituye una de las razones fundamentales a tener en cuenta a la hora de desarrollar aplicaciones web destinadas a usuarios en un entorno inalámbrico.

Por tanto los desarrolladores de sitios web se han visto en la necesidad de encontrar alternativas para disminuir los esfuerzos y costos que supone crear aplicaciones capaces de brindar los mismos servicios tanto a usuarios de teléfonos celulares como de PCs.

Por tanto surge la necesidad de darle solución a la problemática planteada, definiendo como problema científico de la investigación ¿Cómo lograr que los usuarios de dispositivos móviles visualicen y accedan de forma satisfactoria a los contenidos y servicios presentes en la plataforma de gestión de contenidos Gina?

El objeto de estudio lo conforman: los lenguajes de marcado¹, estándares Web, herramientas para el desarrollo de portales WAP y buenas prácticas² para el diseño de portales para terminales móviles, delimitado su campo de acción a los portales WAP para la venta de contenidos en dispositivos móviles.

Para darle solución al problema planteado se definió como objetivo general de la investigación: desarrollar un portal WAP para la venta de contenidos, que garantice que los dispositivos móviles puedan visualizar y acceder de forma satisfactoria a los contenidos y servicios disponibles en Gina.

Para el desarrollo del sitio WAP, se tuvieron en cuenta una serie de interrogantes:

¿Cuáles son las mejores prácticas y métodos para el diseño e implantación de sitios WAP?

¿Qué ventajas ofrecen los portales WAP para la venta de contenidos para móviles y el acceso a la información?

¿Qué ventajas ofrece la utilización de tecnologías basadas en estándares, para el desarrollo de portales web para móviles?

Para darle solución a la situación problemática fue necesario ejecutar un conjunto de tareas de la investigación:

1. Fundamentar tendencias actuales, tecnologías y conceptos más importantes relacionados con el desarrollo de portales WAP.
2. Identificar las principales herramientas open-source y estándares web que aportan soluciones al desarrollo de portales WAP.
3. Identificar e implementar las buenas prácticas para el diseño de portales WAP, para lograr un producto flexible y que cumpla con los estándares establecidos.

Los temas a tratar durante la investigación estarán relacionados al problema de cómo llevar a cabo la implementación de un portal WAP adaptado para la mayoría de los terminales móviles

¹ Lenguaje de Marcado: Es un conjunto de reglas que utilizan una sintaxis predefinida que sirven para estructurar un documento electrónico, utilizando etiquetas al principio y al final de un elemento.

² Buenas Prácticas: Las Buenas Prácticas para la Web Móvil son un conjunto de pautas para la creación y publicación de contenido web para dispositivos móviles.

existentes, que permitirá una navegación satisfactoria a sus usuarios, dando cobertura a las siguientes necesidades de los clientes de telefonía celular en el país: servicio de noticias y reportes del estado del tiempo, acceso a directorios telefónicos y realización de llamadas, avisos que puedan interesar a los clientes y la descarga de contenidos (música, imágenes). Además la propia implementación constituye un aporte en el desarrollo de software en Cuba y es aplicable a cualquier plataforma de gestión de contenidos móviles, así como a cualquier entidad que disponga de una infraestructura creada para proveer sus servicios a través de redes inalámbricas.

Como resultado se demuestra que mediante la utilización de tecnologías que conforman una serie de estándares web, para el desarrollo de sitios adaptados a todo tipo de dispositivo móvil y la utilización del framework MyMobileWeb, es posible lograr una correcta implementación y diseño de sitios adaptados para el acceso mediante móviles.

El contenido de esta investigación está desglosado en 5 capítulos, las conclusiones generales, recomendaciones, referencias bibliográficas y bibliografía utilizada, un glosario de términos y por último los anexos que complementan el cuerpo del trabajo y que son necesarios para su entendimiento.

En el Capítulo 1 “Fundamentación teórica”, se incluyen los resultados del estudio sobre el estado del arte a nivel internacional y nacional sobre cómo realizar sitios web para dispositivos móviles; se describen los conceptos principales que se van a tratar a lo largo del trabajo así como el análisis de las principales herramientas a utilizar en el desarrollo de la aplicación. Durante el Capítulo 2 titulado “Características del Sistema”, se definirá los objetivos estratégicos y se hace una descripción general de la propuesta de sistema y de cómo debe funcionar el negocio a través de procesos. Posteriormente a través del Capítulo 3 “Análisis y Diseño del Sistema”, se presentan los diagramas de clases correspondientes al análisis y diseño, y la realización de los Casos de Uso a través de los diagramas de interacción. Este capítulo está enfocado hacia como construir técnicamente el software. En el Capítulo 4 “Implementación y Prueba”, se describen todos los elementos relacionados con la implementación y prueba de la aplicación y en el desarrollo del Capítulo 5 “Estudio de Factibilidad” se lleva a cabo el estudio de factibilidad del proyecto desarrollado, se describen los beneficios tangibles e intangibles obtenidos y se analizan los costos que supone crear un sistema de este tipo.

CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En este capítulo se describen los principales aspectos y conceptos de relevancia que han sido objeto de análisis a lo largo de la investigación. Las tecnologías y metodología a utilizar, el lenguaje de programación elegido y se expone el estado del arte referente a las tendencias y herramientas que se utilizan para la creación de portales WAP, que permitan un acceso eficiente a los contenidos mediante la utilización de dispositivos móviles.

1.2 Telefonía Móvil

Los teléfonos celulares son el subconjunto más difundido dentro de los terminales móviles y su más importante uso es la transmisión de voz, aunque se pueden utilizar para tomar fotos, videos, mandar mensajes, escuchar música y recientemente para visualizar páginas web a través de Internet y/o Intranets.

Durante los años 50 y 60 se crearon una gran variedad de aparatos de radio, para ser utilizados en la mayoría de los casos por servicios públicos tales como taxis, ambulancias o bomberos. La implementación de estos equipos supuso el comienzo de la evolución hacia los dispositivos denominados móviles de primera generación, surgidos a partir de 1973 y con un tamaño y peso inmanejable, los móviles de primera generación funcionaban sobre redes de transmisión analógicas (1). El hecho de que fueran analógicos traía consigo una serie de inconvenientes, tales como su baja seguridad.

Luego surgió la segunda generación, que con la introducción de una serie de protocolos, permitió realizar más enlaces simultáneos en el mismo ancho de banda y la integración de otros servicios agregados al de transmisión de voz, destacándose el Servicio de Mensajes Cortos (SMS) por sus siglas en inglés (2). De la segunda generación se derivan una serie de limitaciones para las aplicaciones basadas en la transmisión de datos, como son: lento establecimiento de la conexión (sistemas orientado a conexión), insuficiencia a la hora de transmisión de grandes volúmenes de información, poco ancho de banda y un coste excesivo.

El año 2001 fue un año revolucionario en el ámbito de la telefonía móvil con la aparición de los primeros celulares con cámara fotográfica digital, la posibilidad de grabar videos y mandarlos con un sistema de mensajería instantánea evolucionado, juegos 3D, sonido Mp3 o poder

Capítulo 1. Fundamentación Teórica

mantener conversaciones por videoconferencia gracias a una tasa de transferencia de datos más que aceptable, lo que dio lugar a la denominada tercera generación de teléfonos móviles o móviles 3G.

Este acelerado avance no se ha visto evidenciado de igual manera en Cuba, donde la telefonía celular ha debido enfrentar difíciles retos tecnológicos y económicos. A su vez, la telefonía cubana también enfrenta los desafíos provocados por la disparidad de precios entre los servicios celulares y los de la telefonía fija, los cuales en su gran mayoría se cobran en moneda nacional a precios muy bajos.

Unido a esto se halla el problema de no contar hasta el momento dentro de la red nacional, con sistemas que aprovechen las posibilidades que brinda el uso de teléfonos celulares para la navegación a través de sitios WAP, ya que los usuarios finales podrían acceder a la información desde cualquier sitio; en casa, en la oficina o mientras se están moviendo.

Con el objetivo de esclarecer algunos temas tratados durante el desarrollo del presente documento, se hace necesario dar una breve explicación sobre aspectos comunes dentro de los conceptos referentes a las aplicaciones web, los sitios web y los portales WAP.

Las aplicaciones web, son aplicaciones software cuyas interfaces visuales se codifican en la mayoría de los casos en un lenguaje soportado por los navegadores web como HTML (HiperText Markup Language), JavaScript, WML (Wireless Markup Language) y XHTML (Extensible HiperText Markup Language) en la que se encomienda parte de la ejecución de las acciones provocadas por los usuarios al navegador. Un sitio web es un conjunto de documentos digitales, escritos utilizando lenguajes de etiquetas electrónicos, los cuales son interpretados por los navegadores web, también conocidos como clientes HTTP, los sitios pueden ser visualizados o accedidos desde diversos dispositivos con conexión a una red determinada, como computadores personales, computadores portátiles, PDAs y teléfonos móviles.

Los portales o sitios WAP, son un subconjunto dentro de los denominados sitios web y estos a su vez un subconjunto dentro de las aplicaciones web. Un sitio WAP a diferencia de los sitios web convencionales se encuentra sobre una infraestructura de red apropiada para dar servicios a usuarios de equipos con conexión inalámbrica, poder recibir peticiones desde un dispositivo móvil y dar respuestas a estos, gracias a la existencia y desarrollo del Protocolo de Aplicaciones Inalámbricas. Un portal WAP es ajeno a la implementación de las capas inferiores

que conforman la especificación de este protocolo. Por ello estos portales no son más que sitios web desarrollados para cumplir con las exigencias y características dentro de un ambiente destinado a dispositivos móviles.

A continuación se hace referencia a la estructura del protocolo que utilizan generalmente las aplicaciones y teléfonos móviles para comunicarse entre sí, el protocolo WAP, exponiendo las capas sobre las cuales se enmarcan varios puntos relevantes que ayudan a entender algunos temas abordados durante el desarrollo del conocimiento sobre los servicios de valor añadido en la telefonía celular, donde precisamente se encuentra delimitada esta investigación.

1.3 Arquitectura de la tecnología WAP

El protocolo de aplicaciones inalámbricas ha sido un factor clave para la evolución de la telefonía móvil permitiendo la implementación de un gran número de servicios, pues WAP es un estándar bastante seguro que permite que los usuarios accedan a la información a través de dispositivos inalámbricos como PDAs (del inglés: Personal Digital Assistant), teléfonos celulares y los denominados teléfonos inteligentes o smartphones. Está soportado por la mayoría de las redes inalámbricas actuales, siendo un factor clave en el desarrollo de sistemas que brindan servicio a los usuarios que hacen uso de dispositivos con estas características.

En la siguiente figura se pueden observar los diferentes componentes en la arquitectura del protocolo WAP:

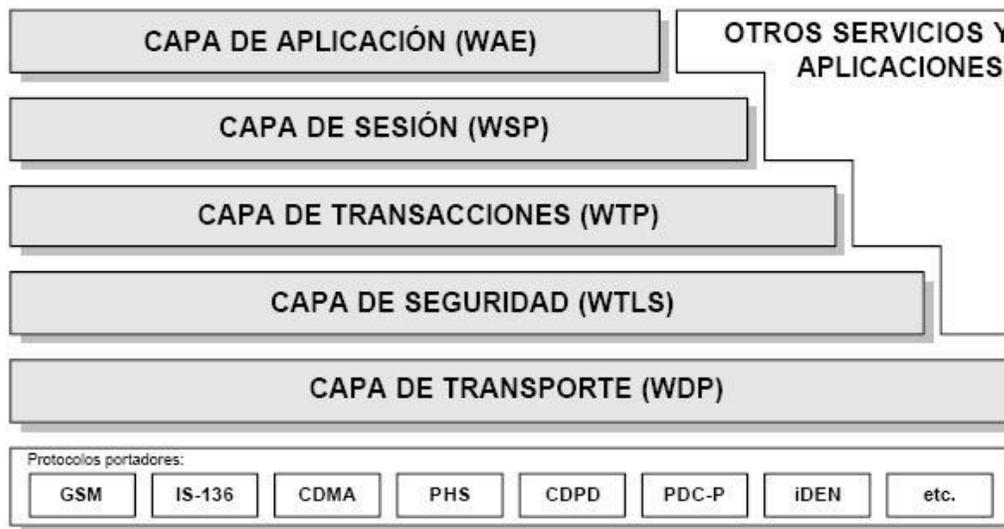


Figura 1 Arquitectura del Protocolo WAP.

Capa de Aplicación

El Entorno Inalámbrico de Aplicación es un entorno de aplicación de propósito general, basado en la combinación del World Wide Web³ y tecnologías de Comunicaciones Móviles.

Este entorno incluye un micro navegador, que posee básicamente las siguientes funcionalidades:

- ✓ Un lenguaje denominado WML (Lenguaje de Marcas Inalámbrico) similar al HTML (Lenguaje de Marcas de Hipertexto), pero optimizado para su uso en terminales móviles.
- ✓ Un lenguaje denominado WMLScript, que es un lenguaje de script para dotar de cierto dinamismo a sus documentos.

Además de un conjunto de formatos de contenido bien definidos entre los que se encuentran imágenes, entradas en la agenda de teléfonos e información de calendario, denominados también como Servicios Telefónicos de Valor Añadido (TeleVAS).

Capa de Transporte

Aquí se encuentra el Protocolo Inalámbrico de Datagramas el cual proporciona un servicio fiable a los protocolos de las capas superiores de WAP y permite la comunicación de forma transparente sobre los protocolos portadores válidos.

Debido a que este protocolo proporciona una interfaz común a los protocolos de las capas superiores, las capas de Seguridad, Sesión y Aplicación pueden trabajar independientemente de la red inalámbrica que dé soporte al sistema.

Ofrece un servicio consistente al protocolo (Seguridad, Transacción y Sesión) de la capa superior de WAP, comunicándose de forma transparente sobre uno de los servicios portadores disponibles.

Este protocolo ofrece servicios a los protocolos superiores, del estilo direccionamiento por número de puerto, segmentación y re-ensamblado y detección de errores opcionales, de forma

³ World Wide Web: Término que designa al subconjunto de Internet que consiste en documentos electrónicos a los que se puede acceder mediante un navegador.

Capítulo 1. Fundamentación Teórica

que se permite a las aplicaciones de usuario funcionar de forma transparente sobre distintos servicios portadores disponibles.

A continuación en la figura 2, se muestra el funcionamiento de WAP cuando se ejecutan peticiones realizadas por usuarios de terminales móviles. El modelo de programación de WAP, está estrechamente alineado con el modelo de programación web, que se caracteriza por un esquema "Pull" (que es donde el cliente solicita el contenido del servidor). Sin embargo, WAP también extiende la arquitectura web mediante la adición de soporte para telefonía y habilita el modelo "Push", donde un servidor puede enviar contenido de manera proactiva al cliente.

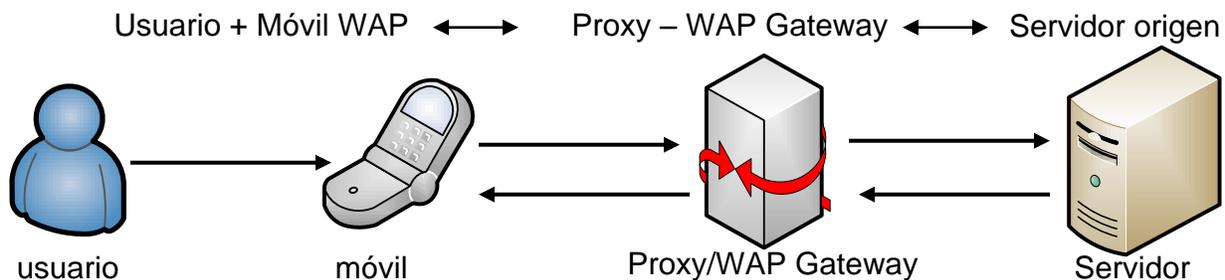


Figura 2 Funcionamiento de WAP.

1. El usuario solicita la página WAP que quiere visualizar.
2. El micro navegador del móvil envía la petición con la dirección de la página solicitada y la información sobre el abonado al WAP Gateway (software capaz de conectarse a la red de telefonía móvil y a Internet).
3. El Gateway examina la petición y la envía al servidor donde se encuentra la información solicitada.
4. El servidor añade la información HTTP (Protocolo de Transferencia de Hipertexto) o HTTPS (Protocolo de Transferencia de Hipertexto Seguro) pertinente y envía la información de vuelta al Gateway.
5. En el Gateway se examina la respuesta del servidor, se valida el código en busca de errores y se genera la respuesta que se envía al móvil.

Capítulo 1. Fundamentación Teórica

6. El micro navegador reconoce la información recibida y si el código es correcto lo muestra en pantalla.

El Proxy/Gateway es la parte más interesante del esquema anterior. Su función es convertir los protocolos de comunicación WAP (ejemplo: WSP, WTP, WTLS) a protocolos HTTP, TCP/IP, y viceversa.

Las especificaciones WAP están en una continua evolución y se está creando un nuevo mundo de posibilidades para los dispositivos y aplicaciones inalámbricos. Con la aparición de la especificación WAP 2.0, el WAP Gateway del esquema anterior no es necesario, ya que la comunicación entre el cliente y el servidor origen se puede realizar utilizando HTTP/1.1. Sin embargo, la implantación de un WAP Proxy puede optimizar los procesos de comunicación y ayudar a ofrecer mejores servicios móviles, como localización y seguridad. Además un WAP Proxy es necesario para poder ofrecer funcionalidades de tipo "Push".

Como se ha visto esta nueva tecnología posee varias ventajas, como el poder acceder a Internet desde cualquier lugar, tiene soporte para HTTP y existe un mayor nivel de seguridad al momento de las transacciones.

WAP 2.0 continúa soportando la antigua pila de protocolos, pero también soporta la pila estándar de Internet con TCP, IP y HTTP/1.1. Sin embargo, se realizaron cuatro cambios a TCP en cuanto al uso de un marco fijo de 64 Kbytes, inicio rápido, una MTU (del inglés: Maximum Transmission Unit) máxima de 1 500 bytes y un algoritmo de retransmisión ligeramente diferente, ver figura 3.

TCP/IP		WAP
HTML RFC-822	Nivel de Aplicación	WAE
HTTP FTP SMTP ...	Nivel de Sesión	WSP
TCP UDP	Nivel de Transacción	WTP
SSL TLS	Nivel de Seguridad	WTLS
IP	Nivel de Transporte	WDP
ETH ATM ...	Nivel de Red	GSM SMS CDMA

Figura 3 WAP 2.0 soporta dos pilas de protocolos.

Otra diferencia técnica con respecto de WAP 1.x es el lenguaje de marcado. WAP 2.0 soporta XHTML Mobile Profile (XHTML-MP) por defecto, el cual amplía el perfil básico de XHTML según la definición del W3C. Este núcleo fue diseñado para ser extensible y la capa de aplicación de WAP se aprovecha de esta capacidad para la definición de las características adicionales del lenguaje, para mejorar la funcionalidad.

Todos estos avances dentro del Protocolo de Aplicaciones Inalámbricas han traído como consecuencia que muchos desarrolladores y entidades alrededor del mundo se interesen en crear aplicaciones con nuevas tecnologías, que permitan llevar la Web al móvil de una manera más sencilla. Para esto hace falta realizar un riguroso examen en cuanto a las herramientas y técnicas que brinden mejores soluciones en cuanto al desarrollo de portales WAP, según los objetivos que se quieran alcanzar con su puesta en marcha.

1.4 Tecnologías y tendencias en el desarrollo de portales WAP

Durante los apartados anteriores se ha mostrado el desarrollo que han tenido las telecomunicaciones móviles y los servicios para aplicaciones inalámbricas, lo cual ha llevado la proliferación en Internet de sitios WAP para la venta y descarga de contenidos. La cantidad de usuarios que hacen uso de estos va en aumento, y se espera que para el 2010 existan unos 1 300 millones de usuarios de computadores convencionales, por 2 500 millones de usuarios móviles (3). A continuación se muestra una figura que representa la cantidad de usuarios en miles de millones según datos de Vodafone.

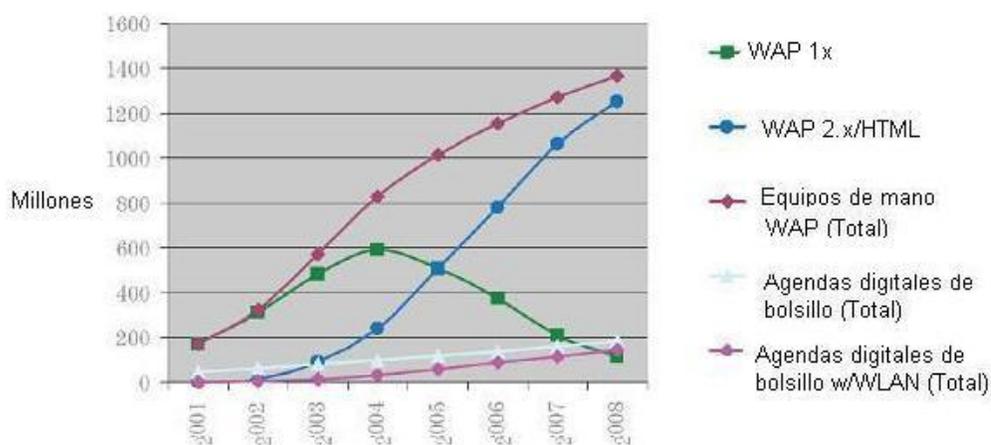


Figura 4 Cantidad de Usuarios por Tecnología.

Capítulo 1. Fundamentación Teórica

Las grandes, pequeñas y medianas empresas con presencia en la red de redes están siendo testigos del inmenso abanico que han abierto las telecomunicaciones actuales en cuanto a mercado se refiere. Contar con una plataforma que brinde servicios a usuarios de terminales móviles por medio de portales WAP, constituye un factor clave para las empresas, ya que tendrían la oportunidad de fortalecer las relaciones con sus actuales y posibles clientes, mientras estos se encuentren en movimiento, a cualquier hora y en cualquier lugar.

Los contenidos digitales, en incremento continuo, comienzan a estar también disponibles en plataformas móviles. Las cuales están evolucionando y se convierten rápidamente en prometedores sistemas de entrega de contenidos.

Los tres primeros tipos de servicios de datos ofrecidos por las redes móviles incluyen:

- ✓ Los servicios de datos basados en comunicación, que abarcan las comunicaciones “peer-to-peer” como la mensajería y el correo electrónico.
- ✓ Los servicios de datos transaccionales, que incluyen los servicios de transacción financieros.
- ✓ Los servicios de datos basados en contenidos, que incluyen la música, el contenido para educación basado en el entretenimiento, los juegos, video, las noticias, la información relacionada con el transporte, el entretenimiento para los adultos.

Como puede observarse existe una gran contradicción, dado que por una parte el uso de contenidos digitales y dispositivos móviles está creciendo vertiginosamente; por otra parte, el acceso a la Web a través de estos es mínimo, principalmente en los países subdesarrollados, tomando como comparación las cifras asociadas a la navegación utilizando computadores personales.

Todo esto se debe a que los sitios web no han sido desarrollados tomando en cuenta las características y limitantes de estos dispositivos en vista de hacer los recursos más accesibles. Así por ejemplo, no se tiene presente que en estos terminales de manera general el despliegue de la información es sumamente limitado, debido a que se tienen pantallas muy pequeñas, con menos resolución y colores, la entrada de información es sumamente difícil, generalmente poseen teclados muy pequeños con escasas teclas y no poseen mecanismos señaladores como el puntero de un mouse por citar un ejemplo, presentan poca capacidad de procesamiento y de almacenamiento, los altos costos de conexión, la mayoría no soporta

Capítulo 1. Fundamentación Teórica

scripts en su navegador, el deficiente ancho de banda y poseer un patrón de navegación muy diferente al empleado en la Web de escritorio o Web convencional, han frenado la visualización de los sitios web a través de dispositivos móviles.

El problema que se presenta al tener pantallas muy pequeñas es que se necesita de una mayor interacción por parte de los usuarios. Dentro de dicho problema destaca el uso excesivo del “scrolling” o desplazamiento en las páginas, donde el usuario tiene que recorrer hacia todas direcciones la pantalla con el objetivo de visualizar el recurso web de forma completa.

Otro problema consiste en que muchos navegadores no soportan algunas características concernientes al lenguaje utilizado en la construcción de las páginas, por lo que no se pueden visualizar y si se visualizan no lo hacen de la manera esperada. Además de diversos factores con respecto al empleo de diferentes lenguajes de marcado por parte de los fabricantes como HTML, CHTML (del inglés: Compact HyperText Markup Language) o WML (del inglés: Wireless Markup Language), los cuales tienen muchas similitudes en su sintaxis, pero poseen numerosos aspectos que los hacen incompatibles entre sí y que no puedan ser interpretados en ocasiones por los micro navegadores de forma correcta.

No existe una uniformidad en los formatos de contenidos digitales (imágenes, sonidos, documentos) soportados por los micro-navegadores, ya que esa capacidad varía, entre modelos de diferentes fabricantes, como también entre modelos de un mismo productor de dispositivos móviles. Esto trae aparejado que se haga muy difícil construir una aplicación donde cada contenido disponible para los usuarios, pueda ser interpretado y visualizado de manera correcta. Ejemplo, existen móviles que por sus características solo pueden visualizar archivos imagen en formato “wbmp”, esto es una imagen monocroma (de un solo color), otros permiten utilizar imágenes animadas, pero con un determinado número de colores y resolución.

Todo esto unido a que los mecanismos de publicidad no funcionan de la misma manera desde la perspectiva móvil y el mal diseño estructural, conlleva a una mala experiencia de navegación por parte de los usuarios y a que elijan métodos alternos de acceso a la Web (3).

Para poder llevar a cabo la implantación de un sitio WAP que propicie el acceso a servicios de diversas características y la venta de contenidos sobre una plataforma de gestión de contenidos para dispositivos móviles, es imprescindible definir cuáles son las tendencias adoptadas, las herramientas y aplicaciones que proveen mejores soluciones y mayor desarrollo

para su implementación, tomando en cuenta los estándares que existen en el mundo, para así poder lograr una total independencia tecnológica en el área de las telecomunicaciones.

1.4.1 Alternativas adoptadas

La única forma de traer orden al caos existente en el desarrollo de aplicaciones móviles, es estableciendo estándares y guías que ayuden a lograr una uniformidad. La Iniciativa para la Web Móvil creada por el W3C ha sido un instrumento con este propósito, sus buenas prácticas para el diseño web y las especificaciones dentro de la herramienta “**mobileOK**” para determinar si un sitio puede funcionar apropiadamente en varios dispositivos móviles o no, son unos de los ejemplos más notables. Hay varias alternativas a tener en cuenta a la hora de realizar un sitio web para un celular, para ello se deben atender principalmente cuatro opciones:

- ✓ **No hacer nada:** solamente presentar el sitio como se está acostumbrado a hacer y dejar que los browsers (navegadores o clientes HTTP) de los terminales móviles se encarguen de la interpretación y visualización de las páginas. Esto puede parecer sorprendente, pero muchos navegadores móviles pueden visualizar los sitios bastante bien en pantallas pequeñas porque cuentan con plugins que hacen zoom a las páginas o transforman algunos contenidos. Las técnicas más notables en este sentido son las utilizadas en los navegadores Opera, en los iPhone y los recientes modelos de la compañía Nokia⁴.

Si se decide no hacer nada, un sitio web puede que sea visible desde un dispositivo móvil, aunque es muy probable que no se visualice y funcione de la manera que se desee y se estaría excluyendo a la inmensa mayoría de los usuarios que aún no cuentan con móviles de tercera generación avanzados.

- ✓ **Eliminar el formato:** esto quiere decir, simplificar el sitio para que se cargue de manera más rápida por los navegadores y use menos ancho de banda, pero dejar el diseño del sitio tal como está. Una de las mayores dificultades de los navegadores de móviles es analizar el código HTML y cargar las páginas, existen incluso varias herramientas que permiten hacer este trabajo más fácil, como IYHY.com⁵ y Skweezer.net⁶, pero los sitios

⁴ Nokia: <http://www.nokia.es/productos>

⁵ IYHY.com: <http://www.iyhy.com/>

⁶ Skweezer.net: <http://www.skweezer.com/>

no tendrán un diseño llamativo con esta práctica, la mayoría de las veces el resultado final son páginas llenas de textos y enlaces, con poca riqueza en el diseño visual.

- ✓ **Diseño basado en CSS:** utilizar un archivo de Hojas de Estilo en Cascada diferente para los visitantes de terminales móviles y definir un formato optimizado para varios elementos de las páginas a través de estos CSS, puede ser una buena práctica, aunque no todos los micro-navegadores soportan completamente los estilos definidos mediante la tecnología CSS.
- ✓ **Crear un nuevo sitio:** desarrollar una versión de su sitio web adaptada para los dispositivos móviles y las personas que hacen uso de ellos mientras están en movimiento. Se debe tener muy en cuenta cuando se hace un sitio para móviles, adaptar no solo el diseño, sino también la estructura del flujo de navegación y el contenido. Se puede extender o adaptar las páginas dependiendo de las capacidades y características de los terminales a los que esté dirigido.

La necesidad es lograr mostrar los contenidos web de manera aceptable a una gran variedad de dispositivos y navegadores; las dos prácticas más comunes para lograr esto son la “**adaptación**” de todos los contenidos y el empleo de lo que comúnmente es denominado por los desarrolladores como “**mínimo común denominador**”. La adaptación con frecuencia se refiere a la entrega de contenido web teniendo en cuenta cada una de las capacidades y características de los distintos tipos de equipos que se utilizan para la navegación. Diferentes técnicas hacen uso de la adaptación incluyendo detección, redirección, establecimiento del tipo MIME adecuado (del inglés: Multipurpose Internet Mail Extensions), cambiando los links de navegación y ajustando o cambiando las dimensiones de los gráficos.

El método “mínimo común denominador” o LCD por sus siglas en inglés, establece un conjunto mínimo de características que son comunes a un grupo de dispositivos y se desarrolla el contenido siguiendo esas directrices. Este enfoque tiene un grave inconveniente, las aplicaciones presentarán un pobre aspecto y diseño, mínima funcionalidad y el resultado final será una mala experiencia por parte de los usuarios, especialmente en dispositivos avanzados con modernas características. Resumiendo, el estilo de programación LCD es muy defensivo porque la mayoría de las capacidades de los dispositivos de tercera generación no pueden ser plenamente explotadas.

Capítulo 1. Fundamentación Teórica

La adaptación tomando en cuenta las particularidades de cada posible dispositivo, es la solución ideal para la entrega de contenidos en la Web móvil, por otra parte el método LCD es con frecuencia más usado por los desarrolladores, pues una adaptación total involucra costos extras en el desarrollo y más complejidad.

A raíz de esto el desarrollo de portales web para móviles se ha visto dividido principalmente en dos vertientes, a través del empleo de las técnicas anteriormente mencionadas. La primera es la creación de sitios bajo el dominio de Internet .mobi ⁷ y la segunda teniendo en cuenta los avances en la estandarización de los lenguajes para páginas web por los fabricantes de teléfonos celulares, es decir crear portales sin la necesidad de contar con un dominio específico, que puedan ser visualizados perfectamente tanto por los equipos de interconexión tradicionales como por dispositivos móviles. Dentro de la primera de estas soluciones adoptada por muchos está la de crear aplicaciones teniendo como objetivo la creación del dominio “.mobi” el cual gracias a una serie de requisitos bien definidos dentro de la especificaciones señaladas en su creación, asegura que los sitios que se encuentren publicados en él sean de un elevado por ciento compatibles con los distintos modelos de terminales existentes.

Dentro de la segunda variante han sido empleadas diversas técnicas por los desarrolladores, para permitir la correcta visualización de los contenidos, como el surgimiento y desarrollo de convertidores de páginas web multiformato, esto consiste en la construcción de las interfaces visuales utilizando como lenguaje de marca HTML, luego se reformatean a un lenguaje soportado por el navegador del dispositivo destino como puede ser WML, XHTML Basic, XML (del inglés: Extensible Markup Language) y texto plano, con el inconveniente que la gran mayoría de las páginas no pueden convertirse de manera correcta para su visualización, sobrecargando a los servidores y provocando que los tiempos de respuesta sean lentos.

Otra de las técnicas empleadas dentro de esta segunda vertiente es la de crear aplicaciones web, tratando de ofrecer una buena experiencia en la navegación a sus usuarios por medio de la utilización de una serie de estándares web y tecnologías bien definidos por el W3C como XML, W-CSS (del inglés: Wireless Cascade Style Sheet), RDF (del inglés: Resource Description Framework), SVG (del inglés: Scalable Vector Graphics), XHTML y XSLT (del inglés: Extensible Stylesheet Language Transformations), que permiten separar contenido y presentación, aumentando así la productividad y el rendimiento en los sistemas.

⁷ .mobi: Dominio de Internet donde los sitios alojados están adaptados para ser correctamente visualizados por la mayoría de los dispositivos móviles.

Como se puede apreciar llegado a este punto, no es necesario disponer de un dominio en particular para que la Web sea compatible con navegadores de terminales móviles.

Otro ejemplo, de cómo las aplicaciones web, conscientes de las limitantes en estos terminales han ido en aumento se puede ver en la figura que aparece a continuación, según MobileOpportunity⁸ donde se evidencia que las aplicaciones basadas en la Web están ganándole la partida a las aplicaciones móviles nativas.

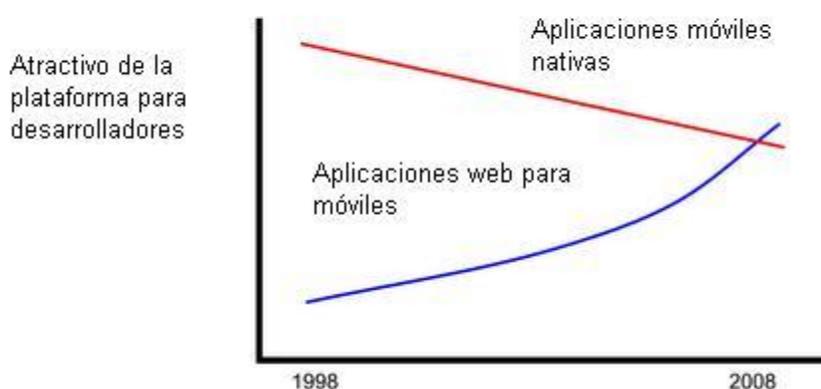


Figura 5 Comparativa de Software en ambientes móviles.

