

*Universidad de las Ciencias Informáticas*  
*Facultad 2*



*Título: Servicio de Información de Festivales de Cine para Celulares.*

*Trabajo de diploma para optar por el título de  
Ingeniero en ciencias informáticas.*

***Autores:***

*Lourdes Veliz Méndez*

*Yalina Rodríguez Rodríguez*

***Tutor: Ing. Yaneisy Cruz Navarro***

Declaramos que somos los únicos autores de este trabajo y autorizamos al <nombre área> de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_

---

Yalina Rodriguez Rodriguez  
Firma del Autor

---

Lourdes Veliz Méndez  
Firma del Autor

---

Yaneisy Cruz Navarro  
Firma del Tutor



*"La educación es el arma más poderosa que tiene el hombre para crear una ética, para crear una conciencia, para crear un sentido del deber, un sentido de la organización, de la disciplina, de la responsabilidad."*

*Fidel Castro Ruz*

*A mis padres Martha y Alfredo por siempre apoyarme en los momentos difíciles, por guiarme en los primeros pasos, por su amor incondicional, por exhortarme siempre a mi superación profesional y espiritual. Gracias por darme la vida los amo.*

*A mi tía Anita, sin su ayuda no hubiera podido realizar mis sueños; gracias por estar ahí para mí, la vida no me alcanza para agradecerte te quiero mucho.*

*A mi hija Pilian Annet por ser el motor impulsor de este trabajo, por su sacrificio de estar lejos de mi espero que valga la pena mi niña, eres mi vida.*

*A mi hermanita Ana Beatriz, por sus cariñitos, por hacerme querer ser mejor persona.*

*A toda mi familia por su cariño y ayuda incondicional.*

*A mis amigos lo de aquí y los de allá gracias por aguantar mi mal carácter y mis malacrianzas, gracias.*

*A mi novio Pasiel por su apoyo en los momentos difíciles, su cariño y su incondicionalidad*

*A mi compañera de tesis por entregarse en todo el desarrollo de este trabajo y aguantarme.*

*A todos los que de una forma u otra hicieron posible este sueño.*

*A ti y a mí.*

*Palina*

*A mis padres, a mis abuelos, mis tíos, en fin a mi familia, por su apoyo incondicional, por inculcarme el amor por el estudio, por ese consejo certero que muchas veces necesité, por hacerme una personita de bien, por guiarme por el camino de la sabiduría, la responsabilidad y la sencillez.*

*A mis amigos por compartir todos estos años, brindarme su mano en momentos difíciles y disfrutar junto a mí en los buenos.*

*A las sexy girls (Palina y Dainelvis) que aguantaron mis malacrianzas de vez en vez y por ser tan lindas e incondicionales conmigo, un besito y mucha salud para ustedes.*

*A Migue por toda su consagración en pos mío, de verdad Migue, gracias por estar ahí para mí.*

*A mi novio Carlos por aguantar mis pesadeces y mis cambios de humor y por ser tan especial, te quiero.*

*A la profe Fahima por confiar en mí. y brindarme un brazo amigo.*

*A todos los que de una u otra forma contribuyeron a que pudiera conquistar mis sueños, de corazón muchísimas gracias. Los quiero a todos.*

*Lourdes*

*A nuestra tutora Faneisy Cruz Navarro por guiarnos durante el desarrollo de este trabajo.*

*A Migue, sin tu ayuda y criticas no hubiéramos podido llegar a la meta.*

*A Iván por todo el empeño e interés que puso en ayudarnos*

*Dedico este trabajo a dos personas muy especiales:*

*Mi abuelita Obelsys que ya no se encuentra entre nosotros, por guiar mis primeros pasos.*

*A mi bebe Fiyi que es mi razón de ser, por estar lejos de mi y aún así darme las fuerzas para seguir adelante.*

*Palina*

*Este trabajo va dedicado a toda mi familia, pero en especial para mis padres Olga y Eduardo, a mi abuelita Lourdes (mima), por ser los primeros en confiar en esa florecita que necesitaba de su amor, cariño y comprensión para crecer y germinar en una Hermosa flor. A todos ustedes por estar tan lejos y estar tan cerca a la vez.*

*Lourdes*

La empresa ETECSA brinda a sus clientes con celulares el servicio de mensajería corta (SMS) el cual les permite comunicarse sin necesidad de realizar una llamada o de que esta comunicación sea necesariamente con otro teléfono. Actualmente existe el servicio para el envío y recepción de SMS con información referente a los festivales de cine, sin embargo este servicio desaprovecha algunas opciones que podían ser de interés para el usuario como lo es el género, el director y los protagonistas de las películas.

En aras de aumentar y mejorar las opciones de este servicio, se propone la creación de una aplicación que incluya estas opciones. En este trabajo se presenta la solución de una aplicación que brinda funcionalidades para la informatización de los procesos que tiene lugar en el envío y recepción de SMS con información referente a los festivales de cine.

En el documento se reflejan los resultados del trabajo realizado, se explican un conjunto de conceptos y se muestran los artefactos generados en cada etapa de construcción del software.

---

**Índice**

Introducción.....	1
Capítulo1. Fundamentación Teórica.....	3
Introducción.....	3
1.1 Servicios de mensajería para dispositivos móviles.....	3
1.1.1 Servicios de mensajería para dispositivos móviles a través de SMS.....	4
1.2 Estado del Arte.....	5
1.2.1 Servicios de Información a través de SMS en el mundo.....	5
1.2.2 Servicios de Información a través de SMS en Cuba.....	6
1.3 Análisis de metodologías y herramientas.....	7
1.3.1 Sistema gestor de Base de Datos.....	7
1.3.2 Lenguaje Unificado de Modelado (UML).....	8
1.3.3 Metodologías de Desarrollo.....	8
1.3.4 Visual Paradigm.....	9
1.3.5 Java.....	10
1.3.6 Apache Tomcat como servidor web.....	11
1.3.7 Entorno de desarrollo.....	11
1.3.8 Framework.....	11
1.3.9 Desarrollo de la interfaz de Usuario.....	13
1.3.10 Metodología y Herramientas seleccionadas.....	14
Conclusiones.....	14
Capítulo2. Características del sistema.....	15
Introducción.....	15
2.1 Objeto de automatización.....	15
2.2 Propuesta del sistema.....	15
2.3 Modelo de dominio.....	17
2.3.1 Descripción de los objetos del dominio:.....	18
2.4 Relación de los Requerimientos (Funcionales, No Funcionales y Adicionales del Sistema).....	18
2.4.1 Requerimientos Funcionales:.....	19
2.4.2 Requerimientos no Funcionales.....	20
2.5 Modelo de Casos de Uso del Sistema.....	21
2.5.1 Definición de los actores del sistema a automatizar:.....	21
2.5.2 Listado de Casos de Uso del sistema (CUS).....	21
2.5.3 Diagramas de Casos de Uso del sistema.....	22
2.5.4 Descripción de los Casos de Uso del sistema.....	22

---

Conclusiones.....	27
Capítulo3. Análisis y Diseño del sistema.....	28
Introducción.....	28
3.1 Diseño.....	28
3.1.1 Patrones utilizados.....	28
3.1.1.1 Patrones de Arquitectura:.....	28
3.1.1.2 Patrones de Diseño:.....	30
3.3 Diagramas de Clases del Diseño.....	33
3.4 Diagramas de Interacción.....	34
3.5 Diseño de la Base de Datos.....	35
3.5.1 Modelo Lógico de Datos (diagrama de clases persistentes).....	36
3.5.2 Modelo Físico de Datos (Modelo de Datos).....	37
Conclusiones.....	37
Capítulo 4. Implementación.....	38
Introducción.....	38
4.1 Diagrama de Despliegue.....	38
4.2 Diagramas de Componentes.....	39
4.2.2 Descripción de los paquetes del Diagrama de componentes Modulo de Servicios.....	39
4.2.2 Descripción de los paquetes del Diagrama de componentes Módulo de Administración.....	40
Conclusiones.....	41
Capítulo 5. Estudio de Factibilidad.....	42
Introducción:.....	42
5.1 Método a utilizar: Método de Estimación por Puntos de Casos de Uso.....	42
5.2 Calcular el costo de todo el proyecto.....	46
5.3 Calcular el tiempo de desarrollo de todo el proyecto.....	46
5.4 Beneficios Tangibles e Intangibles.....	46
5.5 Análisis de costos y beneficios.....	47
Conclusiones.....	47
Conclusiones Generales.....	48
Recomendaciones.....	49
Referencias Bibliográficas.....	50
Bibliografía.....	51
Glosario de términos.....	52
Anexos.....	<b>¡Error! Marcador no definido.</b>
Anexo I. Descripción textual de los Casos de Uso del Negocio en formato expandido.....	<b>¡Error! Marcador no definido.</b>

Anexo II Diagramas de Clases del Diseño.....	¡Error! Marcador no definido.
Anexo III Diagrama de Interacción.....	¡Error! Marcador no definido.

## Índice de las Tablas

Tabla 1 Definición de Actores.....	21
Tabla 2 Descripción del CU Autenticar.....	23
Tabla 3 Descripción del CU Gestionar Película.....	23
Tabla 4 Descripción del CU Buscar Película.....	23
Tabla 5 Descripción del CU Gestionar Presentación.....	24
Tabla 6 Descripción del CU Buscar Presentación.....	24
Tabla 7 Descripción del CU Gestionar Institución Cinematográfica.....	24
Tabla 8 Descripción del CU Buscar Institución Cinematográfica.....	24
Tabla 9 Descripción del CU Gestionar Noticia.....	25
Tabla 10 Descripción del CU Buscar Noticia.....	25
Tabla 11 Descripción del CU Insertar Opinión.....	25
Tabla 12 Descripción del CU Recibir Noticias del Festival.....	26
Tabla 13 Descripción del CU Recibir Cartelera.....	26
Tabla 14 Descripción del CU Recibir Ayuda.....	26
Tabla 15 Descripción del CU Cambiar Contraseña.....	26
Tabla 16 Factor de Peso de los Actores sin Ajustar (UAW).....	42
Tabla 17 Factor de Peso de los CU sin Ajustar (UUCW).....	43
Tabla 18 Factor de Complejidad Técnica.....	44
Tabla 19 Factor de Ambiente.....	44
Tabla 20 Distribución del Esfuerzo entre las actividades del proyecto.....	45

## Índice de las Figuras

Figura 1 Propuesta del sistema (Módulo de Servicios).....	16
Figura 2 Propuesta del Sistema (Módulo de Administración).....	17
Figura 3 Modelo de Dominio.....	18
Figura 4 Diagrama de Caso de Uso del Sistema Módulo de Gestión de Información.....	22
Figura 5 Diagrama de Casos de Uso del Sistema Módulo de Servicios del Sistema.....	22
Figura 6 Arquitectura del módulo de servicios.....	30
Figura 7 Diagrama de Clases del Diseño (Ejemplo para explicar el uso de los patrones).....	33
Figura 8 Diagrama de Diseño del CU Insertar Opinión.....	34
Figura 9 Diagrama de Diseño del CU Recibir Cartelera.....	34
Figura 10 Diagrama de Diseño del CU Recibir Noticias.....	34
Figura 11 Diagrama de Diseño del CU Recibir Ayuda.....	34
Figura 12 Diagrama del Servicio Listar Noticias (Show List News).....	35
Figura 13 Diagrama de Clases Persistentes.....	36
Figura 14 Modelo de Datos.....	37
Figura 15 Diagrama de Despliegue.....	38
Figura 16 Diagrama de Componentes.....	39
Figura 17 Diagrama de Componentes Módulo de Servicios.....	40
Figura 18 Diagrama de Componentes Módulo Administración.....	41
Figura 19 Diagrama de Diseño del CU Recibir Ayuda.....	¡Error! Marcador no definido.

---

Figura 20 Diagrama de Diseño del CU Gestionar Institución Cinematográfica ¡Error! Marcador no definido.

Figura 21 Diagrama de Diseño del Paquete Institution Services ..... ¡Error! Marcador no definido.

Figura 22 Diagrama de Diseño del CU Gestionar Película ..... ¡Error! Marcador no definido.

Figura 23 Diagrama de Diseño del Paquete Film Services ..... ¡Error! Marcador no definido.

Figura 24 Diagrama de Diseño del CU Gestionar Presentación ..... ¡Error! Marcador no definido.

Figura 25 Diagrama de Diseño del Paquete Presentation Services ..... ¡Error! Marcador no definido.

Figura 26 Diagrama de Diseño del CU Gestionar Noticias ..... ¡Error! Marcador no definido.

Figura 27 Diagrama de Diseño del Paquete News Services ..... ¡Error! Marcador no definido.

Figura 28 Diagrama de Diseño del CU Buscar Película ..... ¡Error! Marcador no definido.

Figura 29 Diagrama de Diseño del CU Buscar Institución Cinematográfica ¡Error! Marcador no definido.

Figura 30 Diagrama de Diseño del CU Buscar Presentación ..... ¡Error! Marcador no definido.

Figura 31 Diagrama de Diseño del CU Buscar Presentación ..... ¡Error! Marcador no definido.

Figura 32 Diagrama de Diseño del CU Autenticar ..... ¡Error! Marcador no definido.

Figura 33 Diagrama de Diseño del CU Cambiar Contraseña ..... ¡Error! Marcador no definido.

Figura 34 Diagrama de Diseño del Paquete Usser Services ..... ¡Error! Marcador no definido.

Figura 35 ..... ¡Error! Marcador no definido.

Figura 36 Diagrama del Servicio Listar Presentación (Show List Presentation). ¡Error! Marcador no definido.

Figura 37 Diagrama del Servicio Listar Filmes (Show List Film) ..... ¡Error! Marcador no definido.

Figura 38 Diagrama del Servicio Listar Institución Cinematográfica (Show List Institution Cinematográfica) ..... ¡Error! Marcador no definido.

Figura 39 Diagrama del Servicio Modificar Presentación (Update-Presentation) ¡Error! Marcador no definido.

Figura 40 Diagrama de Secuencia de la Sección Modificar Presentación del CU Gestionar Presentación ..... ¡Error! Marcador no definido.

Figura 41 Diagrama de Secuencia del Servicio Modificar Película (Update Film). ¡Error! Marcador no definido.

Figura 42 Diagrama de Secuencia del Servicio Modificar Noticia (Update News). ¡Error! Marcador no definido.

Figura 43 Diagrama de Secuencia de la Sección Modificar Noticias del CU Gestionar Noticias ... ¡Error! Marcador no definido.

Figura 44 Diagrama de Secuencia del Servicio Modificar Institución Cinematográfica (Update Institution Cinematográfica) ..... ¡Error! Marcador no definido.

Figura 45 Diagrama de Secuencia de la Sección Modificar Institución Cinematográfica del CU Gestionar Institución Cinematográfica ..... ¡Error! Marcador no definido.

Figura 46 Diagrama de Secuencia del Servicio Insertar Presentación (Insert Presentation). ..... ¡Error! Marcador no definido.

Figura 47 Diagrama de Secuencia de la Sección Insertar Presentación del CU Gestionar Presentación. .... ¡Error! Marcador no definido.

Figura 48 Diagrama de Secuencia del Servicio Insertar Película (Insert Film). ¡Error! Marcador no definido.

Figura 49 Diagrama de Secuencia de la Sección Insertar Película del CU Gestionar Película ..... ¡Error! Marcador no definido.

Figura 50 Diagrama de Secuencia del Servicio Insertar Noticia (Insert News). ¡Error! Marcador no definido.