Esto quiere decir que los programas software que son explícitamente instalados sobre un determinado dispositivo, en este caso inalámbrico, tienen el inconveniente que deben estar enfocados también al sistema operativo que posee el terminal, sobre el cual van a funcionar, unido a que existen al menos diez plataformas diferentes (Microsoft por si sola tiene varias: dos versiones de Windows Mobile, el Tablet PC, etc.), lo que dificulta por demás el desarrollo de una misma aplicación para cada uno de estos sistemas operativos. En contrapartida la Web y el desarrollo de las capacidades en los móviles conjuntamente con las redes de transmisión de datos celulares, han despertado un gran interés en los desarrolladores móviles así como en las empresas proveedoras de servicios, ya que ofrece un mejor modelo de negocio.

Naturalmente, las aplicaciones web todavía tienen problemas enormes en los móviles. Las aplicaciones web móviles no funcionan cuando no hay cobertura, son lentas debido a la latencia de la red, y no utilizan las redes inalámbricas de manera eficiente. Pero será más fácil

⁸ MobileOpportunity: <http://mobileoppotunity.blogspot.com>

resolver o soportar estos inconvenientes técnicos que al pasar de los meses se van solucionando, que corregir los problemas estructurales y comerciales fundamentales del mercado de las aplicaciones móviles nativas.

Por otro lado algunas tecnologías son más tenidas en cuenta por parte de los desarrolladores de aplicaciones móviles, según el grado de flexibilidad, desarrollo y potencia que poseen, sobre las cuales se han implementado diferentes servicios que atraen a los usuarios que acceden a Internet.

Los desarrolladores de aplicaciones web se han visto en la necesidad de contar con tecnologías que les permitan disminuir los esfuerzos y costos que supone crear sitios capaces de brindar los mismos servicios tanto a usuarios de equipos tradicionales como a los de dispositivos móviles, para ello deben hacer uso de la información referente a las características de cada dispositivo en concreto, para poder adaptar los contenidos en dependencia de las restricciones que imponga su diseño y capacidades, esta información se encuentra la mayoría de las ocasiones almacenada en determinados repositorios o bases de datos.

1.4.2 Repositorios de Descripción de Dispositivos

La mayoría de los equipos que se utilizan para visualizar páginas web, cuando se conectan a las aplicaciones por medio de un navegador, envían dentro de cada petición un identificador específico, utilizado para resolver sus particularidades, las cuales pueden obtenerse desde bases de datos específicas comúnmente denominadas repositorios de descripción de dispositivos.

Una de las tecnologías empleada para reconocer las características de los dispositivos móviles, es **DeviceAtlas**. Cuenta con un foro en Internet, para brindar asistencia a los desarrolladores de portales WAP que utilicen sus herramientas. Los datos son provenientes de varias fuentes, algunos de estos son directamente proporcionados por los fabricantes, operadores de redes o proveedores de tecnologías en su mayoría de dominio público, como el caso del User-Agent Profile para dispositivos, el cual es mantenido por OMA⁹ y del proyecto WURFL (del inglés: Wireless Universal Resource File), de ellos se hablará más adelante en este apartado.

⁹ OMA: Entidad encargada entre otras cosas de la estandarización dentro de la industria de las telecomunicaciones móviles.

Capítulo 1. Fundamentación Teórica

La información es accedida desde un archivo que utiliza JSON (del inglés: JavaScript Object Notation) como una serialización compacta de los datos, ese es el archivo que contiene la información acerca de nuevos dispositivos, nuevas propiedades, y es utilizado por los desarrolladores para hacer consultas sobre las capacidades de un equipo determinado.

Este repositorio implementa un algoritmo de ponderado para resolver los conflictos que pudieran existir con la incongruencia de los datos provenientes de diversas fuentes. En estos casos si la mayoría de las personas encuestadas están de acuerdo sobre el valor de una determinada propiedad, entonces se ajusta el valor del factor de veracidad correspondiente, bajando su nivel de confianza.

Describe unas setenta propiedades sobre dispositivos móviles y posee una aplicación Web en Internet llamada TA-DA¹⁰ que al iniciarse la conexión mediante un terminal móvil, se activa el mecanismo de reconocimiento y si no se encuentra registrado dentro del repositorio, ejecuta los algoritmos para añadir ese nuevo dispositivo y asignarle a las propiedades los valores apropiados. El inconveniente radica en que es necesario acceder a la aplicación en Internet para lograr que se actualicen los datos de DeviceAtlas con el nuevo dispositivo, de otra manera el mecanismo empleado si no encuentra la correspondencia del User-Agent emitido en la petición HTTP con alguno registrado previamente, toma como valores los del mismo modelo pero de una versión anterior.

Otra característica que presenta esta tecnología es que solo se provee libre de costo para el desarrollo, sin embargo se necesita pagar licencia para la comercialización de aplicaciones que la utilicen, además de acuerdo con el tipo de licencia adquirida se restringe el período de actualización del archivo JSON.

Como se dijo anteriormente parte de su información proviene de proyectos como WURFL, pero desecha su algoritmo de reconocimiento y solamente utiliza una parte de los datos contenidos en él. Es precisamente esta tecnología una de las fuentes en la descripción de dispositivos y de las más utilizadas en el ámbito de las aplicaciones para dispositivos móviles tanto por herramientas open-source como propietarias.

WURFL, es una base de datos en formato XML que constituye un proyecto de software libre con una gran comunidad en Internet que impulsa su desarrollo y mantiene constantemente

¹⁰ TA-DA: <http://www.ta-da.mobi>

Capítulo 1. Fundamentación Teórica

actualizado el fichero wurfl.xml, gracias a la cooperación de la mayoría de los fabricantes. El principal objetivo de dicho fichero es recoger toda la información posible sobre los terminales móviles existentes, basándose en el concepto de árbol genealógico, donde todos los dispositivos descienden de una clase genérica, pero que incluso pueden descender de una familia más especializada. Posee información sobre 5 000 equipos aproximadamente con 523 propiedades, aunque es preciso señalar que estas cifras están sujetas a constantes actualizaciones y al igual que DeviceAtlas cuenta en Internet con una aplicación Web donde los usuarios pueden contribuir a la actualización de nuevos datos.

En esta base de dato XML se emplea un algoritmo de "fall-back" (recurrir a algo), esto es cada objeto <device> mapeado dentro del fichero wurfl.xml posee un atributo fall-back que tiene como valor asociado el identificador del dispositivo del cual hereda. Esto se utiliza para recorrer el árbol de forma recursiva y así ir recuperando los valores de las propiedades de un equipo determinado a medida que se va descendiendo hasta la raíz, que es a su vez un objeto <device> genérico con el que se asegura que ninguna propiedad se quede sin valor. Siguiendo este algoritmo cuando un nuevo dispositivo se da de alta, solo tiene que definir las características que le son diferentes con respecto a su predecesor. (Ver Anexo 1 figura 28).

Una buena parte de los datos recogidos en WURFL provienen de una especificación devenida en estándar creada por la Open Mobile Alliance denominada OMA User Agent Profile, conocida por su acrónimo UAProf.

Esta norma se diseñó como un marco común para que todos los fabricantes de teléfonos móviles pudieran describir las características técnicas y funcionalidad de sus teléfonos mediante el uso de XML. Generalmente los dispositivos móviles envían una cabecera en las peticiones HTTP, que contiene la URL a su perfil. La cabecera usualmente se identifica por el valor "x-wap-profile", "wap-profile" o una serie de parámetros similares con las que las aplicaciones de adaptación de contenidos trabajan para obtener el archivo XML correspondiente al perfil del terminal cuando está realizando una petición. Por otra parte la información recogida no es cien por ciento fiable al igual que en los casos de WURFL y DeviceAtlas, y buena parte de las direcciones no se encuentran disponibles o son erróneas. Además recuperar y analizar los UAProfs en tiempo real desde Internet es lento y puede sobrecargar substancialmente cualquier solicitud Web, exigiendo la creación de un mecanismo de caché y un flujo para su actualización para poder comprobar datos obsoletos.

Todas estas tecnologías son empleadas en mayor o menor grado por la mayoría de los desarrolladores y entidades dedicadas a la creación de aplicaciones Web que adapten sus contenidos y servicios a dispositivos móviles. Muchos utilizan solo una, extendiéndola con implementaciones propias y otros realizan mecanismos más complejos para lograr integrar las funcionalidades de varios módulos, lo que incide directamente en la robustez y el éxito de las soluciones desarrolladas.

1.4.3 Soluciones en la adaptación de contenidos Web

En los países donde ha tenido una gran penetración la telefonía móvil, se ha evidenciado un fuerte impacto de las más novedosas y exitosas aplicaciones para la venta de contenidos en terminales móviles y realización de portales WAP, destacándose entidades como **Ogangi**¹¹, en el contexto Latinoamericano, específicamente en Venezuela, que ofrece a sus clientes la posibilidad de integración a través de diversas redes de operadores móviles y el aprovechamiento de las capacidades de los dispositivos en beneficio de las empresas, gobiernos, proveedores de contenidos y los usuarios, mediante una serie de interfaces web fáciles de usar, que permiten la creación y administración de los contenidos en portales WAP, hace uso de APIs (Application Programming Interface) específicos para lograr la integración entre proveedores de contenidos, empresas y sistemas de pago, posee un Mobile Media Connectivity Gateway que proporciona seguridad para interconectar las diversas redes de operadores móviles, así como el desarrollo de innovadores formatos de entrega de contenidos WAP.

La compañía **mTLD**¹², más conocida por el acrónimo “**dotMobi**” es la organización responsable para la administración y mantenimiento del nombre de dominio de Internet .mobi, la cual está respaldada por una variedad de compañías móviles, fabricantes de dispositivos y proveedores de contenidos, entre los que se incluyen las grandes compañías Ericsson, Google, GSM Association, Microsoft, Nokia, Orascom Telecom, Samsung Electronics, Syniverse, T-Mobile, Telefónica Móviles, TIM, Visa y Vodafone. Mediante su sitio web facilita el alojamiento de portales que podrán ser accedidos por cualquier dispositivo móvil con capacidad de navegación, ocultando a sus clientes aspectos concernientes a la programación, también posee un conjunto de diapositivas que van guiando al cliente en la confección y diseño de su sitio.

¹¹ Ogangi: <http://www.ogangi.com/>

¹² mTLD: <http://www.mtld.mobi/>

Dentro de este marco también se incluye **Volantis Systems**¹³, la cual constituye un participante activo dentro de comunidades para el desarrollo de estándares abiertos. Directamente involucrado en grupos de trabajo como la “Mobile Web Initiative” del W3C y la OMA. Brinda a sus clientes un conjunto de herramientas para el desarrollo de portales WAP o la transformación de los portales Web convencionales ya existentes, para permitir una buena experiencia a sus usuarios en la navegación. Apoyándose en un lenguaje propio para la definición de las presentaciones de los sitios, independiente de los dispositivos y la tecnología utilizada, denominado XDIME (del inglés: XML Device Independent Markup Extensions), muy parecido a XHTML y basado en el lenguaje estándar DIAL (del inglés: Device Independent Authoring Language).

Volantis provee herramientas de diseño en forma de plugins para el IDE Eclipse, una base de datos propia que describe más de 6 000 perfiles de dispositivos y navegadores registrados, un Media Adaptation Proxy (MAP) para la transformación del contenido e imágenes y un Message Preparation Server (MPS), el cual tiene la funcionalidad de transformar distintos formatos de mensajes acorde a las características y capacidades de cada dispositivo, como pueden ser WAP Push, MMS/SMIL (Multimedia Messaging Service), (Synchronized Multimedia Integration Language), SMS y e-mails HTML. También permite la integración y adaptación de servicios de mensajería presentes en redes de terceros. (Ver figura 21 Anexo 1).

Además de las diversas soluciones aportadas por varias compañías del sector móvil, muchas veces estas no satisfacen todas las necesidades que pudiera tener un cliente específico dentro de un determinado dominio. Es por ello que en muchas ocasiones los desarrolladores prefieren utilizar herramientas que le permitan cierta autonomía en el proceso de creación de sus portales WAP y no estar atados a las restricciones impuestas por soluciones de terceros.

1.4.4 Herramientas para el desarrollo de portales WAP

Existen varias herramientas que permiten el desarrollo de portales WAP y la adaptación de sitios para diferentes dispositivos. Dentro del software propietario se encuentra la plataforma .NET con la que se pueden crear aplicaciones web móviles, apoyándose en el uso de una extensión del Framework .NET denominada Microsoft Mobile Internet Toolkit y el entorno de desarrollo integrado Visual Studio. Construido sobre Microsoft ASP.NET Web Forms, se

¹³ Volantis Systems: <http://www.volantis.com/>

encuentra **ASP.NET Mobile Control**, que posee un conjunto de controles para la creación de sitios web para terminales móviles siguiendo el estilo de la programación basada en eventos, aunque también permite el trabajo con código nativo como XSLT (del inglés: Extensible Stylesheet Language Transformations), integración con servicios Web y soporte para datos XML.

Estas tecnologías generan el código en el lenguaje de marcado que soporta el dispositivo que realiza la petición, no realiza un eficiente proceso de adaptación de contenidos, pues no está pensado para hacer la transformación en tiempo de ejecución de los archivos multimedia como las imágenes en caso de ser necesario, pues captura las cabeceras que vienen dentro de la petición que realiza el navegador móvil para descubrir qué soporta ese terminal en específico. Esto trae consigo el inconveniente de que los desarrolladores se ven obligados a realizar esfuerzos extras para extender el componente y lograr que todos los contenidos presentes en las páginas sean correctamente visualizados, ya que no existen mecanismos que agilicen este proceso dentro de esta herramienta.

En el control de imagen, el desarrollador puede incluir una lista de archivos destinos con diferentes tamaños y formatos, lo que hace tedioso el proceso de edición, con el inconveniente de que podría existir algún dispositivo que lo visualice de una forma inesperada. No posee mecanismos de transformación de contenidos multimedia, por lo que es preciso incorporar otros sistemas que realicen esta tarea.

Utiliza un espacio de nombres propios llamado **System.Web.Mobile** que contiene las funciones fundamentales, incluyendo la autenticación y el control de errores, necesarias para generar aplicaciones web asp.net, además posee un tratamiento de estilos propios mediante clases predefinidas en el Framework que tienen la capacidad de ajustarse a las características de los dispositivos haciendo uso de hojas de estilos (CSS). Una desventaja que posee es que no explota todas las potentes funcionalidades que le brindaría el trabajo con varios mecanismos de reconocimiento de dispositivos y la dificultad para lograr extenderlo por parte de los desarrolladores.

Aunque el framework de la plataforma .NET para aplicaciones web móviles, pueda brindar algunas ventajas como poder desarrollar en varios lenguajes de programación, en el denominado software open-source o de código abierto se encuentran algunas herramientas con mayor desarrollo en el ámbito de la creación de portales WAP, como ejemplo se tiene una

Capítulo 1. Fundamentación Teórica

extensión basada en el framework estándar JSF (del inglés: Java Server Faces) desarrollado por la compañía Ericsson que se denomina Mobile JSF Kit.

Mobile JSF Kit provee a los desarrolladores de una guía detallada con códigos de ejemplo y soluciones adicionales para aplicaciones móviles, tales como el uso de AJAX (acrónimo para: Asynchronous JavaScript And XML), seguridad, transformación de los sitios Web convencional que han sido desarrollados conforme a la especificación JSF y un plugin para el IDE NetBeans. Cuenta con un mecanismo automático que convierte el formato de las páginas al utilizado por los navegadores de los dispositivos móviles que se conectan a las aplicaciones creadas con esta tecnología y soporta la transformación de los archivos multimedia como imágenes, utilizados en los sitios Web.

El mecanismo para el reconocimiento de las características que poseen los terminales emplea como fuente de datos ficheros XML como WURFL, pero solo extrae la información necesaria para conformar seis atributos sobre los que trabaja (User-Agent, lista de formatos de imágenes soportados, lenguaje de marcado, nombre del dispositivo, lenguaje de script y dimensión de la pantalla), lo que lleva a pensar que su adaptador de contenidos para la correcta visualización de las presentaciones no realiza un proceso de adaptación eficiente.

Otra de sus particularidades es que posee pocos controles de interfaz para enriquecer las presentaciones, está más enfocado a la utilización de AJAX como solución para brindar dinamismo y enriquecimiento visual de las páginas, en contraste con el gran número de navegadores móviles que no tienen soporte para javascript. Esto restringe las posibilidades de los desarrolladores para mostrar información en distintos formatos.

Mobile JSF Kit es una buena alternativa dentro del software libre para acelerar los procesos de creación de portales WAP, pero todavía carece de madurez en su desarrollo, sin embargo en muchas ocasiones se hace indispensable contar con otras opciones para adaptar un sitio Web a los dispositivos móviles. Dentro de las herramientas open-source existentes se encuentra un API que puede ser utilizado en varios lenguajes con tecnologías Web, llamado **WALL** (del inglés: Wireless Abstraction Library) el cual es muy sencillo de utilizar y brinda una solución bastante eficiente para adaptar algunos contenidos Web en los sitios destinados al ambiente móvil.

WALL se encarga de la transformación de un conjunto de etiquetas con un prefijo bien definido (`<wall:*>`) a etiquetas propias de lenguajes como WML, CHTML, XHTML y HTML

respectivamente, según el soportado por el navegador del dispositivo que realiza la petición, teniendo en cuenta las propiedades presentes en WURFL. Cabe señalar que si bien las aplicaciones son relativamente fáciles de implementar bajo esta tecnología posee la debilidad de no realizar adaptación de contenidos multimedia y es limitado el desarrollo de su lenguaje declarativo de interfaz de usuario, por lo que no está enfocado a realizar aplicaciones ricas en cuanto al diseño de las presentaciones, ejemplo no posee una etiqueta para representar controles tabla en la páginas ni su adaptación.

Su utilización más común es representar algunos componentes de las páginas de un sitio para varios lenguajes de marcado de destino y la selección automática de archivos como imágenes de forma similar a como lo hace el control imagen de la tecnología .NET.

Una alternativa al uso de WALL lo constituye **HAWHAW** (del inglés: HTML and WML Hybrid Adapted Webserver), el cual posee características que lo hacen interesante y útil en el desarrollo de aplicaciones Web. HAWHAW es un lenguaje de script open-source escrito en PHP (del inglés: Hypertext Preprocessor) que posee un servidor Web y genera el código en varios lenguajes de marcado como HTML, WML, XHTML, CHTML, HDML, MML (Mail Markup Language). Mediante el uso de esta herramienta se pueden crear páginas usando el lenguaje común PHP o utilizando XML. Otra característica interesante de HAWHAW es que se pueden generar respuestas en las aplicaciones mediante interacción por voz empleando para ello el lenguaje VoiceXML. Esta herramienta todavía no cuenta con una madurez consolidada en el desarrollo de su tecnología, pues algunos componentes carecen de un especializado desarrollo en los temas sobre los que trabaja.

Aunque hay varias características que hacen de HAWHAW una potente herramienta, muchas entidades del sector de las telecomunicaciones móviles establecen entre sus requisitos implementar las aplicaciones bajo el lenguaje JAVA, ya que es el más difundido y en el que poseen generalmente conocimiento previo. Como ejemplo está el framework anteriormente citado Mobile JSF Kit y otro con un mayor desarrollo denominado MyMobileWeb.

Este Framework es desarrollado por Telefónica I+D e incluido en el seno de una comunidad española de software libre llamada MORFEO¹⁴. Brinda un conjunto de herramientas que facilitan el desarrollo de portales Web adaptados a todo tipo de dispositivo móvil, con un plugin para el IDE Eclipse que facilita su uso y acelera el desarrollo de aplicaciones, así como un SDK

¹⁴ MORFEO: <http://morfeo-project.org>

Capítulo 1. Fundamentación Teórica

(Kit de Desarrollo de Software) que implementa un Modelo Vista Controlador (MVC) sobre tecnologías Java para el manejo del flujo de eventos dentro de las aplicaciones, enfocado a la adaptación de contenidos y orientado sobre la implementación de un conjunto de estándares especificados por el W3C.

Posee un eficiente mecanismo de reconocimiento de las capacidades de los dispositivos utilizando en su beneficio a UAProf, WURFL y extensiones propias de este y puede ser fácilmente extensible e incorporar nuevos mecanismos de reconocimiento como DeviceAtlas, cuenta con módulos de seguridad y validación tanto del lado del cliente como del lado servidor, utilizando para ello javascript y XML para navegadores que no soporten script. Además para aumentar el rendimiento pre-genera las páginas JSP (del inglés: JavaServer Pages) por cada tecnología de marcado, las cuales son especificadas por el programador mediante documentos XML con un conjunto de etiquetas que utilizan un prefijo bien definido (`<mywm:*>`), ocultando a las aplicaciones toda la complejidad relacionada con la gestión de múltiples contextos de entrega.

Hace uso de un lenguaje declarativo de interfaz de usuario muy rico en controles visuales, extendiendo al lenguaje DIAL estandarizado por el W3C. Además tiene un mecanismo para añadir al lenguaje nuevos controles visuales que pudieran ser implementados por los desarrolladores. Las aplicaciones creadas bajo esta herramienta poseen características que las hacen más flexibles de incorporar mecanismos de obtención de contenidos mediante servicios web, bases de datos y CMS (del inglés: Content Management System) que tengan implementación basada en la especificación JSR-170 (del inglés: Java Specification Request).

Provee de librerías que integran las aplicaciones con sistemas de transformación de contenidos multimedia como Alembik Transcoder¹⁵ y plugins para conectar las aplicaciones con el sistema de administración de contenidos de Alfresco Lab¹⁶.

Se pueden establecer gestores de flujos de eventos propios en las aplicaciones o hacer uso del implementado por el Framework el cual posee soporte para la utilización del lenguaje SCXML como fundamento para tratar completamente los eventos que ocurran dentro de las

¹⁵ Alembik Transcoder: <http://alembik.sourceforge.net/>

¹⁶ Alfresco Lab: <http://www.alfresco.com/es/>

aplicaciones móviles. Constituye el único framework de adaptación de contenidos/aplicaciones de software libre recomendado por el **mTLD** para el desarrollo de portales dotMobi.

Como característica de valor añadido, MyMobileWeb incorpora algunos módulos experimentales capaces de explotar la semántica en un entorno móvil, implementando el concepto de Web Móvil Semántica¹⁷ (4).

Existen varias empresas que implementan aplicaciones con MyMobileWeb como son Telefónica en la creación de una aplicación móvil de fuerzas de trabajo para operación y mantenimiento, Turismo Andaluz en su sitio andalucía.mobi, Revista EducaMadrid, Comunidad de Madrid, Mobile Google Earth Murcia (4).

Para dar cumplimiento al problema y los objetivos planteados, se seleccionó las herramientas que brinda la plataforma MyMobileWeb, para el desarrollo de la solución, ya que este trabajo responde a la necesidad de automatización de procesos a través de un portal WAP.

1.5 Metodología de desarrollo

Se entiende por metodología de desarrollo una colección de documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software (5). Una metodología es necesaria para el proceso de desarrollo de software; para obtener un software con calidad, en el tiempo esperado y con el coste planificado.

RUP (Rational Unified Procces)

El proceso unificado de desarrollo RUP es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software. Sus aspectos definitorios son dirigidos por casos de uso, centrado en la arquitectura, iterativo e incremental.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software. Los procesos son mucho más controlados con numerosas políticas y normas, es la metodología idónea para el desarrollo del proyecto.

¹⁷ W3C Semantic Web Activity: <http://www.w3.org/2001/sw>

La figura muestra la arquitectura global de RUP, en dos dimensiones:

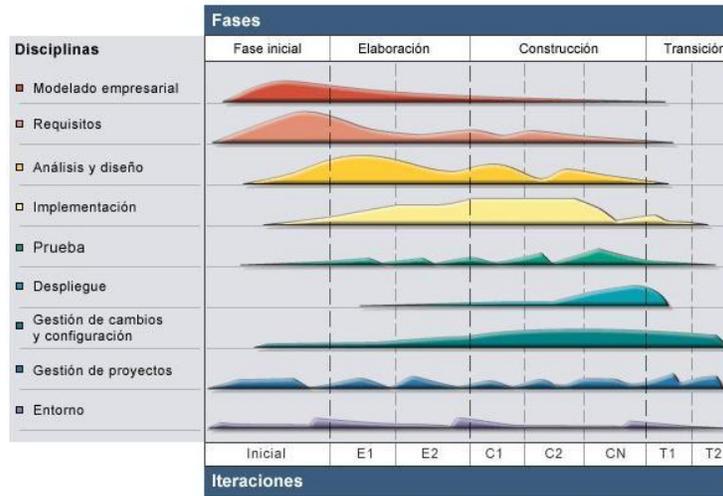


Figura 6 Proceso Unificado de Desarrollo de Software.

El proceso unificado de desarrollo de software utiliza el Lenguaje de Modelado Unificado (UML, Unified Modeling Language), para preparar todos los esquemas de un sistema de software (5). UML es un lenguaje para visualizar, especificar, construir y documentar un sistema de software, ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. La orientación a objetos es un complemento perfecto de UML (6).

Para modelar los procesos existente en el proyecto se decidió utilizar como metodología de desarrollo de software la metodología RUP pues es la que más se adapta a las condiciones del equipo de trabajo, un equipo grande con un cronograma de trabajo bastante amplio. Esta metodología genera la documentación que se necesita para validar los artefactos con el cliente, es una metodología muy organizativa y posee varios elementos de planificación que permiten llevar el control del desarrollo del proyecto. Asociado a RUP se escogió UML que es el lenguaje de modelado propuesto por esta metodología.

1.6 Herramienta CASE

Las herramientas CASE (Computer Aided Software Engineering) son un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de desarrollo de un Software permiten incrementar la

productividad y el control de calidad en cualquier proceso de elaboración de software, pues transforman la actividad de desarrollar software en un proceso automatizado.

Los objetivos fundamentales de estas herramientas son:

1. Permitir la aplicación práctica de metodologías, lo que resulta muy difícil sin emplear herramientas.
2. Facilitar la realización de prototipos y el desarrollo conjunto de aplicaciones.
3. Facilitar la reutilización de componentes de software (7).

Visual Paradigm para UML (VP-UML)

Visual Paradigm para UML es una herramienta que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Posee entre sus principales características las siguientes:

- ✓ **Generación de código e ingeniería inversa:** Brinda la posibilidad de generar código a partir de los diagramas, para plataformas como .Net, Java y PHP, así como obtener diagramas a partir del código.
- ✓ **Generación de documentación:** Brinda la posibilidad de documentar todo el trabajo sin necesidad de utilizar herramientas externas.
- ✓ **Disponibilidad en múltiples plataformas:** Microsoft Windows (98, 2000, XP, o Vista), Linux, Mac OS X, Solaris o Java.
- ✓ **Integración con distintos Ambientes de Desarrollo Integrados (IDE):** Se integra fácilmente con varios IDEs, entre ellos el de Visual Studio y el Eclipse.
- ✓ **Interoperabilidad con otras aplicaciones:** Brinda la posibilidad de intercambiar información mediante la importación y exportación de ficheros con aplicaciones como por ejemplo Visio y Rational Rose. Además permite importar y exportar XML.

Teniendo en cuenta lo antes expuesto la herramienta Case seleccionada para el modelado es Visual Paradigm, posee perfecta integración con el Eclipse, es una herramienta multiplataforma su principal ventaja para el proyecto es la licencia comercial que presenta, constituye una herramienta que facilita la realización de prototipos y el desarrollo de la aplicaciones (8).

1.7 Plataforma de Desarrollo

Se entiende por plataforma el entorno de software común en el cual se desenvuelve la programación de un grupo definido de aplicaciones, ya sea de hardware o software, sobre el cual un programa puede ejecutarse. Ejemplos típicos incluyen: arquitectura de hardware, sistema operativo, lenguajes de programación, Interfaz de programación de aplicaciones y sus librerías de tiempo de ejecución.

Plataforma Java

La plataforma Java es el nombre de un entorno o plataforma de computación originaria de la compañía Sun Microsystems, capaz de ejecutar aplicaciones desarrolladas usando el Lenguaje de programación Java u otros lenguajes. En este caso, la plataforma no es un hardware específico o un sistema operativo, sino más bien una máquina virtual encargada de la ejecución de aplicaciones, y un conjunto de librerías estándar que ofrecen funcionalidad común.

Hoy en día esta plataforma ha evolucionado en concordancia con el avance tecnológico y se ha convertido en una de las plataformas de programación más usadas por los desarrolladores. Su principal ventaja es que su entorno de desarrollo es independiente de la plataforma sobre la que se trabaje, es decir, sus aplicaciones son funcionales independientemente del sistema operativo sobre el que estén operando.

Es toda una tecnología orientada al desarrollo de software con el cual podemos realizar cualquier tipo de programa. Ha cobrado mucha importancia en el ámbito web con su plataforma JEE (Java Enterprise Edition). Java sigue creciendo en sus avances en la industria para dispositivos móviles logrando una gran acogida para este lenguaje. Es un lenguaje orientado a objetos que resuelve los problemas en la complejidad de los sistemas. Constituye una referencia a nivel mundial para el desarrollo de aplicaciones destinadas a un entorno para terminales móviles y especialmente para el framework MyMobileWeb. En el epígrafe siguiente se muestran varias características del marco de trabajo de Java.

1.8 Entorno de desarrollo

Un entorno de desarrollo integrado (IDE) es un ambiente de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica de usuario (GUI).

Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, Python, Java, C#, Delphi, Visual Basic. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto, como es el caso de Smalltalk u Objective-C.

Es posible que un mismo IDE pueda funcionar con varios lenguajes de programación. Este es el caso de Eclipse.

Eclipse

Eclipse es un software para integrar herramientas de desarrollo que corren sobre un amplio rango de sistemas operativos, con una arquitectura abierta y basada en plug-ins. La característica clave del Eclipse es su extensibilidad a través de los plug-ins que no son más que la mínima unidad de la plataforma que puede ser desarrollado por separado y que le aporta una nueva funcionalidad (9). La versión actual de Eclipse dispone de las siguientes características: pruebas unitarias con JUnit, control de versiones con CVS (del inglés: Concurrent Versions System), asistentes (wizards para creación de proyectos, clases, tests), refactorización.

Para darle solución al objetivo general se seleccionó Java como lenguaje de programación, con el entorno de desarrollo Eclipse Ganymede.

1.9 Herramienta para el Control de Versiones

Se llama control de versiones a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo. Los sistemas de control de versiones facilitan la administración de las distintas versiones de cada producto desarrollado, así como las posibles especializaciones realizadas. Un sistema de control de versiones proporciona mecanismos de almacenamiento y posibilita la edición para cada objeto bajo

control, además permite almacenamiento de la línea de cambios por los que ha pasado, de forma tal que posibilite ir a una versión anterior en condiciones determinadas.

SVN

Subversion es uno de los sistemas de control de versiones más modernos y utiliza un sistema con repositorio centralizado y fue diseñado como reemplazo del sistema más utilizado hasta la fecha, CVS (del inglés Concurrent Versions System). Algunas de sus principales características son: Se sigue la historia de los archivos y directorios a través de copias y renombrados. Las modificaciones son automáticas. Se envían solo las diferencias en ambas direcciones. Maneja eficientemente archivos binarios. Permite selectivamente el bloqueo de archivos (10).

Cliente de Subversion utilizado

RapidSVN es un cliente gratuito de software libre para el sistema de control de versiones Subversion. Los ficheros se almacenan en un repositorio central. En el repositorio se guardan todos los cambios que se hayan hecho a sus ficheros y directorios. Esto permite que pueda recuperar versiones antiguas de sus ficheros y examinar la historia de cuándo y cómo cambiaron sus datos, y quién fue el que realizó el cambio.

1.10 Conclusiones del Capítulo

En este capítulo se realizó un estudio sobre las principales herramientas para la creación y adaptación de sitios Web, para dispositivos móviles, teniendo en cuenta sus ventajas y desventajas. Se realizó un análisis de las soluciones brindadas por varias entidades a la problemática planteada, señalando sus principales características.

Después de realizar este proceso de investigación se definieron las herramientas y el framework provistos por MyMobileWeb como la tecnología a utilizar a lo largo del desarrollo del sistema propuesto, se fundamentó la elección de JAVA como el lenguaje de programación escogido, utilizando como metodología para el desarrollo de software RUP, con UML como lenguaje de modelado y utilizar la herramienta CASE Visual Paradigm para UML en la modelación de todo el proceso de desarrollo.

CAPÍTULO 2 CARACTERÍSTICAS DEL SISTEMA

2.1 Introducción

En este capítulo se tratarán las características generales del sistema, se realiza el proceso de captura de requisitos funcionales y no funcionales a los que se debe dar cumplimiento, se analizan los temas referentes a las necesidades de los clientes para obtener las bases y definir el objeto de automatización y la propuesta de solución que será desarrollada.

2.2 Objetivos Estratégicos

Cuba avanza paulatinamente en la informatización de la sociedad, como lo demuestra la extensión de servicios de telefonía móvil, o la digitalización de las comunicaciones en el país. Según datos de ETECSA, en Cuba existen más de 479 861 líneas de telefonía celular activas, incluyendo en ellas tanto las comercializadas en moneda libremente convertible, como las asignadas por interés social, entre ellas las de los Teléfonos Fijos Alternativos. A su vez, los planes de desarrollo prevén que en los próximos cinco años se llegue a más de 1,6 millones de usuarios de este servicio, para lo cual se han invertido cifras millonarias en el desarrollo de las infraestructuras necesarias (11) y en extender la cantidad de radio bases para lograr una mayor cobertura (ver figura 19 y 20, Anexo 1).

La cobertura de la red alcanza alrededor del 67,33 por ciento del territorio cubano y el 75,80 por ciento de la población, implementada sobre la base de tecnología GSM (del inglés: Global System for Mobile Communications). GSM constituye la red con mayor extensión en el mundo (13). A pesar de esto, la mayoría de los usuarios sólo cuentan con servicios básicos de voz y mensajería, estando reservada la navegación por Internet para uso corporativo. La generalidad de los usuarios no cuentan con equipos dotados con la tecnología suficiente para ello por sus altos precios, debido a que el país no puede hasta el momento y a causa del injusto bloqueo y leyes impuestas por el gobierno de los Estados Unidos de América, adquirirlos de fabricantes internacionales de reconocido prestigio como Nokia, Motorola o Samsung y tener que comprarlos a través de terceros.

Por otra parte, los sitios especializados en la venta de contenidos para celulares se encuentran distribuidos en Internet, lo que provoca que el acceso sea lento por las características en la conexión a Internet que posee Cuba y no contar con portales de este tipo dentro de la red nacional.