---

Figura 51 Diagrama de Secuencia del Servicio Insertar Institución Cinematográfica (Insert Institution Cinematográfica)..... **¡Error! Marcador no definido.**

Figura 52 Diagrama de Secuencia de la Sección Insertar Institución Cinematográfica del CU Gestionar Institución Cinematográfica..... **¡Error! Marcador no definido.**

Figura 53 Diagrama de Secuencia del Servicio Eliminar Presentación (Delete Presentation). ..... **¡Error! Marcador no definido.**

Figura 54 Diagrama de Secuencia de la Sección Eliminar Presentación del CU Gestionar Presentación ..... **¡Error! Marcador no definido.**

Figura 55 Diagrama de Secuencia del Servicio Eliminar Película (Delete Film).**¡Error! Marcador no definido.**

Figura 56 Diagrama de Secuencia de la Sección Eliminar Película del CU Gestionar Película ..... **¡Error! Marcador no definido.**

Figura 57 Diagrama de Secuencia de la Sección Eliminar Noticia del CU Gestionar Noticia ..... **¡Error! Marcador no definido.**

Figura 58 Diagrama de Secuencia del Servicio Eliminar Noticia (Delete News).**¡Error! Marcador no definido.**

Figura 59 Diagrama de Secuencia de la Sección Eliminar Institución Cinematográfica del CU Institución Cinematográfica..... **¡Error! Marcador no definido.**

Figura 60 Diagrama de Secuencia del Servicio Eliminar Institución Cinematográfica (Delete Institution Cinematográfica()). ..... **¡Error! Marcador no definido.**

Figura 61 Diagrama de Secuencia del Servicio Cambiar contraseña (Change Password) ..... **¡Error! Marcador no definido.**

Figura 62 Diagrama de Secuencia del Servicio Buscar Presentación (Search Presentation)..... **¡Error! Marcador no definido.**

Figura 63 Diagrama de Secuencia CU Buscar Presentación ..... **¡Error! Marcador no definido.**

Figura 64 Diagrama de Secuencia del Servicio Buscar Película (Search Film).**¡Error! Marcador no definido.**

Figura 65 Diagrama de Secuencia del CU Buscar Película..... **¡Error! Marcador no definido.**

Figura 66 Diagrama de Secuencia del Servicio Buscar Noticias (Search News).**¡Error! Marcador no definido.**

Figura 67 Diagrama de Secuencia del CU Buscar Noticia ..... **¡Error! Marcador no definido.**

Figura 68 Diagrama de Secuencia del Servicio Buscar Institución Cinematográfica (Search Institution Cinematográfica)..... **¡Error! Marcador no definido.**

Figura 69 Diagrama del CU Buscar Institución Cinematográfica ..... **¡Error! Marcador no definido.**

Figura 70 Diagrama de Secuencia del Servicio Autenticar (Authenticate). **¡Error! Marcador no definido.**

Figura 71 Diagrama de Secuencia del CU Autenticar ..... **¡Error! Marcador no definido.**

Figura 72 Diagrama de Secuencia del CU Recibir Ayuda ..... **¡Error! Marcador no definido.**

Figura 73 Diagrama de Secuencia del CU Buscar Noticia ..... **¡Error! Marcador no definido.**

Figura 74 Diagrama de Secuencia del CU Buscar Presentación..... **¡Error! Marcador no definido.**

Figura 75 Diagrama de Secuencia de la Sección Insertar Opinión del CU Gestionar Opinión ..... **¡Error! Marcador no definido.**

Figura 76 Diagrama de Secuencia de la Sección Modificar Película del CU Gestionar Película ... **¡Error! Marcador no definido.**

Figura 77 Diagrama de Secuencia de la Sección Insertar Noticias del CU Gestionar Noticias..... **¡Error! Marcador no definido.**

Figura 78 Diagrama de Secuencia del CU Cambiar contraseña ..... **¡Error! Marcador no definido.**



### Introducción.

El gran avance de las tecnologías ha permitido que los celulares tengan funciones que hasta hace poco parecían imposibles como son: juegos, reproducción de música y videos, envío de correo electrónico y el servicio de mensajería corta (SMS). Cada vez es más frecuente el uso de la telefonía celular en Cuba gracias al gran avance que ha dado nuestro país en las telecomunicaciones. A pesar de esto la empresa ETECSA no explota a profundidad las posibilidades que brindan los celulares.

El festival de cine es un evento de importancia en nuestro país, por lo que es una prioridad mantener informados a los usuarios de lo que acontece en el momento que se desarrolla el mismo. Una vía para que el público conozca rápidamente las noticias referentes al festival es el envío de mensajes SMS, actualmente ETECSA cuenta con el servicio de mensajería para informar de los Festivales de Cine pero dentro de la información que brindan no tienen en cuenta información que en ocasiones es de interés para los usuarios. Muchas veces se ve una película sin darle importancia al título porque gusta el director, los protagonistas o el género, es por esto que dentro de las opciones que brinda este servicio se debería hacer referencia a esta información para que este servicio sea más completo y atraiga un mayor número de usuarios.

Partiendo de la situación problemática anterior se tiene como **problema científico**: ¿Cómo diseñar una aplicación que utilizando el servicio de SMS inserte nuevas funcionalidades al servicio de mensajería para los Festivales de Cine que estén aconteciendo en el momento?

Para ello la investigación centra **su objeto de estudio** en el servicio de mensajería para dispositivos móviles y **el campo de acción** en el servicio de mensajería para dispositivos móviles a través de SMS.

Se propone como **objetivo general**: Desarrollar un sistema informático que permita a los usuarios acceder a la información referente al Festival de Cine que se esté efectuando en ese momento, desde su celular.

A partir del análisis del objetivo general se derivaron los siguientes **objetivos específicos**:

- Analizar la información relacionada con los Festivales de Cine.
- Gestionar la información referente a los Festivales de Cine.

Para el cumplimiento de los objetivos propuestos se realizaron una serie de **tareas**, dentro de las cuales se encuentran:

- Definir qué métodos teóricos se van a emplear en el Diseño metodológico de la investigación.
- Valorar que metodología de desarrollo de software a utilizar, plataforma, lenguaje y el conjunto de herramientas de desarrollo definidas para la elaboración de la aplicación.
- Identificar las funcionalidades que debe poseer el sistema a través del levantamiento de requisitos.

- Realizar el diseño del software teniendo en cuenta las funcionalidades antes descritas.
- Identificar y seleccionar los patrones de arquitectura y diseño más apropiados para la elaboración del producto de software.
- Definir la organización del código e implementar los elementos de diseño en términos de componentes de implementación.

Como **posible resultado**: los usuarios con celulares podrán disponer de un sistema de comunicación rápida para el conocimiento de la información referente al Festival de Cine que tenga lugar en ese momento.

Como **métodos científicos** vamos a utilizar los **métodos teóricos**:

- Como **métodos teóricos** el **analítico – sintético**, logrando con este una mejor comprensión del trabajo, dividiéndolo en partes para su solución e integrándolas para lograr el resultado final. Otro método seleccionado fue el **histórico – lógico** ya que partiendo de la observación y análisis de los antecedentes del objeto de estudio se dio inicio a esta investigación.

El trabajo está **estructurado** en 5 capítulos además de las correspondientes conclusiones, recomendaciones, referencia bibliográfica, bibliografía, anexos y glosario de términos:

- **Capítulo 1- Fundamentación teórica:**

Estudio del estado del arte de los temas a tratar, permitiendo comprender los aspectos teóricos y asuntos de interés más relevantes para crear la base de lo que se abordará en los capítulos siguientes.

- **Capítulo 2 - Características del sistema:**

Se hace una descripción de cómo se realiza el proceso objeto de estudio de nuestra investigación. Se exponen los artefactos que permiten comprender la estructura y la dinámica del proceso abordado según la metodología utilizada. Se presentan las funcionalidades que tendrá el sistema y se definen los casos de uso.

- **Capítulo 3 - Análisis y diseño del sistema :**

Se exponen los diagramas de clases del diseño, los diagramas de secuencias, el diagrama Entidad Relación de la BD, la descripción de las tablas y se explican de manera general las clases utilizadas.

- **Capítulo 4 – Implementación:**

Se describe el sistema que se propone a través de una óptica de programación, basada en diagramas de componentes y diagrama de despliegue.

- **Capítulo 5 - Estudio de Factibilidad:**

Se analiza el método que se va a utilizar para realizar la estimación de esfuerzos y de este modo calcular el esfuerzo de todo el proyecto dado en meses-hombres.

## **Capítulo 1. Fundamentación Teórica.**

### **Introducción**

En el presente capítulo se hará referencia a los aspectos teóricos necesarios que permitan aclarar, especificar todo lo relacionado al problema a resolver. Se realizará un análisis de las herramientas a utilizar en el desarrollo de la solución, sus ventajas, desventajas y el por qué de la utilización de las mismas, así como referencia a otras herramientas que se pudieron haber empleado. Se abordarán otras aplicaciones con características similares o iguales implementadas y usadas a nivel nacional e internacional, desde un punto de vista crítico. Se hará alusión, mediante una breve descripción, a los conceptos más importantes vinculados al dominio del problema.

### **1.1 Servicios de mensajería para dispositivos móviles.**

La telefonía móvil en sus inicios fue creada estrictamente para la voz, hoy es capaz de ofrecer otros tipos de servicios tales como datos, audio, video y servicios de mensajería. Existen diversos tipos de servicios de mensajería para dispositivos móviles entre ellos se destacan:

- El Servicio de Mensajería Corta (Short Message System o SMS) el cual es un servicio de telefonía móvil universalmente aceptado, que posibilita la transmisión de mensajes cortos (SM) alfanuméricos entre usuarios móviles y sistemas externos como correo electrónico, paging y sistemas de correos de voz.
- El Servicio de Mensajes Multimedia (Multimedia Message Service o MMS), que es una tecnología de comunicaciones que permite a los usuarios intercambiar mensajes multimedia entre teléfonos móviles y otros dispositivos. MMS es una extensión de SMS, que define una manera de enviar y recibir, casi instantáneamente, mensajes que incluyen imágenes, audio y videos clip además de texto.
- El servicio de notificaciones WAP (WAP Push). Estas típicamente hacen referencia a un MMS, a un correo electrónico o a una descarga de contenidos como imágenes, juegos, entre otros. Una notificación Wap puede estar formada por un mensaje de texto y una URL, o solamente por un URL, donde el teléfono móvil receptor del mensaje muestra el texto, en caso de que exista, y provee al usuario la opción de conectarse al URL con solo presionar una tecla.

### 1.1.1 Servicios de mensajería para dispositivos móviles a través de SMS.

**SMS:** El Servicio de Mensajes Cortos (SMS) posibilita el envío y recepción de pequeños mensajes de texto que no superen los 160 Bytes a través de la red inalámbrica. Estos tienen como principal característica la descripción de una información de una manera corta y clara. Aunque han evolucionado de manera tal que hace posible en algunos casos, enviar casi cartas. Por la intención de su contenido se pueden clasificar en cuatro clases fundamentales, lo que no quita que se puedan mezclar una con otra, todo depende de la intencionalidad del emisor del mensaje:

- **Personales:** En ellos se incluyen los mensajes de saludo, felicitaciones, deseos, etc.; en fin se trata de una comunicación que puede ser más o menos íntima, y en general con todas las características de las manifestaciones de cariño amistosas. Igualmente, también se pueden incluir los recordatorios de citas
- **Promocionales:** Son aquellos que llegan a los teléfonos móviles divulgando algún servicio en especial, o invitando a las personas a que marquen a algún código numerado con el fin enviar una palabra y tener la posibilidad de acceder a alguna rifa o concurso. Algunas empresas dedicadas a estudiar las tendencias de este servicio concluyen que este tipo de mensajes son los más comunes en el sector tecnológico.
- **Institucionales:** Aquellos que son generados por la compañía de telefonía móvil a la cual está suscrita la persona usuaria del celular.
- **Los amenazantes:** Estos sólo sirven para extorsionar y fomentar la violencia y la injusticia desde cualquier ángulo de la sociedad [1].

#### Aplicaciones para SMS:

Los SMS fueron inicialmente diseñados para soportar mensajes de tamaño limitado, en la mayoría de los casos notificaciones o páginas alfanuméricas, pero se están descubriendo nuevos usos, que han hecho que este mercado tenga mayor auge.

- **Servicios de notificación:** Los servicios de notificación son unos servicios SMS ampliamente utilizados. Ejemplos de servicios de notificación usando SMS son los mensajes de notificación de correo de voz, notificación de correo electrónico, recordatorio de citas, horarios de reuniones, etc.

- **Servicios de localización:** La habilidad de rastrear la localización de un objeto móvil, o de un usuario, es muy valiosa tanto para los proveedores como para los clientes. Esta aplicación, de nuevo, solo necesita un intercambio de pequeñas cantidades de información, tales como la longitud y latitud en un momento preciso del día, y quizás otros parámetros como velocidad, temperatura o humedad.
- **Servicios de información:** Se puede proporcionar una amplia variedad de servicios de información, incluyendo partes meteorológicos, información del tráfico, información de entretenimiento (cines, teatros, conciertos), información financiera (cotizaciones de bolsa, servicios bancarios, servicios de corretaje, etc.), y directorios.

## 1.2 Estado del Arte

### 1.2.1 Servicios de Información a través de SMS en el mundo.

En todo el mundo los SMS son utilizados para ofrecerle información a los usuarios ya sea cultural, deportivas o de otra índole. Actualmente existen diferentes empresas líderes en el marco de la telefonía móvil, mediante las cuales se han podido extender los servicios móviles a diversas partes del mundo, todas ellas ampliamente reconocidas por sus servicios, entre algunas de las más prestigiosas se encuentran:

**Movistar:** Representa a la compañía líder de telecomunicaciones de habla hispana más grande del mundo con más de 100 millones de clientes y presente en 15 países. Presenta un servicio de envío de titulares, por ejemplo, titulares de las últimas noticias, del canal de deportes, del canal informativo de Economía, del canal informativo Cultura y espectáculos y de todos los canales informativos. Movistar e-mocion es el servicio multiacceso que Telefónica Movistar pone al alcance de todos sus clientes para ofrecerles de forma sencilla todo tipo de información, entretenimiento y contenidos a través del teléfono móvil, para buscar el cine que les interesa, la cartelera y horarios de las películas [3].

**Orange:** Forma parte del Grupo France Telecom (el quinto mercado europeo de telecomunicaciones), ofrece servicios de comunicación para grandes clientes, multinacionales y Administraciones Públicas. Su objetivo es contribuir al desarrollo de la Sociedad de la Información desarrollando una infraestructura para la creación de una oferta de nuevos servicios y aplicaciones

para particulares y empresas. Este al igual que Movistar ofrece servicios de información sobre noticias, economía y deportes, además de tiempo, horóscopos, revistas, alertas en Latinoamérica, cocina, y cartelera cinematográfica [4].

**Motorola:** Es una empresa estadounidense especializada en la electrónica y las telecomunicaciones, establecida en Schaumburg, Illinois, Chicago. Los productos más conocidos de la compañía incluyen microprocesadores, teléfonos móviles y sistemas de conexión a redes de telefonía móvil e inalámbrica como Wimax y Wifi. Motorola ofrece variados servicios como un servicio de noticias tanto nacionales como internacionales, así como de un servicio información de la cartelera de cine [5].

### **1.2.2 Servicios de Información a través de SMS en Cuba.**

En la actualidad Cuba se está abriendo paso en el mundo de la telefonía celular. En el 2009 como resultado del trabajo de la división de Telecomunicaciones de Desoft, una empresa perteneciente al Ministerio de la Informática y las Comunicaciones de Cuba surgen nuevos servicios de información entre estos se pueden mencionar:

1. El servicio que permite conocer el estado de los partidos de beisbol de la Serie Nacional. Al enviar un SMS al número 8888 con el texto PELOTA RESULTADOS. Para conocer el resultado de un juego en particular se debe enviar un mensaje con la palabra PELOTA y las iniciales de un equipo participante.
2. El servicio para las votaciones de los premios Lucas.  
Breve descripción del servicio:  
Por medio de este servicio se puede votar por los nominados al video más popular del año de los premios Lucas 2009. Para votar por un nominado se debe enviar un sms al número 8888 con la palabra clave LUCAS y con el número de la canción nominada.
3. El servicio que brinda información referente a Festivales de Cine el cual permite a los usuarios conocer las ultimas noticias, la cartelera y opinar del desarrollo del Festival a través de SMS. Dichos servicios utilizan una plataforma tecnológica llamada "Blue-Eye", desarrollada al 100 por ciento por expertos de la empresa cubana de desarrollo de software Desoft.

Esta aplicación funciona de la siguiente forma:

- Todos los servicios se activan al enviar un mensaje de texto al número corto 8888 con una sintaxis predeterminada.

Convenciones de notación:

- Las palabras o frases encerradas entre comillas (Ej. "palabra" o "varias palabras") significan que el usuario tiene que escribir la palabra o frase literalmente en esta parte del cuerpo del SMS.
- Las palabras o frases encerradas entre llaves angulares <> (Ej. <palabra> o <varias palabras>) significan que el usuario puede escribir una palabra o frase variable en esta parte del cuerpo del SMS.
- Cada elemento (palabra o frase, variable o literal) del cuerpo del mensaje debe estar separado por un espacio.

### 1.3 Análisis de metodologías y herramientas.

#### 1.3.1 Sistema gestor de Base de Datos.

**SGBD:** Un Sistema Gestor o Manejador de Bases de Datos (SGBD) es un conjunto de programas que permite a los usuarios crear y mantener una BD. Es un software de propósito general que facilita el proceso de definir, construir y manipular la BD para diversas aplicaciones. Se compone de un lenguaje de definición de datos, un lenguaje de manipulación de datos y de un lenguaje de consulta [6].

#### **PostgreSQL**

Es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia Distribución de Software Berkeley (BSD) y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado. Utiliza un modelo cliente/servidor y usa multiprocesos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. La estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido las características que más se han tenido en cuenta durante su desarrollo. PostgreSQL funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema [7].