Capítulo 2. Características del Sistema

Todo esto evidencia la gran necesidad de crear portales WAP que permitan brindar múltiples servicios a los usuarios a través de sus móviles, (ejemplo la descarga de contenidos multimedia), y proveer de una alternativa práctica al acceso de la información, lo cual constituye una gran ventaja, pues la tasa de intercambio de datos no se vería afectada por los inconvenientes anteriormente mencionados, así como para las empresas debido a las posibilidades de trasladar sus contenidos y servicios de valor añadido presentes en sitios web convencionales a los terminales móviles, lo cual les permitirá llegar a un mayor número de clientes.

ETECSA necesita ampliar los servicios destinados a sus clientes y lograr la implantación de un mayor número de aplicaciones y servicios de valor agregado con el objetivo de obtener mayores beneficios económicos. Las grandes ganancias que reportan, entre ellos el de navegación y descarga de contenidos a través de portales WAP utilizando dispositivos móviles, han hecho que un gran número de compañías dirijan su atención hacia este mercado.

Para poder realizar una propuesta de sistema que supla las necesidades presentes en nuestro país y la entidad ETECSA, se hace necesario extraer los conceptos de mayor importancia dentro del dominio del problema planteado anteriormente.

2.3 Modelo de Dominio

Como se ha venido explicando, en Cuba no existe un portal WAP que brinde servicios a los usuarios de teléfonos celulares, por tanto se propone la realización de un Modelo de Dominio donde se mostrarán los principales conceptos a utilizar en el desarrollo de la aplicación. Esto ayuda a los usuarios, clientes, desarrolladores e interesados en general, a utilizar un vocabulario común para poder entender el contexto en que se ubica el sistema, para capturar correctamente los requisitos y poder construir una aplicación robusta que brinde los beneficios que de ella se esperan.

La figura muestra los principales conceptos y sus relaciones, presentes en el modelo de dominio.

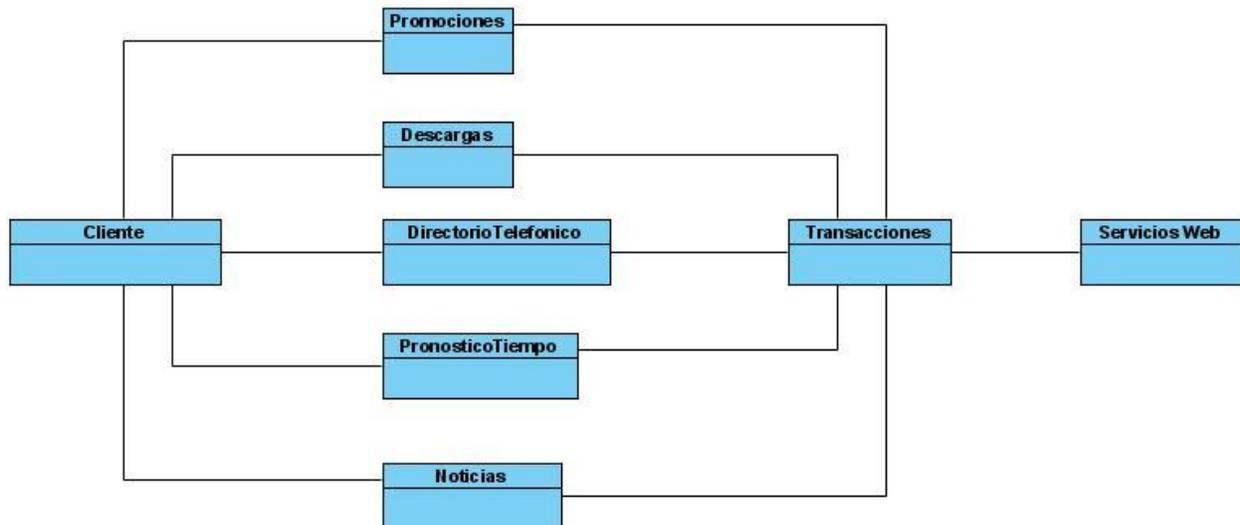


Figura 7 Modelo de Dominio.

2.3.1 Conceptos del Modelo de Dominio

Descripción de los conceptos representados en el Modelo del Dominio, los cuales simbolizan objetos del mundo real que se encuentran dentro del dominio del negocio, con la necesidad de comprender las relaciones que se establecen entre las diferentes clases.

Cliente: Este objeto representa el rol que juegan los diferentes usuarios de dispositivos móviles que cuentan con el servicio de navegación WAP.

Promociones: Define el conjunto de promociones y/o avisos que puedan ser de interés para los clientes.

Capítulo 2. Características del Sistema

Descargas: Se refiere al conjunto de contenidos multimedia que son gestionados por la plataforma y pueden ser descargados por los clientes hacia su dispositivo móvil.

Noticias: Representa las diferentes noticias a las que pudieran tener acceso los clientes que lo soliciten.

Directorio Telefónico: Está relacionado con la búsqueda de información sobre un directorio telefónico, específicamente sobre las principales entidades del país, (páginas amarillas) que pone a disposición la empresa ETECSA.

Pronóstico del Tiempo: Representa al conjunto de información vinculada con los reportes sobre el pronóstico del tiempo, emitido por el Instituto Nacional de Meteorología, que puedan ser de interés para los clientes.

Transacciones: Define la comunicación e intercambio de información que se establece entre las distintas clases que utilizan los servicios web gestionados por los diferentes componentes de la plataforma.

Servicios Web: Representa algunos servicios que se brindan en la plataforma, además de los servicios relacionados con las noticias, promociones, directorio telefónico, pronóstico del tiempo.

Una vez definidos los conceptos que forman parte del modelo de dominio, el siguiente paso es la formulación de una propuesta de sistema para guiar el proceso de elaboración de la solución.

2.4 Propuesta de Sistema

Se desea que los clientes con servicio de navegación accedan a través de un portal WAP de manera rápida y eficiente a los contenidos presentes en la Plataforma de Gestión de Contenidos para Dispositivos Móviles Gina, lo cual constituye el objeto de automatización dentro de la presente investigación.

Este portal será desarrollado sobre el framework MyMobileWeb el cual implementa el patrón arquitectónico Modelo Vista Controlador (MVC). Este framework se encargará de la coordinación entre los diferentes componentes de la aplicación móvil, donde la vista son las páginas que se visualizan en el navegador y los componentes que proveen los datos dinámicos a éstas. El modelo es la representación específica de la información la cual es administrada por

Capítulo 2. Características del Sistema

el sistema. El controlador es el responsable de recibir los eventos de entrada desde la vista, posibilitando diferentes ventajas para el sistema por ejemplo: la aplicación estará implementada modularmente, sus vistas siempre mostrarán información actualizada, las modificaciones a las vistas no afectan en absoluto a los demás módulos de la aplicación. Utilizando este patrón arquitectónico para el diseño del sistema se obtendrá un resultado final con gran flexibilidad y mantenibilidad.

En la siguiente figura se muestra la arquitectura propuesta para el portal WAP.

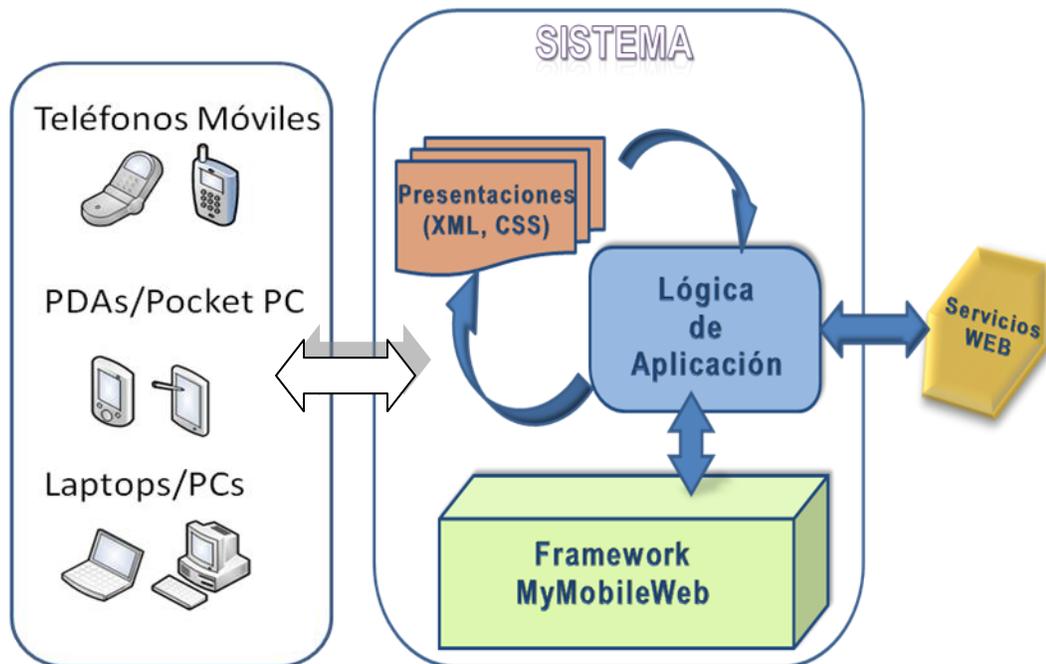


Figura 8 Arquitectura del portal WAP.

Se ilustra una representación de todos los posibles dispositivos que podrían visualizar y acceder al portal WAP, realizando un conjunto de peticiones a través de los protocolos de comunicación HTTP, las cuales serán capturadas por el sistema que posee la capacidad de procesar y transferir a las demás capas de la aplicación datos asociados a los eventos lanzados por el cliente, así como recuperar la información imprescindible para el reconocimiento de dispositivos mediante APIs específicas dentro del framework MyMobileWeb, el cual cuenta con un mecanismo que permite capturar la información presente en varios repositorios de dispositivos.

Entiéndase como lógica de aplicación todos los componentes que se encargan de gestionar el intercambio de información con los distintos servicios web y manipular el flujo de la aplicación.

Capítulo 2. Características del Sistema

A continuación se hace una descripción detallada de la arquitectura de ejecución que sigue el sistema propuesto:

En el lado del dispositivo, cuando se efectúa una interacción del usuario con la aplicación se genera un evento a través de una petición HTTP que debe ser tratado en el lado servidor, esa petición contiene:

- ✓ El identificador del control visual que ha lanzado el evento.
- ✓ El identificador del evento.
- ✓ Los datos que pueden haber sido introducidos por el usuario.

En el lado del servidor, la petición es inicialmente procesada por un componente controlador, el cual reconoce el dispositivo (si no se ha hecho anteriormente). A continuación, los datos son validados y almacenados en el contexto. El contexto es un contenedor que almacena los datos del modelo, y está estructurado jerárquicamente en diferentes ámbitos (aplicación, sesión, caso de uso, presentación). Existen dos tipos de eventos lanzados desde los navegadores, están los eventos específicos de la aplicación y los eventos propios de MyMobileWeb. Los eventos de aplicación tienen que ver con la funcionalidad del sistema y son tratados por métodos de clases Java, proporcionados por los desarrolladores.

Por otra parte existen eventos que son comunes a todas las aplicaciones y están relacionados con las funcionalidades proporcionadas por MyMobileWeb. Son manejados automáticamente por el framework, así que los desarrolladores de aplicaciones no tienen que preocuparse por ellos. Por ejemplo, un evento de página siguiente se lanza cuando un usuario está paginando sobre los contenidos de una tabla.

Los gestores de eventos concretos de la lógica de la aplicación deciden como procesar las peticiones entrantes. Por lo general, llamarán a métodos de clases java para obtener más datos que poner en el modelo y finalmente, realizarán una redirección a la siguiente vista (identificada por su nombre lógico). En este punto MyMobileWeb se encargará de localizar la página JSP (del inglés: Java Server Page) o Página Servidora de Java apropiada, de acuerdo con el dispositivo destino. Esta página JSP resolverá toda la inyección de datos y contenidos relacionados con la interfaz. El runtime del framework también resolverá aspectos comunes, tales como la paginación de contenidos, selección de las mejores imágenes, llamadas al transcodificador de imágenes y tratar con el gestor de literales de la aplicación.

2.5 Requerimientos del Sistema

Una vez que se realiza el modelado del negocio se puede comenzar a ejecutar el proceso de captura de requisitos del sistema. Los requisitos son la condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente software para satisfacer un contrato, estándar, u otro documento impuesto formalmente. A continuación se exponen los siguientes requisitos que debe cumplir la aplicación a desarrollar. La identificación de estos se ha realizado a partir de las necesidades reales planteadas por ETECSA.

2.5.1 Requerimientos Funcionales

Los requerimientos funcionales no alteran la funcionalidad del producto, lo cual quiere decir que estos se mantienen invariables sin importarle con que propiedades o cualidades se relacionen. Después de un análisis de las principales necesidades de los clientes y las características que el sistema debe cumplir se definió un conjunto de requisitos funcionales que se muestran a continuación.

RF1. Mostrar Contenido para Descargar

- RF1.1 Mostrar el listado de categorías disponibles.
- RF1.2 Mostrar el listado por clasificación de contenidos.
- RF1.3 Mostrar listado de contenidos.
- RF1.4 Mostrar las características del contenido seleccionado.
- RF1.5 Ejecutar petición de descarga.

RF2. Mostrar Directorio Telefónico

- RF2.1 Mostrar listado por categorías.
- RF2.2 Mostrar listado de provincias con entidades dentro de una categoría.
- RF2.3 Mostrar listado de entidades dado una provincia.
- RF2.4 Mostrar listado de números telefónicos y dirección de la entidad.
- RF2.5 Realizar llamada.

RF2.6 Realizar búsqueda.

RF2.6.1 Permitir insertar criterio de búsqueda en forma de texto.

RF3 Mostrar Promociones

RF3.1 Mostrar un listado de promociones.

RF3.2 Mostrar detalles de la promoción.

RF4 Mostrar Noticias

RF4.1 Mostrar listado de noticias por categorías.

RF4.2 Mostrar listado de resúmenes de noticias.

RF4.3 Mostrar texto de la noticia completa.

RF5 Mostrar Pronóstico del Tiempo

F5.1 Mostrar detalles del pronóstico del tiempo (fecha, hora, resumen, descripción detallada, imagen asociada).

2.5.2 Requerimientos no Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener para desarrollar un producto atractivo y con la calidad requerida.

Requerimientos de Software

Requerimientos mínimos para la máquina servidora:

- ✓ Disponer de Sistema Operativo Windows Server 2003 ó distribución Debian del sistema operativo Linux.
- ✓ Tener instalado la Máquina Virtual de Java versión 1.6.0 o superior.
- ✓ Contenedor Web que soporte la especificación Java Servlet 2.3 y JSP 1.2 o superiores.

Requerimientos mínimos para los terminales móviles:

- ✓ Contar con un navegador Web o cliente HTTP.

Requerimientos de Hardware

Requerimientos mínimos para la máquina servidora:

- ✓ Procesador Pentium IV 3.6 GHz o superior.
- ✓ Memoria RAM de 1GB.
- ✓ Espacio libre en disco duro 20 GB.

Restricciones en el diseño y la implementación

Para garantizar un producto de acuerdo con las necesidades planteadas por los clientes:

- ✓ Se utilizará el lenguaje de programación Java, el cual constituye un lenguaje muy extendido en aplicaciones web y particularmente para móviles.
- ✓ Mostrar las páginas en el lenguaje de marcado soportado por el navegador del dispositivo móvil destino como XHTML, HTML, WML, CHTML.
- ✓ El análisis y el diseño de la aplicación se harán bajo las condiciones que impone la Metodología RUP, utilizando como lenguaje de modelado a UML.
- ✓ Como herramienta CASE se utilizará Visual Paradigm para realizar el modelado de los artefactos que se obtienen como consecuencia de los flujos de trabajo que define RUP.

Requerimientos de apariencia o interfaz externa

- ✓ El contenido que solicita el usuario debe estar limitado a las restricciones del dispositivo.
- ✓ La interfaz externa del producto debe estar regida a las propuestas por las buenas prácticas de diseño Web planteadas por el W3C.
- ✓ Estar adaptada al ambiente de Cuba debe cumplir los siguientes aspectos, idioma en español, el logo de la entidad aparecer en la parte superior de las presentaciones.
- ✓ Usar la menor cantidad de imágenes y en el formato más idóneo de acuerdo a las características del móvil, además de tener buena calidad visual y ser comprensibles.
- ✓ Se contará en la página principal con la información necesaria para guiar al cliente en su navegación básica, propiciando que el usuario tenga éxito.

- ✓ Cuando se usan menús o listas seleccionables con un gran número de opciones deben ser paginadas y mostrar al usuario el total de páginas sobre las que puede iterar.

Requerimientos de Usabilidad

- ✓ La información debe poder ser accedida por todos los usuarios de móviles que tengan habilitado servicio de navegación, sin poder modificar el contenido de las páginas.
- ✓ Proporcionar solamente una navegación mínima en las presentaciones, cualquier otro elemento de navegación secundario se puede poner al pie de la página si realmente es necesario.
- ✓ Es importante que los usuarios puedan ver el contenido de una vez cuando la misma haya cargado, sin la necesidad de realizar scroll y paginados excesivos.
- ✓ No generar “pop-ups” (cambios de ventana) sin avisarle al usuario, muchos dispositivos móviles no pueden soportar más de una ventana abierta y en ocasiones al usuario puede incomodarle esta situación.

Requerimientos de Rendimiento

- ✓ La aplicación debe emplear mecanismo de cache sobre el contexto de cada dispositivo y los archivos multimedia que se hallan transcodificado, para mejorar el rendimiento de sobrecarga en los servidores.
- ✓ El tiempo de respuesta debe estar comprendido entre 0 a 5 segundos (acotar que el rendimiento está dado por el nivel de latencia en la red, características del móvil y del browser).

A manera de resumen podemos expresar que los requisitos especificados representan la base para el Modelo de Casos de Uso del sistema. En el próximo epígrafe se enumeran los actores del sistema y una descripción de los caso de uso identificados.

2.6 Modelo del Sistema

Una vez que se identificaron y se clasificaron los requerimientos del sistema es necesario identificar los actores y casos de usos. Los casos de usos son priorizados y detallados, realizando la descripción de cada uno de ellos.

2.6.1 Definición de los actores del sistema

En el sistema se identificaron tres actores, los cuales están relacionados con los diferentes casos de usos beneficiándose de ellos.

Tabla 1 Definición de los Actores del Sistema.

Actor	Descripción
Cliente	Visualiza los servicios y contenidos que ofrece la plataforma beneficiándose con los mismos. Puede navegar por el Portal WAP y hacer uso de los servicios disponibles en él.
M_Contenido	Constituye un sistema automatizado externo que interactúa con la aplicación, encargado de proveer los contenidos vinculados al servicio de descargas.
M_EntregaContenido	Constituye un sistema automatizado externo que interactúa con la aplicación, encargado de ejecutar la acción de descarga de contenidos.
Recurso	Constituye un sistema automatizado externo que interactúa con el sistema, proporciona los contenidos presentes en la plataforma Gina, exceptuando los relacionados a las funcionalidades de descarga. Estos contenidos son (Noticias, Promociones, Pronóstico del Tiempo, Directorio Telefónico)

Una vez descrito los actores se realiza el proceso de identificación de los casos de uso apoyados en los requisitos del sistema, los casos de uso identificados fueron:

Casos de uso base.

- ✓ CU_MostrarNoticias.
- ✓ CU_MostrarPromociones.
- ✓ CU_MostrarContenidoDescarga.
- ✓ CU_MostrarDirectorioTelefonico.
- ✓ CU_MostrarPronosticoTiempo.

Casos de uso extendidos.

- ✓ CU_Buscar.
- ✓ CU_RealizarLlamada.
- ✓ CU_EjecutarDescarga.

2.6.2 Diagrama de Casos de Uso del sistema

Una vez que se han identificado los actores y casos de usos se puede estructurar el diagrama de casos de uso del sistema. Este representa gráficamente a los procesos y su interacción con los actores.

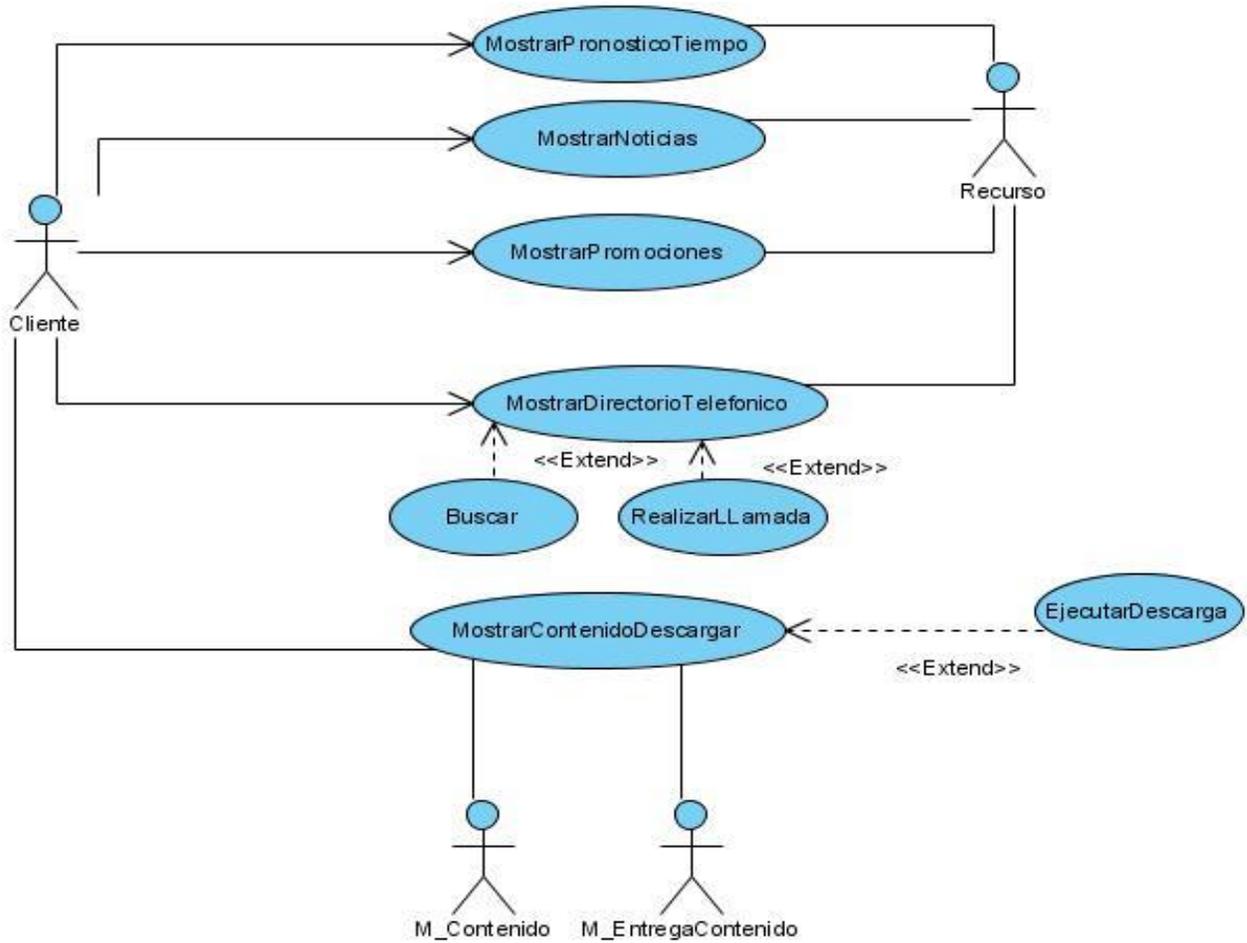


Figura 9 Diagrama de Casos de Uso del Sistema.

2.6.3 Descripción Detallada de los Casos de Uso del Sistema

Para entender la funcionalidad asociada a cada caso de uso no es suficiente con la representación gráfica del diagrama de casos de uso. La descripción puede ser elaborada de forma breve o extendida para lograr un mayor entendimiento sobre lo que debe realizar el sistema y ayudar a su mejor comprensión.

Tabla 2 Descripción CU_MostrarPronosticoTiempo.

Caso de Uso:	MostrarPronosticoTiempo.	
Actores:	Cliente, Recurso.	
Resumen:	Permite al cliente navegar por los contenidos referentes al estado del tiempo,	
Referencias:	R.5	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El cliente accede al portal WAP.	2. Muestra una interfaz que contiene las siguientes opciones: <ul style="list-style-type: none"> • Noticias. • Teléfonos. • El Tiempo. • Descargas. • Promociones. 	
3. El cliente selecciona la opción del menú principal de navegar hacia el "El Tiempo".	4. Obtiene el id de la opción seleccionada.	
	5. Solicita al actor Recurso los siguientes datos del pronóstico del tiempo para ese día (la solicitud se ejecuta a través de la llamada a un método del web service): <ul style="list-style-type: none"> • Fecha. • Hora. • Resumen. • Descripción detallada. • Imagen asociada. 	
6.El Recurso envía la información correspondiente del pronóstico del tiempo: <ul style="list-style-type: none"> • Fecha. • Hora. 	7. Muestra una interfaz con la información enviada.	

Capítulo 2. Características del Sistema

<ul style="list-style-type: none"> • Resumen. • Descripción detallada. • Imagen asociada. 	
--	--

Tabla 3 Descripción CU_MostrarContenidoDescargar.

Caso de Uso:	MostrarContenidoDescargar, EjecutarDescarga	
Actores:	Cliente, M_Contenido, M_EntregaContenido.	
Resumen:	Permite al cliente navegar por todos los contenidos disponibles y realizar su descarga.	
Referencias:	R.1	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema
	1. El cliente accede al portal WAP.	2. Muestra una interfaz que contiene las siguientes opciones: <ul style="list-style-type: none"> • Noticias. • Teléfonos. • El Tiempo. • Descargas. • Promociones.
	3. El cliente selecciona la opción de “Descargas” del menú principal.	4. Obtiene el id de la opción seleccionada.
		5. Solicita al actor M_Contenido obtener el listado de todas las categorías de contenidos disponibles <ul style="list-style-type: none"> • Ringtones. • Images. (la solicitud realizada se ejecuta a través de la llamada a un método del web service).
	6. El M_Contenido envía el listado de categorías solicitada por el sistema. <ul style="list-style-type: none"> • Ringtones. • Images. 	7. Muestra una interfaz que contiene todas las categorías de contenidos disponibles para realizar la descarga. <ul style="list-style-type: none"> • Ringtones. • Images.
	8. El cliente selecciona una categoría.	9. Obtiene el id de la categoría seleccionada.
		10. Solicita al actor M_Contenido obtener el

Capítulo 2. Características del Sistema

	<p>listado de todas las clasificaciones de contenidos disponibles asociada a una categoría</p> <p>Imges:</p> <ul style="list-style-type: none">• Art.• Nature.• Buildings.• Space.• Stuff. <p>Ringtones:</p> <ul style="list-style-type: none">• Pop.• Rock.• Hip-Hop.• Misc. <p>(La Solicitud realizada se ejecuta a través de la llamada a un método del web service).</p>
<p>9. El M_Contenido envía un listado con las clasificaciones de contenidos pertenecientes a una categoría.</p> <p>Imges:</p> <ul style="list-style-type: none">• Art.• Nature.• Buildings.• Space.• Stuff. <p>Ringtones:</p> <ul style="list-style-type: none">• Pop.• Rock.• Hip-Hop.• Misc.	<p>10. Muestra una interfaz con el listado de clasificaciones pertenecientes a una categoría.</p>
<p>11. El cliente selecciona una clasificación de contenidos.</p>	<p>12. Obtiene el id de la clasificación.</p>
	<p>13. Solicita al actor M_Contenido el listado de nombres de contenidos disponibles que pertenecen a la clasificación seleccionada por el usuario.</p> <p>(La solicitud se ejecuta a través de la llamada a</p>

Capítulo 2. Características del Sistema

	un método del web service).
14. El M_Contenido envía al sistema un listado con los nombres de todos los contenidos pertenecientes a la clasificación seleccionada.	15. Muestra la interfaz con el listado de nombres de los contenidos disponibles que pertenecen a la clasificación seleccionada por el usuario.
16. El cliente selecciona un contenido para obtener su información detallada.	17. Obtiene el id del contenido.
	16. Solicita al M_Contenido el contenido a través del identificador obtenido. (La solicitud se ejecuta a través de la llamada a un método del web service al cual se le pasa como parámetro el id).
17. El M_Contenido le envía el contenido solicitado el cual coincide con el identificador enviado por el sistema.	18. Muestra una interfaz con los datos relacionados al contenido seleccionado y la opción de "Descargar". <ul style="list-style-type: none"> • Nombre. • Precio. • Autor. • Imagen.
Flujo Normal de Eventos Sesión 1(EjecutarDescarga)	
1.1 El cliente solicita descargar un contenido.	1.2 Conformar la solicitud de descarga con los datos correspondientes <ul style="list-style-type: none"> • Número telefónico. • User_agent. • id_cont.
	1.3 Envía la solicitud conformada hacia el M_EntregaContenido de la plataforma.
	1.4 Muestra una interfaz que contiene un texto de congratulaciones.

Capítulo 2. Características del Sistema

Tabla 4 Descripción CU_MostrarNoticias.

Caso de Uso:	MostrarNoticias
Actores:	Cliente, Recurso.
Resumen:	Permite al cliente navegar por todas las noticias y obtener un breve resumen de ellas.
Referencias:	R4
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El cliente accede al portal WAP.	2. Muestra una interfaz que contiene las siguientes opciones: <ul style="list-style-type: none"> • Noticias. • Teléfonos. • El Tiempo. • Descargas. • Promociones.
3. El cliente selecciona la opción "Noticias" del menú principal.	4. Obtiene el id de la opción seleccionada.
	5. Solicita al actor Recurso el listado de las noticias por categorías las cuales pueden ser: <ul style="list-style-type: none"> • Deportivas. • Nacionales. • Internacionales. • Culturales. • Economía. • Ciencia. (La solicitud se ejecuta a través de la llamada a un método del web service).
6. El cliente selecciona una categoría.	7. Obtiene el id de la categoría seleccionada.
	6. Solicita al actor Recurso el listado de noticias correspondiente a la categoría seleccionada mediante el identificador. (La solicitud se ejecuta a través de la llamada a un método del web service al cual se le pasa como parámetro el id de categoría).
7. El Recurso le envía el listado con las noticias correspondientes a la categoría seleccionada.	8. Muestra una interfaz con un listado de noticias conformado:

Capítulo 2. Características del Sistema

	<ul style="list-style-type: none"> • Resumen. • Opción “Detalle de la noticia”.
9. El cliente selecciona una noticia.	10. Obtiene el id de la noticia seleccionada.
	11. Solicita al actor Recurso la noticia seleccionada mediante el identificador. (La solicitud se ejecuta a través de la llamada a un método del web service al cual se le pasa como parámetro el id de noticia).
11. El Recurso le envía el texto completo de la noticia correspondiente.	10. Muestra el texto completo de la noticia seleccionada en una interfaz.

Tabla 5 Descripción CU_MostrarPromociones.

Caso de Uso:	MostrarPromociones
Actores:	Cliente, Recurso.
Resumen:	Permite al cliente navegar por todos las promociones publicadas en el portal WAP.
Referencias:	R3
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El cliente accede al portal WAP.	2. Muestra una interfaz que contiene las siguientes opciones: <ul style="list-style-type: none"> • Noticias. • Teléfonos. • El Tiempo. • Descargas. • Promociones.
3. El cliente selecciona la opción “Promociones” del menú principal.	4. Obtiene el id de la opción seleccionada.
	5. Solicita al actor Recurso el listado de las promociones disponibles. (La solicitud se ejecuta a través de la llamada a un método del web service).
5. El actor Recurso le envía los detalles de las promociones solicitadas. <ul style="list-style-type: none"> • Título. • Texto. • Ccompleto. 	6. Muestra la interfaz con el listado de títulos de las promociones disponibles.

Capítulo 2. Características del Sistema

7. El cliente selecciona la promoción que desea ver.	8. Se muestra en una interfaz la promoción solicitada (Texto).
---	--

Tabla 6 Descripción CU_MostrarDirectorioTelefonico.

Caso de Uso:	MostrarDirectorioTelefonico	
Actores:	Cliente, Recurso.	
Resumen:	Permite al cliente navegar por el directorio telefónico, realizar búsquedas y llamadas.	
Referencias:	R2	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema
	1. El cliente accede al portal WAP.	2. Muestra una interfaz que contiene las siguientes opciones: <ul style="list-style-type: none"> • Noticias. • Teléfonos. • El Tiempo. • Descargas. • Promociones.
	3. El cliente accede a la opción “Teléfonos” presente en el menú principal del portal WAP.	4. Obtiene el id de la opción seleccionada.
		5. Solicita al actor Recurso los datos relacionados con el listado de categorías de entidades disponibles en el directorio telefónico de páginas amarillas de ETECSA. (La solicitud se ejecuta a través de la llamada a un método del web service).
	5. Recurso envía el listado con las categorías de las entidades.	6. Muestra la interfaz con un listado de las categorías de entidades que están presentes en el directorio telefónico (Estas entidades están ordenadas en orden alfabético y son las mismas que se encuentran en las páginas amarillas de las guías telefónicas proporcionadas por ETECSA).
	7. El cliente selecciona la categoría que desea.	8. Obtiene el id de la categoría seleccionada.
		9. Solicita al actor Recurso el listado de las provincias que poseen entidades que se

Capítulo 2. Características del Sistema

	<p>incluyen dentro de esa categoría mediante el identificador de la categoría seleccionada.</p> <p>(La solicitud se ejecuta a través de la llamada a un método del web service al cual se le pasa el id de la categoría como parámetro).</p>
10. Recurso envía el listado de provincias que poseen alguna entidad dentro de la categoría seleccionada.	11. Muestra una interfaz donde aparece el listado de provincias que poseen entidades que se incluyen dentro de la categoría elegida por el cliente.
12. El cliente selecciona la provincia.	13. Obtiene el id de la provincia seleccionada.
	<p>14. Solicita al actor Recurso el listado de entidades que pertenecen a la categoría seleccionada mediante el identificador de la categoría y de la provincia a la que pertenece (id de la provincia, id de categoría).</p> <p>(La solicitud se ejecuta a través de la llamada a un método del web service al cual se le pasa el id de la categoría y el id de la provincia como parámetros).</p>
15. Recurso envía el listado de las entidades correspondiente a esa provincia dentro de la categoría seleccionada.	<p>16. Muestra una interfaz con el conjunto de entidades relacionadas a una categoría perteneciente a la provincia seleccionada por el cliente.</p> <ul style="list-style-type: none"> • Nombre. • Dirección.
17. El cliente selecciona una entidad.	18. Obtiene el id de la entidad seleccionada.
	<p>19. Solicita al actor Recurso los datos de la entidad seleccionada a través del identificador.</p> <ul style="list-style-type: none"> • Nombre. • Dirección. • Listado de teléfonos. <p>(La solicitud se ejecuta a través de la llamada a un método del web service al cual se le pasa el id de la entidad como parámetro).</p>
20. El Recurso envía los datos solicitados.	21. Muestra una interfaz con los datos.
<ul style="list-style-type: none"> • Nombre. 	<ul style="list-style-type: none"> • Nombre.

Capítulo 2. Características del Sistema

<ul style="list-style-type: none">• Dirección.• Listado de teléfonos.	<ul style="list-style-type: none">• Dirección.• Listado de teléfonos.
Flujo Normal de Eventos Sesión 1 (Realizar Búsqueda)	
Acción del Actor	Respuesta del Sistema
1.1 El cliente solicita realizar búsqueda a través del directorio telefónico.	1.2 Se muestra un interfaz con los controles visuales que permiten introducir el criterio de búsqueda (mínimo 3 caracteres).
1.3 El cliente introduce un texto como criterio y solicita realizar la búsqueda.	1.4 Según el texto introducido solicita al actor Recurso un listado con todas las provincias que cuenten con alguna entidad cuyo nombre contenga el texto entrado por el usuario.
1.5 El Recurso envía el listado de provincias.	1.6 Muestra la interfaz con un listado de las provincias que tienen entidades cuyo nombre cuenta con esta palabra (Seguir el flujo normal de eventos a partir de 11).
Flujo Normal de Eventos Sesión 2 (Ejecutar Llamada)	
Acción del Actor	Respuesta del Sistema
2.1 El cliente activa la opción de realizar llamada mediante el número de la entidad.	2.2 Activa la función de llamada hacia el número telefónico seleccionado por el usuario.

2.7 Conclusiones del Capítulo

A lo largo de este capítulo se ha desarrollado la propuesta de solución basado en un análisis crítico de la situación a nivel internacional y en Cuba en cuanto a la implementación de portales WAP. Se obtuvieron los principales conceptos del dominio y se registraron las funcionalidades que deben ser desarrolladas en el proceso de captura de requisitos, realizándose una descripción de cada uno de estos. Se identificaron y describieron los actores y casos de uso estableciéndose las relaciones correspondientes entre cada uno de ellos en el diagrama de casos de uso.

CAPÍTULO 3 ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 Introducción

El objetivo principal de este capítulo es realizar la descripción de los procesos que se llevan a cabo en la disciplina de análisis y diseño, obteniendo como resultado los artefactos más importantes para modelar el sistema, teniendo en cuenta los patrones de diseño que aportan soluciones concretas a problemas específicos, para lograr un diseño eficaz en el software orientado a objetos.

3.2 Modelo de Análisis

El Modelo de análisis contiene clases del análisis y sus objetos organizados en paquetes que colaboran. Las clases de análisis se centran en los requisitos funcionales y son evidentes en el dominio del problema porque representan conceptos y relaciones del dominio. Se inserta además la realización de casos de uso del análisis que describe cómo se lleva a cabo y se ejecuta un caso de uso determinado en término de las clases del análisis y de sus objetos en interacción. Contiene además los paquetes de análisis que organizan los artefactos del análisis en piezas manejables así como la descripción de la arquitectura (vista del modelo de análisis) que muestra los artefactos significativos para la arquitectura del sistema propuesto.

Para modelar el sistema se realizó un diagrama de clases de análisis y diagrama de interacción por cada caso de uso presente en el dominio del problema. A continuación se muestra una representación del diagrama de clase del análisis correspondiente al caso de uso (Mostrar Contenido para Descarga), el cual constituye el caso de uso principal del sistema.

3.2.1 Diagrama de Clases de Análisis

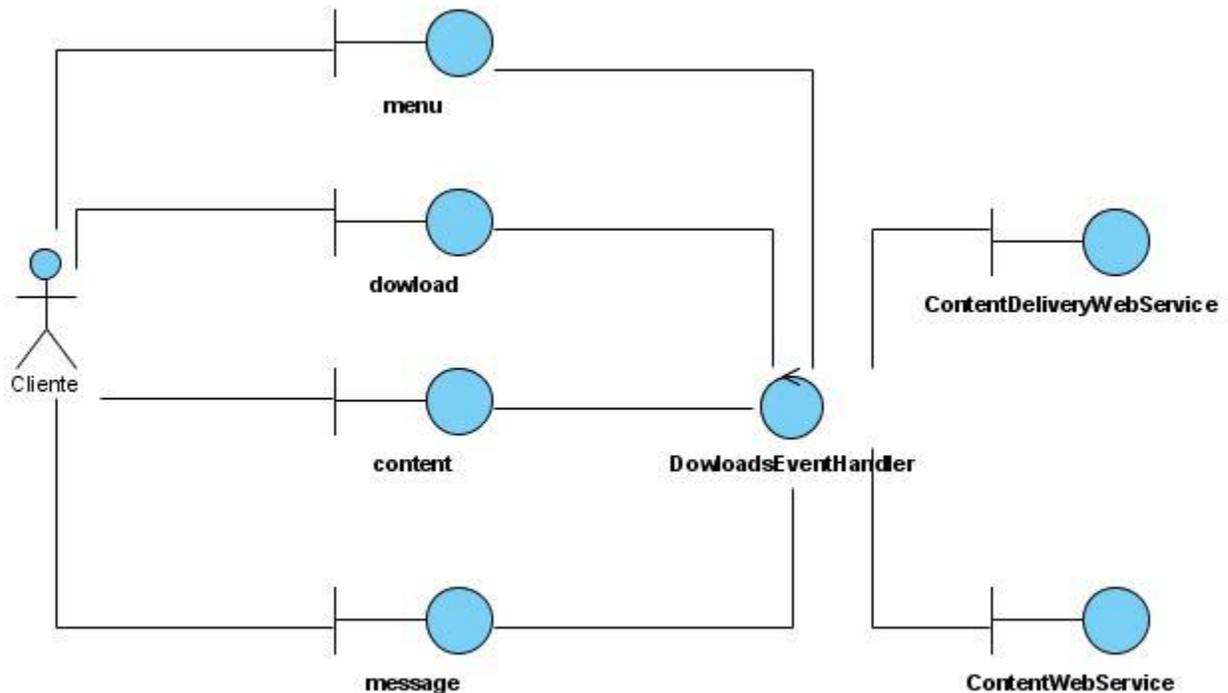


Figura 10 Diagrama de Clases de Análisis CU_MostrarContenidoParaDescargar.

Para establecer una vista dinámica de la interacción entre las clases, las relaciones y las llamadas entre cada una de las clases se realizan los diagramas de secuencia y colaboración del análisis. (Anexo 2. Diagramas de Secuencia del Análisis.)

3.3 Modelo de Diseño

El modelo de diseño, es un modelo de objetos que describe la realización física de los casos de uso, centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Además el modelo de diseño sirve de abstracción de la implementación del sistema y es de ese modo utilizado como una entrada fundamental de las actividades de implementación. Los casos de uso son realizados por las clases de diseño y sus objetos. Esto se representa por colaboraciones en el modelo de diseño y denota realización de caso de uso-diseño (5).

El modelo de diseño está compuesto por los artefactos clases del diseño, realización caso de uso-diseño, diagramas de interacción, diagramas de clases, subsistemas del diseño, interfaz, descripción de la arquitectura (vista del modelo de diseño), modelo de despliegue, descripción de la arquitectura (vista del modelo de despliegue).

A continuación se muestran dos de los diagramas de clases del diseño Web arquitectónicamente significativos para la realización del sistema.

3.3.1 Diagrama de Clases del Diseño

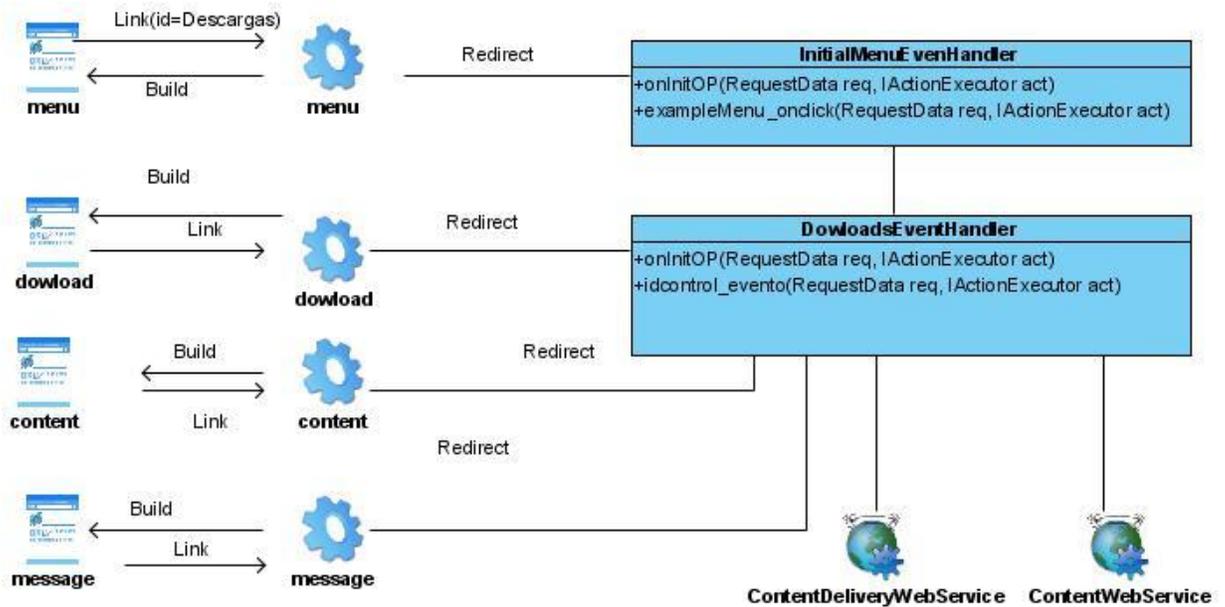


Figura 11 Diagrama de Clases de DiseñoWeb CU_MostrarContenidoParaDescargar.

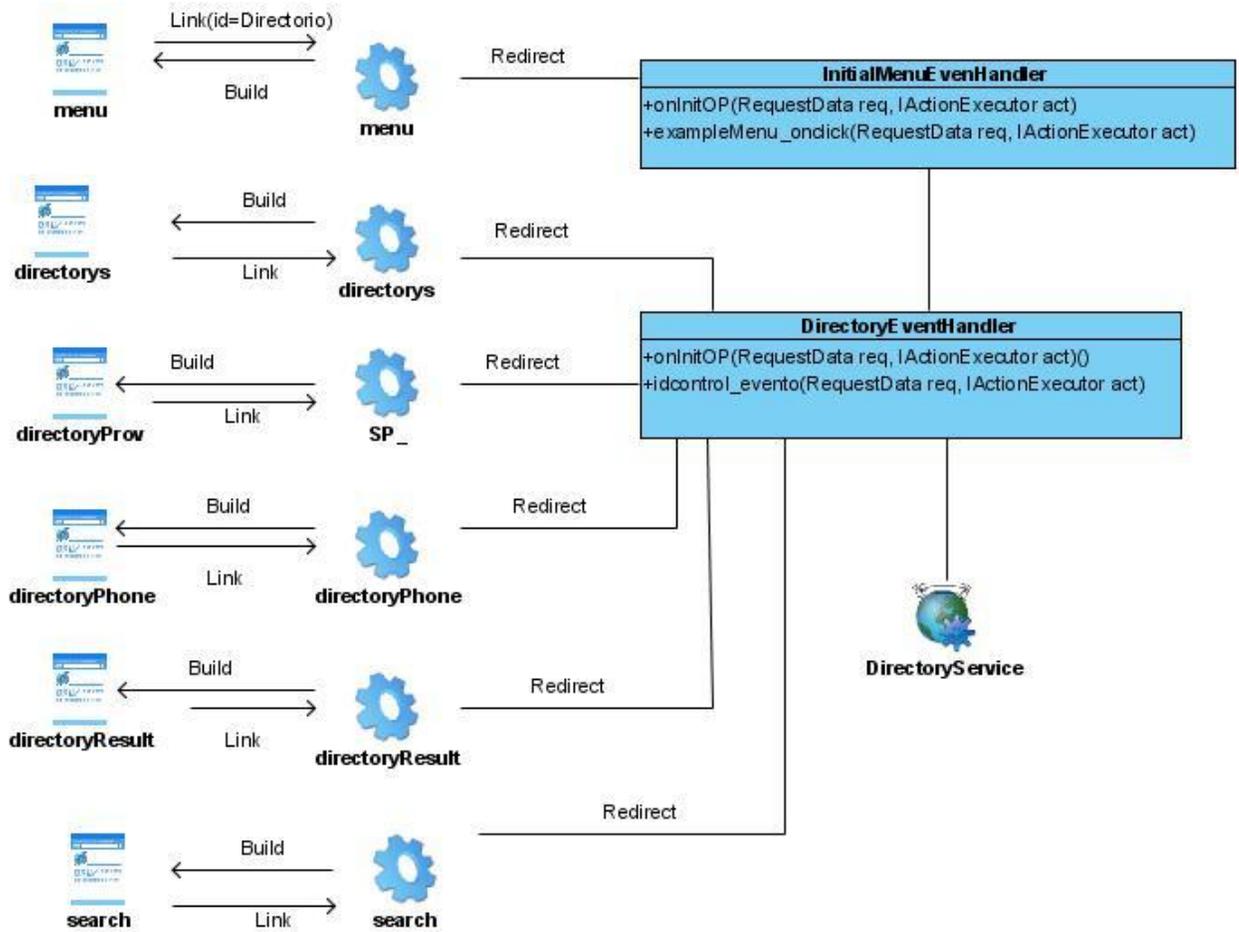


Figura 12 Diagrama de Clases de DiseñoWeb CU_MostrarDirectorioTelefonico.

En este trabajo se presentan los diagramas de clases del diseño web y los diagramas de secuencia, para cada uno de los casos de uso identificados, con el fin de mostrar una vista de la interacción entre los objetos, y sus relaciones con los mensajes que se envían entre ellos. (Anexo 2. Diagramas de Secuencia del Diseño y Diagramas de clases del Diseño Web). La creación de los diagramas de interacción requiere la aplicación de los principios para la asignación de responsabilidades y el uso de los principios y patrones de diseño. Es por ello que en el epígrafe siguiente se explican los patrones de diseño utilizados en el desarrollo del sistema.

3.4 Patrones de Diseño

El patrón es una pareja de problema/solución con un nombre y que es aplicable a otros contextos, con una sugerencia sobre la manera de usarlo en situaciones nuevas (12).

Un sistema orientado a objetos se compone de objetos que envían mensajes a otros objetos para que lleven a cabo las operaciones. Las decisiones poco acertadas dan origen a sistemas y

componentes frágiles y difíciles de mantener, entender, reutilizar o extender. Una buena implementación se funda en los principios cardinales que rigen un buen diseño orientado a objetos. Los patrones GRASP (patrones generales de software para asignar responsabilidades) se aplican al preparar los diagramas de interacción, cuando se asignan las responsabilidades o durante ambas actividades (12).

Patrones GRASP utilizados en el desarrollo del sistema.

Bajo Acoplamiento:

El acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases, como las conoce y como recurre a ellas. Una clase con bajo (o débil) acoplamiento no depende de muchas otras. Una clase con alto (o fuerte) acoplamiento recurre a muchas otras. Este tipo de clases no es conveniente porque presentan los siguientes problemas:

- ✓ Los cambios de las clases afines ocasionan cambios locales.
- ✓ Son más difíciles de entender cuando están aisladas.
- ✓ Son más difíciles de reutilizar porque se requiere la presencia de otras clases de las que dependen.

Alta Cohesión:

En la perspectiva del diseño orientado a objetos, la cohesión es una medida de cuan relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme. Una clase con baja cohesión hace muchas cosas no afines o un trabajo excesivo. No conviene este tipo de clases pues presentan los siguientes problemas:

- ✓ Son difíciles de comprender, de reutilizar, de conservar.
- ✓ Son delicadas, las afectan constantemente los cambios.

Controlador:

Encargado de manejar los eventos del sistema que representan la visión general del sistema o de un caso de uso. Si se recurre a un diseño orientado a objetos, hay que elegir los controladores que manejen esos eventos de entrada. Este patrón ofrece una guía para tomar decisiones apropiadas que generalmente se aceptan. Un defecto frecuente al diseñar controladores consiste en asignarles demasiada responsabilidad. Normalmente un controlador debería delegar a otros objetos el trabajo que ha de realizarse mientras coordina la actividad.

Patrones GoF

Otros patrones que brindan soluciones para sistema orientados a objetos son los denominados patrones "Gang of Four" (GoF) conocidos con ese nombre por ser cuatro los autores del libro "Design Patterns". Se clasifican en tres grupos, los de creación que abstraen el proceso de creación de instancias, los estructurales se ocupan de cómo clases y objetos son utilizados para componer estructuras de mayor tamaño, y de comportamiento que tratan sobre los algoritmos y la asignación de responsabilidades entre objetos.

Patrones Creacionales

Singleton (Instancia única): Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. Se evidencia en la clase ConfigurationServices, encargada de crear las instancias de cada servicio web que se consumen en la aplicación. Aquí se utiliza la clase SingletonHelper para hacer transparente en toda la aplicación el acceso o creación de una instancia única de la clase ConfigurationServices.

Patrones de Comportamiento

Chain of Responsibility (Cadena de responsabilidad): Permite establecer la línea que deben llevar los mensajes para que los objetos realicen la tarea indicada. Evita el acoplamiento entre el emisor y el receptor de una petición brindando la posibilidad de que más de un objeto pueda manejar la petición, envía la petición a través de la cadena de objetos hasta que alguno de ellos maneje la petición. Se evidencia en la transferencia de las responsabilidades de las clases manejadoras de evento dentro de la aplicación, ya que el cliente conoce a un gestor, en este caso el gestor de flujo de la aplicación. Éste es el que lanza la petición a la cadena, hasta que alguien la trate, siguiendo una jerarquía (manejador de presentación, manejador de caso de uso, manejador de aplicación).

En el desarrollo del portal, se decidió el uso de presentaciones agradables y sencillas, con un sistema de navegación que oriente al usuario, a continuación se muestran un conjunto de prototipos de interfaz de usuario los cuales muestran las presentaciones que serán visualizadas por el usuario.

3.6 Prototipos de Interfaz

Para el desarrollo de la aplicación se decidió crear una interfaz de usuario amigable, sencillo y con un apropiado patrón de navegación para dispositivos móviles. Los prototipos propuestos se rigen por un conjunto de patrones de diseño Web conocidos como Buenas Prácticas para el diseño Web Móvil, especificado por el W3C. Para lograr un sistema que satisfaga los requerimientos de los clientes se tuvo en cuenta siempre colocar en la parte superior central de las presentaciones el logo de la empresa, al pie de todas las páginas presenta un menú básico de navegación (atrás, inicio), un fondo blanco para lograr una buena visibilidad de los contenidos. Las imágenes en varios formatos y tamaños dependiendo de las características del dispositivo móvil.

Se limita lo mayor posible la entrada de texto por parte de los usuarios para lograr una mayor aceptación. Los menús de navegación que presentan gran número de atributos son paginados mostrando siempre en la parte inferior enlaces de navegación hacia algunos elementos dentro del menú y el número total de pantallas que lo conforman. Las imágenes se mostrarán siempre que el dispositivo destino las soporte y se cambiarán sus dimensiones y formatos a los adecuados para su visualización en los navegadores móviles.

Figura 13 Presentación con el Logo.



Figura 15 Menú Principal.



Figura 17 Directorio Telefónico.



Figura 14 Presentación de Búsqueda.



Figura 16 Menú de Descargas.



Figura 18 Descargar Contenido.



El prototipo se encuentra diseñado para dar cumplimiento a los requerimientos tanto funcionales como no funcionales expuestos en el apartado “2.7 Requerimientos del Sistema”, en cuanto a apariencia y usabilidad se refiere, así lo demuestra la siguiente figura que describe el mapa de navegación del sistema en la que se evidencia una buena distribución de los elementos que lo conforman para lograr que los usuarios accedan al objeto de su interés de forma rápida y sin ambigüedades.

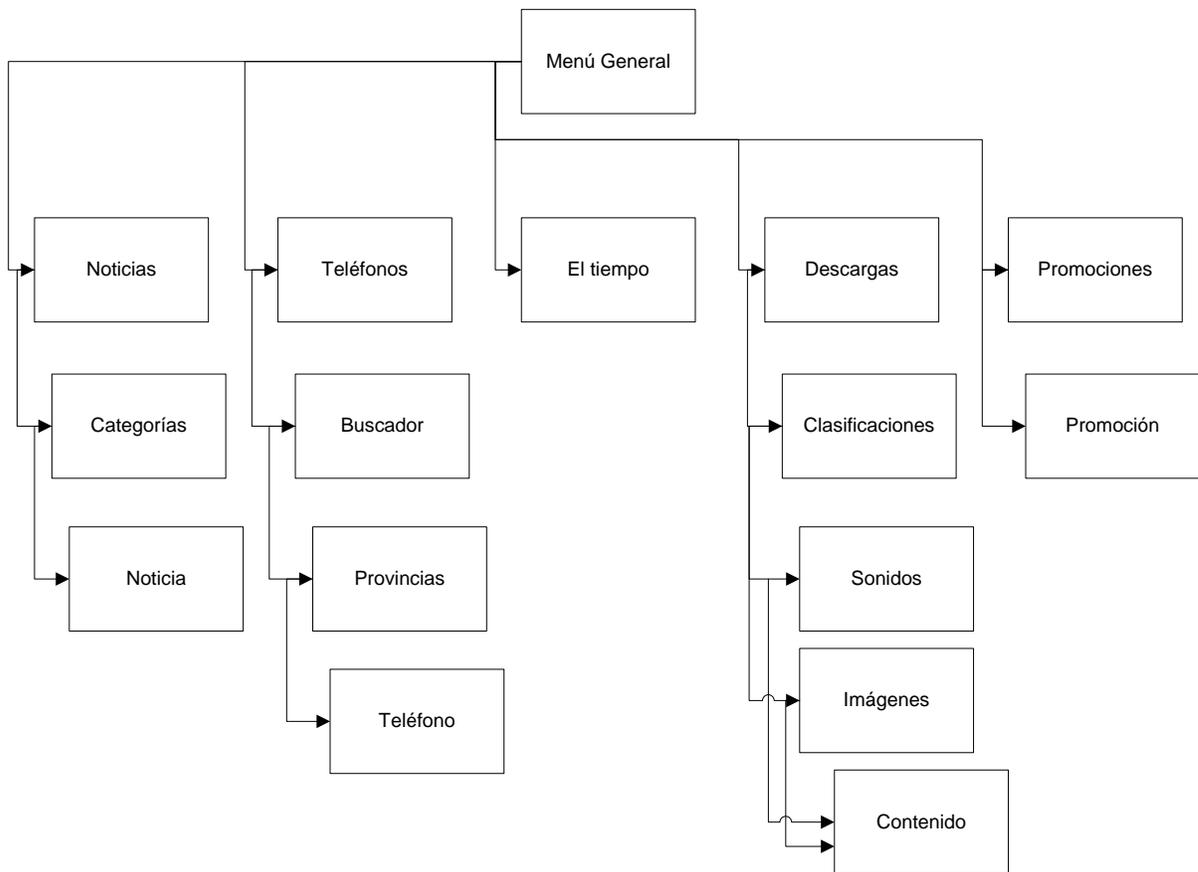


Figura 19 Mapa de Navegación del Portal WAP.

3.7 Conclusiones del Capítulo

En este capítulo se expusieron los principales artefactos del análisis y diseño del sistema propuesto como solución a la problemática planteada, principalmente los diagramas de clases e interacción del análisis y el diseño. Además se describieron los patrones de diseño utilizados y los prototipos de interfaz de usuario.

CAPÍTULO 4 IMPLEMENTACIÓN Y PRUEBA DEL SISTEMA

4.1 Introducción

En este capítulo se hace referencia a la disciplina de implementación y prueba. Se describe el modelo de implementación utilizado y se muestran los diagramas de componente y de despliegue. Se realiza, además, el modelo de prueba a través de los casos de prueba aplicados en el desarrollo del sistema propuesto.

4.2 Modelo de Implementación

El resultado principal de la implementación es el modelo de implementación, el cual describe, al igual que los elementos del modelo de diseño, cómo las clases se implementan en términos de componentes, ficheros de código fuente, ejecutable, scripts, librerías, APIs, dentro del flujo de implementación que propone RUP. El modelo de implementación está conformado por los diagramas de despliegue y componentes, éstos describen los componentes a construir y su organización y dependencia entre nodos físicos en los que funcionará el sistema. El modelo de implementación es la entrada principal de las etapas de prueba que siguen a la implementación.

4.2.1 Diagrama de Componentes

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre esos elementos. Es un grafo de componentes unidos a través de relaciones que pueden ser de compilación o de ejecución. A continuación se muestran los diagramas de componentes generados durante el modelado del sistema.

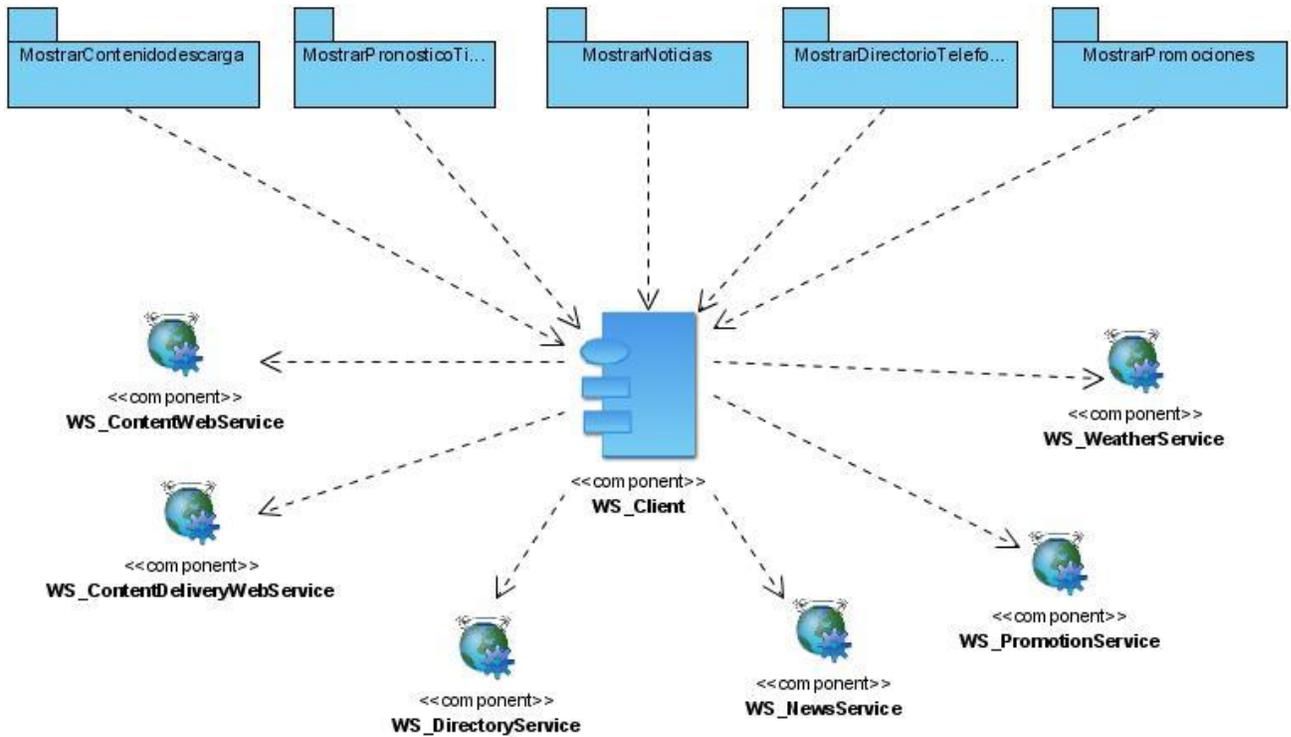


Figura 20 Diagrama de Componentes.

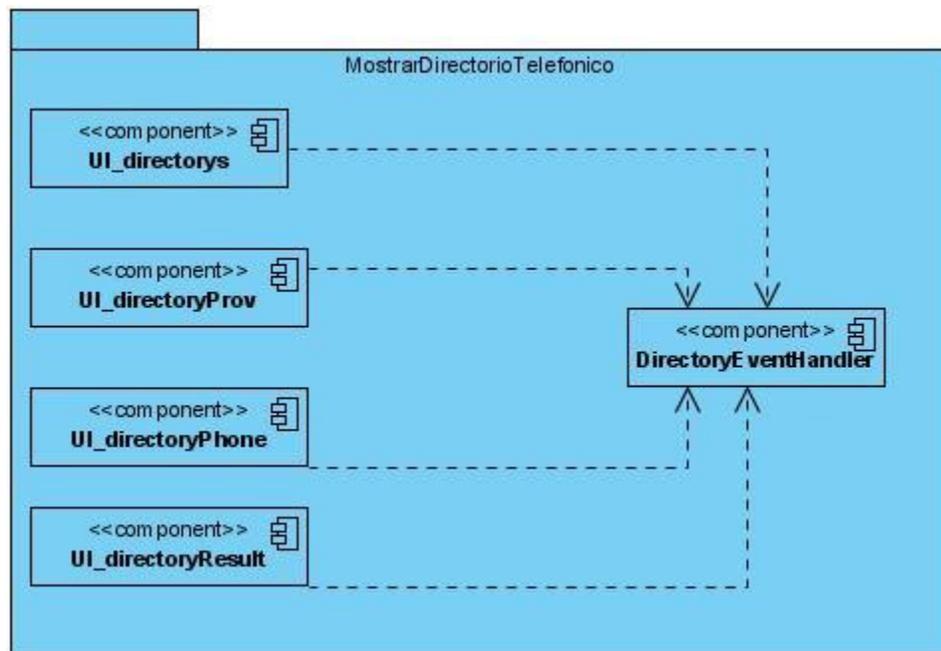


Figura 21 DiagramaComponentes CU_MostrarDirectorioTelefonico.

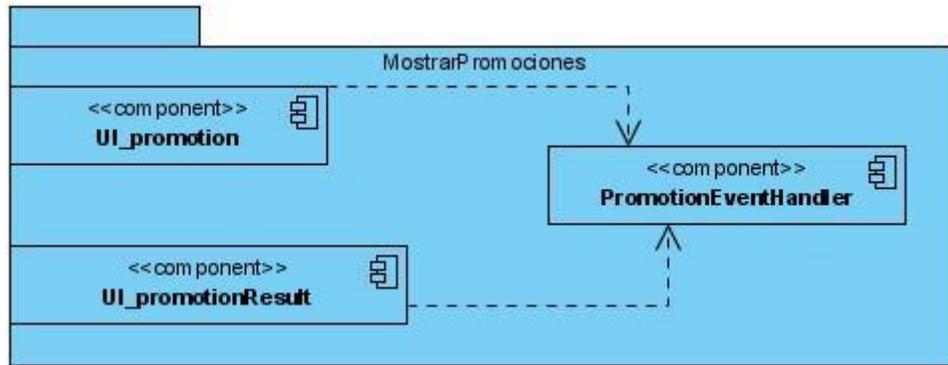


Figura 22 DiagramaComponentes CU_MostrarPromociones.

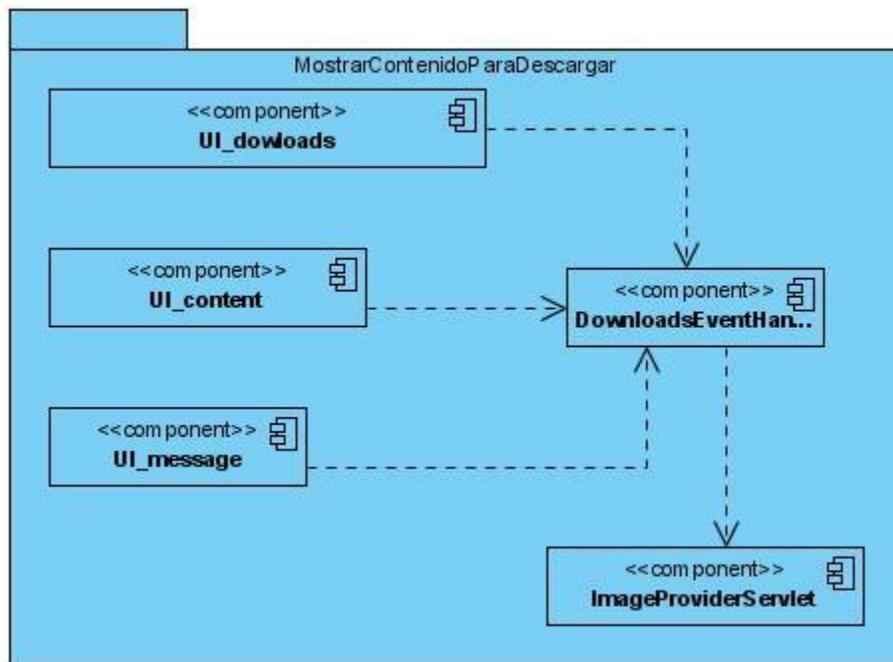


Figura 23 DiagramaComponentes CU_MostrarContenidoParaDescargar.

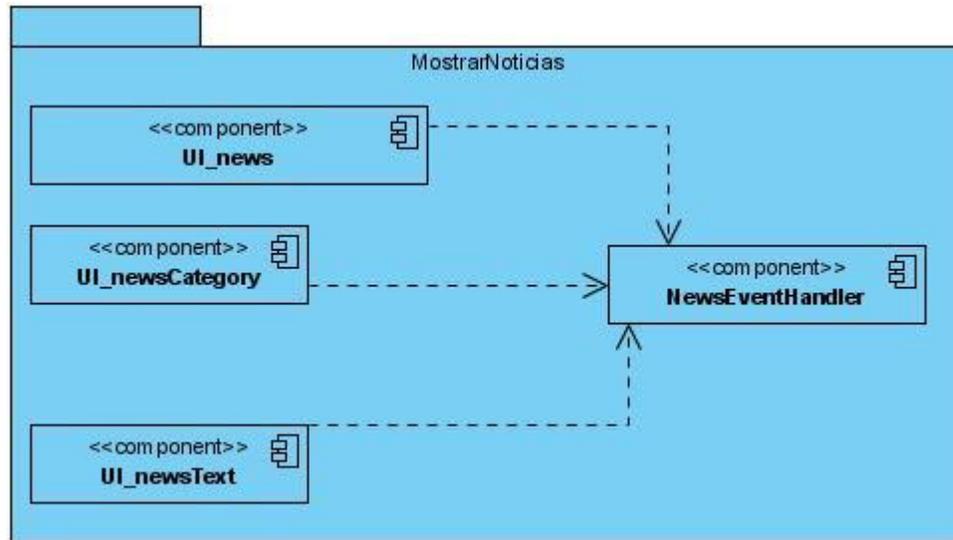


Figura 24 DiagramaComponentes CU_MostrarNoticias.