#### **Oracle**

Oracle es un sistema de bases de datos relacional que se destaca por ofrecer soporte de transacciones, estabilidad, escalabilidad, además de ser multiplataforma. Por estas características se le considera como uno de los más completos. Las últimas versiones han sido certificadas para poder

trabajar sobre plataforma Linux (Expertos, 2009). Su principal defecto es su enorme precio, que es de varios miles de euros.

### **1.3.2 Lenguaje Unificado de Modelado (UML).**

Es un lenguaje de modelado que permite visualizar, especificar y documentar cada una de las partes que comprende el desarrollo del software. Permite modelar elementos conceptuales como son los procesos de negocio, las funciones de sistema, clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables, así como distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. Dentro de sus objetivos tiene: brindar un material de apoyo que le permita al lector poder definir diagramas propios como también entender diagramas ya existentes. Prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que significan estos diagramas y símbolos [8].

### **1.3.3 Metodologías de Desarrollo**

#### **BPM:**

Es el entendimiento, gestión e innovación de procesos bajo estándares internacionales, alineados con la estrategia de negocio para asegurar la efectividad del proceso y crear valor a la cadena productiva de la empresa y su sector. Constituye un nuevo paradigma para abordar procesos de mejoramiento que aumenta la eficiencia. Se lleva a la práctica integrando la estrategia, los procesos y la tecnología, la cual emplea estándares de modelado para permitir una comunicación fluida y con menor esfuerzo entre procesos de negocio y las compañías del sector. Constituye una alternativa para la búsqueda de ventajas competitivas, para afrontar escenarios de competencia en mercados nacionales e internacionales. Se perfila como una nueva línea de pensamiento que atiende necesidades tangibles de las empresas y abre nuevos nichos de mercado para nuevas empresas que se enfoquen en temas de gestión empresarial.

#### **Proceso Unificado de desarrollo de Software (RUP)**

Es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo, a través del lenguaje UML y el trabajo de muchas metodologías utilizadas por los clientes. La versión que se ha estandarizado se creó en 1998 y se conoció en sus inicios como Proceso

Unificado de Rational 5.0, de ahí las siglas con las que se identifica a este proceso de desarrollo. El mismo está conformado por cuatro fases: inicio, elaboración, construcción y transición. Además también propone 9 flujos de trabajo los cuales están comprendidos en dos grupos 6 de ingeniería: modelación del negocio, requerimientos, análisis y diseño, implementación, prueba e instalación y 3 de soporte: administración del proyecto, administración de configuración y cambios y por último ambiente [9].

### Características de RUP:

1. **Dirigido por casos de uso:** Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí los casos de uso guían el proceso de desarrollo debido a que los modelos que se obtienen, como resultado de los diferentes flujos de trabajo, representan la realización de los casos de uso.
2. **Centrado en la arquitectura:** La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción. RUP se desarrolla mediante iteraciones, comenzando por los casos de uso relevantes desde el punto de vista de la arquitectura.
3. **Iterativo e Incremental:** RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros.

RUP divide el proceso de desarrollo en ciclos, teniendo un producto funcional al final de cada ciclo, cada ciclo se divide en fases que finalizan con un hito donde se debe tomar una decisión importante. Es fácilmente adaptable al ambiente de desarrollo de software deseado, por lo que puede configurarse como una metodología ágil.

### 1.3.4 Visual Paradigm.

Es una herramienta de Ingeniería asistida por Computadora (CASE) que utiliza UML como lenguaje de modelado. Es básicamente una herramienta diseñada para cumplir las mismas funcionalidades que el

Rational Rose, con la diferencia de que es compatible con sistemas Linux. Algunas de las principales ventajas que presenta son:

- Presenta un diseño centrado en casos de uso y enfocado al negocio.
- Usa un lenguaje estándar común a todo el equipo de desarrollo y facilita la comunicación.
- Tiene capacidades de ingeniería directa (versión profesional) e inversa.
- Tiene modelos y códigos que permanecen sincronizados en todo el ciclo de desarrollo.
- Presenta disponibilidad de múltiples versiones, para cada necesidad y múltiples plataformas [10].

### 1.3.5 Java

Es un lenguaje de programación por objetos creado por Sun Microsystems, Inc. que permite crear programas que funcionan en cualquier tipo de ordenador y sistema operativo. Incorpora dinámicamente objetos a una aplicación activa, elimina muchos de los problemas derivados de actualizar software, ya que se puede disponer de un servidor central de aplicaciones, del que se recogen módulos a medida que se necesitan.

#### Java Web

Arranca aplicaciones Java que están en un servidor web mediante enlaces en una página web o bien a través de enlaces en el escritorio del cliente. Comprueba previamente si el cliente tiene la versión de dicha aplicación actualizada y se asegura de que ésta se distribuye siempre en su última versión, descargándola y ejecutándola.

#### Maquina Virtual de Java (JVM)

Es un emulador de la ejecución de un código nativo. La gran ventaja de la máquina virtual java es aportar portabilidad al lenguaje. La arquitectura de la JVM se basa en el concepto de una implementación que no es específica de una máquina. Esto es, la arquitectura misma no asume nada acerca de la máquina o de las características físicas y de construcción sobre la cual es implementada. De esta manera, la JVM es una entidad autónoma y única que ejecuta los archivos de clases. La JVM se separa en cinco unidades de funcionalidad distintas, las que se dedican a la tarea de ejecutar los archivos de clases. Las tareas principales de la JVM son las siguientes:

- Reservar espacio en memoria para los objetos creados.
- Liberar la memoria no usada (garbage collector).
- Asignar variables a registros y pilas.

- Llamar al sistema huésped para ciertas funciones, como los accesos a los dispositivos.
- Vigilar el cumplimiento de las normas de seguridad de las aplicaciones Java [11].

### **JEE**

Java Platform, Enterprise Edition o Java EE (anteriormente conocido como Java 2 Platform, Enterprise Edition o J2EE hasta la versión 1.4), es una plataforma de programación parte de la Plataforma Java para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java con arquitectura de N niveles distribuida, basándose ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones.

#### **1.3.6 Apache Tomcat como servidor web.**

Tomcat es un servidor web con soporte de servlets y JSPs. Puede funcionar como servidor web por sí mismo. Apache Tomcat y las tecnologías afines proporcionan a los programadores de Java un completo conjunto de herramientas para crear de forma rápida sofisticadas aplicaciones web. Tomcat es mantenido y desarrollado por miembros de la Apache Software Foundation y voluntarios independientes. Es usado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad. Dado que Tomcat fue escrito en Java, funciona en cualquier sistema operativo que disponga de la máquina virtual Java [12].

#### **1.3.7 Entorno de desarrollo**

##### **Eclipse**

Eclipse es un entorno de desarrollo multiplataforma, integrado de código abierto. La arquitectura basada en plug-ins que posee le permite integrar diversos lenguajes sobre un mismo IDE e introducir otras aplicaciones accesorias. Es soportado por los principales sistemas operativos. Al igual que en cualquier otro software, Eclipse permite la instalación de plug-ins destinados a mejorar las funcionalidades del propio IDE y a extenderse en cada vez más tecnologías. Tiene un muy buen soporte de refactorización así como ayuda a la codificación [13].

#### **1.3.8 Framework**

Estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. Puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre

aplicaciones para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

### **Spring**

Es un framework de código abierto muy potente en cuanto a la gestión del ciclo de vida de los componentes y fácilmente ampliable. Su objetivo central es permitir que objetos de negocio y de acceso a datos sean reutilizables. Este framework simplifica el manejo de errores, reduce considerablemente la cantidad de código necesario e incluye amplias capacidades de integración con otros frameworks como Hibernate, JDO e iBatis SQL Maps en términos de soporte a implementaciones DAO. Ofrece a los desarrolladores un diseño y arquitectura sólidos y confiables, así como una configuración de aplicaciones rápida y sencilla. Presenta una potente gestión de configuración basada en JavaBeans, aplicando los principios de Inversión de Control (IoC). El uso de un contenedor de Inversión de Control reduce grandemente la complejidad del código a interfaces, más que a clases. Proporciona una respuesta coherente, un modelo de programación simple en muchas áreas. [14].

Spring está compuesto por 7 módulos, sin embargo la flexibilidad con que está diseñado e implementado le brinda a los desarrolladores la posibilidad de escoger y utilizar sólo los módulos necesarios para el desarrollo de la aplicación e ignorar el resto:

**Módulo Core container:** Provee las funcionalidades fundamentales del framework. En este módulo se encuentra la clase BeanFactory, el corazón de las aplicaciones basadas en spring. BeanFactory aplica la Inversión de Controles (Invertion of Control o IoC), específicamente la Inyección de Dependencias (Dependency Injection), para separar la configuración de la aplicación y sus especificaciones de dependencias entre objetos de la lógica de negocio de la aplicación.

**Módulo MVC (Modelo- Vista- Controlador):** Brinda soporte al desarrollo de aplicaciones web basando el intercambio de los clientes (dígase navegadores web) con el servidor, en la filosofía que sigue el patrón arquitectónico del mismo nombre. Esta facilidad también emplea la Inversión de Controle para lograr separar limpiamente la lógica de controlador de los objetos de negocio, también permite unir declarativamente, parámetros de una petición a objetos de negocio. Además incorpora un mecanismo de validación de formularios muy útil en la comprobación de los datos provenientes del cliente [15].

### **Hibernate**

Es software libre, distribuido bajo los términos de la licencia GNU LGPL. Permite desarrollar clases persistentes a partir de clases comunes, incluyendo asociación, herencia, polimorfismo, composición y

colecciones de objetos. El lenguaje de consultas de Hibernate HQL (Hibernate Query Language), diseñado como una mínima extensión orientada a objetos de SQL, proporciona un puente elegante entre los mundos objetual y relacional. Hibernate también permite expresar consultas utilizando SQL nativo o consultas basadas en criterios. Soporta todos los sistemas gestores de bases de datos SQL y se integra de manera elegante y sin restricciones con los más populares servidores de aplicaciones J2EE y contenedores web [16].

### **IBatis**

Es un framework que facilita la implementación de la capa de persistencia utilizada en las aplicaciones. Es conocido como un mapeador de datos, lo que significa que mueve los datos entre los objetos y la base de datos manteniendo independientes el uno del otro y del propio mapeador [16]. Se divide en dos elementos:

**IBatis Data Mapper (SQL Maps):** proporciona un modo simple y flexible de mover los datos entre los objetos Java y la base de datos relacional. Este framework mapea clases a sentencias SQL usando un descriptor XML muy simple.

**IBatis Data Access Object (DAO):** Implementación realizado por el framework del patrón DAO. IBatis DAO es una capa de abstracción que oculta los detalles de la capa de persistencia y proporciona un API común para el resto de la aplicación. Con los DAOs se permite configurar una aplicación dinámicamente para usar distintos mecanismos de persistencia. [17]

### **1.3.9 Desarrollo de la interfaz de Usuario**

#### **Google Web Toolkit (GWT)**

Es un marco abierto de desarrollo de Java y su mejor característica es que no hay que preocuparse acerca de las incompatibilidades entre los navegadores Web y las plataformas. En GWT, se escribe el código en Java y luego el compilador lo traducirá en JavaScript y HTML. Proporciona un marco de programación que es similar a la utilizada por los desarrolladores para crear aplicaciones Java, utilizando uno de los kits de herramientas GUI como el Swing, AWT, o SWT.

Por otra parte, hace que la reutilización de fragmentos de programa sea fácil. Esto reduce en gran medida el número de tecnologías diferentes que se necesita para el desarrollo puede usarse cualquier IDE de Java para escribir y depurar una aplicación AJAX GWT. GWT no es sólo una interfaz de programación; proporciona un conjunto de herramientas que permiten desarrollar funcionalidades JavaScript de alto rendimiento en el navegador del cliente. Eso significa por ejemplo que se pueden

poner los puntos de interrupción en el código y depurar a la perfección desde el lado del cliente para el servidor. Se puede desplegar las aplicaciones en cualquier contenedor de servlets, así crear y ejecutar pruebas de unidad GWT es utilizado por muchos productos de Google, como Google Wave y la nueva versión de AdWords. Es de código abierto, completamente libre, y utilizado por miles de desarrolladores de todo el mundo [18].

### 1.3.10 Metodología y Herramientas seleccionadas.

La herramienta CASE a emplear será **Visual Paradigm** por ser multiplataforma, generar código para Java y exportar como HTML e integrarse con herramientas Java como el Eclipse.

El sistema Gestor de Base Datos escogido **PostgreSQL** ya que funciona en todos los sistemas operativos importantes, incluyendo Linux, UNIX y Windows. Además su código fuente está disponible para todos sin costo, permitiendo personalizarlo o extenderlo.

Se utilizará como framework **Spring** debido a que permite que objetos de negocio y de acceso a datos sean reutilizables, simplifica el manejo de errores y reduce considerablemente la cantidad de código a interfaces, más que a clases. Para dar soporte a implementaciones DAO se usará **Hibernate** ya que está diseñado para ser flexible en cuanto al esquema de tablas de una manera muy rápida y optimizada por lo que puede generarse la base de datos en cualquiera de los entornos soportados.

Se hará uso de **Java** como lenguaje de programación, del **Eclipse** como entorno de desarrollo y del **Apache Tomcat** como servidor web.

Para la realización de la interfaz de usuario hará uso de **GWT** hace que sea aún más fácil diseñar una aplicación AJAX utilizando sólo el lenguaje de programación Java.

## Conclusiones

En este capítulo se realizó un estudio de los servicios de mensajería para dispositivos móviles haciendo énfasis en los que utilizan el servicio de SMS, además de la utilización de estos servicios en Cuba y el mundo. Se describieron las principales herramientas a utilizar para la implementación de la solución que se propone y se abordaron un conjunto de conceptos asociados al dominio del problema que pudieran ser de ayuda a los interesados en conocer más a fondo la presente investigación.

## **Capítulo 2. Características del sistema.**

### **Introducción**

En este capítulo se describe la propuesta de un servicio web que permitirá el envío y recepción de mensajes durante el festival de cine. Se muestra un modelo de dominio como alternativa al modelo de negocio, modelando y relacionando los principales conceptos que se identificaron en el campo de acción. Se enumeran también los requisitos funcionales y no funcionales del sistema que se propone. Por último se identifican los actores y casos de usos del sistema así como su descripción.

### **2.1 Objeto de automatización.**

Se pretenden automatizar los procesos que intervienen en el envío y recepción de información a celulares referente a los Festivales de Cine.

### **2.2 Propuesta del sistema.**

Con el propósito de darle cumplimiento al problema planteado y de acuerdo a los estudios realizados, se propone el desarrollo de un servicio web para el envío y recepción de mensajes con información referente a los Festivales de Cine. Este servicio se activará al enviar un mensaje con una sintaxis predeterminada, con el texto de los mensajes se buscará la información almacenada referente a los festivales de cine que coincida con los datos del mensaje y se le enviará dicha información al usuario.

Los usuarios podrán:

- Conocer las carteleras de los principales cines del país.
- Conocer las noticias publicadas relacionadas con el Festival de Cines.
- Opinar sobre el desarrollo del Festival.
- En caso de no saber cómo enviar un mensaje pedir ayuda.

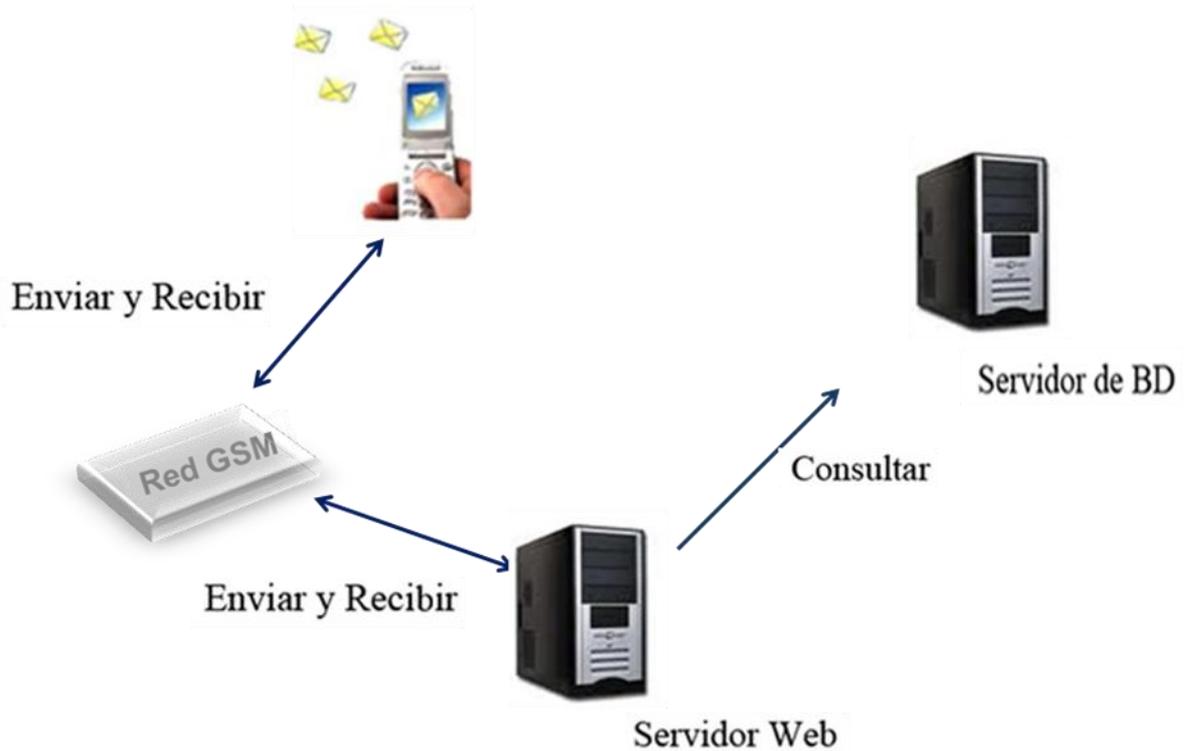


Figura 1 Propuesta del sistema (Módulo de Servicios)

Para poder brindarles a los usuarios la Información a los Festivales de Cine será necesario desarrollar una aplicación web auxiliar que permitirá actualizar esta información para ello esta aplicación brindará las opciones de añadir, eliminar y modificar:

- Las películas.
- Las noticias.
- Las instituciones cinematográficas.
- Las presentaciones.

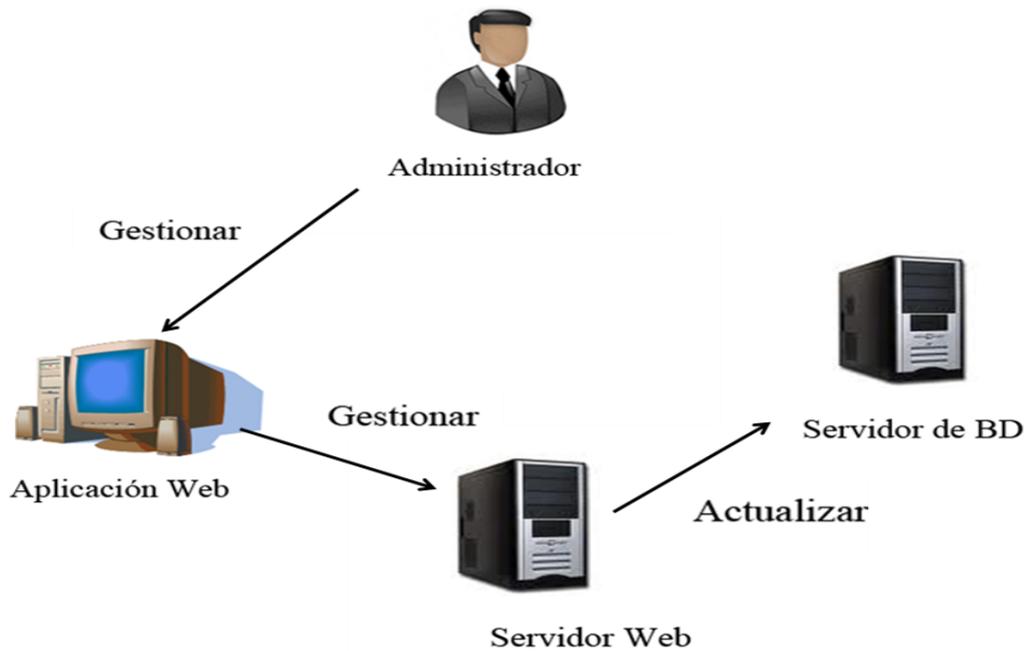


Figura 2 Propuesta del Sistema (Módulo de Administración)

### 2.3 Modelo de dominio.

Se propone un modelo del dominio, ya que en el sistema los procesos del negocio no están bien definidos, no son visibles y las fronteras no están bien establecidas. Además permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. Esto ayuda a los usuarios, clientes, desarrolladores e interesados, a utilizar un vocabulario común para poder entender el contexto en que se emplaza el sistema. Este modelo va a contribuir posteriormente a identificar algunas clases que se utilizarán en el sistema. Ver figura 1.

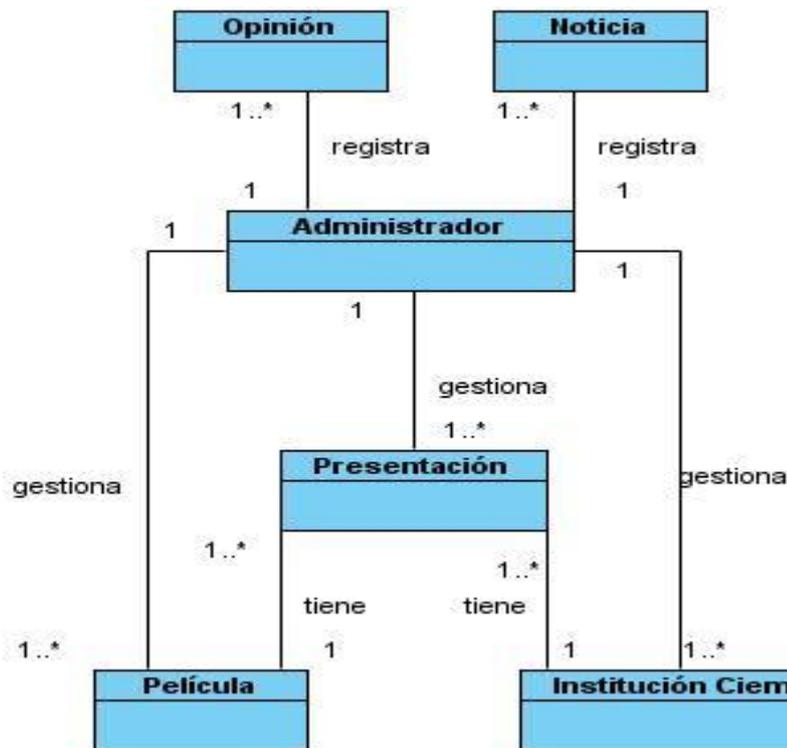


Figura 3 Modelo de Dominio

### 2.3.1 Descripción de los objetos del dominio:

**Administrador:** Gestiona la información del festival.

**Institución Cinematográfica:** Representa los lugares en los que se proyectan las películas del festival.

**Película:** Representan todas las películas que se proyectan durante el festival.

**Presentación:** Representan las películas que se van a presentar y la Institución Cinematográfica donde se va a presentar.

**Noticias:** Representa las noticias emitidas referentes al festival de cine.

**Opinión:** Representa las opiniones efectuadas por los usuarios sobre el festival de cine.

## 2.4 Relación de los Requerimientos (Funcionales, No Funcionales y Adicionales del Sistema).

Un requerimiento es una condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.

Los requisitos se pueden clasificar en: funcionales y no funcionales.

### **2.4.1 Requerimientos Funcionales:**

R1. Administrar los datos de Películas:

- 1.1 Adicionar películas al sistema.
- 1.2. Eliminar películas del sistema.

R2. Buscar Películas:

- 2.1. Permitirá buscar todas las películas según el identificador el mismo.
- 2.2. Permitirá buscar todas las películas almacenadas en el sistema.

R3. Administrar los datos de las Presentaciones:

- 3.1. Permitirá a los administradores adicionar presentaciones al sistema.
- 3.2. Permitirá a los administradores eliminar presentaciones del sistema.

R4. Buscar Presentaciones:

- 4.1. Permitirá buscar todas las presentaciones según el identificador del mismo.
- 4.2. Permitirá buscar todas las presentaciones almacenadas en el sistema.

R5. Administrar los datos de las Instituciones Cinematográficas:

- 5.1. Permitirá a los administradores adicionar Instituciones Cinematográficas al sistema
- 5.2. Permitirá a los administradores eliminar Instituciones Cinematográficas del sistema.

R6. Buscar Instituciones Cinematográficas:

- 4.1. Permitirá buscar todas las Instituciones Cinematográficas según el identificador del mismo.
- 4.2. Permitirá buscar todas las Instituciones Cinematográficas almacenadas en el sistema.

R6. Registrar Opiniones:

- 6.1. Permitirá almacenar todas las opiniones en el sistema.

R7. Administrar los datos de las Noticias:

- 7.1. Permitirá a los administradores adicionar noticias al sistema.
- 7.2. Permitirá a los administradores eliminar noticias del sistema.

R8. Buscar Noticias:

- 8.1. Permitirá buscar todas las noticias según el identificador de la misma.
- 8.2. Permitirá buscar todas las noticias almacenadas en el sistema.

R9. Mostrar Información:

- 9.1. Permitirá mostrar las presentaciones solicitadas por los usuarios.
- 9.2. Permitirá mostrar las noticias almacenadas en el sistema.

R10. Recibir ayuda:

- 10.1 Permitirá al usuario conocer como se realiza el intercambio de información con el sistema.

R11. Autenticar Usuario.

11.1. Permitirá al administrador autenticarse en la aplicación para poder gestionar la información.

R12 Cambiar contraseña

12.1 Permitirá al administrador modificar su contraseña.

### 2.4.2 Requerimientos no Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener.

#### Software:

En la PC servidor:

- Se necesita tener instalada la máquina virtual de Java (JVM) JRE 6.0 y la JDK 6.0.
- Se necesita tener instalado el Sistema Gestor de Base de Datos PostgreSQL 8.4.
- Se necesita tener instalado el servidor web Apache Tomcat 6.0.

En la PC cliente:

- Se necesita tener instalado un navegador web.

#### Hardware:

- Se necesita como requerimientos mínimos una PC Pentium IV para que se ejecute el servidor.
- Requiere como mínimo que la PC destinada a ser servidor posea una memoria RAM de 1 GB.

#### Portabilidad:

- El sistema deberá ser portable en cualquier sistema operativo.

#### Usabilidad:

- Se deben tener conocimientos previos del manejo de aplicaciones web. La aplicación podrá ser manejada por cualquier usuario con conocimientos básicos del tema.
- Los mensajes de error deberán ser lo suficientemente informativos para dar a conocer la severidad del error. Estos mensajes no deben revelar información interna.

#### Seguridad:

- La información de la aplicación deberá ser protegida de acceso no autorizado haciendo uso de mecanismos de autenticación y autorización que garanticen la protección de los datos que la requieran.
- La autenticación se realizará mediante el nombre de usuario y la contraseña
- El sistema debe estar disponible las 24 horas del día.

#### Requerimientos de Interfaz:

- La estructura general del sistema debe estar bien definida y la información que se muestre debe ser fácil de entender.
- Validar la entrada de información del usuario, e identificar los errores que se detecten para que el usuario los corrija.

#### Soporte:

- Garantizar la instalación y prueba del sistema, así como proveer la preparación e instrucción necesaria a los usuarios que trabajarán con la aplicación.

## **2.5 Modelo de Casos de Uso del Sistema.**

A continuación se presenta el modelo de casos de uso del sistema que comprende los actores del sistema, casos de uso que estos inicializan, el diagrama de casos de uso del sistema y una descripción detallada de estos casos de uso.

### **2.5.1 Definición de los actores del sistema a automatizar:**

#### **Definición de Actores**

<b>Actores</b>	<b>Justificación</b>
Administrador	Representa la persona que gestiona la información referente al festival de cine, como: las presentaciones, las películas, los cines y las noticias acerca del desarrollo de dicho festival.
Usuario	Representa las personas que a través de sus celulares accedan a los servicios que brinda el sistema.

**Tabla 1 Definición de Actores.**

### **2.5.2 Listado de Casos de Uso del sistema (CUS)**

Los casos de uso son artefactos narrativos que describen el comportamiento del sistema desde el punto de vista del usuario. Establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema. A continuación se muestran los casos de uso del sistema identificados teniendo en cuenta los requisitos funcionales.

CUS 1.Gestionar Película.

CUS 2.Buscar Película.

CUS 3.Gestionar Presentaciones.

CUS 4.Buscar Presentaciones.

CUS 5.Gestionar Instituciones Cinematográficas.

CUS 6.Buscar Instituciones Cinematográficas.

CUS 7.Gestionar Noticias.

CUS 8.Buscar Noticias.

CUS 9.Insertar Opinión.

CUS 10.Recibir Noticias del Festival.

CUS 11.Recibir Cartelera del Festival.

CUS 12.Recibir Ayuda.

CUS 13 Autenticar.

CUS 14 Cambiar Contraseña.

### 2.5.3 Diagramas de Casos de Uso del sistema

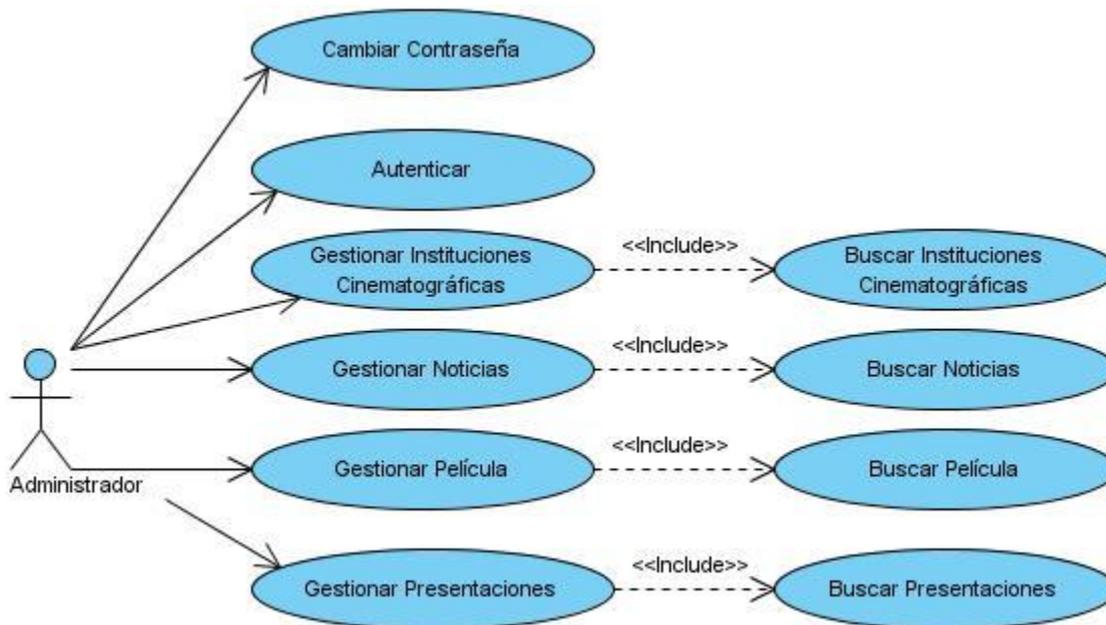


Figura 4 Diagrama de Caso de Uso del Sistema Módulo de Gestión de Información.

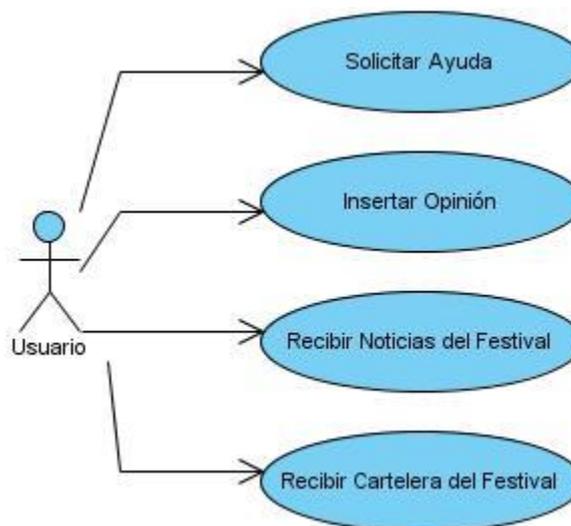


Figura 5 Diagrama de Casos de Uso del Sistema Módulo de Servicios del Sistema

### 2.5.4 Descripción de los Casos de Uso del sistema

<b>Caso de Uso:</b>	<b>Autenticar</b>
<b>Actores:</b>	<b>Administrador</b>
<b>Resumen:</b>	El usuario solicita la opción autenticarse y el sistema confirma que es el usuario señalado y

	le asigna los permisos
<b>Referencia</b>	R11
<b>Precondiciones:</b>	----
<b>Poscondiciones</b>	Queda registrado el usuario en la aplicación.

Tabla 2 Descripción del CU Autenticar

<b>Caso de Uso:</b>	<b>Gestionar Película</b>
<b>Actores:</b>	<b>Administrador</b>
<b>Resumen:</b>	El administrador selecciona la opción que permite gestionar películas. Se muestran vistas para insertar, eliminar y modificar películas, se realiza el proceso correspondiente a cada una de estas vistas y se actualiza la información en la BD.
<b>Referencia:</b>	R1.
<b>Precondiciones:</b>	El administrador debe autenticarse.
<b>Poscondiciones</b>	Queda la película gestionada.