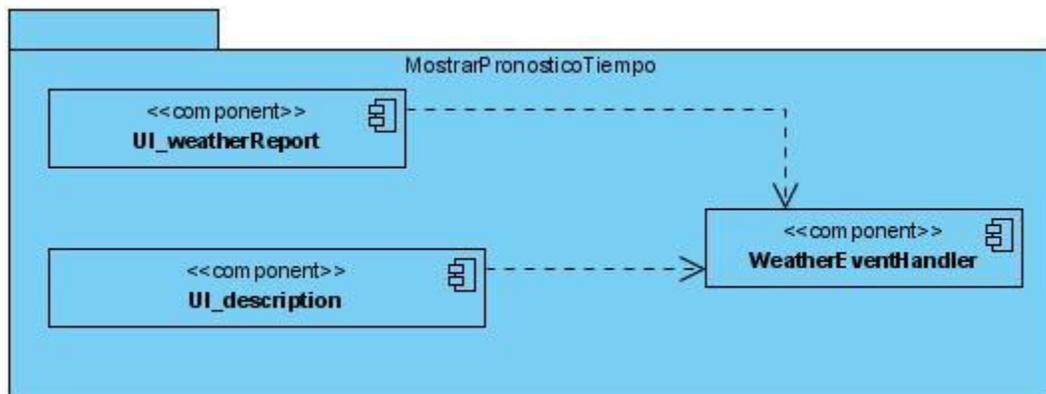


Figura 25 DiagramaComponentes CU_MostrarPronosticoTiempo.

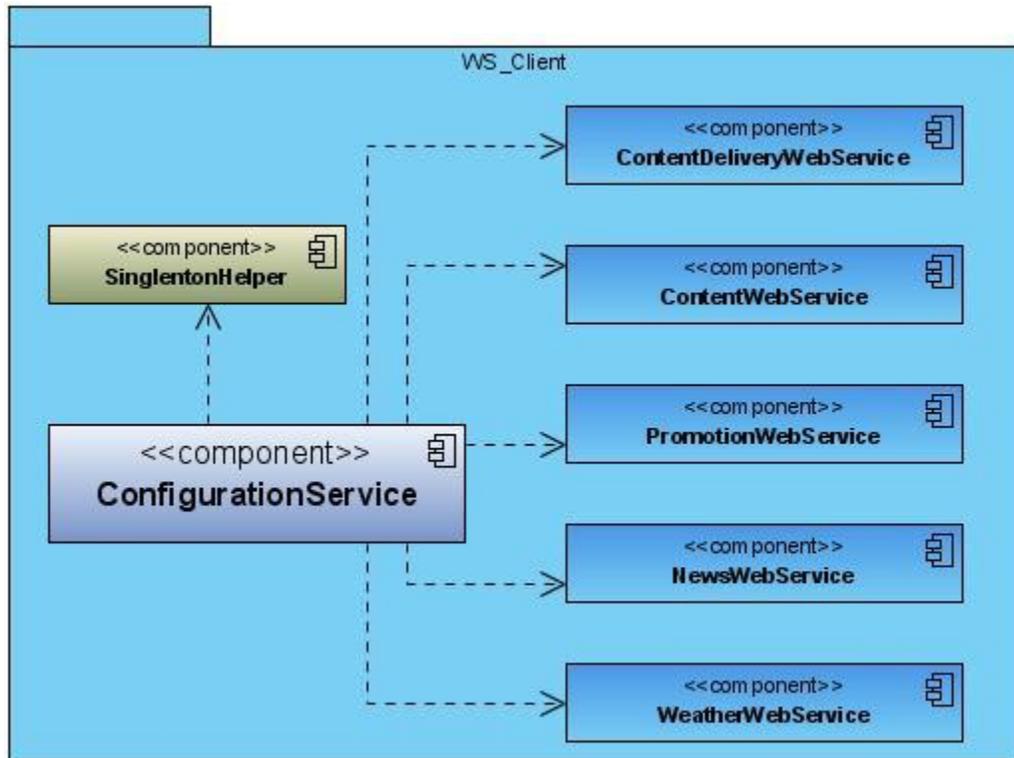


Figura 26 DiagramaComponentes PaqueteComponenteWS_Client.

4.2.2 Diagrama de Despliegue

En el diagrama de despliegue se indica la situación física de los componentes lógicos desarrollados. Un diagrama de despliegue es un grafo de nodos unidos por conexiones de comunicación. Un nodo se representa como un cubo y es el elemento donde se ejecutan los componentes, representan el despliegue físico de éstos.(5)

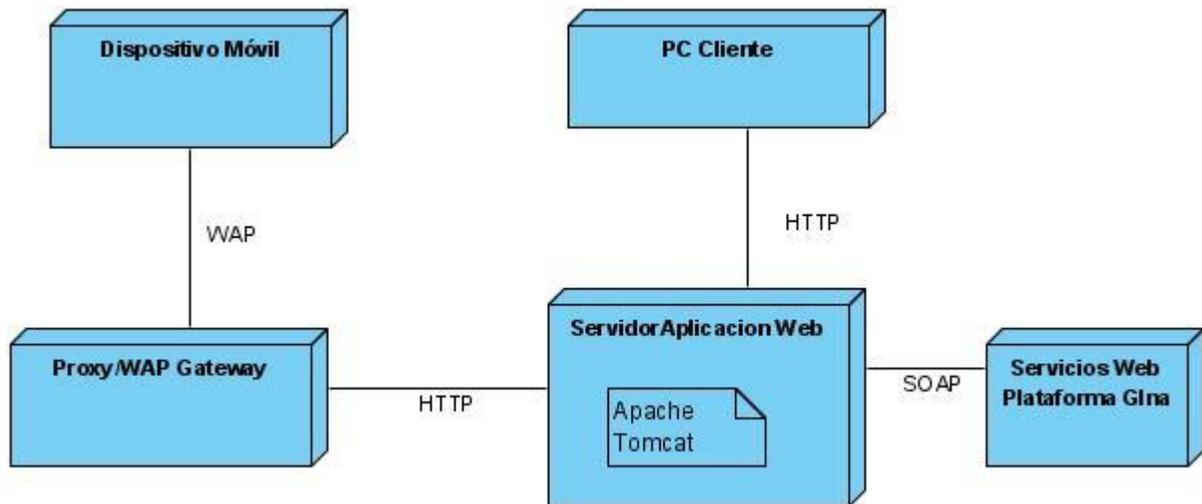


Figura 27 Diagrama de Despliegue.

Un aspecto a tener en cuenta en el ciclo de desarrollo del software son las pruebas pues permiten detectar los errores desde etapas tempranas, aumentando de esta forma la calidad del producto final y la aceptación por parte de los usuarios.

4.3 Modelo de Pruebas

Durante la disciplina de prueba es objetivo verificar que el sistema cumpla con los requerimientos de los clientes y usuarios finales para lograr una mayor calidad, el principal objetivo es realizar y evaluar las pruebas como se describen en el modelo de pruebas.

El modelo de pruebas describe principalmente cómo se prueban los componentes ejecutables en el modelo de implementación. El modelo de pruebas es una colección de casos de prueba, procedimientos de prueba y componentes de prueba.

4.3.1 Métodos de Prueba

Principales tipos de prueba, prueba de caja negra y prueba de caja blanca.

Prueba de Caja Negra

Pruebas que se llevan a cabo sobre la interfaz de usuario, se centra principalmente en los requisitos funcionales del software. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Examina aspectos del modelo principalmente del sistema sin tener mucho en cuenta la estructura interna del software.

Prueba de Caja Blanca

Las pruebas de caja blanca realizan un seguimiento del código fuente según va ejecutando los casos de prueba, de manera que se determinan de manera concreta las instrucciones en las que existen errores. Requieren del conocimiento de la estructura interna del programa y son derivadas a partir de las especificaciones internas de diseño o el código.

4.3.2 Diseño de Casos de Prueba

Un caso de prueba específica una forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse (2). Para comprobar el funcionamiento del sistema se realizaron diferentes casos de prueba, a continuación se muestra un caso de prueba de caja negra.

Prueba de Caja negra

Caso de uso: MostrarDirectorioTelefonico

Caso de prueba: CPR2.6 Realizar Búsqueda

Descripción del flujo:

1. El usuario introduce el texto como criterio de búsqueda.

Precondiciones:

El usuario tiene que haber seleccionado la opción de mostrar directorio telefónico.

Tabla 7 Caso de Prueba de CU_MostrarDirectorioTelefonico.

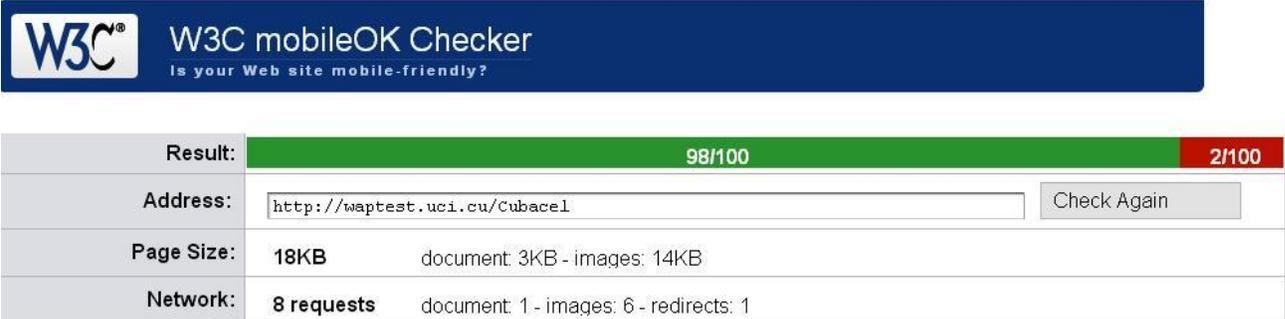
Clase Válida	Clase no Válida	Resultado Esperado	Resultado
Se introduce el texto con más de 3 caracteres. Se inicializa el evento de inicializar búsqueda.		Se realiza la búsqueda.	Satisfactorio.
	Se introduce el texto con menos de 3 caracteres.	Se muestra un mensaje de advertencia "Debe introducir al menos 3	Satisfactorio.

		caracteres”.	
--	--	--------------	--

La etapa de prueba es importante puesto que en ella se refleja la calidad con que ha sido llevada a cabo el desarrollo del sistema. Aplicando el diseño de casos de prueba, el ingeniero del software puede conseguir una prueba más completa y descubrir y corregir así el mayor número de errores antes de que la aplicación sea puesta a prueba por el cliente final. Para realizar pruebas a los portales WAP existen un conjunto de herramientas que proveen entidades como W3C y mTLD (mobileOK¹⁸ y ready.mobi¹⁹) respectivamente, las cuales son empleadas por la mayoría de los desarrolladores de sitios móviles, para así tener una medida de cómo funcionaría el sistema en un ambiente real y que sus aplicaciones cumplen con una serie de estándares sobre requisitos de diseño a continuación se muestran los datos proyectados por estas herramientas una vez realizada las pruebas al portal WAP.

Pruebas realizadas con la herramienta **mobileOK**.

mobileOK Checker.



The screenshot shows the W3C mobileOK Checker interface. At the top, there is a blue header with the W3C logo and the text "W3C mobileOK Checker" and "Is your Web site mobile-friendly?". Below the header, there is a table with the following data:

Result:	98/100		2/100
Address:	<input type="text" value="http://waptest.uci.cu/Cubacel"/>	<input type="button" value="Check Again"/>	
Page Size:	18KB	document: 3KB - images: 14KB	
Network:	8 requests	document: 1 - images: 6 - redirects: 1	

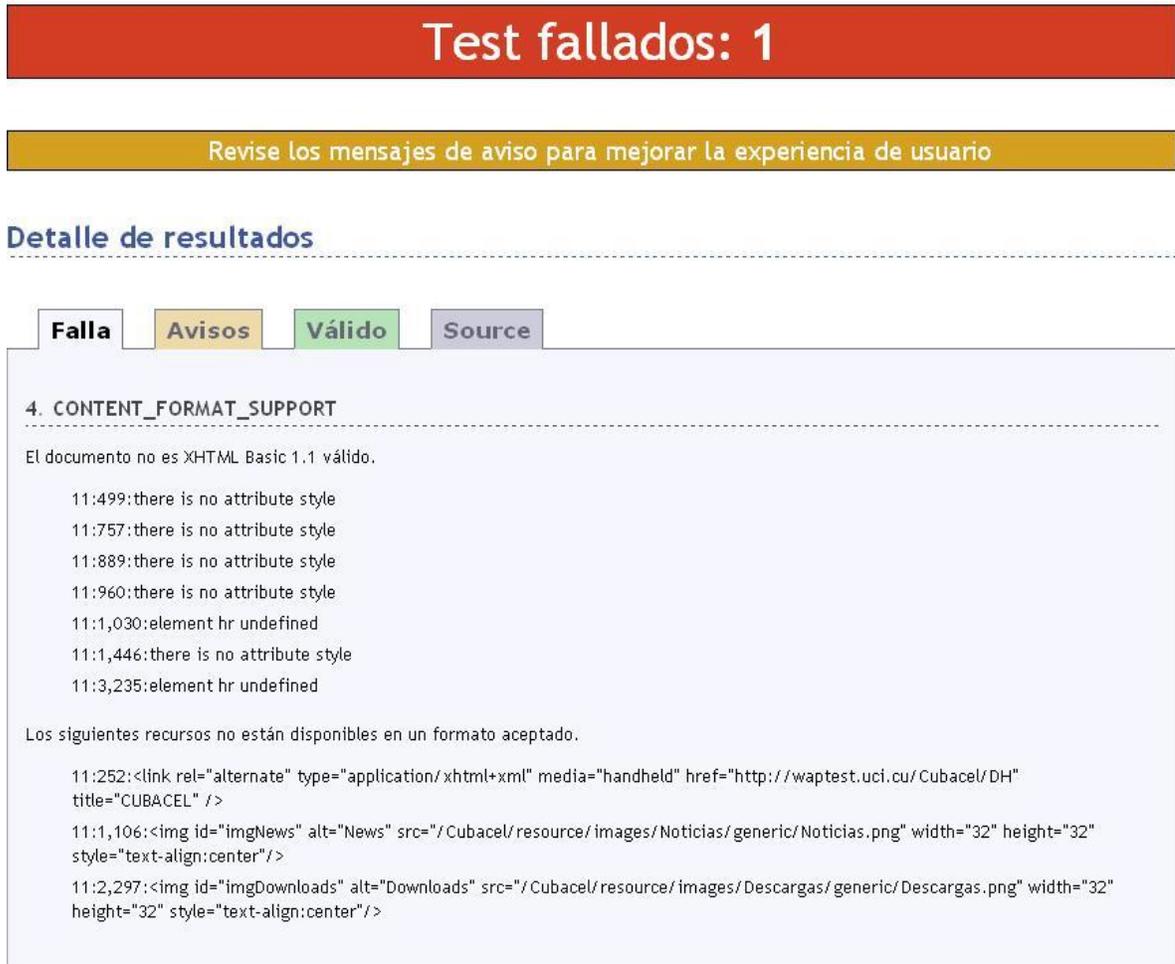
Figura 28 Resultado de la prueba realizada con mobileOK Checker.

¹⁸ mobileOK: <http://validator.w3.org/mobile/>

¹⁹ ready.mobi: <http://ready.mobi>

mobileOK Validator.

Es un analizador de buenas prácticas para sitios web para móviles. En concreto comprueba la adecuación de un contenido móvil a nivel mobileOK Basic²⁰ según se define en las Buenas Prácticas para la Web Móvil 1.0.



The screenshot shows the mobileOK Validator interface. At the top, a red banner displays "Test fallados: 1". Below it, a yellow banner suggests "Revise los mensajes de aviso para mejorar la experiencia de usuario". The main section is titled "Detalle de resultados" and contains a tabbed interface with "Falla" selected. The error details for "4. CONTENT_FORMAT_SUPPORT" are as follows:

```
4. CONTENT_FORMAT_SUPPORT
-----
El documento no es XHTML Basic 1.1 válido.

11:499:there is no attribute style
11:757:there is no attribute style
11:889:there is no attribute style
11:960:there is no attribute style
11:1,030:element hr undefined
11:1,446:there is no attribute style
11:3,235:element hr undefined

Los siguientes recursos no están disponibles en un formato aceptado.

11:252:<link rel="alternate" type="application/xhtml+xml" media="handheld" href="http://waptest.uci.cu/Cubacel/DH"
title="CUBACEL" />
11:1,106:
11:2,297:
```

Figura 29 Detalles de la falla encontrada.

Se muestran los resultados válidos obtenidos por esta herramienta al ejecutar las pruebas.

1. AUTO_REFRESH and REDIRECTION

No se utilizan refrescos meta automáticos en la página.

No se utilizan refrescos (cabecera Refresh) automáticos en la página.

²⁰mobileOK Basic: <http://www.w3.org/TR/mobileOK-basic10-tests/>

5. DEFAULT_INPUT_MODE

Los controles de texto de la página definen inputmode.

6. EXTERNAL_RESOURCES

El número total de recursos incluidos es menor de 10

7. GRAPHICS_FOR_SPACING

No se utilizan imágenes como espaciadores.

8. IMAGE_MAPS

No se utilizan mapas de imagen.

9. IMAGES_RESIZING and IMAGES_SPECIFY_SIZE

Todas las imágenes definen los atributos width y height.

Los tamaños de las imágenes concuerdan con los tamaños especificados en el documento.

10. LINK_TARGET_FORMAT

Todos los enlaces enlazan a recursos aceptados por el dispositivo.

11. MEASURES

No se han detectado tamaños de fuente absolutos en las hojas de estilo.

No se han detectado anchos absolutos en las hojas de estilo.

12. MINIMIZE

La página contiene 0% caracteres extra.

13. NO_FRAMES

No se han detectado frames.

14. NON_TEXT_ALTERNATIVES

Todas las imágenes tienen texto equivalente.

16. PAGE_SIZE_LIMIT

Peso del código 3Kb (límite 10Kb) y el peso de las páginas es 18Kb (límite 20Kb).

17. PAGE_TITLE

Se ha proporcionado un título a la página.

18. POP_UPS

No se han detectado enlaces que abran en nueva ventana.

19. PROVIDE_DEFAULTS

Los botones de selección disponen de una opción seleccionada por defecto.

20. STYLE_SHEETS_SUPPORT

No se utilizan propiedades de posicionamiento en las hojas de estilo.

22. TABLES_ALTERNATIVES

No se usan tablas en la página.

23. TABLES_LAYOUT

No se han detectado tablas de posicionamiento en la página.

24. TABLES_NESTED

No hay tablas anidadas en la página.

Pruebas realizadas con la herramienta **ready.mobi**.



Figura 30 Resumen de las pruebas.

Page test results

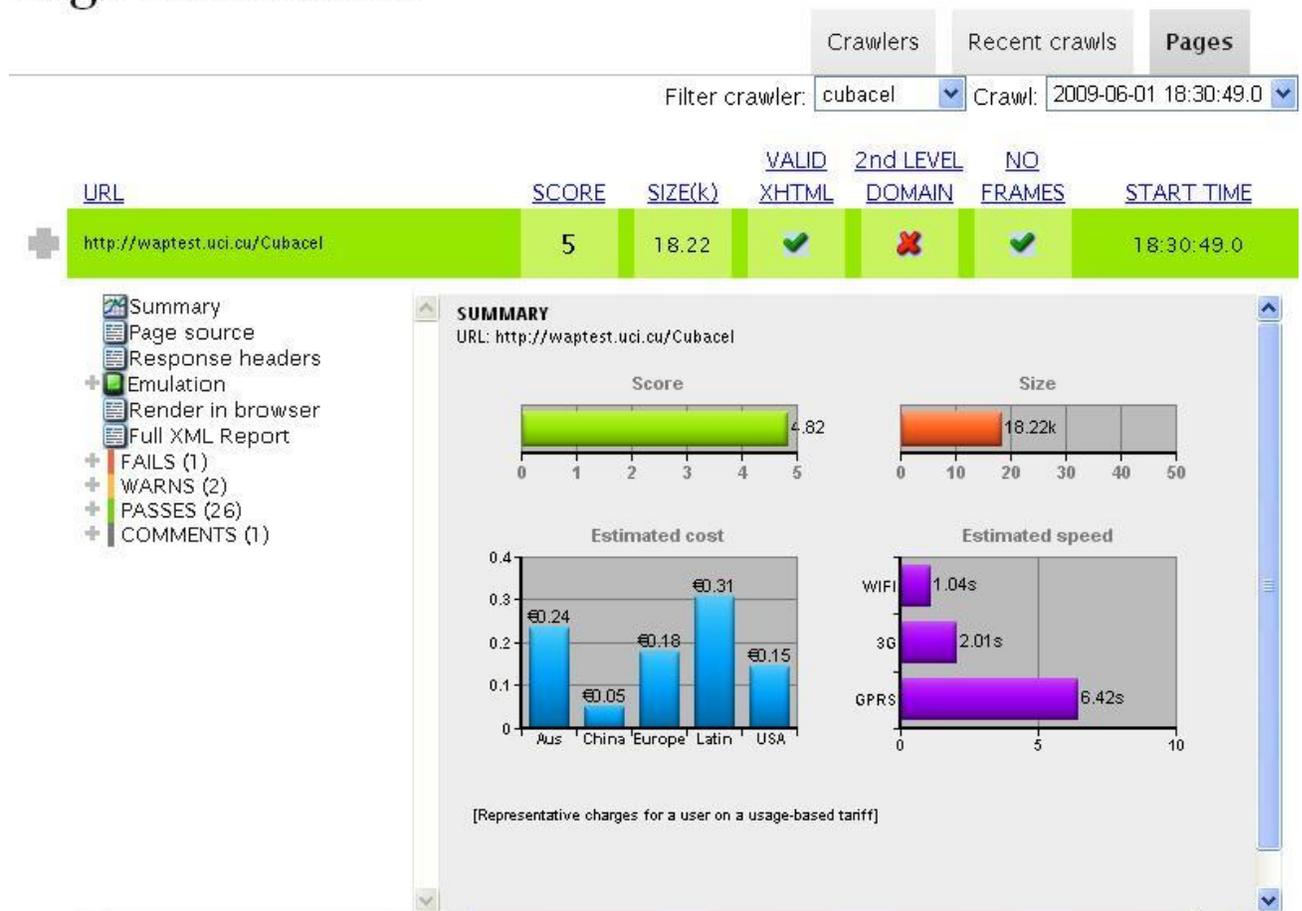


Figura 31 Resultado de la prueba con ready.mobi.

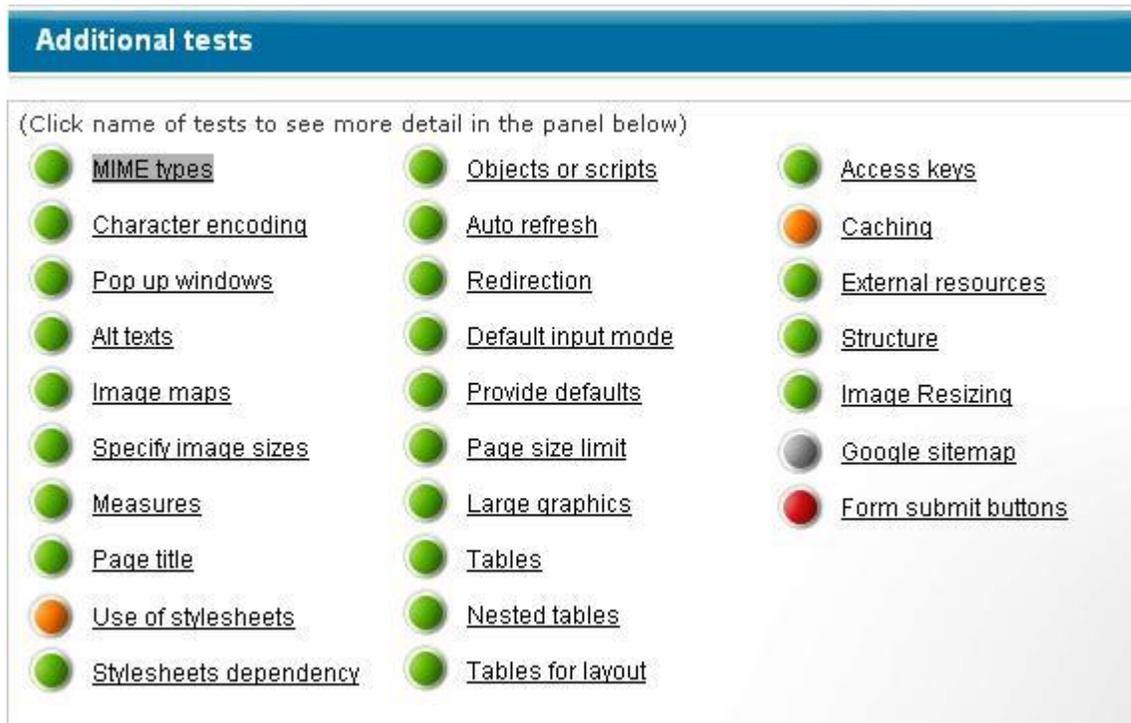


Figura 32 Pruebas realizadas.

4.4 Conclusiones del Capítulo

En este capítulo se mostraron los principales artefactos de las disciplinas de implementación y prueba, en específico los diagramas de despliegue y de componentes. Además de una descripción de los casos de prueba realizados al sistema utilizando la técnica de caja negra permitiendo de esta forma la detección de errores desde los primeros ciclos de desarrollo del proyecto. También se realizaron pruebas con herramientas como ready.mobi y mobileOK mostrando resultados satisfactorios que dan respuesta a los objetivos propuestos.

CAPÍTULO 5 ESTUDIO DE FACTIBILIDAD

5.1 Introducción

En este capítulo se lleva a cabo la planificación del proyecto con el objetivo de obtener una aproximación de los recursos necesarios así como del esfuerzo, costo y tiempo necesario para culminar el proyecto. Mediante este estudio se podrá obtener la cantidad de personas que intervendrán y sobre todo, cuánto debe ser invertido por parte de las empresas interesadas en el producto.

5.2 Estimación basada en Puntos de Casos de Uso

La estimación basada en puntos de casos de uso sigue tres pasos fundamentales los cuales son descritos a continuación.

Paso 1. Cálculo de los Puntos de casos de uso sin ajustar $UUCP = UAW + UUCW$ Donde:

UUCP: Puntos de casos de uso sin ajustar.

UAW: Factor de peso de los actores sin ajustar.

UUCW: Factor de peso de los casos de uso sin ajustar.

Paso 2. Cálculo de los Puntos de casos de uso ajustados $UCP = UUCP \times TCF \times EF$ Donde:

UCP: Puntos de Casos de Uso ajustados.

UUCP: Puntos de Casos de Uso sin ajustar.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

Paso 3. Estimación de esfuerzo a través de los puntos de casos de uso $E = UCP \times CF$ Donde:

UCP: Puntos de Casos de Uso ajustados.

CF: Factor de Conversión.

E: Esfuerzo estimado en horas hombres.

Paso 4. Calcular el esfuerzo de todo el proyecto.

Las siguientes tablas muestran una descripción, así como los cálculos de los valores necesarios para dar solución al método de la estimación basada en puntos de casos de uso del sistema:

5.2.1 Cálculo de los Puntos de casos de uso sin ajustar

Factor de peso de los actores sin ajustar

Tabla 8 Descripción de tipos de Actores.

Tipo de Actor	Factor de Peso	Descripción
Simple	1	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)
Medio	2	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto
Complejo	3	Una persona que interactúa con el sistema mediante una interfaz gráfica

Tabla 9 Factor de peso de los actores sin ajustar.

Nombre del Actor	Identificación	Actores	Factor de peso	Total
Sistema Módulo de Contenidos	M	1	2	2
Sistema Módulo de Descargas	M	1	2	2
Cliente	C	1	3	3
UAW = $\Sigma(\text{Factor} * \text{Actores})$				7

La complejidad de los Casos de Uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo.

Factor de peso de los casos de uso sin ajustar

Tabla 10 Descripción de tipos de casos de uso.

Tipo de CU	Descripción	Peso	Cant CU	Total
Simple	El Caso de Uso contiene de 1 a 3 transacciones	5	0	0
Medio	El Caso de Uso contiene de 4 a 7 transacciones	10	2	20
Complejo	El Caso de Uso contiene más de 8	15	3	45

Capítulo 5. Estudio de Factibilidad

	transacciones			
	$UUCW = \sum(\text{Factor} * \text{CantCU})$			65

UUCW toma valor 65, por lo tanto se realiza la suma dando un valor de 72 a UUCP. (UAW = 7; UUCW = 65; UUCP = 72).

5.2.2 Cálculo de los puntos de Casos de Uso ajustados

El Factor de Complejidad Técnica (TCF) se determina mediante la cuantificación de un conjunto de factores y predeterminados que determinan la complejidad técnica del sistema.

Tabla 11 Factor de Complejidad Técnica.

Factor	Nombre del factor	Peso	Relevancia	Producto
			(0 irrelevante, 5 esencial)	
1.	Sistema distribuido	2	0	0
2.	Tiempos de Respuesta	1	5	5
3.	Eficiencia del usuario final (on line)	1	5	5
4.	Procesamiento interno complejo	1	3	3
5.	Código que debe ser reusable	1	4	4
6.	Fácil de instalar	0.5	2	1
7.	Fácil de usar	0.5	3	1.5
8.	Portable	2	3	6
9.	Fácil de cambiar	1	4	4
10.	Concurrente	1	0	0
11.	Incluye características especiales de seguridad	1	5	5
12.	Proporciona acceso directo a terceras partes	1	0	0
13.	Requiere facilidades de entrenamiento especial a usuario	1	3	3
	$TCF = 0.6 + 0.01 * \sum (\text{Peso} * \text{Valor})$			0.975

El Factor de Ambiente (EF) está relacionado con las habilidades y entrenamiento del equipo de desarrollo.

Tabla 12 Factor de Ambiente.

Factor	Descripción del factor	Peso	Nivel	Producto
--------	------------------------	------	-------	----------

Capítulo 5. Estudio de Factibilidad

1.	Familiaridad con RUP (con el modelo del proyecto).	1.5	4	6
2.	Experiencia en la aplicación.	0.5	3	1.5
3.	Experiencia OO.	1	3	3
4.	Capacidad del analista líder.	0.5	3	1.5
5.	Motivación.	1	4	4
6.	Estabilidad de los Requerimientos.	2	0	0
7.	Trabajador a tiempo parcial.	-1	0	0
8.	Dificultad con el lenguaje de programación.	-1	3	-3
EF = 1.4 - 0.03 * Σ(Peso * Valor)				1.01

Después del resultado se hace el cálculo, para completar la ecuación.

$$UCP = UUCP * TCF * EF = 72 * 0.975 * 1.01 = 70.90.$$

5.2.3 Estimación de esfuerzo

Según lo planteado en este método, para el cálculo del esfuerzo $E=UCP*CF$ el factor de conversión CF, se obtiene contando cuantos valores de los que afectan el Factor Ambiente, están por debajo de la media 3, y los que están por arriba de la media.

- ✓ Si el total es 2 o menos se utiliza el factor de conversión 20 Horas-Hombre / Punto de Casos de uso.
- ✓ Si el total es 3 o 4 se utiliza el factor de conversión 28 Horas-Hombre / Punto de Casos de uso.
- ✓ Si el total es mayor o igual que 5 se recomienda efectuar cambios en el proyecto o desecharlo, por su alto riesgo de fracaso.

Sobre lo planteado se puede decir que CF toma un valor 20 horas-hombre/Punto de Casos de Uso. ($E = 70.90*20 = 1418.04$ horas-hombres).

Estimación del esfuerzo total del proyecto.

Si además se considera que este esfuerzo representa un porcentaje del esfuerzo total del proyecto, de acuerdo con los valores de porcentaje de la tabla se obtiene:

Tabla 13 Esfuerzo total del Proyecto.

Actividad	Porcentaje	Esfuerzo Horas-Hombre
Análisis	10%	354.51
Diseño	20%	709.02
Implementación	40%	1418.04
Pruebas	15%	531.77
Sobrecarga (OA)	15%	531.77
Total	100%	3545.10

Si $ET = 3545.10$ horas-hombre y se estima que cada mes tiene como promedio 240 horas laborables, eso daría un $ETh = 14.77$ mes-hombre.

Esto quiere decir que 1 persona puede realizar la tarea en 14 meses aproximadamente.

Costo del Proyecto

Se asume como salario promedio mensual \$100, personal 2 (CH), dada la estimación del tiempo (total del proyecto), se obtiene, el costo-hombres-mes ($CHM = 2 * \text{Salario Promedio}$) 200 \$/mes, el costo total del proyecto $\text{Costo} = CHM * ETh = 200.00 * 14.77 = 2954$.

De los resultados obtenidos se interpreta que, con 2 hombres trabajando en el proyecto, el mismo se desarrolla en 7 meses aproximadamente y su costo total se estima que sea \$2954.

Debido a que los contenidos y los servicios accesibles a través de portales WAP es un acontecimiento reciente, es difícil buscar datos públicos disponibles sobre las proyecciones de ingresos. Muchas veces, los datos específicos a este tipo de mercado no están separados de datos agregados. Los estudios auspiciados por la industria, solo están enfocados a ciertos mercados o tecnologías, lo que hace difícil comparar datos entre estudios. Por consiguiente, se realiza una evaluación de acuerdo a los posibles beneficios que trae la implementación de un sistema como el propuesto para un determinado sector, teniendo en consideración el estudio realizado y los conocimientos adquiridos por los desarrolladores.