Tabla 3 Descripción del CU Gestionar Película

<b>Caso de Uso:</b>	<b>Buscar película (CU incluido del CU Gestionar Película)</b>
<b>Actores:</b>	<b>Administrador.</b>
<b>Resumen:</b>	El sistema realiza una búsqueda en la BD para obtener el listado de las películas que estén almacenadas en la BD.
<b>Referencia</b>	R2
<b>Precondiciones:</b>	
<b>Poscondiciones</b>	Se muestran los datos de las películas almacenadas en la BD.

Tabla 4 Descripción del CU Buscar Película

<b>Caso de Uso:</b>	<b>Gestionar Presentación.</b>
<b>Actores:</b>	<b>Administrador</b>
<b>Resumen:</b>	El administrador selecciona la opción que permite gestionar Presentaciones. Se muestran vistas para insertar, eliminar y modificar Presentaciones, se realiza el proceso correspondiente a cada una de estas vistas y se actualiza la información en la BD.
<b>Referencia:</b>	R3.
<b>Precondiciones:</b>	El administrador debe autenticarse.

<b>Poscondiciones</b>	Queda la Presentación gestionada.
-----------------------	-----------------------------------

Tabla 5 Descripción del CU Gestionar Presentación

<b>Caso de Uso:</b>	<b>Buscar Presentación (CU incluido del CU Gestionar Presentación).</b>
<b>Actores:</b>	<b>Administrador.</b>
<b>Resumen:</b>	El sistema realiza una búsqueda en la BD para obtener un listado de las presentaciones almacenadas en la BD.
<b>Referencia</b>	R4
<b>Precondiciones:</b>	
<b>Poscondiciones</b>	Se muestran los datos de las Presentaciones almacenadas en la BD.

Tabla 6 Descripción del CU Buscar Presentación

<b>Caso de Uso:</b>	<b>Gestionar Institución Cinematográfica</b>
<b>Actores:</b>	<b>Administrador</b>
<b>Resumen:</b>	El administrador selecciona la opción que permite gestionar las Instituciones Cinematográficas. Se muestran vistas para insertar, eliminar y modificar Instituciones Cinematográficas, se realiza el proceso correspondiente a cada una de estas vistas y se actualiza la información en la BD.
<b>Referencia:</b>	R5.
<b>Precondiciones:</b>	El administrador debe autenticarse.
<b>Poscondiciones</b>	Queda la Institución Cinematográfica gestionada.

Tabla 7 Descripción del CU Gestionar Institución Cinematográfica

<b>Caso de Uso:</b>	<b>Buscar Institución Cinematográfica (CU incluido del CU Gestionar Institución Cinematográfica).</b>
<b>Actores:</b>	<b>Administrador.</b>
<b>Resumen:</b>	El sistema realiza una búsqueda en la BD para obtener un listado de las instituciones cinematográficas almacenadas en la BD.
<b>Referencia</b>	R6
<b>Precondiciones:</b>	
<b>Poscondiciones</b>	Se muestran los datos de las Instituciones Cinematográficas almacenadas en la BD.

Tabla 8 Descripción del CU Buscar Institución Cinematográfica

<b>Caso de Uso:</b>	<b>Gestionar Noticia</b>
<b>Actores:</b>	<b>Administrador</b>
<b>Resumen:</b>	El administrador selecciona la opción que permite gestionar las Noticias. Se muestran vistas para insertar, eliminar y modificar Noticias, se realiza el proceso correspondiente a cada una de estas vistas y se actualiza la información en la BD.
<b>Referencia:</b>	R7.
<b>Precondiciones:</b>	El administrador debe autenticarse.
<b>Poscondiciones</b>	Queda la Noticia gestionada.

Tabla 9 Descripción del CU Gestionar Noticia

<b>Caso de Uso:</b>	<b>Buscar Noticias (CU incluido del CU Gestionar Noticia).</b>
<b>Actores:</b>	<b>Administrador.</b>
<b>Resumen:</b>	El sistema realiza una búsqueda en la BD para obtener un listado de las noticias almacenadas en la BD.
<b>Referencia</b>	R8
<b>Precondiciones:</b>	
<b>Poscondiciones</b>	Se muestran los datos de las Noticias almacenadas en la BD.

Tabla 10 Descripción del CU Buscar Noticia

<b>Caso de Uso:</b>	<b>Insertar Opinión</b>
<b>Actores:</b>	<b>Usuarios</b>
<b>Resumen:</b>	El usuario envía un mensaje con la palabra opina y el sistema almacena el contenido del mismo.
<b>Referencia</b>	R6
<b>Precondiciones:</b>	----
<b>Poscondiciones</b>	Queda registrada la Opinión en la BD.

Tabla 11 Descripción del CU Insertar Opinión

<b>Caso de Uso:</b>	<b>Recibir noticias del Festival</b>
<b>Actores:</b>	<b>Usuarios</b>
<b>Resumen:</b>	El usuario envía un mensaje solicitando las noticias del día, el sistema busca y envía las noticias con la fecha en que recibió el mensaje.
<b>Referencia</b>	R8, R9

<b>Precondiciones:</b>	----
<b>Poscondiciones</b>	Noticias enviadas.

Tabla 12 Descripción del CU Recibir Noticias del Festival

<b>Caso de Uso:</b>	<b>Recibir Cartelera</b>
<b>Actores:</b>	<b>Usuarios</b>
<b>Resumen:</b>	El usuario solicita la cartelera, el sistema busca la misma y envía al usuario un mensaje con la cartelera solicitada.
<b>Referencia</b>	R11,R4
<b>Precondiciones:</b>	----
<b>Poscondiciones</b>	Cartelera enviada.

Tabla 13 Descripción del CU Recibir Cartelera

<b>Caso de Uso:</b>	<b>Recibir Ayuda</b>
<b>Actores:</b>	<b>Usuarios</b>
<b>Resumen:</b>	El usuario envía un mensaje con palabras claves "Ayuda" y <De que es la ayuda>, el sistema responde enviando un mensaje.
<b>Referencia:</b>	R10
<b>Precondiciones:</b>	----
<b>Poscondiciones</b>	Ayuda enviada.

Tabla 14 Descripción del CU Recibir Ayuda

<b>Caso de Uso:</b>	<b>Cambiar Contraseña</b>
<b>Actores:</b>	<b>Administrador</b>
<b>Resumen:</b>	El usuario selecciona la opción de cambiar contraseña y el sistema cambia la contraseña almacenada.
<b>Referencia</b>	R11
<b>Precondiciones:</b>	Tiene que haberse logueado previamente el usuario.
<b>Poscondiciones</b>	Queda cambiada la contraseña del usuario.

Tabla 15 Descripción del CU Cambiar Contraseña

2.5.4.1 Descripción textual de los Casos de Uso del Sistema en formato expandido, (VER ANEXO I).

## **Conclusiones**

En este capítulo se realizó una descripción de la solución propuesta, se representó el modelo de dominio con los principales conceptos del sistema, así como los requisitos funcionales y no funcionales. Se realizó el diagrama de casos de uso del sistema, así como la descripción detallada de estos.

## **Capítulo3. Análisis y Diseño del sistema.**

### **Introducción**

Este capítulo está dirigido fundamentalmente al diseño del sistema propuesto. Se explican los patrones de diseño y arquitectura empleados, los diagramas de clases del diseño, los diagramas de interacción y el diseño de la base de datos mediante el modelo de datos.

### **3.1 Diseño**

El diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, o sea, cómo cumple el sistema sus objetivos. RUP como metodología de desarrollo del software propone la realización del diseño como iteración fundamental de este flujo de trabajo siendo el análisis opcional en dicha fase. Con el diseño debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades.

#### **3.1.1 Patrones utilizados**

Un patrón define una solución a un problema en un contexto determinado e indica cómo puede ser utilizado en diferentes circunstancias. Pueden organizarse por categorías según la escala o nivel de abstracción entre las que pueden mencionarse:

**3.1.1.1 Patrones de Arquitectura:** Aquellos que expresan un esquema organizativo estructural fundamental para sistemas software.

##### **Patrón Modelo-Vista-Controlador**

Modelo Vista Controlador (MVC) En la mayoría de las aplicaciones web la lógica de un interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio. Con el modelo vista controlador se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos. Elementos del patrón:

Modelo: Datos y reglas de negocio

Vista: Muestra la información del modelo al usuario

Controlador: Gestiona las entradas del usuario

Un modelo puede tener diversas vistas, cada una con su correspondiente controlador.

##### **1. El modelo es el responsable de:**

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.

- Definir las reglas de negocio (la funcionalidad del sistema).
- Llevar un registro de las vistas y controladores del sistema.

### 2. El controlador es responsable de:

- Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- Contener reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada a actualizar.

### 3. Las vistas son responsables de:

- Recibir datos del modelo y mostrarlos al usuario.
- Tener un registro de su controlador asociado.
- Poder dar el servicio de actualizar, para que sea invocado por el controlador o por el modelo.

Este patrón se utilizó para desacoplar la lógica de la vista con respecto a la de modelo y minimizar el impacto de los cambios entre la interfaz y el negocio; donde las clases vistas son las View, las controladoras son los Controller y las clases principales del modelo son los Manager.

### Patrón n Capas.

La arquitectura en Capas está concebida con el objetivo de organizar el modelo de diseño en capas, que pueden estar físicamente distribuidas, en otras palabras, los componentes de una capa solo pueden hacer referencia a componentes en capas inmediatamente inferiores. La ventaja principal de este patrón es que simplifica la comprensión y la organización del desarrollo de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no son conscientes de ningún detalle o interfaz de las superiores, es decir en caso de que se realice algún cambio solo se afectará la capa donde se realizó la modificación.

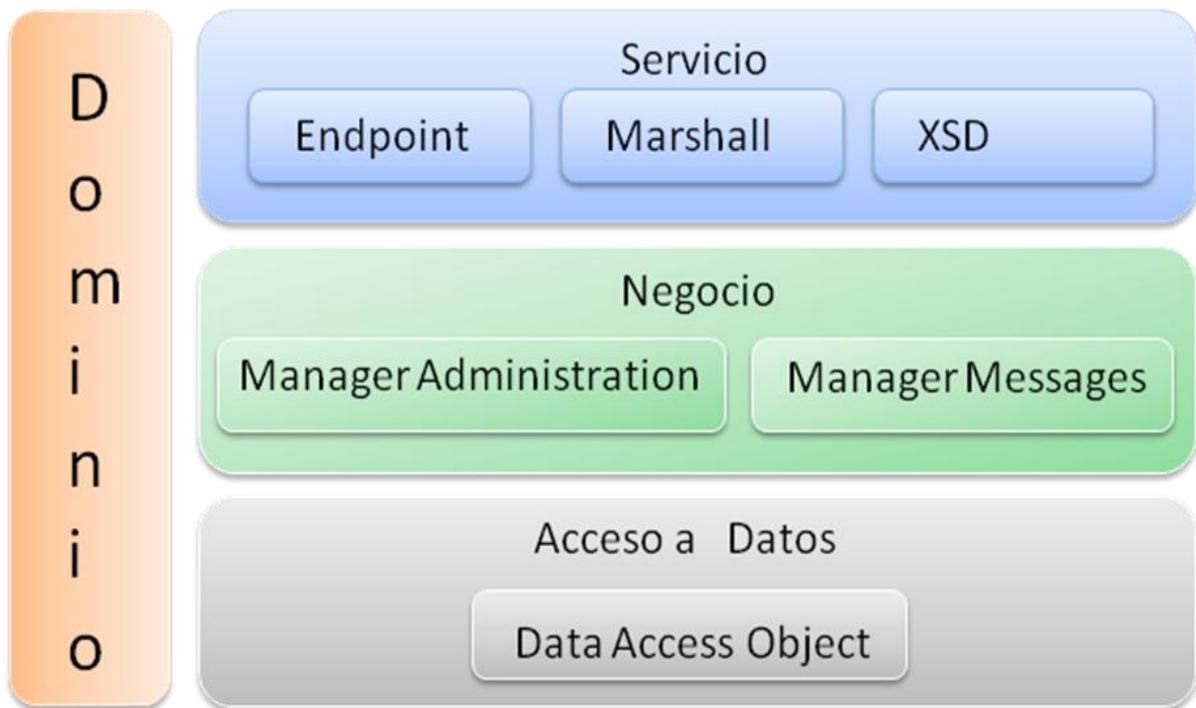


Figura 6 Arquitectura del módulo de servicios.

- En la capa Servicios se encuentran los XSD (Schema Definition Lenguaje), las clases marshall y los endpoint. Los XSD permiten asociar tipos de datos con elementos y atributos, es decir definir como estará estructurada la petición que se va a atender y el modo en que se va a responder dicha petición. Las clases marshall soportan los servicios web. Por último los endpoint que son los encargados de definir dada una petición que clase será la encargada de atender dicha petición.
- En la capa Negocio se encuentran los manager del módulo de administración y del módulo de mensajería, los mismos manejan toda la lógica del negocio.
- En la capa de Acceso a Dato se encuentran todas las clases que manejan los procedimientos, esto permite abstraerse de cómo se guardan físicamente los datos en la BD.
- El Dominio representa las clases que permiten el flujo de datos a través de las capas, desde la capa de Servicios hasta la capa de Acceso a Datos y viceversa.

**3.1.1.2 Patrones de Diseño:** Son directrices y principios estructurados que describen un problema común y entregan una buena solución ya probada a la que le dan un nombre. Estos ayudan a diseñar correctamente en menos tiempo, ayudan a construir problemas reutilizables y facilitan la documentación.

**Patrones GRASP (General Responsibility Assignment Software Patterns)**

Los patrones GRASP permiten describir los principios fundamentales de asignación de responsabilidades a objetos. Estos patrones son esenciales en el diseño eficaz de un software. Por las características de estos patrones, se consideró necesaria su utilización en este trabajo para obtener un producto de mayor calidad.

### **Controlador**

Este patrón GRASP establece la asignación de la responsabilidad de manejar los eventos a una clase que será la encargada de controlarlos. Permite la reutilización de componentes al delegar las responsabilidades en una clase, lo que facilita que aunque existan cambios desde el punto de vista de la interfaz del sistema las funcionalidades se mantengan intactas y puedan ser reutilizadas. Aplicado en el sistema, en la aplicación web este patrón se ve reflejado en las clases controladoras, las cuales son las encargadas de manejar las peticiones del cliente, en el caso de los servicios web los controladores serían los Endpoint, encargados de manejar las peticiones XML.

### **Experto**

Este patrón de diseño define asignar la responsabilidad a la clase que cuente con la información necesaria para manejar esta responsabilidad. Es necesario que la asignación de responsabilidades a las clases se lleve a cabo de una forma adecuada, lo que permitirá que el sistema sea más fácil de entender así como que se puedan reutilizar componentes. Con su aplicación se conserva el encapsulamiento, ya que los objetos se valen de su propia información para hacer lo que se les pide. El comportamiento se distribuye entre las clases que cuentan con la información requerida permitiendo la creación de clases sencillas y más cohesivas. En la aplicación este patrón se ve reflejado en las clases del negocio y en las clases de acceso a datos, donde cada una realiza funciones específicas.

### **Alta Cohesión**

Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un gran trabajo. Cada elemento del diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificable. En el sistema se tiene bien definido las responsabilidades de cada clase, por ejemplo, se definen clases del negocio por cada caso de uso, para evitar la sobrecarga de funcionalidades en una clase.

### **Bajo Acoplamiento**

El acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases, con que las conoce y con que recurre a ellas. Un alto acoplamiento tendría como consecuencia que las clases sean más difíciles de entender cuando estén aisladas, más difíciles de reutilizar porque se requiere la presencia de otras clases de las que dependen. Por lo que un bajo acoplamiento plantea que debe haber pocas dependencias entre las clases. Spring plantea el uso de interfaces para evitar un alto acoplamiento, por lo que en el sistema este patrón se ve claramente reflejado,

ya que tanto la capa de negocio como la capa de acceso a datos tienen interfaces que son implementadas por otras clases.

### **Patrón Inversión de Control (IoC) / Inyección de Dependencia (DI)**

Inversión de Control es un principio que generalmente siguen los frameworks, en lugar de tener las clases de la aplicación el control sobre la creación de objetos, le asigna esta responsabilidad a un agente externo como un contenedor. Spring proporciona un contenedor que maneja el ciclo de vida de los objetos y resuelve las dependencias entre dichos objetos. Se suelen confundir los términos de Inversión de Control e Inyección de Dependencia, pero no son lo mismo. Mientras que la Inversión de Control es un principio de diseño general, la Inyección de Dependencia es un patrón de diseño concreto que encarna este principio. La Inyección de Dependencia es la más típica realización de la Inversión de Control. El patrón Inyección de Dependencia, es una técnica que busca facilitar la resolución de dependencias entre objetos. Consiste en inyectar objetos a una clase en lugar de ser la propia clase quien cree el objeto. La forma habitual de implementar este patrón es mediante un contenedor. El contenedor inyecta a cada objeto los objetos necesarios según las relaciones plasmadas en un fichero de configuración. Típicamente este contenedor es implementado por un framework externo a la aplicación (como Spring). El uso de este patrón permite construir aplicaciones mucho más flexibles y robustas.