5.3 Beneficios tangibles e intangibles

El portal WAP es un software con gran valor comercial para las empresas proveedoras de servicios y contenidos de valor agregado para dispositivos móviles (VASP). Puede ser utilizado por cualquier entidad en conjunción con los operadores móviles para ofrecer a sus clientes diversos servicios como: descargas de contenidos (imágenes, música, juegos), servicio de noticias, consultar

reportes sobre el clima, directorios. El beneficio fundamental consiste en contar con un sistema que independientemente de las redes de transmisión de datos inalámbricas y las capacidades y características de los terminales móviles, sea capaz de presentar un sitio Web bien estructurado y diseñado para todo tipo de dispositivos, ya sean móviles o no.

Como beneficio tangible se puede mencionar que el ingreso generado a través de portales WAP en Internet e Intranets logró importes ascendentes a 5.5 billones de dólares en el año 2008 y se estima que para el 2013 alcancen la cifra de 22.4 billones de dólares, según estudios realizados por Juniper Research.

Algunos de los beneficios intangibles que provee el portal WAP:

Entre los beneficios intangibles que se esperan obtener con la realización de este proyecto se encuentran:

- ✓ Mejorar los tiempos de repuesta y accesos a páginas Web desde dispositivos móviles.
- ✓ Visualización de páginas Web de forma adecuada, de tal forma que ayude a una mejor interacción con el usuario, independientemente de las limitantes de los terminales móviles.
- ✓ La facilidad de administración y de programación, al no estar centrado en la creación de páginas distintas para cada tecnología, sino por medio del empleo de documentos XML se define una única presentación que luego es pre-generada automáticamente en varias JSPs para cada tecnología de marcado.
- ✓ Incrementar la cuota del mercado y el alcance de la audiencia que hace uso de un servicio determinado.
- ✓ Contenidos reutilizables por múltiples móviles.

5.4 Análisis de costos y beneficios

El framework y las tecnologías elegidas para el desarrollo del sistema son distribuidas bajo licencias de Código Abierto (Open-Source) y algunas constituyen estándares o están en vías de estandarización por parte de los comité encargados de ello en organizaciones internacionales como el W3C y OMA, por lo que no se hace necesario la adquisición de licencias para la utilización de estas herramientas, lo cual constituye una ventaja dadas las características de Cuba de ser un país económica y tecnológicamente bloqueado.

Por tanto durante el análisis realizado en los apartados anteriores sobre los costes que supondría el desarrollo de un sistema de este tipo, se debe determinar la influencia de varios factores, como el impacto que trae consigo un sistema similar dentro de un determinado sector económico o

social, la reducción de los costos y tiempo de desarrollo, comparados con el empleo de otras herramientas para lograr los mismos objetivos.

Tomando todo esto en cuenta, se puede señalar que la puesta en marcha de este sistema, mejorará el desarrollo de servicios orientados a estas tecnologías y generará una gran fuente de ingresos, además permitirá su empleo con fines educativos y sociales en muchos sectores de nuestra sociedad.

5.5 Conclusiones del Capítulo

En este capítulo se obtuvieron los estimados de horas por hombre que se requieren para desarrollar la solución, y el costo aproximado. Se realizó un análisis de los distintos indicadores y se concluyó que era factible la realización del portal WAP.

CONCLUSIONES GENERALES

Con el propósito de darle cumplimiento al objetivo general en la realización de esta investigación, se profundizó en el conocimiento de las herramientas para la creación de portales para dispositivos móviles, así como en el desarrollo de un portal WAP. Se modeló mediante la metodología RUP, para guiar el proceso de desarrollo de software. El diseño y la implementación se rigieron por un conjunto de estándares internacionales establecidos para la elaboración de portales con capacidad de ser visualizados de manera correcta en dispositivos móviles. Además de estar enfocado a las características que posee Cuba en cuanto a tecnología y servicios.

Se puede concluir que se ha cumplido satisfactoriamente el objetivo trazado para este trabajo enfatizando en los siguientes puntos:

- ✓ Se eligieron las tecnologías propicias dentro del espectro de tendencias actuales y se identificó MyMobileWeb como el camino idóneo para desarrollar un sistema que cumpla con los requerimientos planteados.
- ✓ Se implementaron las funcionalidades para el acceso de los servicios presentes en la plataforma Gina mediante el portal.
- ✓ Se logro adaptar los contenidos y servicios de manera eficiente alcanzando una correcta visualización mediante el Portal WAP.
- ✓ Se provee la posibilidad de realizar descargas de contenido al usuario, de consultar noticias (internacionales, nacionales, deportivas), promociones, posibilidad de acceder al directorio telefónico el cual cuenta con páginas amarillas.

RECOMENDACIONES

A continuación se indican algunas recomendaciones con el objetivo de efectuar un seguimiento y perfeccionamiento de esta investigación.

- ✓ Agregarle nuevas funcionalidades a Gina, para brindar un mayor número de servicios a los usuarios de móviles en Cuba.
- ✓ Mantener constantemente actualizado el repositorio de descripción de los dispositivos móviles, debido que los teléfonos celulares están en constante evolución y siempre surgen nuevos modelos o se les agregan nuevas características a modelos existentes.
- ✓ Realizar implementaciones para extender con nuevos controles el lenguaje DIAL utilizado por MyMobileWeb.
- ✓ Explotar más las capacidades de los dispositivos, para personalizar los contenidos y servicios a través del empleo de tecnologías como AJAX.

BIBLIOGRAFÍA

1. **Basterretche, Juan Felix.** *Dispositivos Móviles.* Corrientes - Argentina : Universidad Nacional del Nordeste, Facultad de Ciencias Exactas, Naturales y Agrimensura, 2007.
2. **Pascual Ramos, Francisco.** Radiocomunicaciones y fibra óptica. [En línea] [Citado el: 25 de Septiembre de 2008.] http://www.radioptica.com/Radio/telefonía_movil.asp.
3. **Rebelde, Juventud.** Juventud Rebelde. [En línea] [Citado el: 14 de Junio de 2008.] <http://www.juventudrebelde.cu/2007>.
4. **J. Carlos Olivares R, J. Gabriel González S., Víctor J. Sosa S. y Azucena Montes R.** *Uso del Diseño Accesible de Páginas Web para la Correcta Visualización de la Web en Dispositivos Móviles.* s.l. : IEEE LATIN AMERICA TRANSACTIONS, VOL. 5, NO. 2, MAY, 2007.
5. **Diego Berrueta, Ignacio Marín, Sergio Fernández, José M. Cantera, Mario Arias, Juan J. Hierro, Miguel Jiménez, Javier Soriano.** *Aplicación de las tecnologías de la Web Semántica a la problemática de cumplimentación automática de formularios en la Web Móvil.* s.l. : Proyecto Tractor MORFEO-MyMobileWeb (FIT-350401-2006-2), 2006.
6. **Booch Grady, Jacobson Ivar, Rumbaugh James.** *El Proceso Unificado de Desarrollo de Software.* s.l. : Addison Wesley, 2000.
7. **Martin Fowler, Kendall Sccott.** *UML Gota a Gota.* 1999.
8. **LONDOÑO, M. Y. O.** Herramientas C.A.S.E. [En línea] [Citado el: 2 de Febrero de 2009.] <http://personales.com/cuba/cardenas/internet/case.html>.
9. **PARADIGM, C. V.** [En línea] [Citado el: 12 de Febrero de 2009.] <http://www.visual-paradigm.com/product/vpum/>.
10. **Foundation Eclipse.** [En línea] 2007. [Citado el: 12 de Mayo de 2008.] <http://www.eclipse.org/>.
11. **Tigris.org.** [En línea] [Citado el: 20 de Enero de 2009.] <http://subversion.tigris.org>.
12. **Larman, Craig.** UML y Patrones. *UML y Patrones.* s.l. : Prentice Hall, 2002.
13. **Mobile Web Best Practices 1.0.** [En línea] <http://www.w3.org/TR/2008/REC-mobile-bp-20080729/>.
14. **Armstrong, Chris.** *Modeling Web Services with UML.* 2002.
15. **López, Manrique.** *Lenguajes de marcado para Independencia Web de dispositivo.*
16. **Guía dominios .mobi para Desarrolladores.** *mobiForge.* [En línea] 2007. [Citado el: 3 de Noviembre de 2008.] <http://www.mobiforge.com>.

17. *INFORME SOBRE EL DESARROLLO MUNDIAL DE LAS TELECOMUNICACIONES 2003. Indicadores de acceso para la sociedad de la información.* Ginebra : UIT, 2003.
18. **Metha, Nirav.** *Mobile Web Development.* BIRMINGHAM - MUMBAI : Packt Publishing, 2008.
19. **AdMob.** *AdMob Mobile Metrics Report.* 2008.
20. *The Best & Worst of the Mobile Web.* [pdf] s.l. : mobiThinking, 2008.
21. **ROJAS, JUAN CARLOS OLIVARES.** *MoviWeb: Plataforma para Solucionar el Problema de la Visualización de Sitios Web en Dispositivos Móviles Heterogéneos.* CUERNAVACA, MORELOS : cenidet, 2006.
22. **GALÁN, IGNACIO SÁNCHEZ.** *Perspectivas de la telefonía móvil.* [pdf]
23. **Hernán Galperin, Judith Mariscal.** *Pobreza y Telefonía Móvil en América Latina y el Caribe.* [pdf] Ottawa, Canadá : DIRSI, 2007.
24. **Junquera, Rafael A.** *Servicios de valor Agregado en las Redes Móviles Latinoamericanas: Casos de Operadores.* [pdf] 2006.
25. **Jeff Axup, B.S.** *Methods of Understanding and Designing For Mobile Communities.* [pdf] 2006.
26. **Research, Juniper.** *Share, Collaborate, Exploit ~ Defining Mobile Web 2.0.* [pdf] 2008.
27. **Systems, Volantis.** *Volantis Mobile Content Framework™.* [pdf]
28. —. *Volantis Mobile Content Transcoder™.* [pdf]
29. **Cruz, Cristian Rodríguez de la.** *Entregable D.4.3 Especificación de formalismos para el content binding.* [pdf] 2006.
30. —. *Entregable D.4.2 Especificación de Formalismos para el Data Binding.* [pdf] 2006.
31. *MyMobileWeb MANUAL DE USUARIO DEL COMPONENTE DE GESTIÓN DE DISPOSITIVOS.* [pdf] s.l. : Telefónica Investigación y Desarrollo, S.A. Unipersonal, 2006.
32. *MyMobileWeb GUÍA DE REFERENCIA DEL LENGUAJE DE DEFINICIÓN DE PRESENTACIONES.* [pdf] s.l. : Telefónica Investigación y Desarrollo, S.A. Unipersonal, 2006.
33. *MyMobileWeb MANUAL DE USUARIO DE LAS HERRAMIENTAS DE DESPLIEGUE DE PRESENTACIONES.* [pdf] s.l. : Telefónica Investigación y Desarrollo, S.A. Unipersonal, 2006.
34. **Bulbrook, Dale.** *WAP A Beginner's Guide.* s.l. : McGraw-Hill, 2001. 0-07-219455-3.
35. **Patrick Coelho, David Jorgensen, Ken Cox, Norman Gragasin, Steve Milroy, Amit Kalani, Laura Barker, Doug Safford.** *.NET Mobile Web Developer's Guide.* s.l. : Syngress Publishing, 2002. 1-928994-56-3.

REFERENCIAS BIBLIOGRÁFICAS

1. **Basterretche, Juan Felix.** *Dispositivos Móviles.* Corrientes - Argentina : Universidad Nacional del Nordeste, Facultad de Ciencias Exactas, Naturales y Agrimensura, 2007.
2. **Pascual Ramos, Francisco.** Radiocomunicaciones y fibra óptica. [En línea] [Citado el: 25 de Septiembre de 2008.] http://www.radioptica.com/Radio/telefonía_movil.asp.
3. **J. Carlos Olivares R, J. Gabriel González S., Víctor J. Sosa S. y Azucena Montes R.** *Uso del Diseño Accesible de Páginas Web para la Correcta Visualización de la Web en Dispositivos Móviles.* s.l. : IEEE LATIN AMERICA TRANSACTIONS, VOL. 5, NO. 2, MAY, 2007.
4. **Diego Berrueta, Ignacio Marín, Sergio Fernández, José M. Cantera, Mario Arias, Juan J. Hierro, Miguel Jiménez, Javier Soriano.** *Aplicación de las tecnologías de la Web Semántica a la problemática de cumplimentación automática de formularios en la Web Móvil.* s.l. : Proyecto Tractor MORFEO-MyMobileWeb (FIT-350401-2006-2), 2006.
5. **Booch Grady, Jacobson Ivar, Rumbaugh James.** *El Proceso Unificado de Desarrollo de Software.* s.l. : Addison Wesley, 2000.
6. **Martin Fowler, Kendall Scott.** *UML Gota a Gota.* 1999.
7. **LONDOÑO, M. Y. O.** Herramientas C.A.S.E. [En línea] [Citado el: 2 de Febrero de 2009.] <http://personales.com/cuba/cardenas/internet/case.html>.
8. **PARADIGM, C. V.** [En línea] [Citado el: 12 de Febrero de 2009.] <http://www.visual-paradigm.com/product/vpuml/>.
9. **Foundation Eclipse.** [En línea] 2007. [Citado el: 12 de Mayo de 2008.] <http://www.eclipse.org/>.
10. **Tigris.org.** [En línea] [Citado el: 20 de Enero de 2009.] <http://subversion.tigris.org>.
11. **Valle, Amaury E. del.** Telefonía celular en Cuba El futuro pasa por el móvil. [En línea] 19 de marzo de 2009. [Citado el: 21 de marzo de 2009.] <http://www.juventudrebelde.cu/>.
12. **Larman, Craig.** UML y Patrones. *UML y Patrones.* s.l. : Prentice Hall, 2002.
13. **Association, GSM.** GSM World. *GSM World Coverage 2008.* [En línea] <http://www.gsmworld.com/>.

ANEXOS

Anexo1

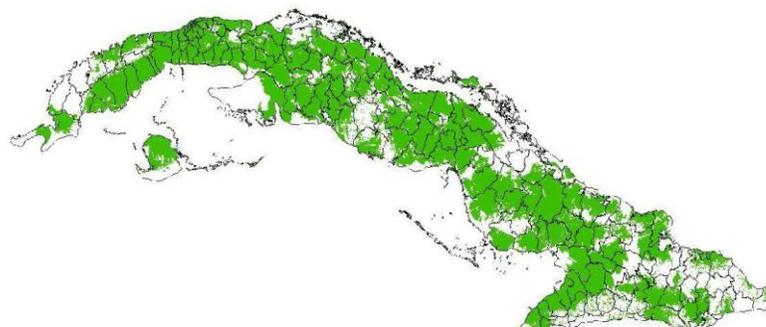


Figura 33 Mapa de Cobertura GSM en Cuba 2008.



Figura 34 Cobertura Celular en Cuba 2007.

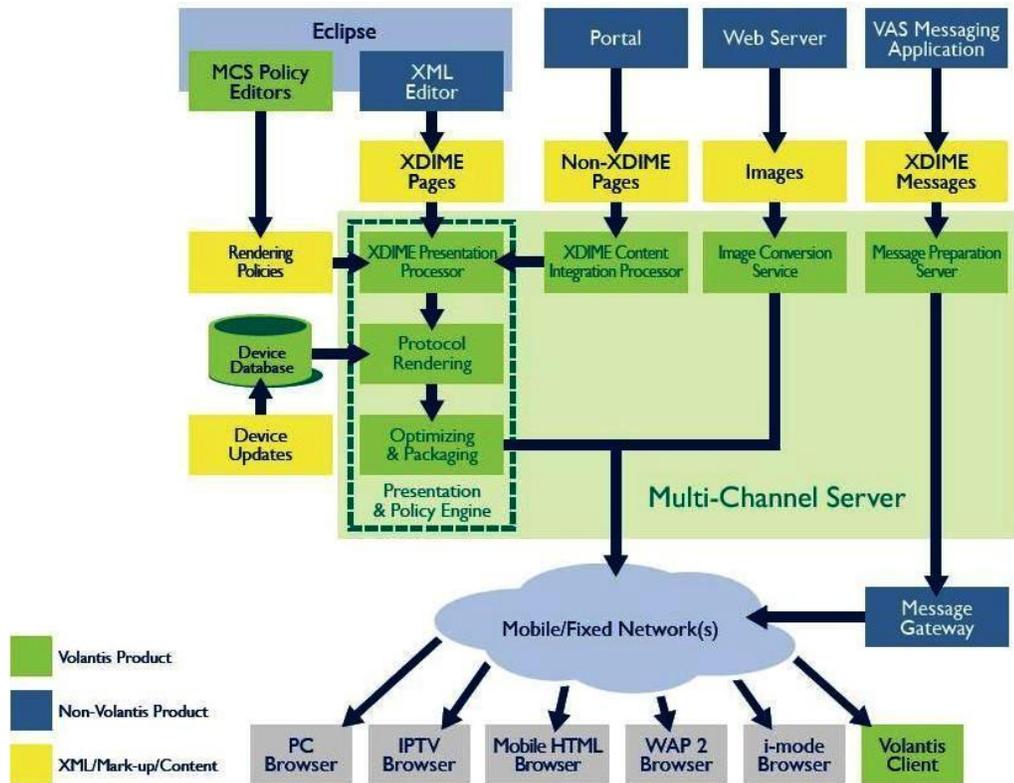


Figura 35 Solución brindada por Volantis Systems.

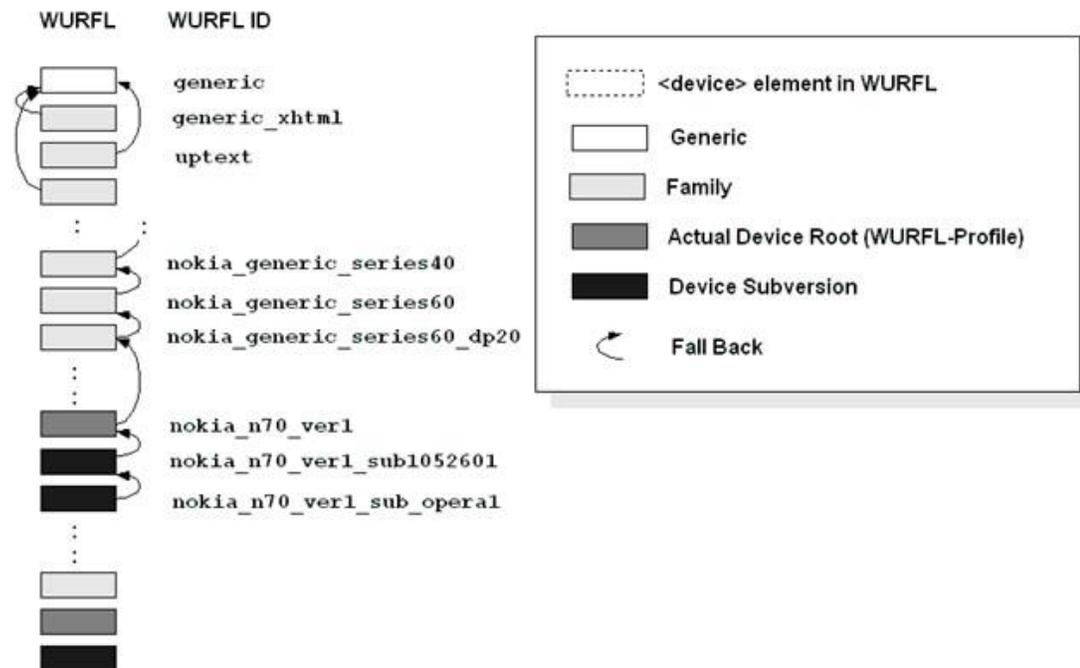


Figura 36 Estructura utilizada por WURFL.

Resumen de Principios: Buenas Prácticas en Web Móvil 1.0, propuestas por el W3C para la creación de sitios Web para móviles.

1. [[CONSISTENCIA DEL CONTENIDO](#)]

Asegúrese de que el contenido obtenido accediendo a una URI proporciona una consistencia de contenido coherente cuando se accede desde diferentes dispositivos.

2. [[CARACTERÍSTICAS](#)]

Aproveche las características del dispositivo para proporcionar una experiencia al usuario satisfactoria.

3. [[DEFICIENCIAS](#)]

Tome los pasos razonables acerca de la puesta en marcha de las deficiencias.

4. [[PRUEBAS](#)]

Realice las pruebas en dispositivos reales así como en emuladores.

5. [[URIS](#)]

Mantenga las URIs de las direcciones de acceso cortas.

6. [[BARRA DE NAVEGACIÓN](#)]

Proporcione una barra de navegación mínima al principio de la página.

7. [[BALANCE](#)]

Considere las ventajas entre tener muchos vínculos en una página y preguntar al usuario para seguir muchos vínculos hasta encontrar lo que está buscando.

8. [[NAVEGACIÓN](#)]

Proporcione mecanismos de navegación consistentes.

9. [[TECLAS DE ACCESO RÁPIDO](#)]

Asigne teclas de acceso rápido a los vínculos del menú de navegación y las funcionalidades que se usan más frecuentemente.

10. [[IDENTIFICAR DESTINO DEL VÍNCULO](#)]

Identifique claramente el destino de cada vínculo.

11. [[FORMATO DE DESTINO DEL VÍNCULO](#)]

Avise del formato del vínculo de destino a menos que sepa que el dispositivo lo soporta.

12. [[MAPAS DE IMÁGENES](#)]

No use mapas de imágenes a menos que sepas que el dispositivo los soporta eficazmente.

13. [[VENTANAS EMERGENTES](#)]

No provoque que se abran ventanas emergentes y no cambie la ventana actual sin informar al usuario.

14. [[ACTUALIZACIÓN AUTOMÁTICA](#)]

No cree páginas que se actualicen automáticamente, a menos que informe al usuario y proporcione medios para pararlo.

15. [[REDIRECCIÓN](#)]

No utilice etiquetas para Redirigir a páginas automáticamente. En su lugar, configure el servidor para realizar redirecciones por medio de código HTTP 3xx.

16. [[RECURSOS EXTERNOS](#)]

Mantenga el número de vínculos a recursos externos al mínimo.

17. [[ADECUADO](#)]

Asegúrese que el contenido es adecuado para su uso en el contexto de dispositivos móviles.

18. [[CLARIDAD](#)]

Use un lenguaje claro y simple.

19. [[LIMITACIÓN](#)]

Limite el contenido a lo que ha solicitado el usuario.

20. [[TAMAÑO DE PÁGINA USABLE](#)]

Divida las páginas en porciones usables pero de tamaño limitado.

21. [[LÍMITE DE TAMAÑO DE PÁGINA](#)]

Asegúrese de que el tamaño total de la página es apropiado para las limitaciones de memoria de los dispositivos.

22. [[SCROLL](#)]

Límite el scroll a una dirección, a menos que no se pueda evitar el scroll secundario.

23. [[MATERIA PRINCIPAL](#)]

Asegúrese de que el contenido principal de la página precede al contenido que no lo es.

24. [[IMÁGENES PARA ESPACIAR](#)]

No use imágenes para crear espacios en blanco.

25. [[IMÁGENES GRANDES](#)]

No use imágenes que no pueden ser mostradas en el dispositivo. Evite las imágenes grandes o imágenes con gran resolución excepto para información crítica que se perdería de otra manera.

26. [[USO DE COLOR](#)]

Asegúrese que la información transmitida por color lo está también disponible sin color.

27. [[CONTRASTE DE COLOR](#)]

Asegúrese de que la combinación del color del primer plano y el color de fondo proporciona suficiente contraste.

28. [[LEGIBILIDAD DE IMAGEN DE FONDO](#)]

Cuando use imágenes de fondo, asegúrese de que el contenido sigue siendo legible en el dispositivo.

29. [[TÍTULO DE PÁGINA](#)]

Proporcione un título corto pero descriptivo de la página.

30. [[SIN MARCOS](#)]

No use marcos.

31. [[ESTRUCTURA](#)]

Utilice los caracteres del código para indicar una estructura lógica en el documento.

32. [[SOPORTE DE TABLAS](#)]

No use tablas a menos que sepa que el dispositivo las soporte correctamente.

33. [[TABLAS ANIDADAS](#)]

No use tablas anidadas.

34. [[TABLAS PARA MAQUETAR](#)]

No use tablas para maquetar.

35. [[ALTERNATIVAS A LAS TABLAS](#)]

Cuando sea posible, use una alternativa para la presentación tabular.

36. [[ALTERNATIVAS NO-TEXTUALES](#)]

Ofrezca una alternativa textual para cada elemento no-textual.

37. [[OBJETOS O SCRIPTS](#)]

No confíe en objetos embebidos o scripts.

38. [[TAMAÑO ESPECIFICO DE IMÁGENES](#)]

Especifique un tamaño de imagen en el código, si tiene un tamaño intrínseco.

39. [[REDIMENSIONAMIENTO DE IMÁGENES](#)]

Redimensione las imágenes en el servidor, si tienen un tamaño intrínseco.

40. [[CÓDIGO VÁLIDO](#)]

Cree documentos que validen las gramáticas formales publicadas.

41. [[MEDIDAS](#)]

No utilice medidas en pixeles y no utilice unidades absolutas en valores de los atributos del código y en los valores de las propiedades de las hojas de estilo.

42. [[USE HOJAS DE ESTILO](#)]

Use hojas de estilos para controlar la disposición y presentación, a menos que sepa que el dispositivo no las soporta.

43. [[SOPORTE DE HOJAS DE ESTILO](#)]

Organice os documentos para que en caso de necesidad puedan ser leídos sin hojas de estilos.

44. [[TAMAÑO DE HOJAS DE ESTILO](#)]

Mantenga las hojas de estilos ligeras.

45. [[MINIMIZE](#)]

Use un código conciso y eficiente.

46. [[SOPORTE DEL FORMATO DEL CONTENIDO](#)]

Envíe el contenido en un formato que sepa que va a ser soportado por el dispositivo.

47. [[FORMATO DEL CONTENIDO PREFERIDO](#)]

Cuándo sea posible, envíe el contenido en un formato preferido.

48. [[SOPORTE DE LA CODIFICACIÓN DE CARACTERES](#)]

Asegúrese de que el contenido está codificado usando un juego de caracteres que conoce que va a ser soportado por el dispositivo.

49. [[JUEGO DE CARACTÉRES UTILIZADO](#)]

Indique en la respuesta del servidor el juego de caracteres que está siendo utilizado.

50. [[MENSAJES DE ERROR](#)]

Proporcione mensajes de error informativos y los mecanismos de navegación lejos del mensaje de error para volver a la información útil.

51. [[COOKIES](#)]

No confíe en que las cookies estén disponibles.

52. [[CACHE](#)]

Proporcione información de caché en las respuestas HTTP.

53. [[FUENTES TIPOGRÁFICAS](#)]

No confíe en el soporte de las fuentes tipográficas indicadas en los estilos.

54. [[MINIMIZE PULSACIONES DE TECLAS](#)]

Mantenga el número de pulsaciones de teclas al mínimo.

55. [[EVITE INTRODUCCIÓN DE TEXTO](#)]

Evite la libre entrada de texto cuando sea posible.

56. [[PROPORCIONE VALORES POR DEFECTO](#)]

Ofrezca valores por defecto preseleccionados cuando sea posible.

57. [[MODO DE ENTRADA POR DEFECTO](#)]

Especifique una forma por defecto de introducir texto, idioma y/o forma de entrada, si se sabe que el dispositivo lo soporta.

58. [[ORDEN DE TABULACIÓN](#)]

Cree un orden lógico de navegación entre vínculos, controles de formulario y objetos.

59. [[ETIQUETADO DE CONTROLES](#)]

Asigne etiquetas a todos los controles de formulario de forma apropiada y asocie explícitamente las etiquetas con los controles de formulario.

60. [[CONTROL DE POSICIÓN](#)]

Posicione las etiquetas para que se muestren correctamente en relación con los controles de formulario a los que se refieren.

Anexo 2

Diagramas de clases de análisis.

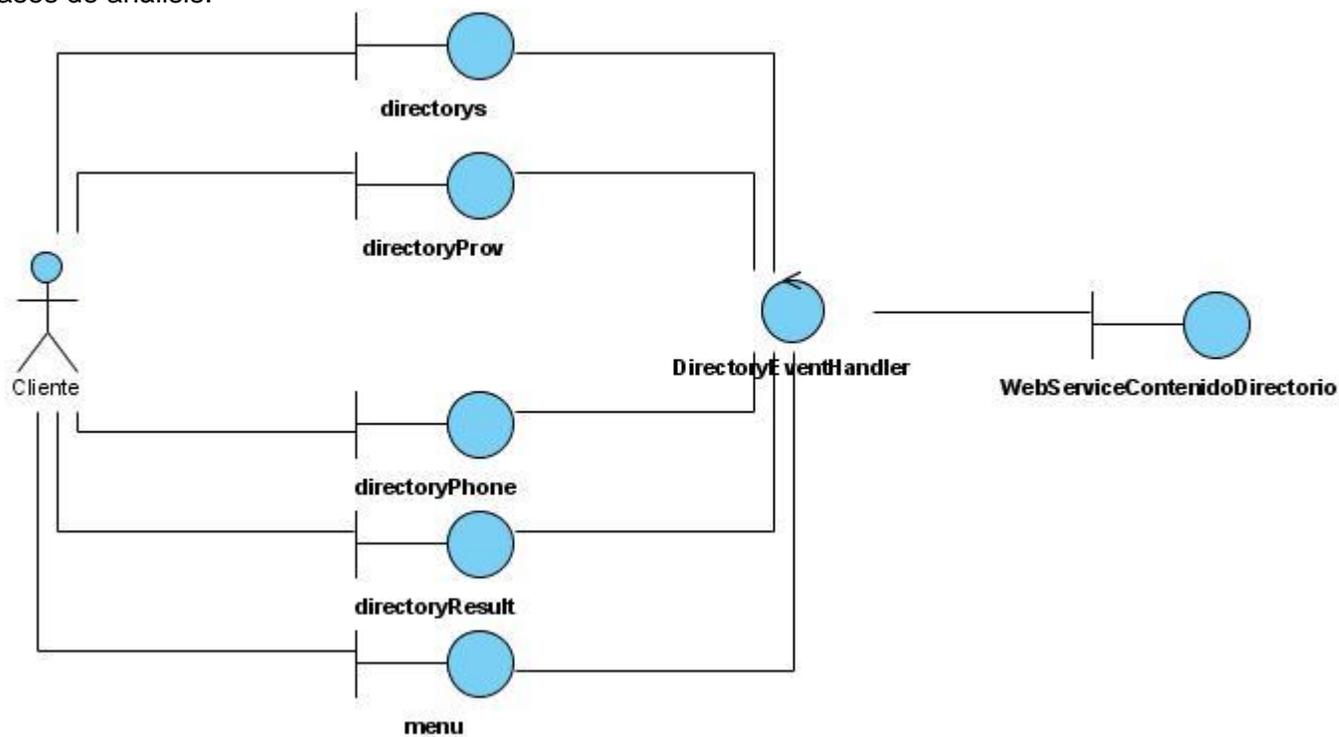


Figura 37 Diagrama de clases de análisis CU_MostrarDirectorioTelefonico.

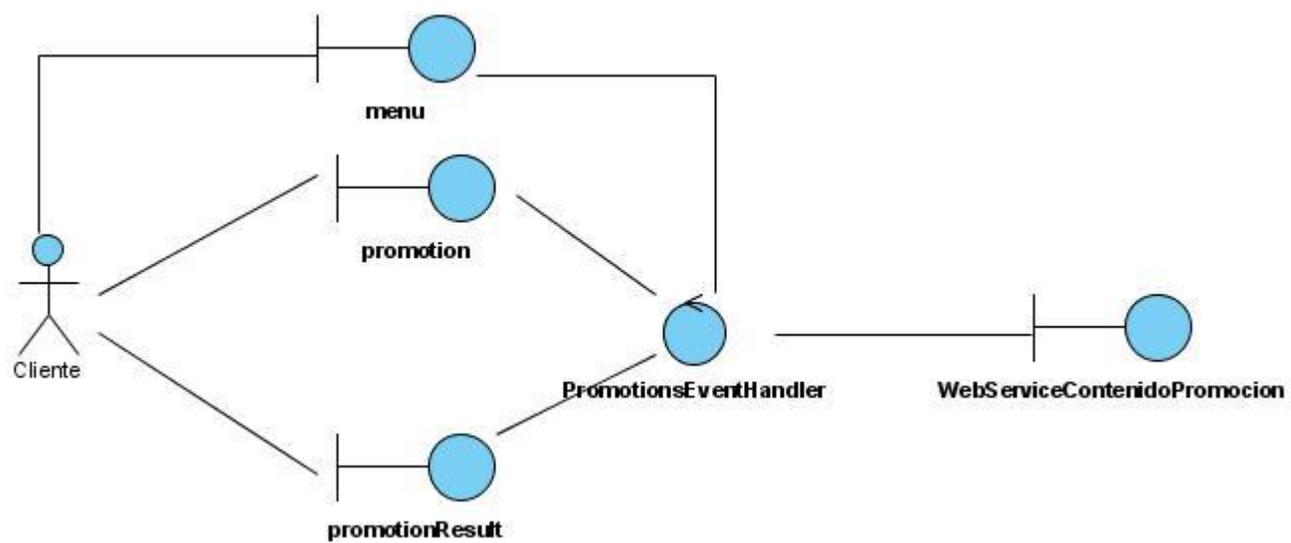


Figura 38 Diagrama de clases de análisis CU_MostrarPromociones.