### **Patrón Fachada**

El patrón fachada trata de simplificar la interfaz entre dos sistemas o componentes de software ocultando un sistema complejo detrás de una clase que hace las veces de pantalla o fachada.

La idea principal es la de ocultar todo lo posible la complejidad de un sistema, el conjunto de clases o componentes que lo forman, de forma que solo se ofrezca un (o unos pocos) punto de entrada al sistema tapado por la fachada.

Una ventaja más de usar una clase fachada para comunicar las dos partes o componentes, es la de aislar los posibles cambios que se puedan producir en alguna de las partes. Si cambias, por poner un ejemplo, el medio de comunicación o de almacenamiento de una de las partes, la otra, que por ejemplo hace la presentación, no tiene que enterarse, y viceversa.

### **Ejemplo Práctico del uso de los patrones en la aplicación**

La clase EntityFacade sirve de intermediaria entre la clase controladora y la clase manager facilitando así la comunicación entre las mismas, además excluye los posibles cambios que se puedan producir en alguna de esas clases.

La clase EntityController es responsable de manejar las peticiones del cliente facilitando que aunque existan cambios en la interfaz del sistema, las funcionalidades se van a mantener intactas y podrán ser reutilizadas.

Cada una de las clases se encarga específicamente de sus funciones, por lo que las clases son sencillas y más cohesivas.

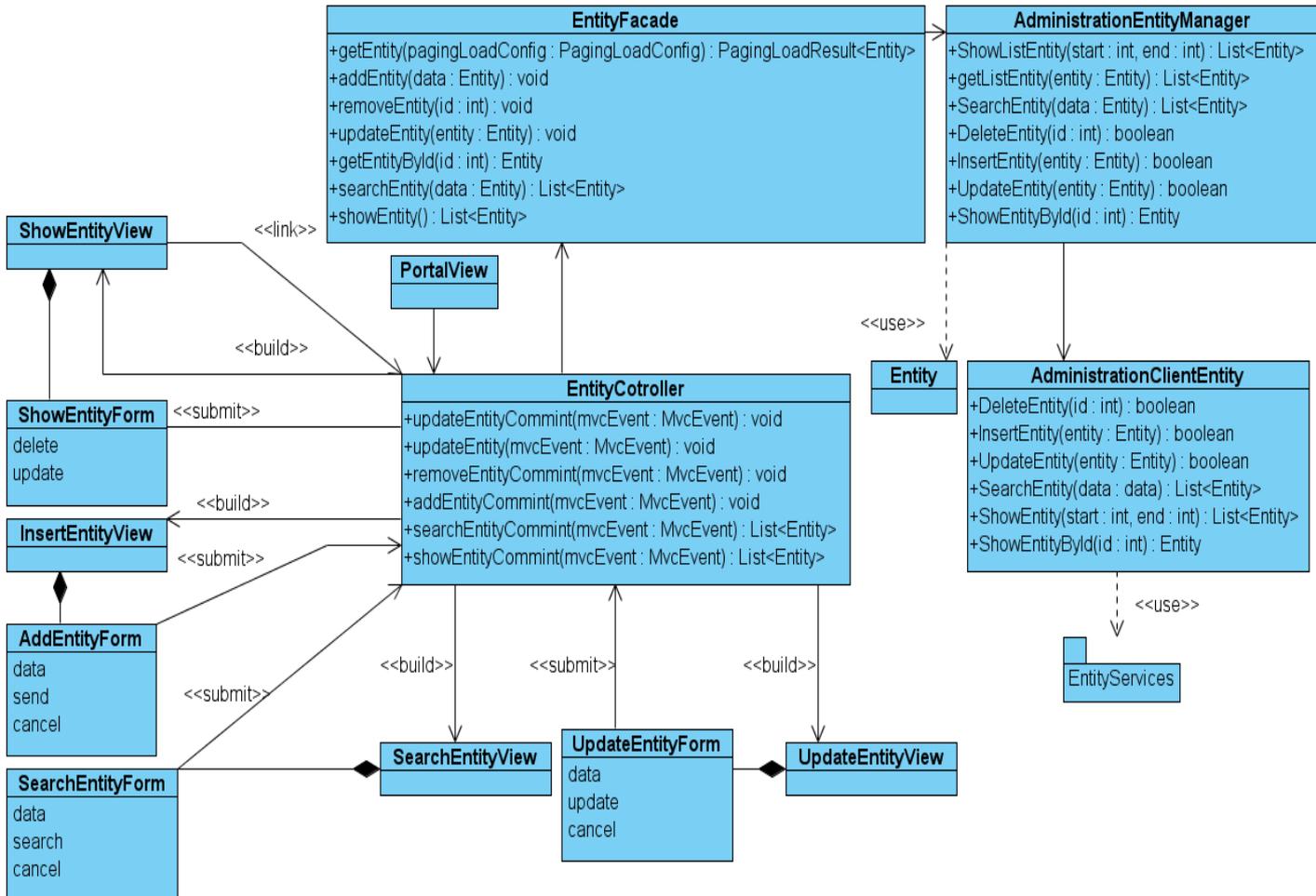


Figura 7 Diagrama de Clases del Diseño (Ejemplo para explicar el uso de los patrones).

### 3.3 Diagramas de Clases del Diseño.

El modelo de diseño contribuye a una arquitectura estable y sólida, y a crear un plano del modelo de implementación. En el diseño se modela el sistema y se encuentra su forma, incluida la arquitectura, para que soporte todos los requisitos, incluyendo los no funcionales. El diagrama de clases del diseño contiene clases y las relaciones estructurales entre ellas. Este se obtiene como resultado del refinamiento del modelo conceptual y de los diagramas de secuencia. La definición de clase incluye definiciones para atributos y responsabilidades. Los mismos pueden ser encontrados en el Anexo II.



Figura 8 Diagrama de Diseño del CU Insertar Opinión

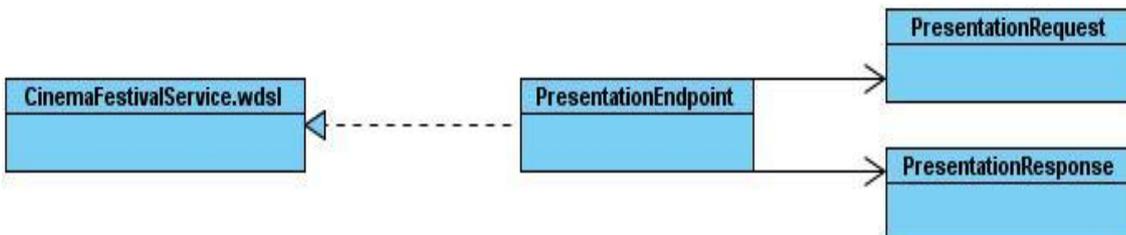


Figura 9 Diagrama de Diseño del CU Recibir Cartelera.

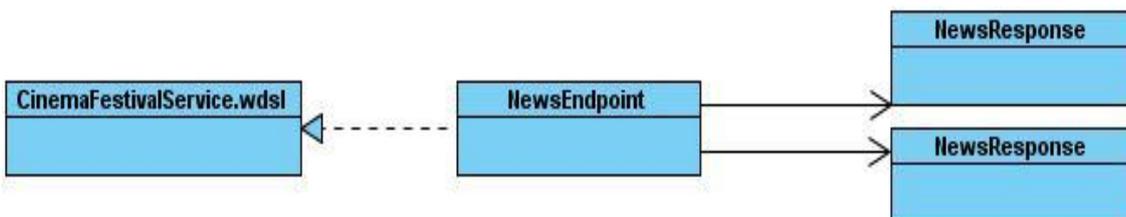


Figura 10 Diagrama de Diseño del CU Recibir Noticias.

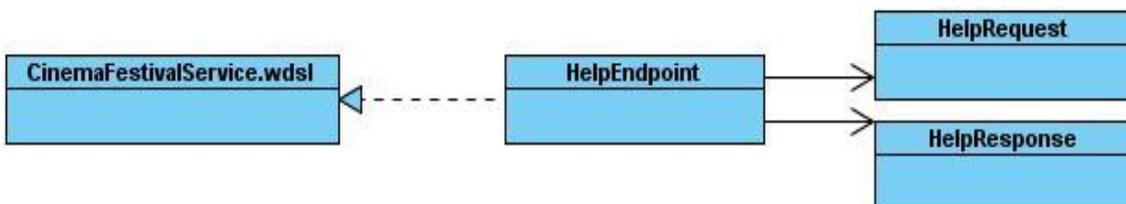


Figura 11 Diagrama de Diseño del CU Recibir Ayuda.

### 3.4 Diagramas de Interacción.

Los diagramas de interacción modelan los aspectos dinámicos del sistema, mostrando las interacciones entre objetos mediante transferencia de mensajes entre ellos. Dentro de estos diagramas están los de secuencia y los de colaboración. En el presente trabajo se emplearon los diagramas de secuencia teniendo en cuenta que muestran las interacciones entre objetos, ordenadas en secuencia temporal durante un escenario concreto. Los mismos pueden ser encontrados en el Anexo III.

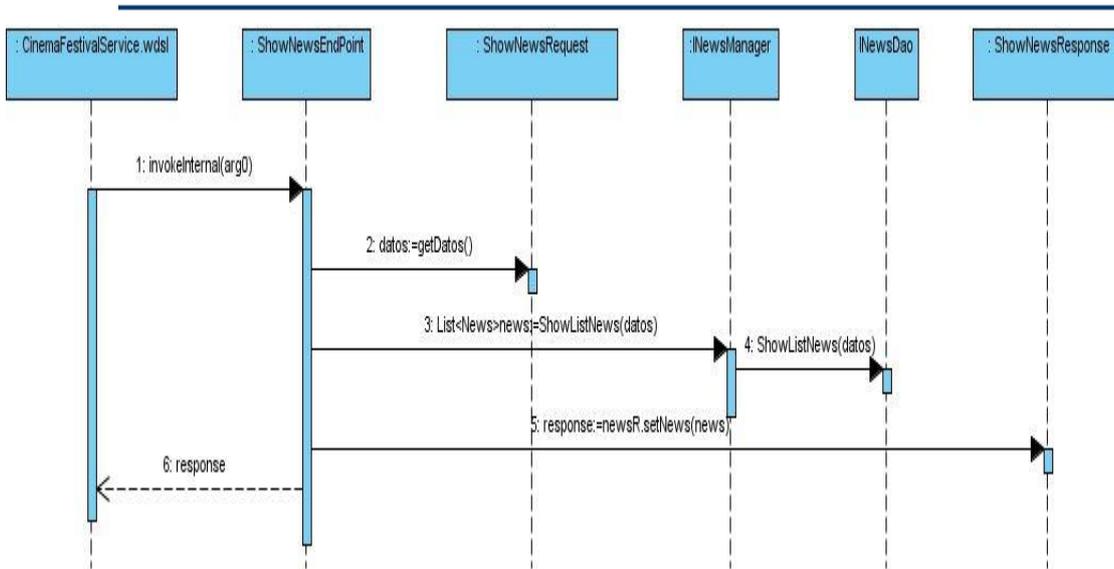


Figura 12 Diagrama del Servicio Listar Noticias (Show List News).

### 3.5 Diseño de la Base de Datos

Una base de datos o banco de datos (del inglés Database) es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. El diseño de la base de datos es una de las tareas más importantes en la construcción de un sistema, ya que una base de datos bien diseñada contendrá información correcta, almacenará los datos más eficientemente y será más fácil de gestionar y de mantener.

3.5.1 Modelo Lógico de Datos (diagrama de clases persistentes)

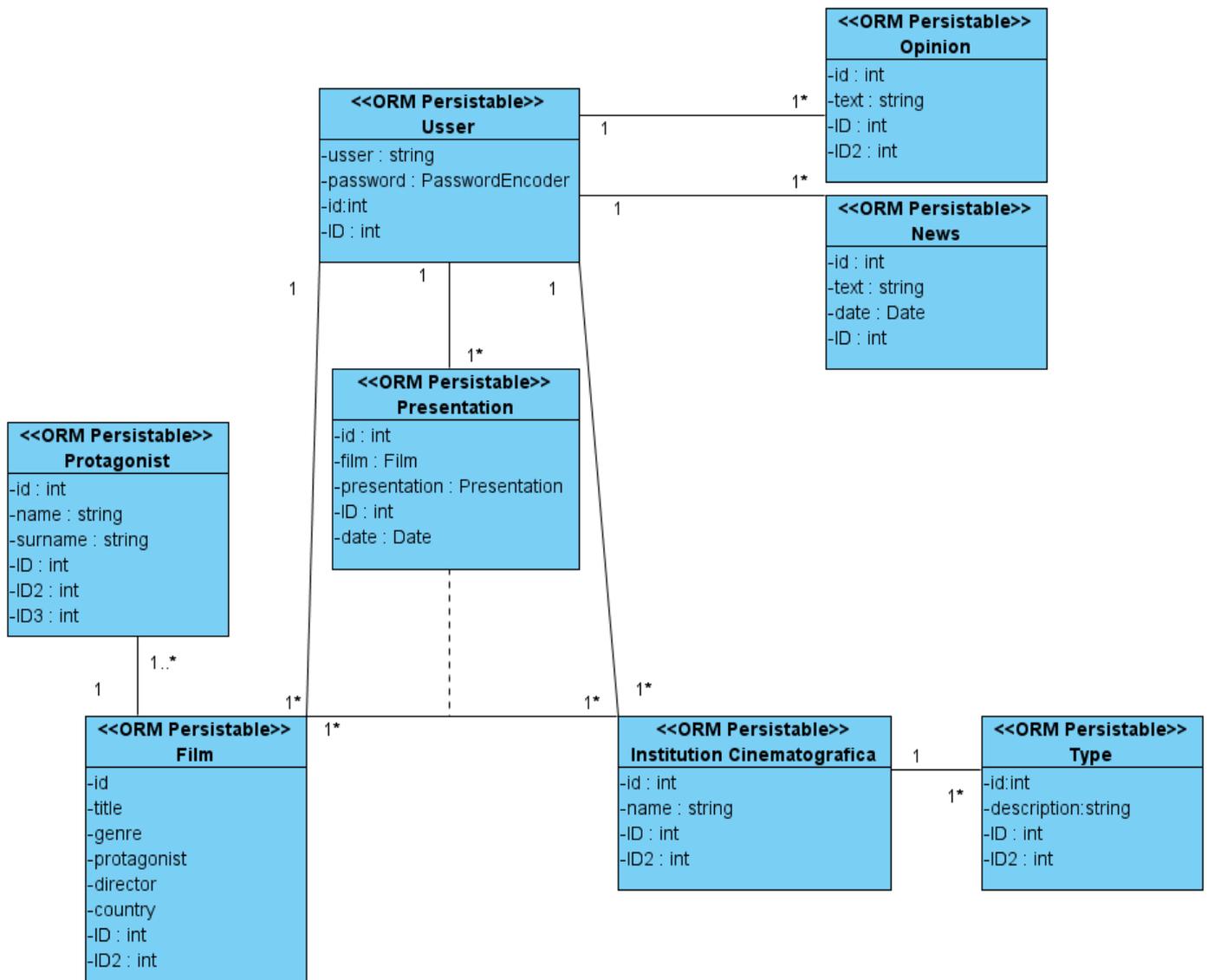


Figura 13 Diagrama de Clases Persistentes

### 3.5.2 Modelo Físico de Datos (Modelo de Datos)

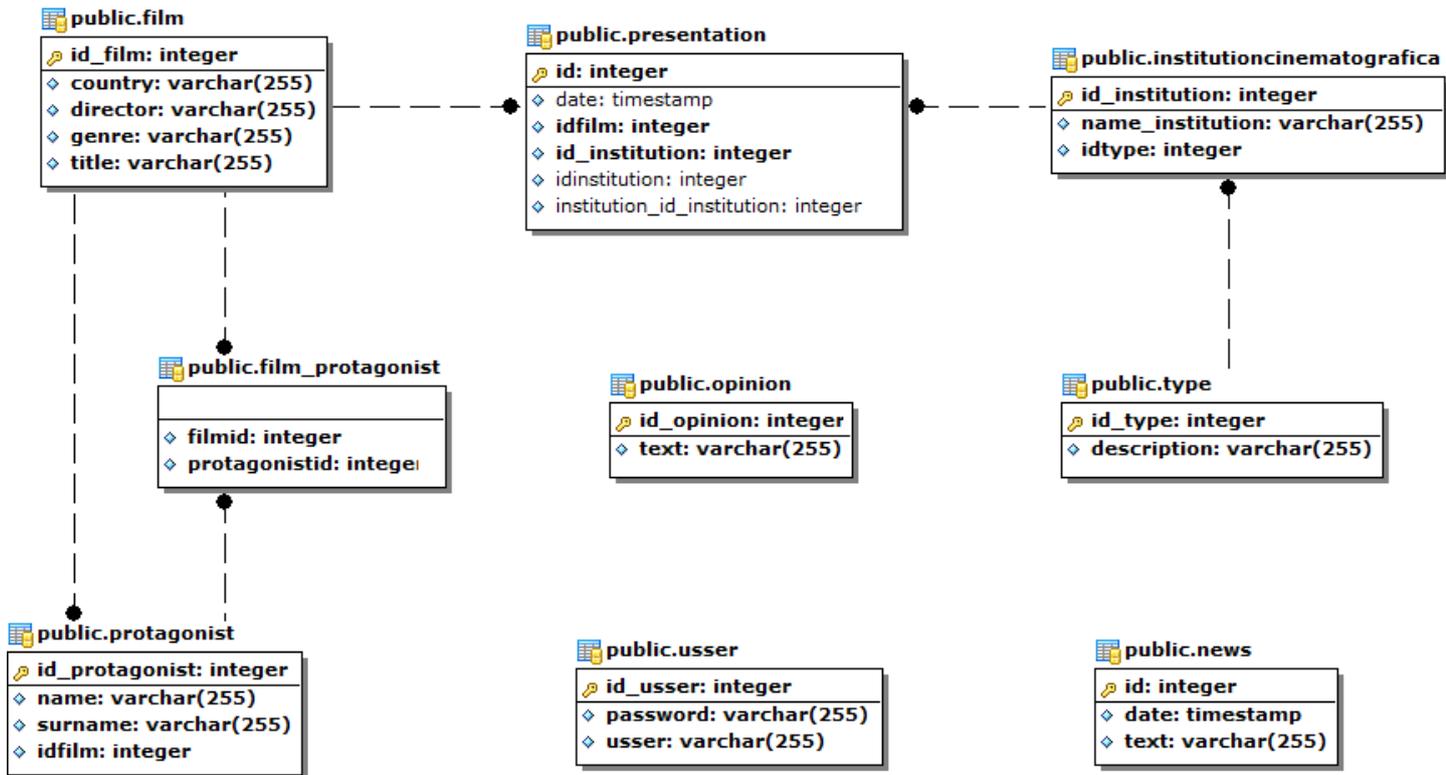


Figura 14 Modelo de Datos.

## Conclusiones

En este capítulo se definió el diseño del sistema propuesto. Se explicaron los patrones de diseño y arquitectura empleados, los diagramas de clases del diseño, los diagramas de interacción y el diseño de la base de datos mediante el modelo de datos.

## Capítulo 4. Implementación.

### Introducción

En el presente capítulo se explica cómo está estructurado físicamente el sistema implementado mediante los modelos de despliegue y de componentes, con los diagramas correspondientes estructurados por paquetes con sus respectivas descripciones.

### 4.1 Diagrama de Despliegue.

El modelo de despliegue contiene los nodos que conforman la topología de hardware sobre la cual se ejecuta el sistema software implementado (servicio web). En la siguiente figura se muestra el Diagrama de Despliegue del sistema desarrollado.

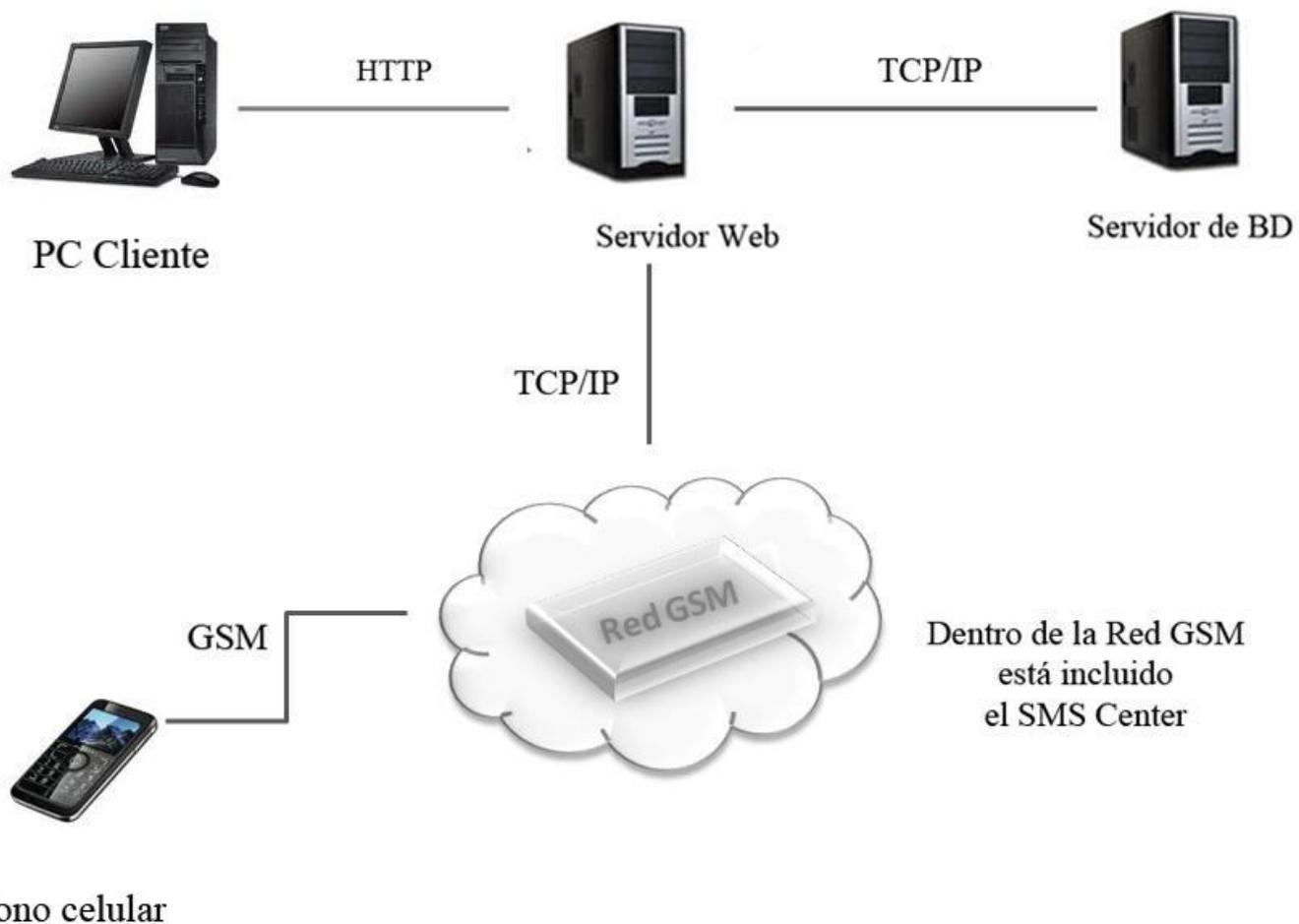


Figura 15 Diagrama de Despliegue.

**En el nodo PC Cliente:** se puede acceder a la aplicación web mediante un navegador.

**En el nodo Servidor Web:** Estará el servidor Web Apache Tomcat, donde se ejecutarán los servicios y la aplicación web.

**En el nodo servidor de base de datos:** se utilizará el Sistema gestor de Base de Datos PostgreSQL para el acceso a la base de datos.

**El nodo Teléfono celular** representa los dispositivos móviles que van a consumir los servicios.

**El nodo Red GSM:** sirve de intermediario entre el teléfono celular y los servicios.

### 4.2 Diagramas de Componentes.

Los Diagramas de Componentes representan la separación del software en componentes físicos (archivos, cabeceras, módulos, paquetes, etc.). Se usan para modelar la estructura del software, incluyendo las dependencias entre los componentes de software, los componentes de código binario, y los componentes ejecutables. A continuación se representa el diagrama de componentes estructurado por paquetes correspondientes al sistema implementado.

#### 4.2.1 Descripción de los paquetes del Diagrama de componentes:

Los paquetes services y ServCineApplication representan las dos aplicaciones que se han desarrollado.

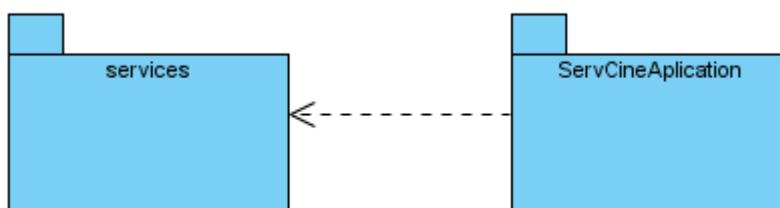


Figura 16 Diagrama de Componentes.

#### 4.2.2 Descripción de los paquetes del Diagrama de componentes Modulo de Servicios

El paquete SRC tiene como propósito agrupar las clases que desarrollan los servicios a desplegar.

El paquete WEB-INF contiene todas las librerías y componentes que deberán ser empleados para la implementación de los servicios

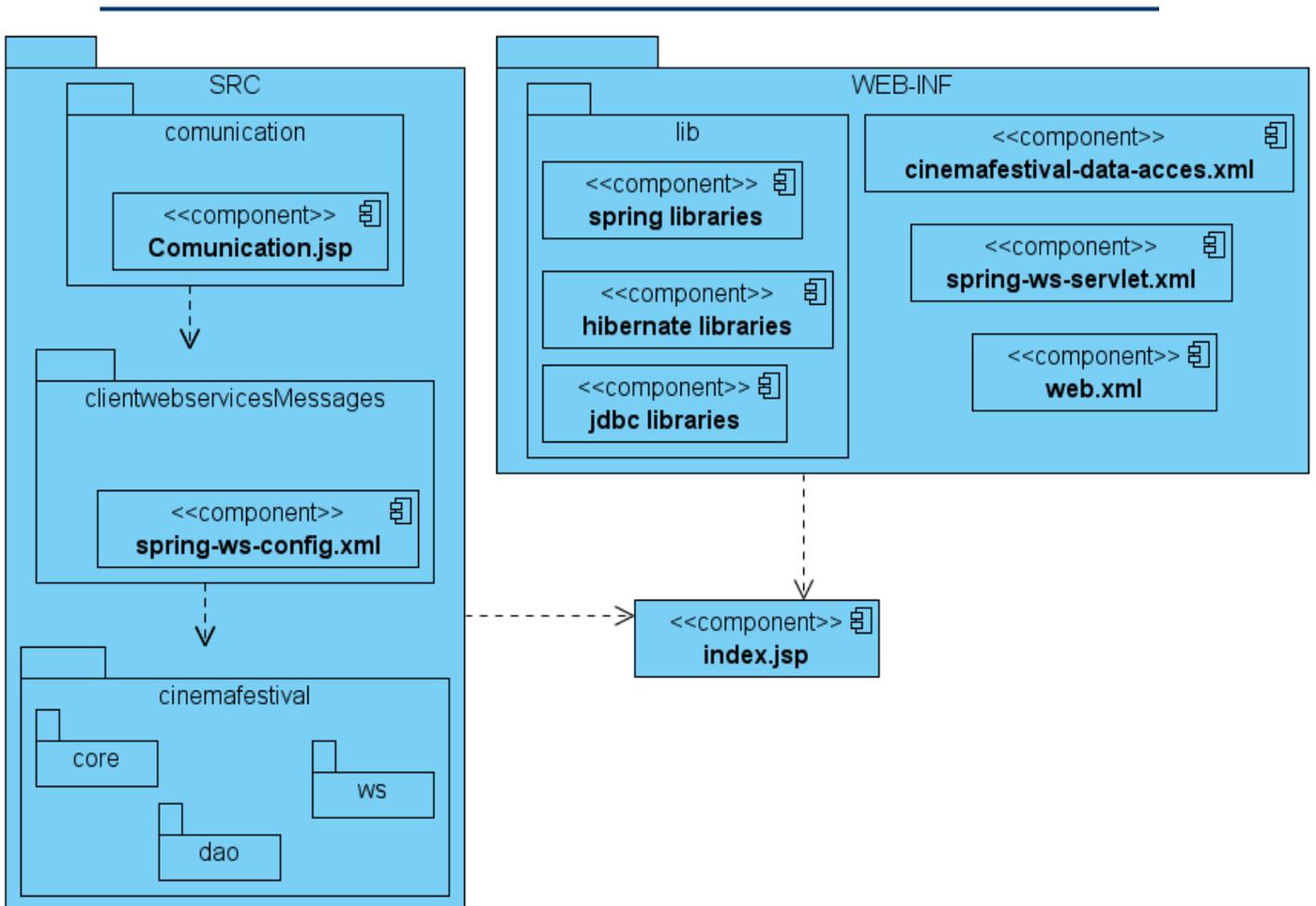


Figura 17 Diagrama de Componentes Módulo de Servicios.

#### 4.2.2 Descripción de los paquetes del Diagrama de componentes Módulo de Administración.

El paquete SRC tiene como propósito agrupar las clases que desarrollan la aplicación a desplegar. El paquete WEB-INF contiene todas las librerías, estilos y componentes que deberán ser empleados para la implementación de la aplicación

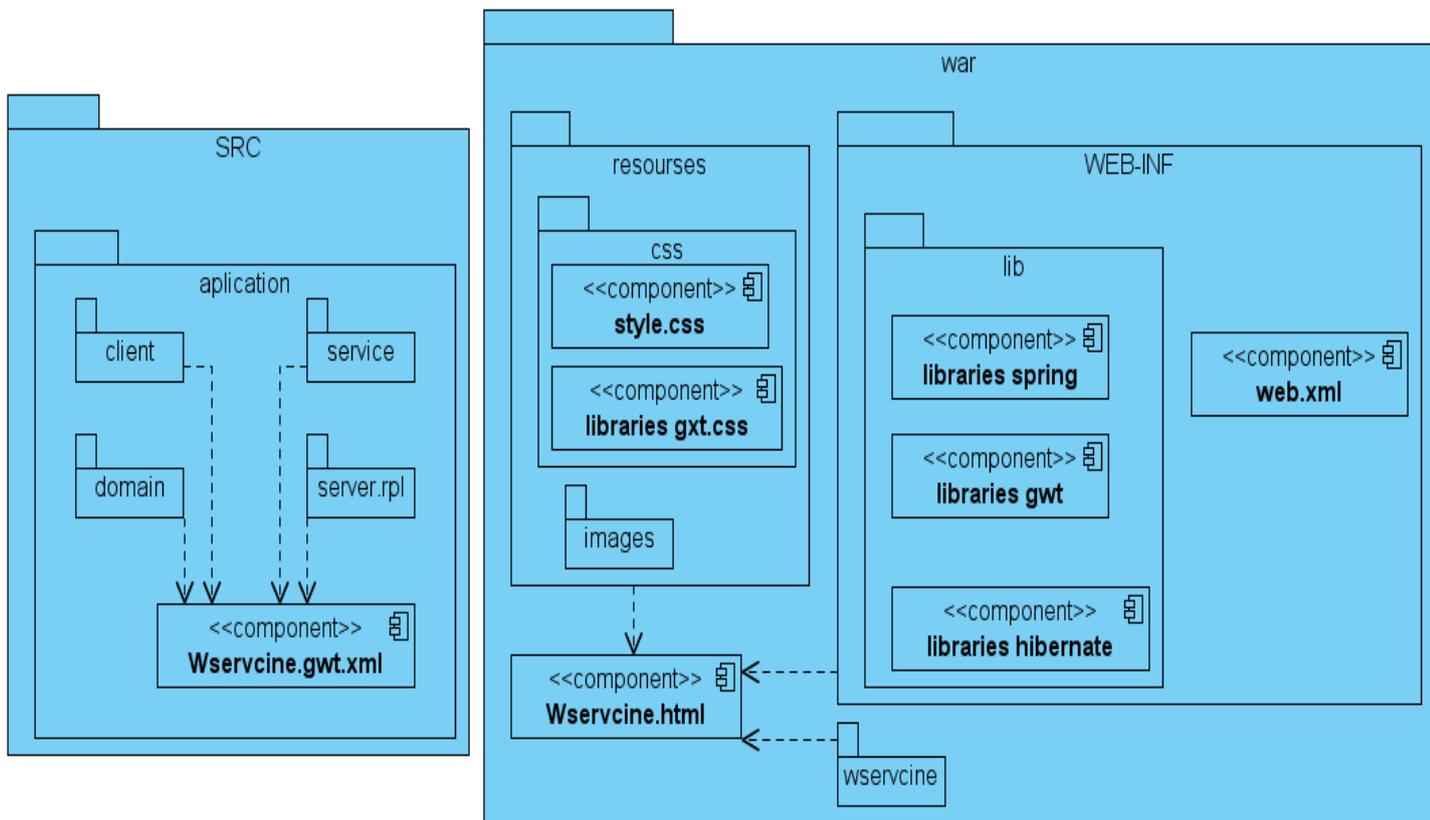


Figura 18 Diagrama de Componentes Módulo Administración

## Conclusiones

En este capítulo se explicó cómo está estructurado físicamente el sistema implementado mediante los modelos de despliegue y de componentes con los diagramas correspondientes estructurados por paquetes.