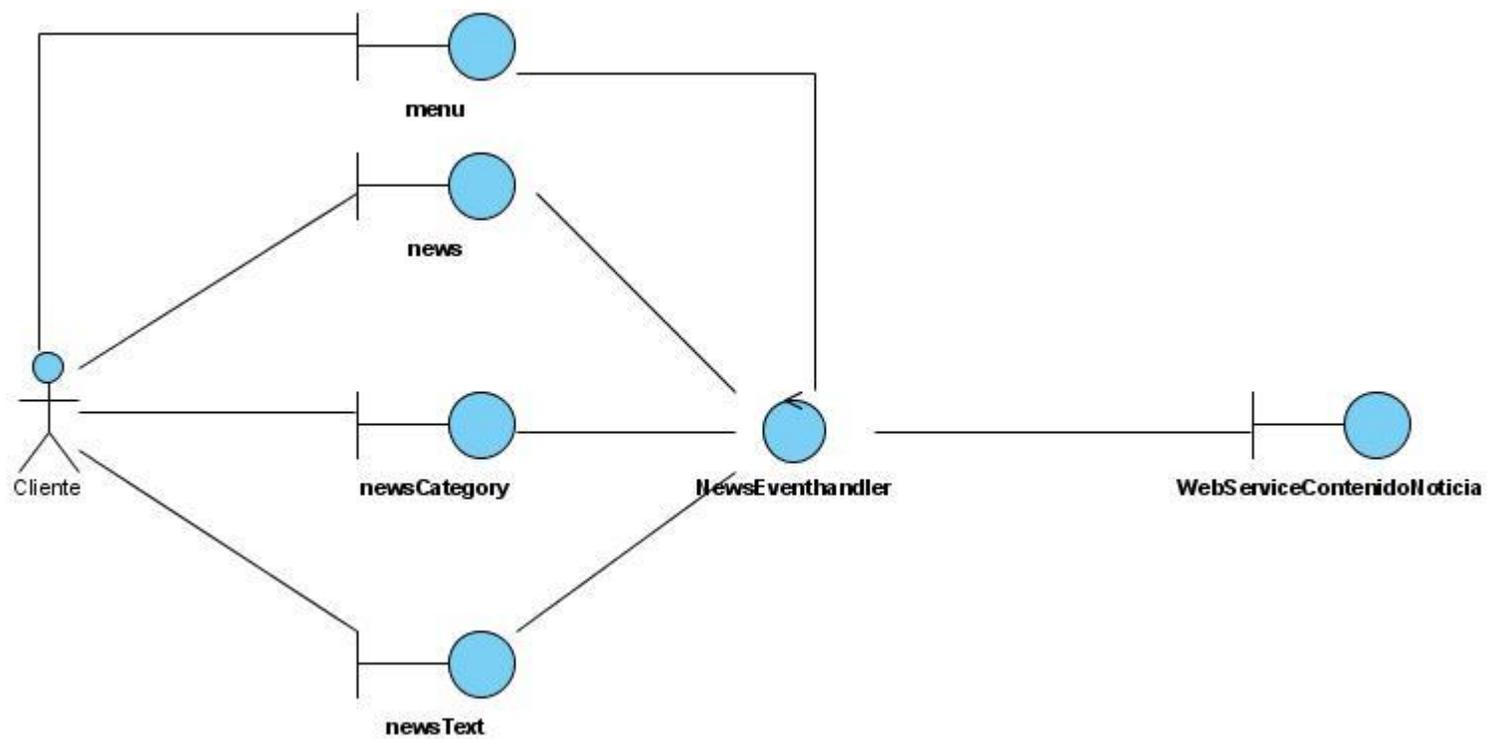


Figura 39 Diagrama de clases de análisis CU_MostrarNoticias.

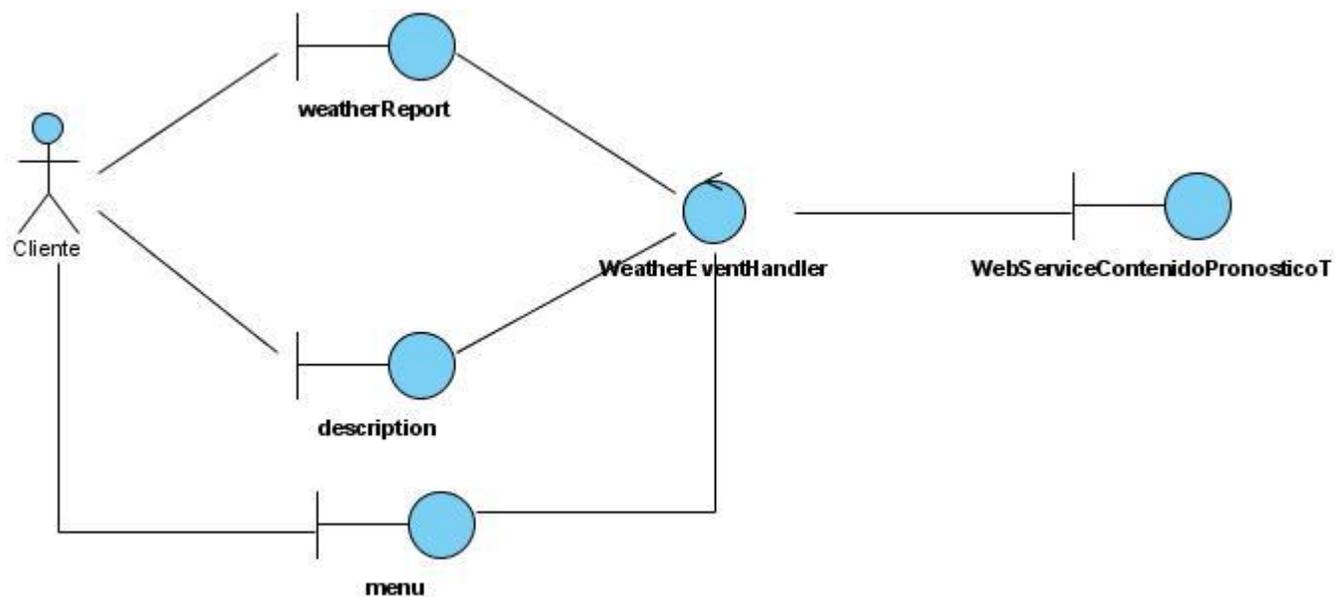


Figura 40 Diagrama de clases de análisis CU_MostrarPronosticoT.

Diagramas de Secuencia de análisis.

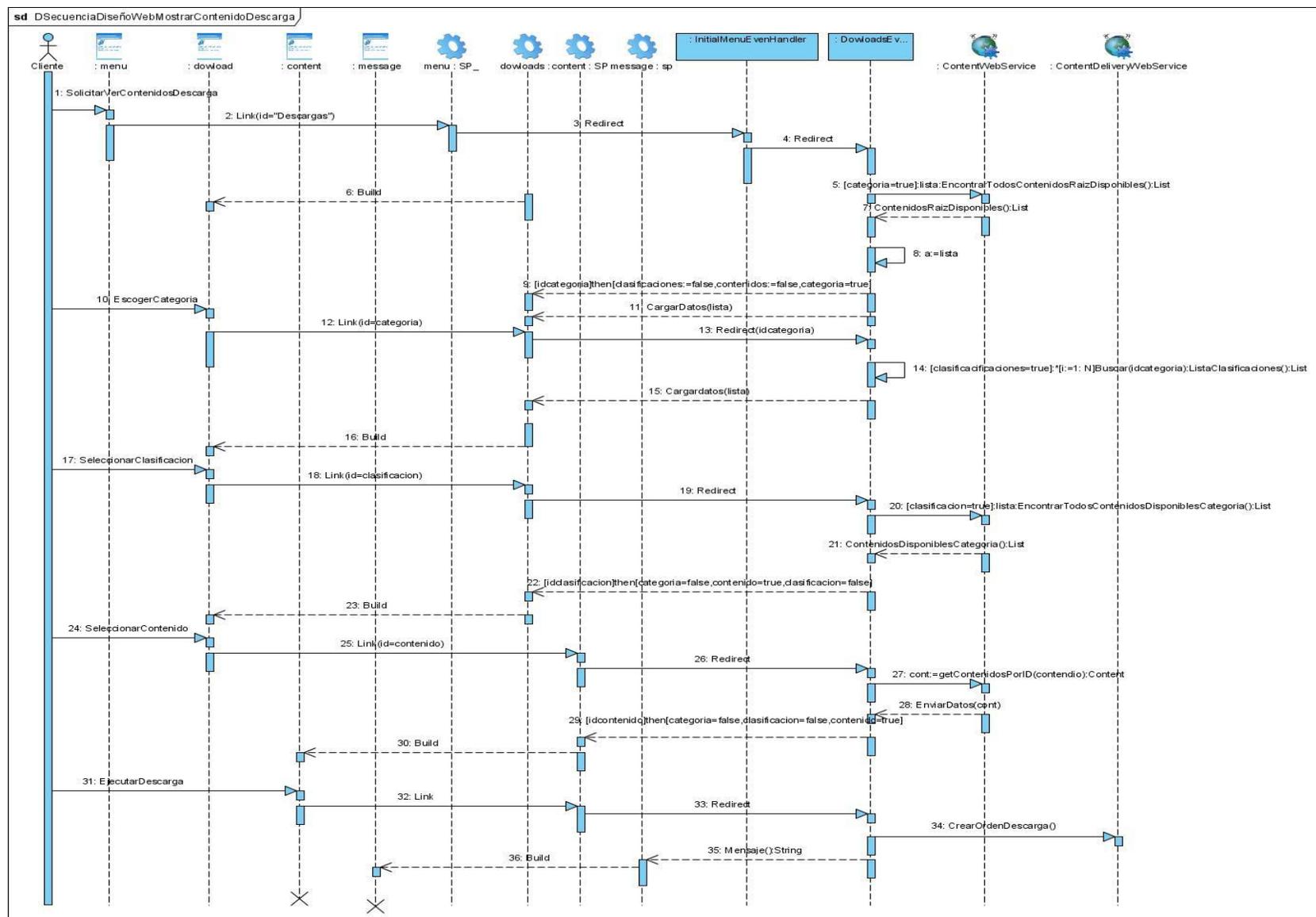


Figura 41 Diagrama de secuencia de análisis CU_MostrarContenidoDescarga.

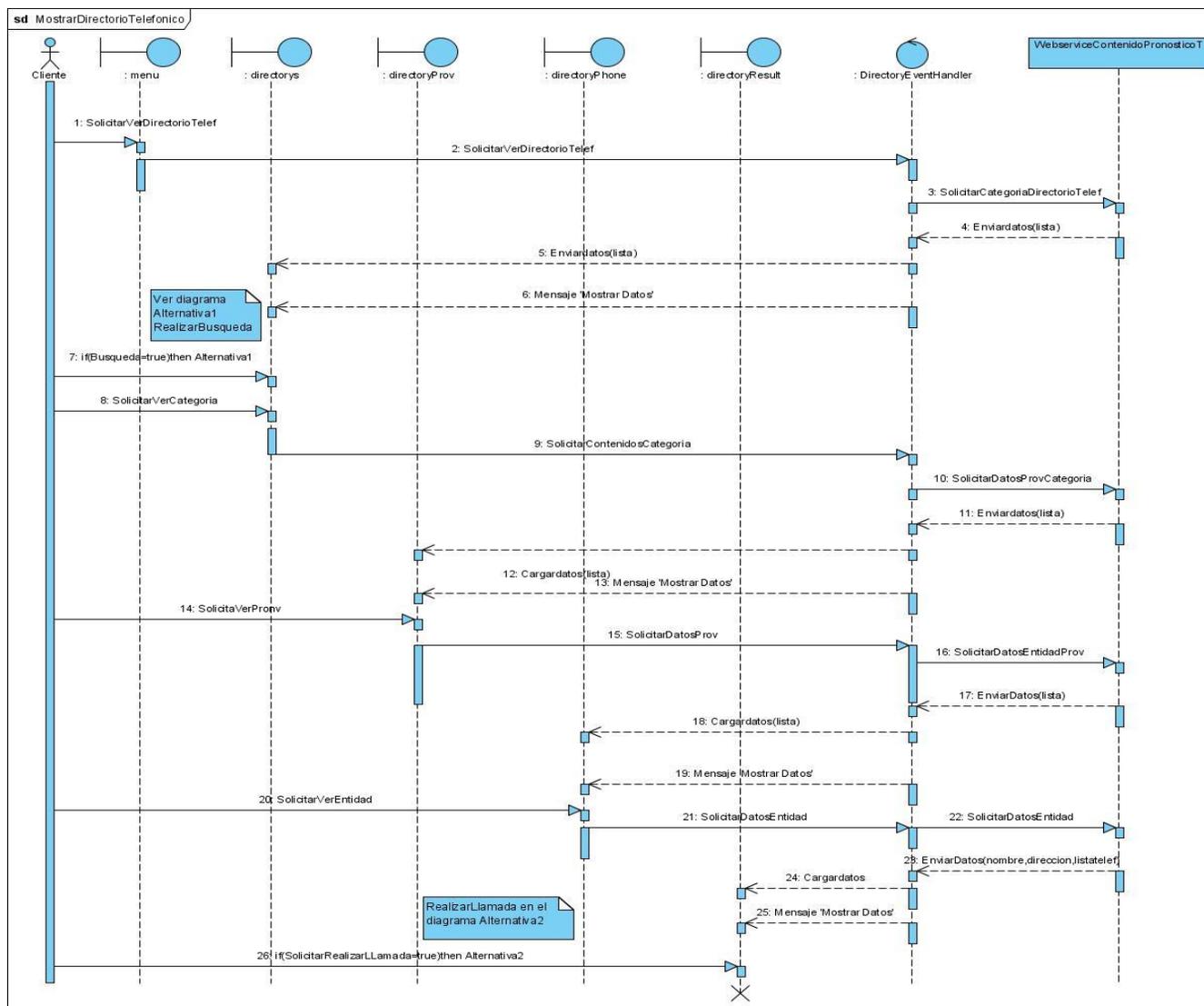


Figura 42 Diagrama de secuencia de análisis CU_MostrarDirectorioTelefonico.

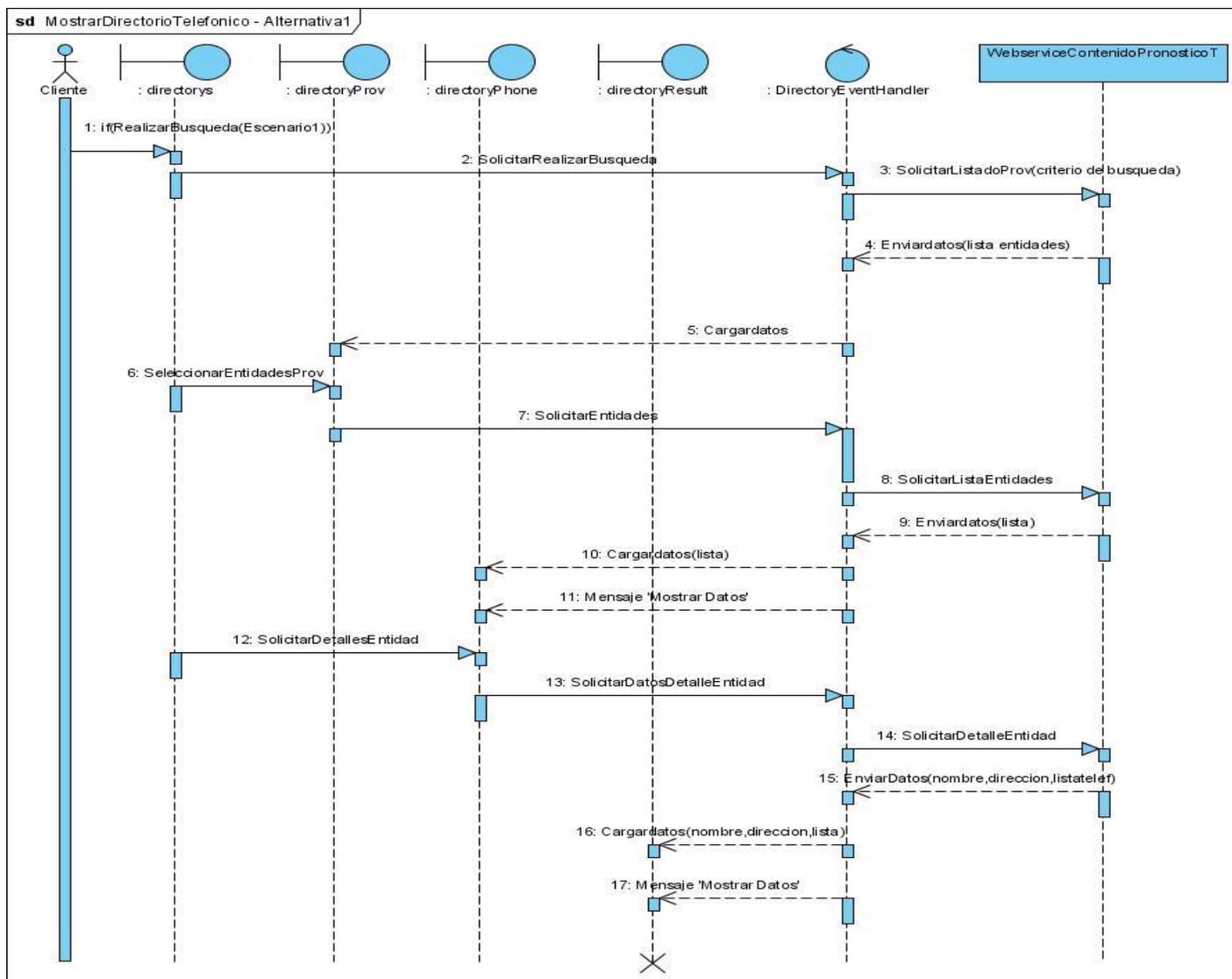


Figura 43 Diagrama de secuencia de análisis CU_MostrarDirectorioTelefonico-Alternativa1..

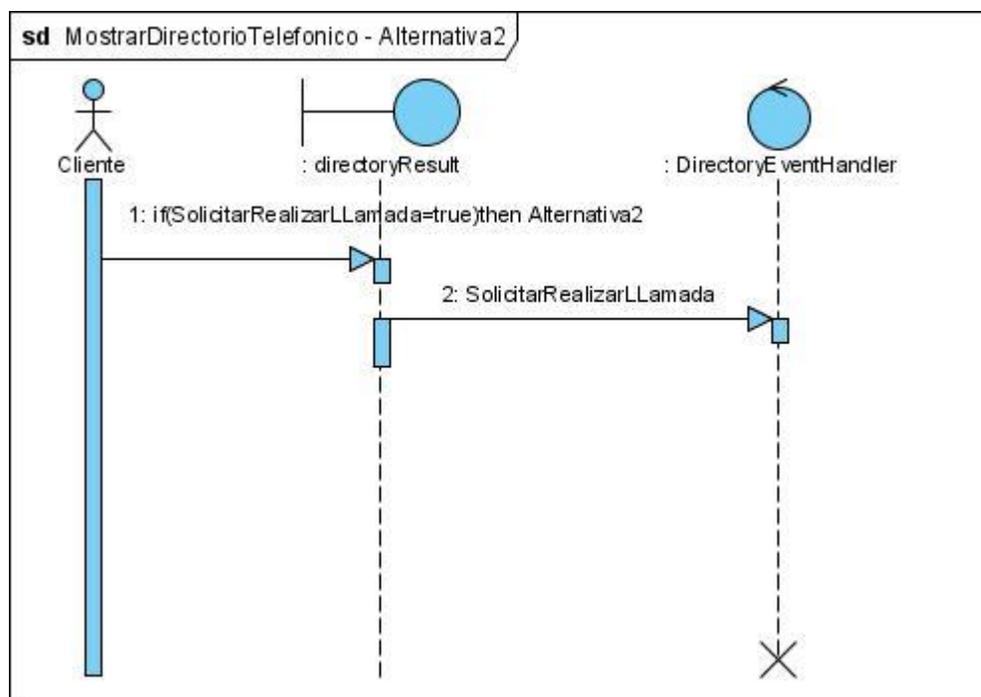


Figura 44 Diagrama de secuencia de análisis CU_MostrarDirectorioTelefonico-Alternativa2.

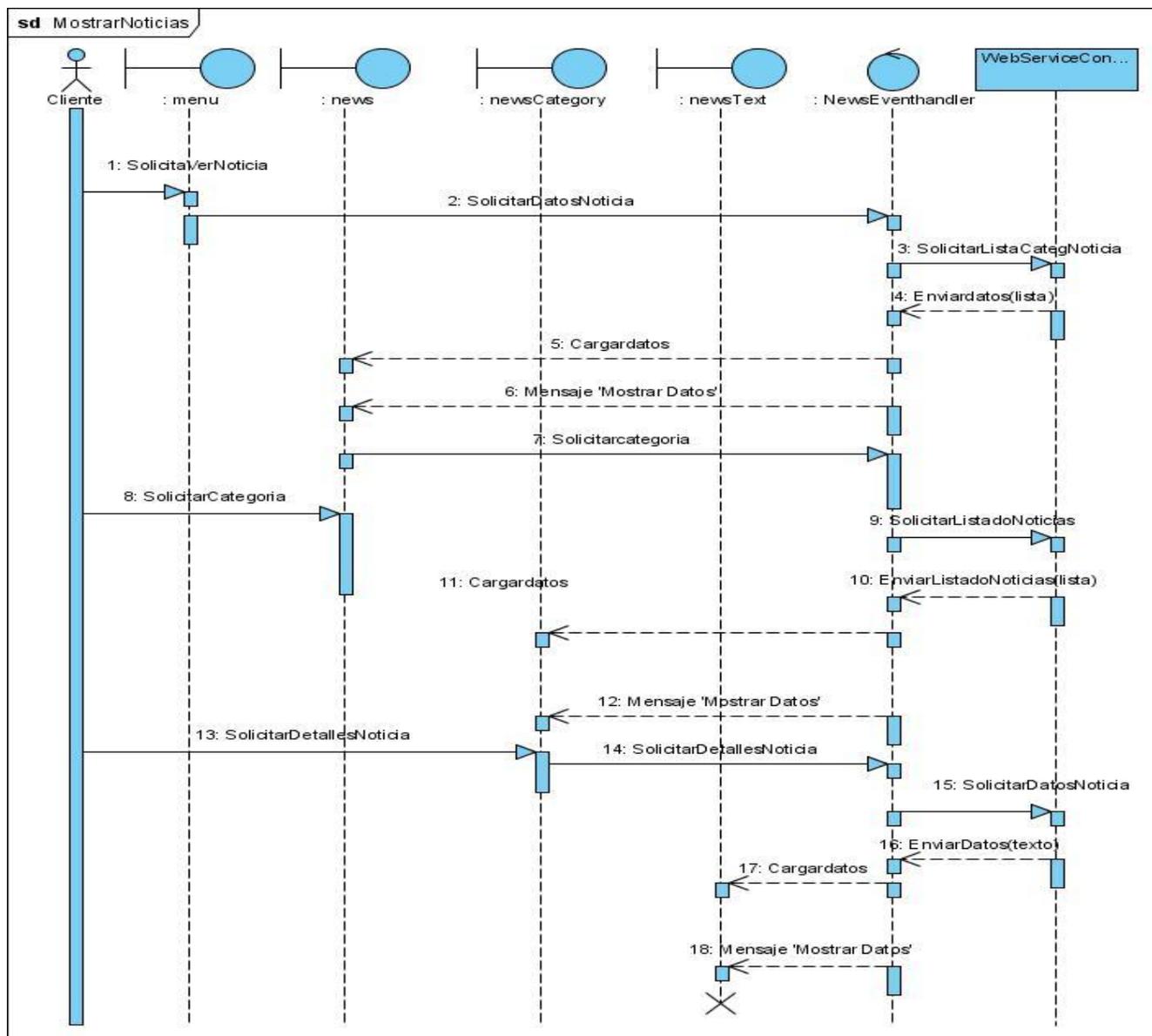


Figura 45 Diagrama de secuencia de análisis CU_MostrarNoticias.

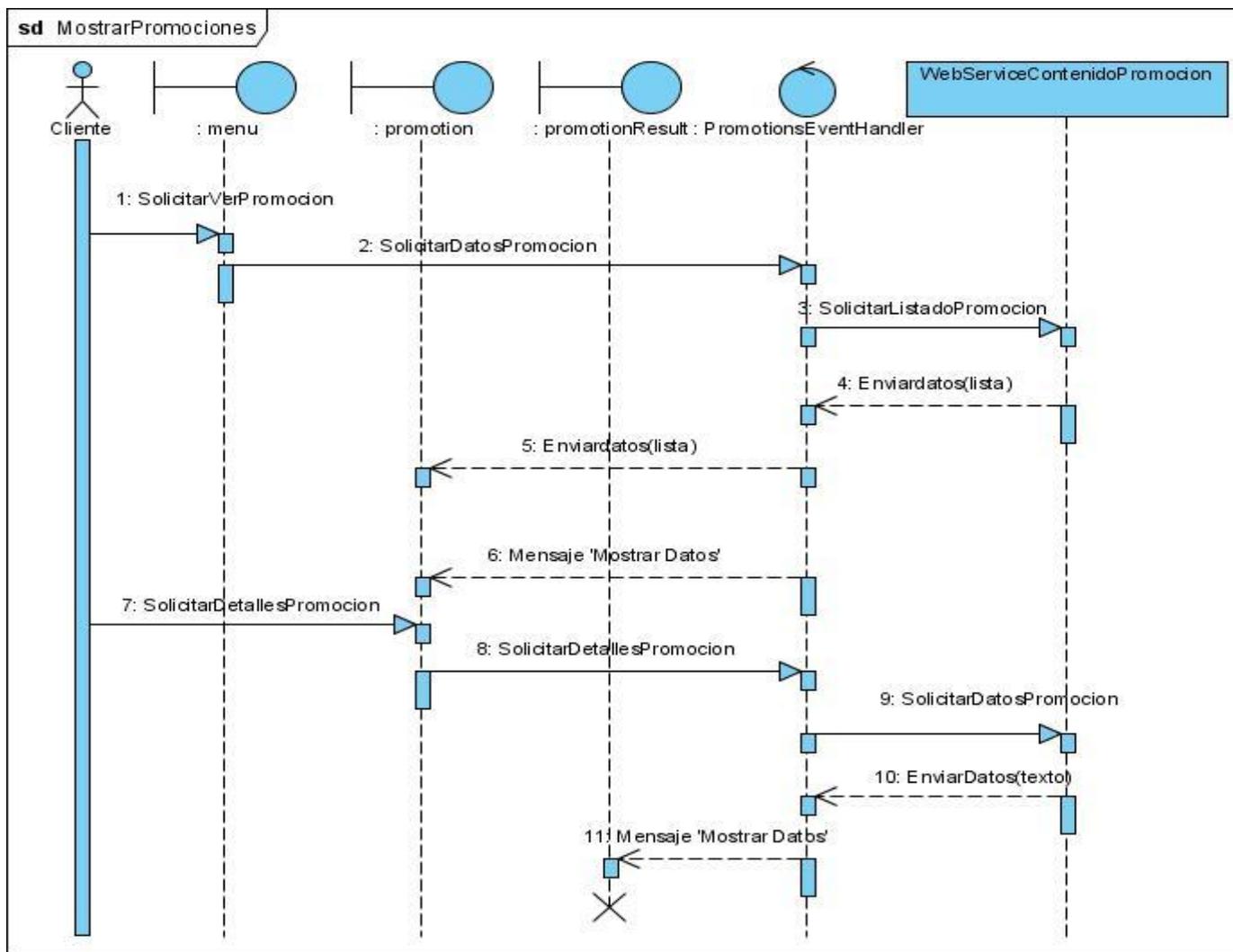


Figura 46 Diagrama de secuencia de análisis CU_MostrarPromociones.

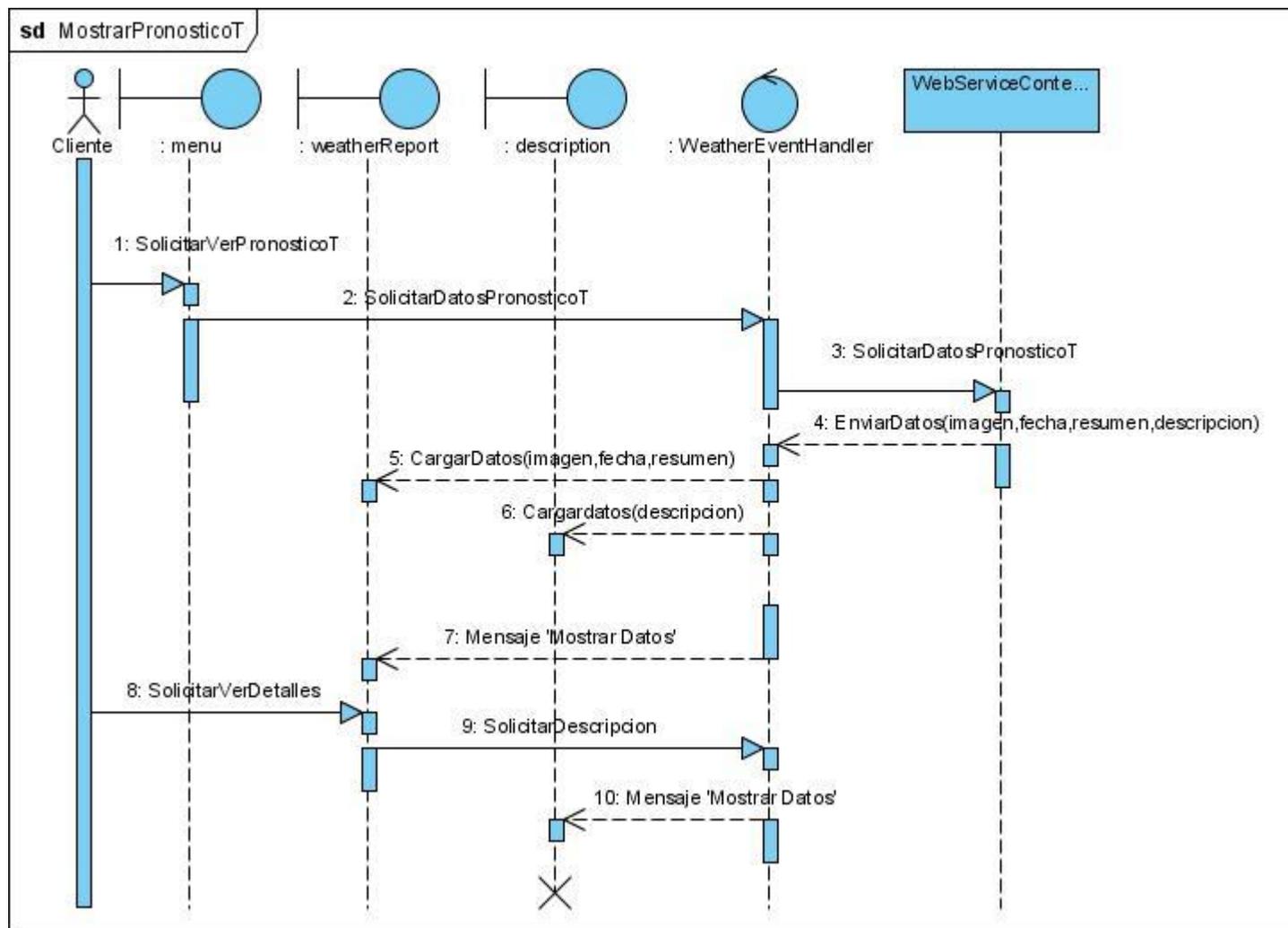


Figura 47 Diagrama de secuencia de análisis CU_MostrarPronosticoTiempo.

Diagramas de clase del Diseño Web.

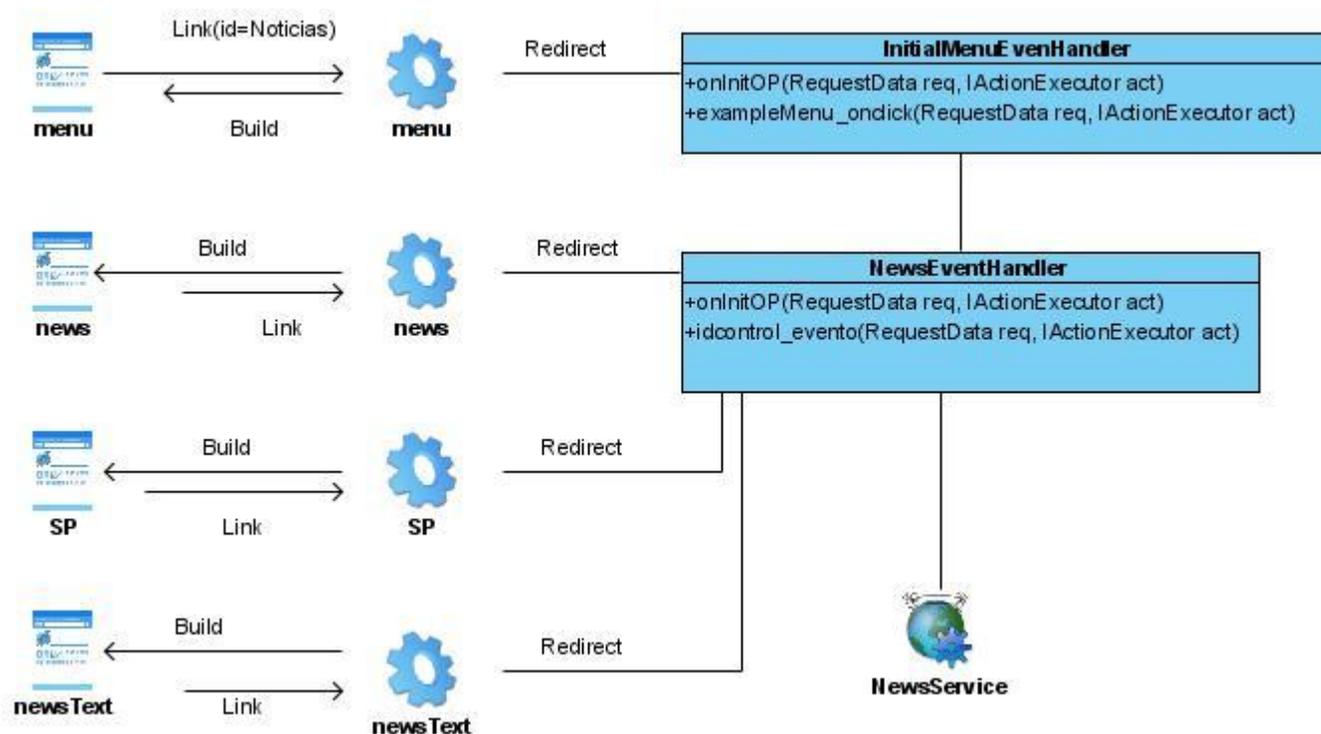


Figura 48 Diagrama de Clases de Diseño Web CU_MostrarNoticias.