## Capítulo 5. Estudio de Factibilidad.

### Introducción:

En este capítulo se realizará mediante el método de Puntos de Caso de Uso una estimación del esfuerzo, costo y tiempo de desarrollo del proyecto. Además se hará un estudio de los costos y beneficios del mismo para llegar a la conclusión si el sistema es factible o no.

### 5.1 Método a utilizar: Método de Estimación por Puntos de Casos de Uso.

#### Paso 1: Cálculo de Puntos de Casos de Uso sin ajustar.

Se calcula a través de la siguiente ecuación:

$$UUCP = UAW + UUCW$$

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.

#### Para determinar el Factor de Peso de los Actores sin ajustar (UAW):

Tipo de actor	Descripción	Factor de peso	Actores	Total
Simple	Sistema con sistema a través de interfaz de programación.	1	0	0
Medio	Sistema con sistema mediante protocolo de interfaz basada en texto.	2	1	2
Complejo	Persona que interactúa con el sistema mediante una interfaz gráfica.	3	1	3
Total				5

Tabla 16 Factor de Peso de los Actores sin Ajustar (UAW)

$$UAW = \sum \text{cant\_actores} * \text{Factor de Peso}$$

$$UAW = 5$$

#### Para determinar el Factor de Peso de los Casos de Uso sin ajustar (UUCW):

Tipo de Caso de Uso	Descripción	Peso	Cantidad*Peso
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	10*5
Medio	El caso de uso tiene de 4 a 7 transacciones.	10	4*10

Complejo	El caso de uso tiene más de 8 transacciones.	15	0*15
Total			90

Tabla 17 Factor de Peso de los CU sin Ajustar (UUCW)

$$UUCW = \sum \text{cant\_CU} * \text{Peso}$$

$$UUCW = 90$$

Los Puntos de Casos de Uso sin ajustar resultan

$$UUCP = UAW + UUCW$$

$$UCCP = 5 + 90$$

$$UCCP = 95$$

### Paso 2. Cálculo de Puntos de Casos de Uso ajustados.

Se calcula mediante la siguiente ecuación:

$$UCP = UUCP \times TCF \times EF$$

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.

**Factor de complejidad técnica:** Este coeficiente se calcula mediante un conjunto de Factores que determinan la complejidad del sistema, cada factor se cuantifica con un valor de 0 a 5. Significado de los valores:

0: No presente o sin influencia.

1: Influencia incidental o presencia incidental.

2: Influencia moderada o presencia moderada.

3: Influencia media o presencia media.

4: Influencia significativa o presencia significativa.

5: Fuerte influencia o fuerte presencia.

Factor	Descripción	Peso	Valor asignado	Total
T1	Sistema distribuido	2	0	0
T2	Tiempo de respuesta	1	3	3
T3	Eficiencia del usuario final	1	3	3
T4	Funcionamiento Interno complejo	1	3	3
T5	El código debe ser reutilizable	1	3	3

T6	Facilidad de instalación	0.5	3	1.5
T7	Facilidad de uso	0.5	5	2.5
T8	Portabilidad	2	5	10
T9	Facilidad de cambio	1	3	3
T10	Concurrencia	1	4	4
T11	Incluye objetivos especiales de seguridad	1	0	0
T12	Provee acceso directo a terceras partes	1	0	0
T13	Se requieren facilidades especiales de entrenamiento de usuarios	1	0	0
Total				33

Tabla 18 Factor de Complejidad Técnica

Se utiliza la siguiente ecuación:

$$TCF = 0.6 + 0.01 \times \sum (\text{Peso}_i \times \text{Valor asignado}_i)$$

$$TCF = 0.6 + 0.01 \times 33$$

$$TCF = 0.6 + 0.33$$

$$TCF = 0.93$$

#### Factor de ambiente (EF):

El factor de ambiente (EF) está relacionado con las habilidades y el entrenamiento del grupo involucrado en el desarrollo del sistema. El cálculo del mismo es similar al cálculo del Factor de complejidad técnica, es decir, se trata de un conjunto de factores que se cuantifican con valores de 0 a 5.

Para determinar el Factor de ambiente (EF):

Factor	Descripción	Peso	Peso * Valor	Total
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	1.5 * 2	3
E2	Experiencia en la aplicación.	0.5	0.5 * 0	0
E3	Experiencia en la orientación a objeto.	1	1 * 4	4
E4	Capacidad del analista líder.	0.5	0.5 * 0	0
E5	Motivación.	1	1 * 5	5
E6	Estabilidad de requerimientos.	2	2 * 2	4
E7	Personal Part-Time.	-1	-1 * 3	-3
E8	Dificultad del lenguaje de programación.	-1	-1 * 3	-3
Total				10

Tabla 19 Factor de Ambiente

$$EF = 1.4 - 0.03 \times \sum (\text{Peso}_i \times \text{Valor asignado}_i)$$

$$EF = 1.4 - 0.03 * 10$$

$$EF = 1.01$$

$$UCP = UUCP \times TCF \times EF$$

$$UCP = 95 * 0.93 * 1.01$$

$$UCP = 89.23$$

**Paso 3. Estimación de esfuerzo a través de los puntos de casos de uso.**

Para calcular el Esfuerzo se utiliza la siguiente ecuación:

$$E = UCP \times CF$$

**E:** Esfuerzo estimado en horas-hombre.

**UCP:** Puntos de Casos de Uso ajustados.

**CF:** Factor de conversión.

**CF = 20** horas-hombre (si Total EF  $\leq$  2)

**CF = 28** horas-hombre (si Total EF = 3 ó Total EF = 4)

**CF = abandonar o cambiar proyecto** (si Total EF  $\geq$  5)

**Total EF = Cant EF < 3** (entre E1 –E6) + **Cant EF > 3** (entre E7, E8)

$$\text{Total EF} = 2 + 0$$

$$\text{Total EF} = 2$$

$$E = UCP \times CF$$

$$E = 89.23 * 20 \text{ horas-hombre}$$

$$E = 1784.6 \text{ horas-hombre}$$

**Paso 4. Distribución del Esfuerzo entre las diferentes actividades de un proyecto**

Actividad	Porcentaje %	Horas-Hombres
Análisis	10	446.15
Diseño	20	892.3
Implementación	40	1784.6
Pruebas	15	669.22
Sobrecarga (otras actividades)	15	669.22
Total	100	4461.5

Tabla 20 Distribución del Esfuerzo entre las actividades del proyecto

El Esfuerzo Total sería 4461.5 horas-hombre, si estimamos teniendo en cuenta que un mes tiene 288 horas laborables, pues se trabajan 12 horas diarias 24 días al mes, entonces el Esfuerzo Total en mes-hombre sería 15.49 mes-hombre.

## **5.2 Calcular el costo de todo el proyecto.**

$$\text{Costo} = \text{CHM} * \text{ET} / \text{CH}$$

Donde:

**CH:** Cantidad de hombres.

**CHM:** Costo Hombre – Mes.

**ET:** Esfuerzo Total.

Si la Cantidad de hombres es 2 y se tiene un Salario Promedio mensual igual a \$100.00. Entonces

$$\text{CHM} = \text{CH} * \text{Salario Promedio}$$

$$\text{CHM} = 2 * 100 \text{ CHM} = 200.00$$

Luego

$$\text{Costo} = \text{CHM} * \text{ET} / \text{CH}$$

$$\text{Costo} = 200.00 * 15.49 / 2$$

$$\text{Costo} = \$1540.$$

## **5.3 Calcular el tiempo de desarrollo de todo el proyecto.**

A partir de los datos conocidos anteriormente se puede decir que una persona puede realizar el trabajo en un tiempo aproximado de 26.70 meses. Pero teniendo en cuenta que la cantidad de hombres para la realización del Servicio de Información de Festivales de Cine para Celulares es 2 el tiempo va a ser:

$$\text{Tiempo} = \text{ET} / \text{CH}$$

$$\text{Tiempo} = 15.49 / 2$$

$$\text{Tiempo} = 7.7 \approx 8 \text{ meses}$$

De los resultados obtenidos se interpreta que con 2 hombres trabajando en el proyecto el mismo se desarrolla en 8 meses y su costo total se estima que sea \$ 1540.

## **5.4 Beneficios Tangibles e Intangibles**

La creación del Servicio de Información de Festivales de Cine para Celulares pretende ofrecer a los clientes del operador celular CUBACEL un nuevo servicio: Información de Festivales de Cine. Actualmente los servicios son un paso de avance en la telefonía celular obteniéndose ganancias y fomentando la popularidad que tienen estos dispositivos a nivel mundial, ya que brindan una serie de comodidades, por ejemplo, conocer desde cualquier lugar Información de los Festivales de Cine. Para

la realización de dicha aplicación se implementa una aplicación web desarrollada sobre la plataforma Java, garantizando posibles migraciones de sistemas operativos o plataformas en un futuro, sin afectar a los beneficiarios.

### **5.5 Análisis de costos y beneficios**

El análisis de los beneficios que aportará el desarrollo de un proyecto, constituye una ayuda importante en la toma de decisiones, ya que frecuentemente brinda la información necesaria para determinar si el producto a desarrollar es factible o no. Con la realización de los Servicios y la aplicación web se reflejaron algunas ventajas que permitieron obtener un costo del proyecto no muy elevado, entre ellas: es un sistema fácil de usar, multiplataforma y que está soportado con tecnologías de software libre. Debido a estas características se considera que es factible la implementación del mismo.

### **Conclusiones**

En este capítulo se llevó a cabo un estudio de la factibilidad del proyecto teniendo en cuenta el posible costo, el tiempo de duración y los posibles beneficios que podrá brindar.

## **Conclusiones Generales**

Después de realizar la investigación se puede concluir que las tareas de investigación y el objetivo general, así como los específicos del trabajo de diploma se cumplieron íntegramente, obteniéndose los siguientes resultados:

- Se analizó la información relacionada con los Festivales de Cine.
- Se realizó un estudio de la metodología, lenguajes, herramientas y tecnologías de desarrollo que se utilizaron en la confección de la solución.
- Se describieron y detallaron los patrones de diseño que se utilizaron en la elaboración de la aplicación.
- Se construyó una aplicación que consta de dos módulos uno de servicios capaz de garantizar rapidez y efectividad a los usuarios con celulares para el conocimiento de la información referente a los Festivales de Cine que tenga lugar en ese momento y otro para gestionar dicha información a través una de una aplicación web.

## **Recomendaciones**

El desarrollo de la aplicación se efectuó satisfactoriamente, sin embargo, con el propósito de alcanzar mejores resultados se recomienda:

- Continuar con el desarrollo de la aplicación adicionando la opción de enviar a los usuarios imágenes referentes a la información solicitada por este utilizando el servicio **MMS**.
- Mejorar las interfaces de usuario del Módulo de Administración.

## Referencias Bibliográficas

- [1] Vida digital-Tecnología Internet y Ciencia, SMS o mensajes de texto <http://www.vidadigitalradio.com/sms-mensajes-texto/>
- [2] IDEC Geoportal, ¿Que son los Web Services? [www.geoportal-idec.net/geoportal/cas/geoserveis/ws-i/](http://www.geoportal-idec.net/geoportal/cas/geoserveis/ws-i/)
- [3] Alierta, César. Movistar. [En línea] Telefónica S.A, 2007. [Citado el: 10 de enero del 2010.] [www.movistar.com.pa/informe2007.pdf](http://www.movistar.com.pa/informe2007.pdf).
- [4] Salazar-Simpson, Luis Alberto. Fundación Orange. [En línea] 2008. [http://www.fundacionorange.es/areas/21\\_funda/funda\\_210.asp](http://www.fundacionorange.es/areas/21_funda/funda_210.asp).
- [5] Zander, Edward. Motorola. [En línea] 2007. [Citado el: 2 de febrero del 2010.] <http://www.motorola.com/>.
- [6] Wesley, Addison. Introducción a los Sistemas de Bases de Datos.
- [7] (PostGre-es.org, 2009)
- [8] Rumbaugh, James, Jacobson, Ivar y Booch, Grady. El lenguaje unificado de modelado. Manual de referencia. 2000.
- [9] Ayuda de la herramienta Case Rational Rose.
- [10] Colectivo de Autores. Visual Paradigm for UML User's Guide.
- [11]Maquina Virtual de Java (JVM) <http://www.dcc.uchile.cl/~rbaeza/cursos/proyarg/lbastias/JVM.html>
- [12]Chopra, Vivek, Li, Sing y Genender, Jeff. Apache Tomcat 6. s.l. : Anaya Multimedia, 2008. ISBN 8441523770.
- [13]Colectivo de Autores. Help - Eclipse SDK. eclipse. [En línea] 2010. <http://help.eclipse.org/ganymede/index.jsp>.
- [14] C. Walls, R. Breidenbach. Spring in Action. s.l. : Manning Publications, 2005.
- [15] Burbeck, Steve. Application programming in Smalltalk-80: How to use Model-View-Controller (MVC). 2007.
- [16] Colectivo de Autores. Página oficial del proyecto Hibernate. [www.hibernate.org](http://www.hibernate.org). [En línea].
- [17] Clinton Begin, Brandon Goodin, Larry Meadors. iBatis in Action. 2007. Manning Publications.
- [18] GWT. [En línea]. <http://code.google.com/webtoolkit>

### **Bibliografía**

- Alierta, César. Movistar. [En línea] Telefónica S.A, 2007. [Citado el: 10 de enero del 2010.] [www.movistar.com.pa/informe2007.pdf](http://www.movistar.com.pa/informe2007.pdf).
- Ayuda de la herramienta Case Rational Rose.
- Burbeck, Steve. Application programming in Smalltalk-80: How to use Model-View-Controller (MVC). 2007.
- Chopra, Vivek, Li, Sing y Genender, Jeff. Apache Tomcat 6. s.l.: Anaya Multimedia, 2008. ISBN 8441523770.
- Colectivo de Autores. Visual Paradigm for UML User's Guide.
- Colectivo de Autores. Help - Eclipse SDK. eclipse. [En línea] 2010.
- C. Walls, R. Breidenbach. Spring in Action. s.l. : Manning Publications, 2005.
- Clinton Begin, Brandon Goodin, Larry Meadors. iBatis in Action. 2007. Manning Publications.
- Clinton Begin, Brandon Goodin, Larry Meadors. IBATIS in Action.
- Gamma, J.M.V.R.H.R.J.E. Design Patterns: Elements of Reusable Object-Oriented Software. 1995.
- Jacobson, Ivar y Booch, Grady, Rumbaugh, James. El Proceso Unificado de Desarrollo de Software. 2000.
- Rumbaugh, James, Jacobson, Ivar y Booch, Grady. El lenguaje unificado de modelado. Manual de referencia. 2000.
- Wesley, Addison. Introducción a los Sistemas de Bases de Datos.

## **Glosario de términos**

**Base de datos:** Un conjunto de información almacenada en memoria auxiliar que permite acceso directo y un conjunto de programas que manipulan esos datos.

**Diagramas:** Facilitan el entendimiento de grandes cantidades de datos y la relación entre diferentes partes de los datos.

**FrameWork:** Es una estructura de soporte definida, en la cual otro proyecto de software puede ser organizado y desarrollado.

**Hibernate:** Framework de capa de persistencia para el lenguaje Java.

**IDE:** Integrated Development Environment. Un entorno de desarrollo integrado es un programa compuesto por un conjunto de herramientas para un programador.

**Java:** Es un lenguaje de programación de alto nivel orientado a objetos.

**Plataforma:** Se refiere al sistema operativo o a sistemas complejos que a su vez sirven para crear programas, como las plataformas de desarrollo.

**Software:** Se refiere al equipamiento lógico o soporte lógico de un computador digital.

**Sistema Operativo:** Es un conjunto de programas de computadora destinados a permitir una administración eficaz de sus recursos.

**Spring:** Framework de la capa de lógica de negocio para Java.

**Tecnologías:** Conjunto de teorías y de técnicas que permiten el aprovechamiento práctico del conocimiento científico.

**Web:** abreviatura de World Wide Web (WWW); interfaz de comunicación en la Internet, que hace uso de enlaces de hipertexto en el interior de una misma página, o entre distintas páginas.