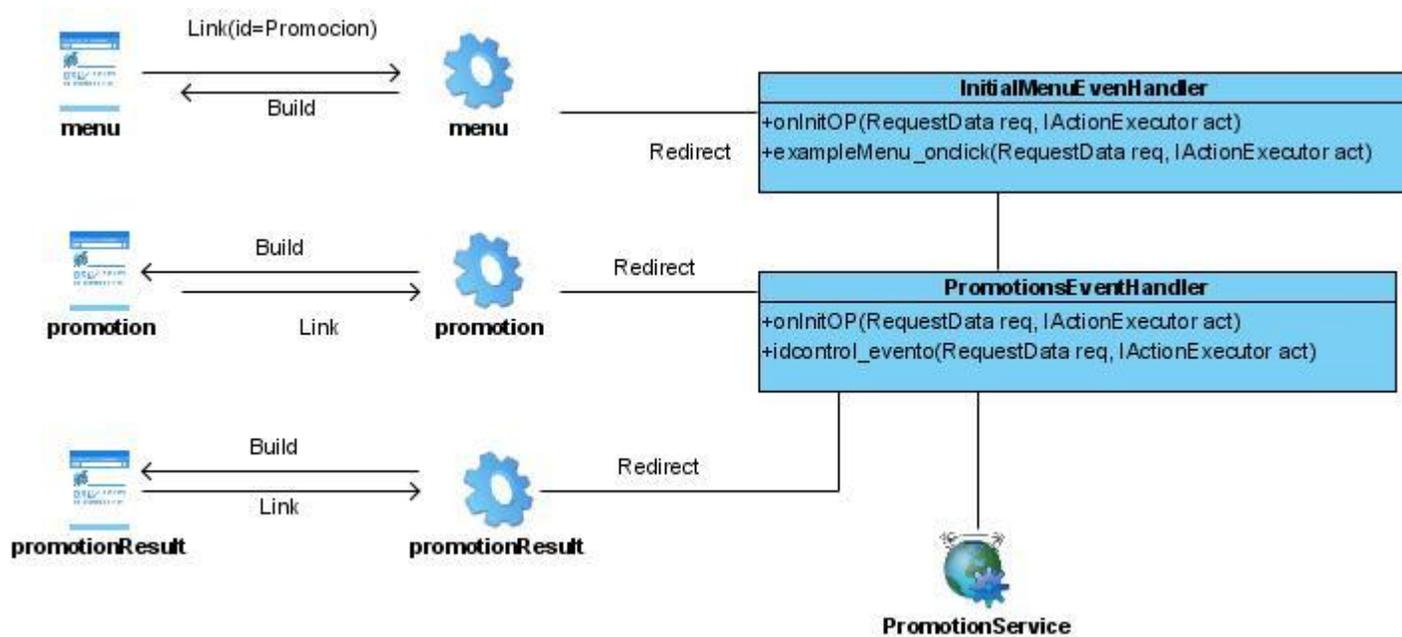


Figura 49 Diagrama de Clases de Diseño Web CU_MostrarPromociones.

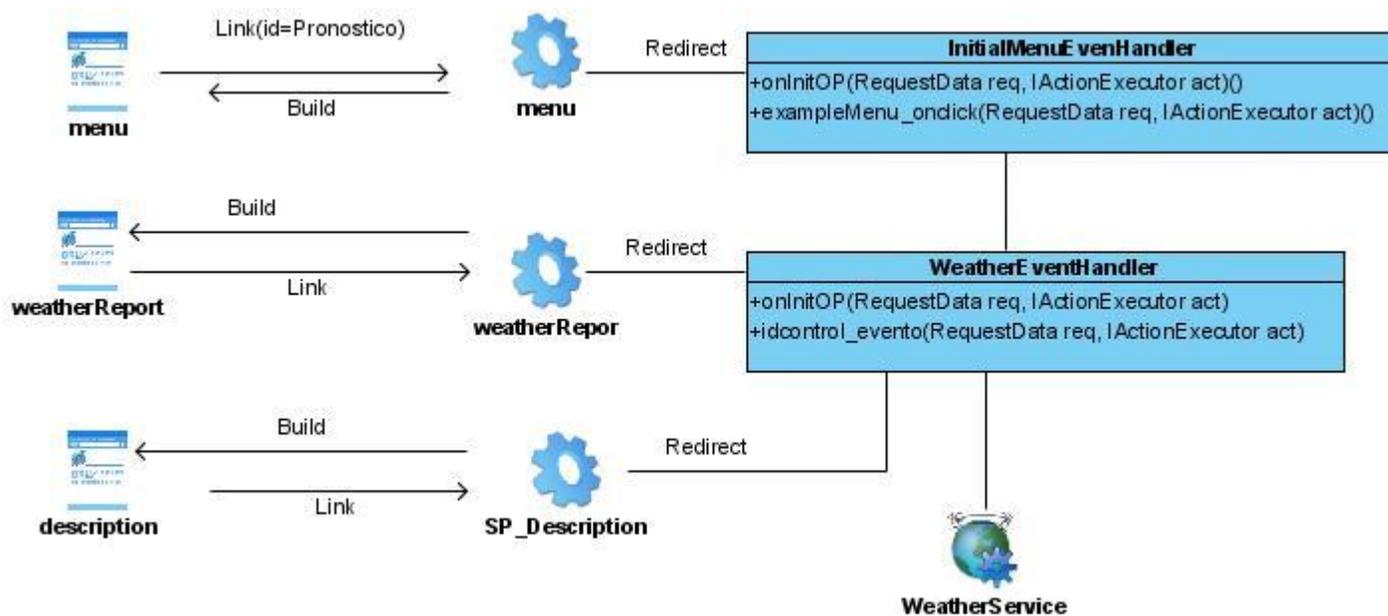


Figura 50 Diagrama de Clases de Diseño Web CU_MostrarPronosticoTiempo.

Diagramas de Secuencia del Diseño Web.

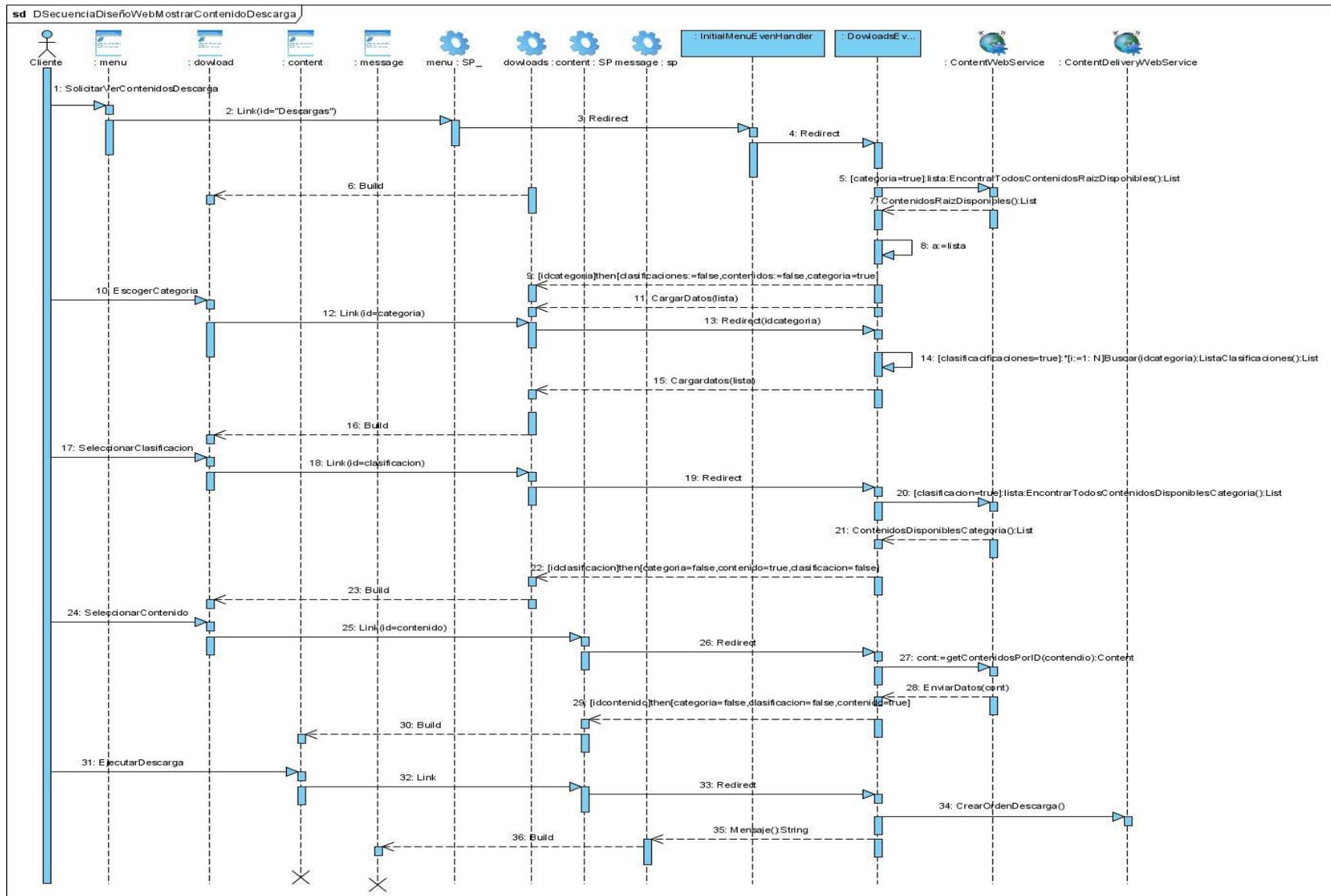


Figura 51 Diagrama de Secuencia de Diseño Web CU_MostrarContenidoDescarga.

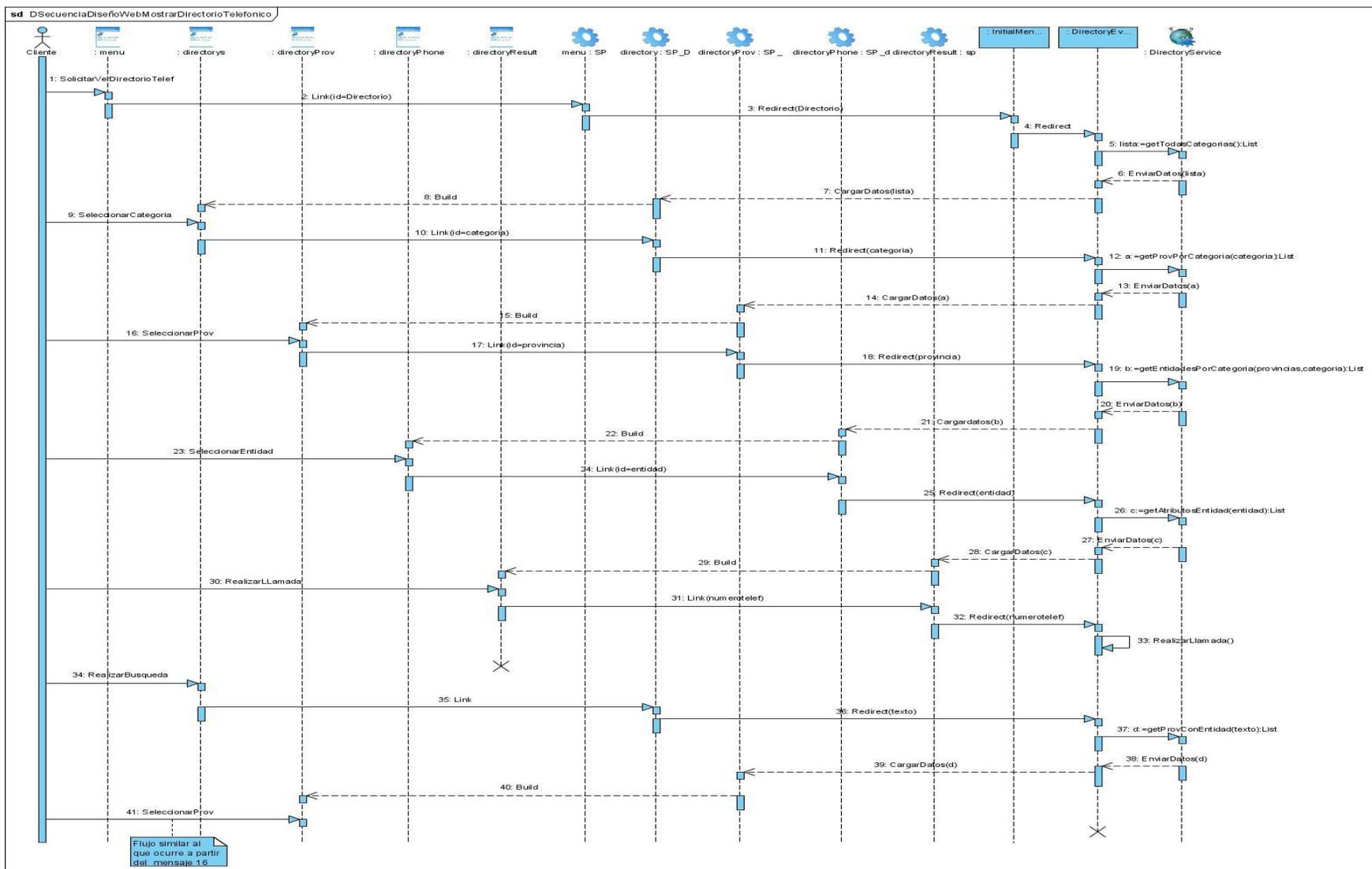


Figura 52 Diagrama de Secuencia de Diseño Web CU_MostrarDirectorioTelefonico.

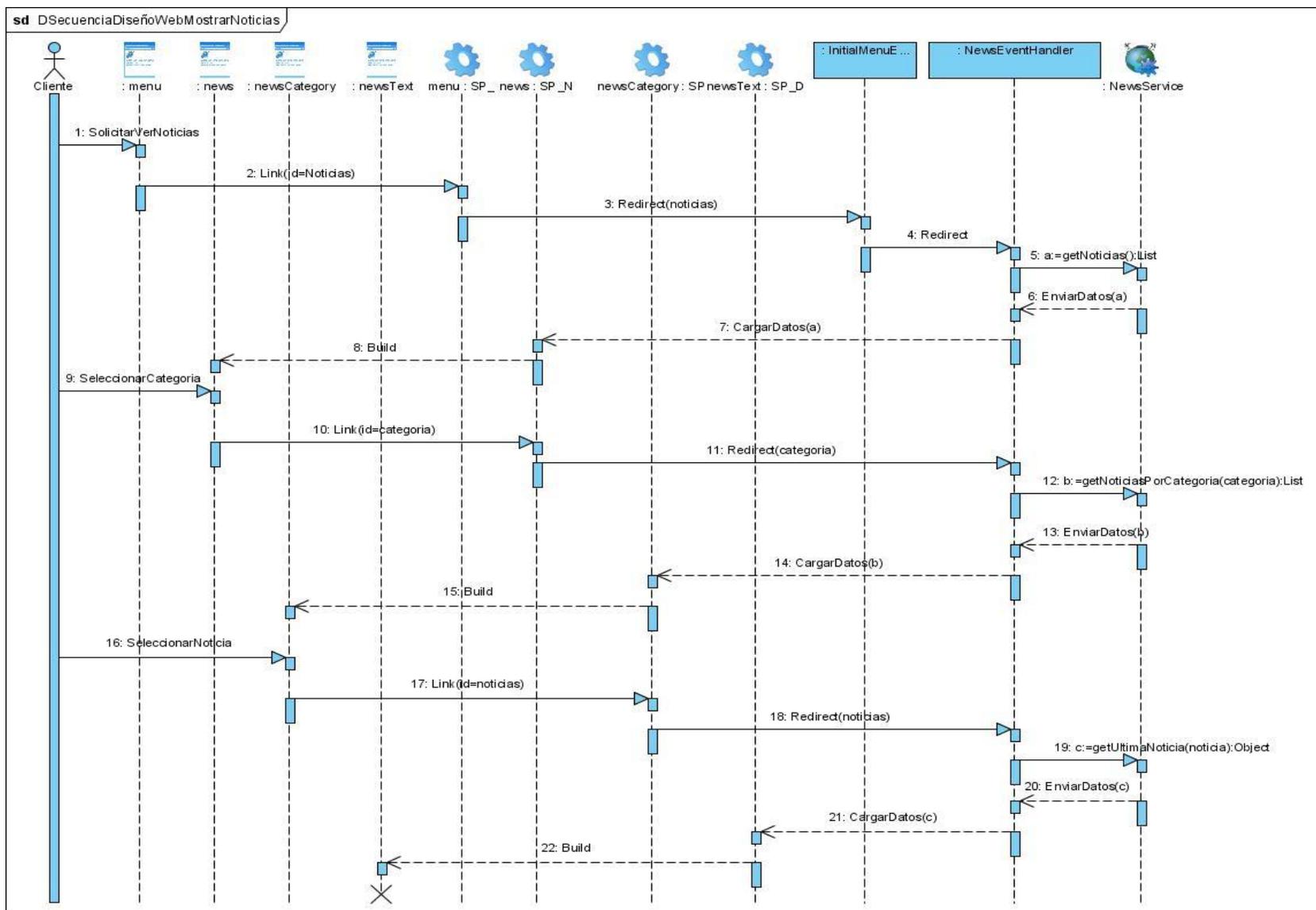


Figura 53 Diagrama de Secuencia de Diseño Web CU_MostrarNoticias.

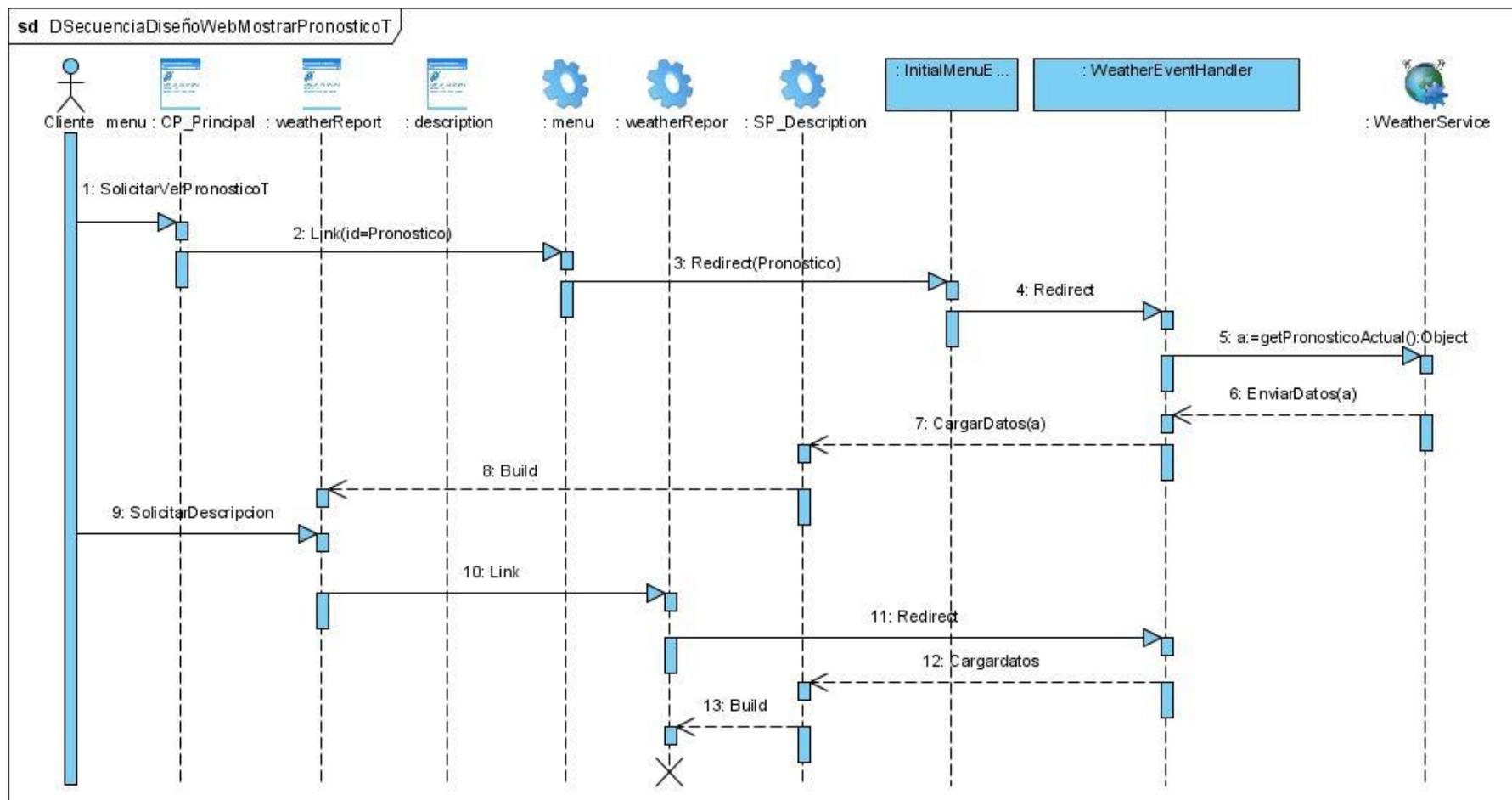


Figura 54 Diagrama de Secuencia de Diseño Web CU_MostrarPronosticoTiempo.

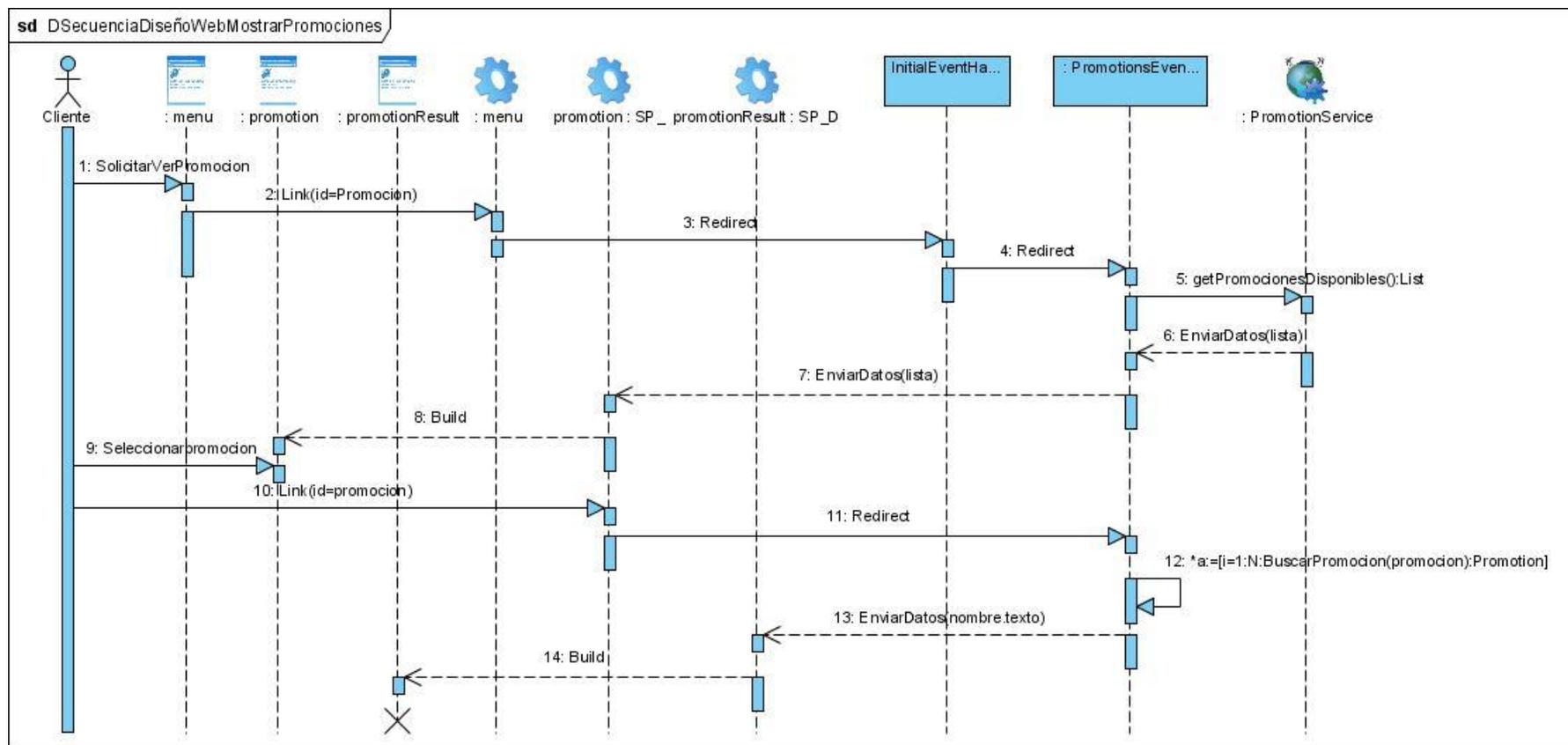


Figura 55 Diagrama de Secuencia de Diseño Web CU_MostrarMostrarPromociones.

GLOSARIO DE TÉRMINOS

AJAX: Acrónimo de Asynchronous JavaScript And XML, es una técnica de desarrollo Web para crear aplicaciones interactivas. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los dispositivos mientras se mantiene la comunicación asíncrona con el servidor en segundo plano.

API: Una interfaz de programación de aplicaciones (Application Programming Interface) es el conjunto de funciones y procedimientos (o métodos, si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Aplicación Web: 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.

Aplicación: En informática es un tipo de software diseñado para facilitar al usuario la realización de una determinada tarea o trabajo.

CMS: Un Sistema de gestión de contenidos (Content Management System) es un programa que permite crear una estructura de soporte (framework) para la creación y administración de contenidos, principalmente en páginas web, por parte de los participantes. Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio.

CSS: Las hojas de estilo en cascada (Cascading Style Sheets) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

CHTML: (Compact HTML) es un subconjunto del lenguaje de marcas HTML que funciona en los teléfonos móviles. Añade varias características que no se encuentra en el estándar HTML, en particular el número de teléfono, accesos directos a enlaces, imágenes y caracteres.

Datagramas: Es un fragmento de paquete que es enviado con la suficiente información como para que la red pueda simplemente encaminar el fragmento hacia el equipo terminal de datos

receptor, de manera independiente a los fragmentos restantes. La estructura de un datagrama es: cabecera y datos.

DIAL: Lenguaje de Generación de Contenido de Independencia de Dispositivo es un perfil de lenguaje basado en vocabularios XML del W3C existentes y módulos CSS. Estos ofrecen mecanismos estándares para representación de estructuras de páginas Web, presentación e interacción de formularios, facilitando la creación de contenido para una extensa gama de dispositivos móviles.

Framework: Es una estructura de soporte definida, representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

Gateway: Es un dispositivo 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.

GSM: Sistema global para la comunicación entre móviles, actualmente el estándar más extendido entre los teléfonos móviles a nivel mundial, con un total estimado de 1800 millones de usuarios repartidos en todo el orbe. Así los operadores telefónicos son capaces de ofrecer un servicio más seguro y con compatibilidad internacional o 'roaming'.

Hipertexto: Es el nombre que recibe el texto que conduce a su usuario a otro texto relacionado. La forma más habitual de hipertexto en documentos es la de hipervínculos o referencias cruzadas automáticas que van a otros documentos.

HTML: Lenguaje de Marcas de Hipertexto (HyperText Markup Language). Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

IDE: Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un

constructor de interfaz gráfica GUI. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes.

Internet: Es un conjunto descentralizado de redes de comunicación interconectadas, que utilizan la familia de protocolos TCP/IP, posee alcance mundial, constituye una red mundial de comunicación.

iPhone: De la compañía Apple Inc., es un teléfono inteligente multimedia que permite la conexión a Internet por medio de una red inalámbrica, posee pantalla táctil y una interfaz de hardware despojada de elementos sobrantes.

JSR-170: Es un emergente estándar industrial creado para definir la manera en la que las aplicaciones de portal y de gestión de contenidos deben de trabajar conjuntamente. Define un interfaz estándar para acceder a un repositorio de contenidos desde una aplicación, esta podrá utilizarse independientemente del proveedor de gestor de contenidos implantado.

LCD: Común Mínimo Denominador, termino empleado por los desarrolladores de software para designar un conjunto de características comunes que cumplen ciertos dispositivos o navegadores web entre sí.

Lenguaje de marca: Es un conjunto de reglas que utilizan una sintaxis predefinida que sirven para estructurar un documento electrónico, utilizando etiquetas al principio y al final de un elemento.

Lenguaje de script: Un lenguaje de script es un pequeño lenguaje de programación cuyo código se inserta dentro del documento HTML. Este código se ejecuta en el navegador del usuario al cargar la página, o cuando sucede algo especial como puede ser el pulsar sobre un enlace. Estos lenguajes permiten variar dinámicamente el contenido del documento, modificar el comportamiento normal del navegador.

Micro navegador: Es un navegador web diseñado para el uso en dispositivos móviles y de reducidas dimensiones, como los PDA o los teléfonos móviles. Los micro navegadores están optimizados para mostrar contenido de internet en pantallas reducidas, y utilizan tamaños de archivo reducidos para ser instalados en dispositivos con memorias de baja capacidad.

MIME: Extensiones de Correo de Internet Multipropósito (Multipurpose Internet Mail Extensions), son una serie de convenciones o especificaciones dirigidas a que se puedan intercambiar a través de Internet todo tipo de archivos (texto, audio, vídeo, etc.) de forma transparente para el usuario. Una parte importante del MIME está dedicada a mejorar las posibilidades de transferencia de texto en distintos idiomas y alfabetos.

MML: Lenguaje de etiquetas similar al XML que permite insertar música en documentos de texto de tipo web que pueden ser interpretados tanto por humanos como por los ordenadores.

MMS: El sistema de mensajería multimedia es un estándar de mensajería que le permite a los teléfonos móviles enviar y recibir contenidos multimedia, incorporando sonido, video y fotos. El límite de cada mensaje multimedia suele ser de 100 o 300 KB, dependiendo de cada móvil, si bien ese límite lo definen el operador o las características del terminal y no el protocolo.

MVC: Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

MyMobileWeb: Es una plataforma de software para móviles de código abierto, proporcionando un entorno avanzado de adaptación de contenidos. MyMobileWeb es la única web para móviles de código abierto motor de adaptación propuesto para el desarrollo compatible con los sitios dotMobi.

Navegador: Un navegador web o cliente HTTP, es un programa que permite interpretar la información y el código que contiene una página web (esté alojada en un servidor dentro de la World Wide Web o en uno local) y presentarla de manera legible a los usuarios.

OMA: La misión de la Open Mobile Alliance es facilitar la adopción mundial de usuarios de los servicios móviles de datos, especificando el mercado móvil de los habilitadores de servicios que garanticen la interoperabilidad de los servicios a través de los dispositivos, geografías, proveedores de servicios, operadores, redes y al mismo tiempo a las empresas para competir a través de la innovación y la diferenciación.

Open-source: Práctica de desarrollo de software que promueve el acceso al código fuente de los sistemas computacionales.

RDF: Marco de Descripción de Recursos (Resource Description Framework) es un framework para metadatos en la World Wide Web, desarrollado y estandarizado por el W3C.

Script: Es un guión o conjunto de instrucciones. Permiten la automatización de tareas creando pequeñas utilidades. Es un trozo de código que puede recibir argumentos y devolver un valor. Los scripts se utilizan para generalizar código repetido y así ahorrar espacio y ganar velocidad.

SCXML: Es un estándar del W3C que permite la definición de máquinas de estados, SCXML es bastante flexible, permite control de abstracción.

Smartphone: Es un dispositivo electrónico que funciona un teléfono celular con características similares a las de un computador personal. Los teléfonos inteligentes permiten la instalación de programas para incrementar el procesamiento de datos y la conectividad.

SMS: Servicio de Mensaje Corto. Disponible en redes digitales como Sistema global para la comunicación entre móviles (GSM) permitiendo enviar y recibir mensajes de texto de hasta 160 caracteres a teléfonos móviles vía el centro de mensajes de un operador de red.

SVG: Scalable Vector Graphics es un lenguaje para describir gráficos vectoriales bidimensionales, tanto estáticos como animados en XML.

Telefonía celular: También llamada telefonía móvil, básicamente está formada por dos grandes partes, una red de comunicaciones y los terminales que permiten el acceso a dicha red.

TeleVAS: Servicios Telefónicos de Valor Añadido, se refiere a los servicios que se brindan por parte de las entidades de telecomunicaciones distintos al de voz, que suponen la utilización de los mismos canales para transmitir datos.

Valor agregado VASP: El valor agregado es el valor que se le adiciona al ya plasmado, en este caso los nuevos servicios que pueden brindar los dispositivos móviles a demás de los que ya poseen.

W3C: El Consorcio World Wide Web, es un consorcio internacional donde las organizaciones miembro, personal a tiempo completo y el público en general, trabajan conjuntamente para desarrollar estándares Web.

WALL: Es una librería de tags JSPs similares al HTML que permite desarrollar páginas para móviles, convirtiéndolas a sus correspondientes en lenguajes como WML, CHTML y XHTML de las peticiones realizadas desde los dispositivos móviles teniendo en cuenta sus características.

WAP Push: Es una tecnología de nueva generación que emplea SMS, el cual se envía al móvil por parte del operador generalmente conteniendo algún tipo de dirección en el cuerpo del mensaje mediante la cual el usuario puede acceder al contenido. Esta especificación permite enviar un mensaje a un dispositivo móvil sin previa autorización del usuario.

WAP: Es el protocolo de aplicaciones inalámbricas, es un estándar global y abierto. El WAP permite que dispositivos móviles, tales como los teléfonos o los PDAs, accedan a información o servicios especialmente creados en Internet.

W-CSS: Especifica un subconjunto de CSS y WAP extensiones específicas. Se puede utilizar para el estilo XHTML así como cualquier otro documento XML.

WML: El Wireless Markup Language es un lenguaje cuyo origen es el XML (eXtensible Markup Language). Este lenguaje se utiliza para construir las páginas que aparecen en las pantallas de los dispositivos móviles.

WMLS: El foro WAP proporciona un lenguaje de escritura llamado WMLScript, es usado junto a WML (o independiente de este) para proporcionar inteligencia en la forma lógica de proceder a los servicios del cliente. WMLScript define funciones de control de flujo de contenidos y estructuras lógicas.

World Wide Web: En informática World Wide Web (o la "Web") o Red Global Mundial es un sistema de documentos de hipertexto accesibles a través de Internet. Con un navegador Web, el usuario puede visualizar las páginas web y navegar a través de ellas mediante hiperenlaces.

WSP: Wireless Session Protocol, hace referencia a la aplicación de más alto nivel que ofrece WAP a través de un interface para servicios de dos sesiones, en la capa de sesión. La primera consistiría en un servicio con conexión que operaría sobre el protocolo del nivel de transacción y el segundo sería sin conexión y operaría sobre el servicio de transporte de información.

WTP: El Protocolo Inalámbrico de Transacción (WTP) hace referencia a la capa de transacciones de la arquitectura WAP.

WURFL: Es un tipo de base de datos en formato XML que contiene información acerca de las capacidades de la mayoría de los dispositivos móviles y varios navegadores Web que existen en la actualidad.

XHTML: Lenguaje eXtensible de Marcado de Hipertexto (eXtensible Hypertext Markup Language), es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web.

XML: Lenguaje extensible de marcas (Extensible Markup Language), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium.