





**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**



FACULTAD DE INGENIERÍA

**ANÁLISIS Y DISEÑO DE UN SISTEMA PARA EL MANEJO DE PUBLICIDAD  
PERSONALIZADA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA  
FACULTAD DE INGENIERÍA  
POR

**RENÉ ELIEZER PALACIOS ROSALES**

ASESORADO POR EL INGENIERO DANIEL CACIÁ RIVAS

AL CONFERÍRSELE EL TÍTULO DE

**INGENIERO EN CIENCIAS Y SISTEMAS**

GUATEMALA, ABRIL DE 2009



UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERÍA



**NÓMINA DE JUNTA DIRECTIVA**

DECANO	Ing. Murphy Olympto Paiz Recinos
VOCAL I	Inga. Glenda Patricia García Soria
VOCAL II	Inga. Alba Maritza Guerrero de López
VOCAL III	Ing. Miguel Ángel Dávila Calderón
VOCAL IV	Br. Milton De León Bran
VOCAL V	Br. Isaac Sultan Mejía
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

**TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO**

DECANO	Ing. Murphy Olympto Paiz Recinos
EXAMINADOR	Ing. Ludwing Altán
EXAMINADOR	Ing. Edgar Estuardo Santos
EXAMINADOR	Inga. Virginia Victoria Tala Ayerdi
SECRETARIA	Inga. Márcia Ivónne Véliz Vargas



**HONORABLE TRIBUNAL EXAMINADOR**

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**Análisis y Diseño de un sistema para el manejo de publicidad personalizada,**

tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, en julio 2006.



René Eliezer Palacios Rosales



Guatemala, 09 de julio de 2008

Ing. Carlos A zurdía  
Escuela de CC y Sistemas  
Facultad de Ingeniería  
USAC  
Presente.

Estimado Ingeniero Carlos A zurdía:

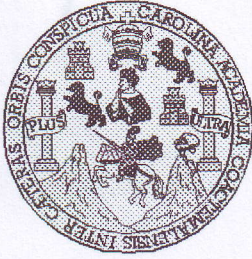
Por medio de la presente le estoy informando que como asesor del trabajo **“Análisis y Diseño de un sistema para el manejo de publicidad personalizada”**, realizado por el estudiante **René Eliezer Palacios Rosales**, identificado con el carné 200010546 he procedido a revisarlo y luego de corroborar que el mismo cumple con los objetivos propuestos al inicio del trabajo, procedo con su aprobación y aceptación de que el trabajo se encuentra finalizado. Aprovecho para informarle que como asesor del trabajo me hago corresponsable del contenido del mismo.

Sin otro particular, me despido,

Atentamente,

  
Daniel Cacia Rivas  
Ingeniero en Ciencias y Sistemas  
Colegiado No. 8882  
Ing. Daniel Cacia Rivas  
Colegiado No. 8882





Universidad San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 08 de Septiembre de 2008

Ingeniero  
**Marlon Antonio Pérez Turk**  
Director de la Escuela de Ingeniería  
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **RENÉ ELIEZER PALACIOS ROSALES** titulado: **"ANÁLISIS Y DISEÑO DE UN SISTEMA PARA EL MANEJO DE PUBLICIDAD PERSONALIZADA"**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,

**Ing. Carlos Alfredo Azurdia**  
Coordinador de Privados  
y Revisión de Trabajos de Graduación





E  
S  
C  
U  
E  
L  
A  
  
D  
E  
  
C  
I  
E  
N  
C  
I  
A  
S  
  
Y  
  
S  
I  
S  
T  
E  
M  
A  
S

UNIVERSIDAD DE SAN CARLOS  
DE GUATEMALA



FACULTAD DE INGENIERÍA  
ESCUELA DE CIENCIAS Y SISTEMAS  
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, de trabajo de graduación titulado "ANÁLISIS Y DISEÑO DE UN SISTEMA PARA EL MANEJO DE PUBLICIDAD PERSONALIZADA" presentado por el estudiante **RENE ELIEZER PALACIOS ROSALES**, aprueba el presente trabajo y solicita la autorización del mismo.*

**"ID Y ENSEÑAD A TODOS"**



*My [Signature]*  
Ing. Marlon Antonio Pérez Turk

Director, Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 27 de abril 2009





El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **ANÁLISIS Y DISEÑO DE UN SISTEMA PARA EL MANEJO DE PUBLICIDAD PERSONALIZADA**, presentado por el estudiante universitario **René Eliezer Palacios Rosales**, procede a la autorización para la impresión del mismo.

IMPRÍMASE.

Ing. Murphy Olimpo Paiz Recinos  
DECANO



Guatemala, abril de 2009

/gdech

## **ACTO QUE DEDICO A:**

<b>Dios</b>	Por ser el ser Supremo que me permite vivir y compartir con las personas que más quiero.
<b>Mis padres José Mamerto Palacios Rodas y Martha Lilia Rosales</b>	Por todos sus sacrificios, consejos y cariño a lo largo de mi vida, este logro no pudo ser posible sin sus enseñanzas.
<b>La familia Catalán Gudiel</b>	Por ser una familia que me brindo su apoyo y amistad en todos mis años de estudio y que se han vuelto parte de mi familia.
<b>La familia Salazar de León</b>	Por convertirme en parte de su familia y abrirme las puertas de su casa.
<b>Mis primos y tíos de las familias Palacios , Palencia y Rosales.</b>	Por aconsejarme y permitirme compartir buenos momentos junto a ustedes.
<b>Mis amigos y amigas de la infancia y la vida, universidad y del trabajo</b>	Por brindarme su amistad, apoyo y poder compartir parte de mi vida con ellos.
<b>Mis amigos de la Escuela de Ciencias y Sistemas</b>	Por haber compartido una época especial, en las buenas y en las malas para poder lograr nuestra meta y esperando que podamos continuar compartiendo.

**Mis amigos que ya no se encuentran** Por brindarme su amistad, nunca los  
**en este mundo** olvidare.





## ÍNDICE GENERAL

<b>ÍNDICE DE ILUSTRACIONES .....</b>	<b>XI</b>
<b>GLOSARIO.....</b>	<b>XV</b>
<b>RESUMEN.....</b>	<b>XXI</b>
<b>OBJETIVOS .....</b>	<b>XXIII</b>
<b>INTRODUCCIÓN.....</b>	<b>XXV</b>
<b>1. EVOLUCIÓN DEL MARKETING.....</b>	<b>1</b>
1.1. Comunicaciones de Marketing .....	2
1.2. Marketing directo .....	2
1.3. Marketing electrónico .....	4
1.4. Las 4 F's del Marketing Online .....	4
1.5. Relaciones públicas.....	5
<b>2. OBJETIVOS AL DISEÑAR UN SISTEMA DE INFORMACIÓN.....</b>	<b>7</b>
2.1. Especificación de los elementos lógicos del diseño .....	7
2.2. Apoyo para las actividades de la empresa .....	9
2.3. Asegurar que las características del sistema cumplan con los requerimientos del usuario .....	10
2.4. Proporcionar un sistema que sea fácil de utilizar .....	11
2.4.1. Ingeniería humana .....	11
2.5. Diseño ergonómico.....	12
2.6. Proporcionar especificaciones detalladas para el desarrollo de software.....	13
2.7. Ajustarse a los estándares de diseño.....	13
<b>3. CARACTERÍSTICAS QUE SE DEBEN DISEÑAR .....</b>	<b>17</b>
3.1. Elementos de diseño .....	17
3.2. Diseño de salida.....	19
3.3. Diseño de archivos.....	20
3.4. Diseño de interacciones con la base de datos .....	21

3.5.	Diseño de la entrada .....	22
3.6.	Diseño de controles.....	23
3.7.	Diseño de procedimientos .....	24
3.8.	Diseño de especificaciones para programas.....	25
<b>4.</b>	<b>PROCESO DE DESARROLLO.....</b>	<b>27</b>
4.1.	Técnicas de desarrollo .....	27
4.2.	Casos de uso .....	28
4.2.1.	Descripción.....	28
4.2.1.1	Especificación de un caso de uso.....	29
4.2.1.2	Diagrama de casos de uso .....	30
4.2.2.	Notación .....	32
4.2.2.1	Caso de uso.....	32
4.2.2.2	Actor .....	33
4.2.2.3	Relación.....	33
4.2.2.4	Paquete .....	34
4.2.2.5	Ejemplo.....	35
4.3.	Diagrama de clases.....	35
4.3.1.	Descripción.....	36
4.3.1.1	Clases.....	36
4.3.1.2	Relaciones.....	38
4.3.1.3	Interfaces.....	40
4.3.1.4	Paquetes.....	40
4.3.2.	Notación .....	40
4.3.2.1	Clases.....	40
4.3.2.2	Relaciones.....	42
4.3.2.3	Interfaces.....	45
4.3.2.4	Paquetes.....	46
4.3.2.5	Ejemplo.....	46
4.4.	Diagrama de componentes .....	47

4.4.1.	Descripción.....	47
4.4.2.	Notación .....	48
4.4.2.1	Componente .....	48
4.4.2.2	Interfaz.....	49
4.4.2.3	Paquete .....	49
4.4.2.4	Relación de dependencia .....	49
4.4.2.5	Ejemplo.....	50
4.5.	Diagrama de despliegue.....	50
4.5.1.	Descripción.....	51
4.5.2.	Notación .....	52
4.5.2.1	Nodo .....	52
4.5.2.2	Conexión.....	52
4.5.2.3	Ejemplo.....	52
4.6.	Diagrama de interacción.....	53
4.6.1.	Descripción.....	53
4.7.	Diagrama de secuencia.....	55
4.7.1.	Descripción.....	55
4.7.2.	Notación .....	56
4.7.2.1	Objeto y línea de vida .....	56
4.7.2.2	Foco de control o activación .....	56
4.7.2.3	Mensaje .....	57
4.7.2.4	Ejemplo.....	58
4.8.	Diagrama de paquetes .....	59
4.8.1.	Descripción.....	59
4.8.2.	Notación .....	60
4.8.2.1	Paquete .....	60
4.8.2.2	Dependencia.....	60
4.8.2.3	Ejemplo.....	60
<b>5.</b>	<b>ESPECIFICACIÓN DE CASOS DE USO .....</b>	<b>63</b>

5.1.	Propósito .....	63
5.2.	Actualizar información de cliente .....	63
5.2.1.	Descripción.....	63
5.2.2.	Flujo básico .....	64
5.2.3.	Flujo alternativo .....	64
5.2.3.1	Tipo de datos erróneo.....	64
5.2.4.	Precondiciones.....	64
5.2.5.	Realización del caso de uso.....	65
5.3.	Agregar preferencias .....	66
5.3.1.	Descripción.....	66
5.3.2.	Flujo básico .....	66
5.3.3.	Flujo alternativo .....	67
5.3.3.1	Preferencia existe .....	67
5.3.4.	Precondiciones.....	67
5.3.5.	Realización del caso de uso.....	68
5.4.	Agregar premio.....	68
5.4.1.	Descripción.....	69
5.4.2.	Flujo básico .....	69
5.4.3.	Precondiciones.....	69
5.4.4.	Realización del caso de uso.....	70
5.5.	Agregar mensaje de publicidad .....	70
5.5.1.	Descripción.....	71
5.5.2.	Flujo básico .....	71
5.5.3.	Flujo alternativo .....	71
5.5.3.1	Mensaje excede tamaño mínimo .....	71
5.5.4.	Precondiciones.....	71
5.5.5.	Realización del caso de uso.....	72
5.6.	Cargar backup.....	72
5.6.1.	Descripción.....	73

5.6.2.	Flujo básico .....	73
5.6.3.	Flujo alternativo .....	73
5.6.3.1	Archivo no existe .....	73
5.6.3.2	Archivo corrupto.....	73
5.6.4.	Precondiciones.....	74
5.6.5.	Realización del caso de uso.....	74
5.7.	Crear usuarios.....	75
5.7.1.	Descripción.....	75
5.7.2.	Flujo básico .....	75
5.7.3.	Flujo alternativo .....	75
5.7.3.1	Usuario, password ya existe .....	75
5.7.4.	Precondiciones.....	76
5.7.5.	Realización del caso de uso.....	76
5.8.	Enviar información a clientes.....	77
5.8.1.	Descripción.....	77
5.8.2.	Flujo básico .....	77
5.8.3.	Flujo alternativo .....	78
5.8.3.1	El sistema de la empresa de comunicación esta fuera de ..... línea.....	78
5.8.4.	Precondiciones.....	78
5.8.5.	Realización del caso de uso.....	79
5.9.	Enviar resultado de encuestas .....	79
5.9.1.	Descripción.....	80
5.9.2.	Flujo básico .....	80
5.9.3.	Precondiciones.....	80
5.9.4.	Realización del caso de uso.....	81
5.10.	Logueo de usuario.....	82
5.10.1.	Descripción.....	82
5.10.2.	Flujo básico .....	82

5.10.3.	Flujo alternativo .....	82
5.10.3.1	Usuario no existe .....	82
5.10.4.	Precondiciones.....	83
5.10.5.	Realización del caso de uso.....	83
5.11.	Modificar mensaje de publicidad .....	84
5.11.1.	Descripción.....	84
5.11.2.	Flujo básico .....	84
5.11.3.	Flujo alternativo .....	85
5.11.3.1	Mensaje excede tamaño mínimo .....	85
5.11.4.	Precondiciones.....	85
5.11.5.	Realización del caso de uso.....	86
5.12.	Modificar preferencias .....	86
5.12.1.	Descripción.....	87
5.12.2.	Flujo básico .....	87
5.12.3.	Flujo alternativo .....	87
5.12.4.	Preferencia existe.....	87
5.12.5.	Precondiciones.....	87
5.12.6.	Realización del caso de uso.....	87
5.13.	Modificar premio.....	88
5.13.1.	Descripción.....	89
5.13.2.	Flujo básico .....	89
5.13.3.	Precondiciones.....	89
5.13.4.	Realización del caso de uso.....	90
5.14.	Modificar usuarios .....	91
5.14.1.	Descripción.....	91
5.14.2.	Flujo básico .....	91
5.14.3.	Precondiciones.....	91
5.14.4.	Realización del caso de uso.....	92
5.15.	Realizar backup.....	93



5.15.1.	Descripción.....	93
5.15.2.	Flujo básico .....	93
5.15.3.	Precondiciones.....	93
5.15.4.	Realización del caso de uso.....	94
5.16.	Llenar encuesta.....	95
5.16.1.	Descripción.....	95
5.16.2.	Flujo básico .....	95
5.16.3.	Precondiciones.....	95
5.16.4.	Realización del caso de uso.....	96
5.17.	Realizar sorteo .....	96
5.17.1.	Descripción.....	97
5.17.2.	Flujo básico .....	97
5.17.3.	Flujo Alternativo.....	97
5.17.3.1	Premio no existe .....	97
5.17.3.2	El sistema de la empresa de comunicación esta fuera de .....	
	línea .....	97
5.17.4.	Precondiciones.....	98
5.17.5.	Realización del caso de uso.....	98
5.18.	Recibir comentario/queja.....	99
5.18.1.	Descripción.....	99
5.18.2.	Flujo básico .....	99
5.18.3.	Precondiciones.....	99
5.18.4.	Realización del caso de uso.....	100
5.19.	Registrar datos de cliente.....	101
5.19.1.	Descripción.....	101
5.19.2.	Flujo básico .....	101
5.19.3.	Flujo alternativo .....	101
5.19.3.1	Usuario, password ya existe .....	101
5.19.4.	Precondiciones.....	102

5.19.5.	Realización del caso de uso.....	102
5.20.	Ver comentarios de clientes .....	103
5.20.1.	Descripción.....	103
5.20.2.	Flujo básico .....	103
5.20.3.	Precondiciones.....	103
5.20.4.	Realización del caso de uso.....	104
5.21.	Ver mensajes de publicidad .....	104
5.21.1.	Descripción.....	105
5.21.2.	Flujo básico .....	105
5.21.3.	Precondiciones.....	105
5.21.4.	Realización del caso de uso.....	105
5.22.	Analizar preferencias.....	106
5.22.1.	Descripción.....	106
5.22.2.	Flujo básico .....	106
5.22.3.	Precondiciones.....	106
5.22.4.	Realización del caso de uso.....	107
<b>6.</b>	<b>ARQUITECTURA DE SOFTWARE.....</b>	<b>109</b>
6.1.	Propósito .....	109
6.1.1.	Alcance.....	109
6.1.2.	Definiciones, acrónimos y abreviaturas .....	109
6.2.	Representación de la arquitectura.....	110
6.2.1.	Vista de casos de uso .....	112
6.2.1.1	Realizar sorteo.....	114
6.2.1.2	Enviar información .....	115
6.2.1.3	Realizar backup .....	116
6.2.1.4	Cargar backup .....	116
6.2.2.	Vista lógica .....	117
6.2.2.1	Vista global .....	117
6.2.2.1.1	Capa de presentación .....	119

6.2.2.1.1.1	Subcapa aplicación.....	119
6.2.2.1.1.2	Subcapa ajax.....	122
6.2.2.1.2	Reglas del negocio.....	122
6.2.2.1.3	Capa de datos.....	124
6.2.2.2	Realización de casos de uso.....	125
6.2.2.2.1	Realizar sorteo.....	125
6.2.2.2.2	Enviar información.....	126
6.2.2.2.3	Realizar backup.....	127
6.2.2.2.4	Cargar backup.....	128
6.2.3.	Vista de procesos.....	128
6.2.3.1	Realizar sorteo.....	128
6.2.3.2	Enviar información.....	129
6.2.3.3	Realizar backup.....	130
6.2.3.4	Cargar backup.....	131
6.2.4.	Vista de componentes.....	132
6.2.4.1	Realizar sorteo.....	132
6.2.4.2	Enviar información.....	133
6.2.4.3	Realizar backup.....	134
6.2.4.4	Cargar backup.....	134
6.2.5.	Vista de despliegue.....	135
6.2.5.1	Servidor de aplicaciones.....	135
6.2.5.1.1	Publinnovate.....	135
6.2.5.1.2	Manejador de base de datos MySql.....	136
6.2.5.2	Cliente.....	136
	<b>CONCLUSIONES.....</b>	<b>137</b>
	<b>RECOMENDACIONES.....</b>	<b>139</b>
	<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>141</b>
	<b>BIBLIOGRAFÍA.....</b>	<b>143</b>



# ÍNDICE DE ILUSTRACIONES

## FIGURAS

1 . Ejemplo diagrama de casos de uso.....	31
2 . Ejemplo de una relación extiende.....	32
3 . Ejemplo de cómo se representa un caso de uso.....	32
4 . Ejemplo de cómo se representa un actor. ....	33
5 . Ejemplo de un diagrama de casos de uso con relaciones. ....	34
6 . Ejemplo de cómo se representa un paquete.....	34
7 . Ejemplo de un caso de uso de una biblioteca. ....	35
8 . Ejemplo de cómo se representa una clase. ....	42
9 . Ejemplo de una relación entre clases.....	43
10 . Ejemplo de tipos de relaciones entre clases. ....	44
11 . Ejemplo de cómo se representa una interfaz.....	46
12 . Ejemplo de un diagrama de clases de una biblioteca.....	47
13 . Ejemplo de cómo se representa un componente.....	48
14 . Ejemplo de cómo se presersenta una interfaz de un componente.....	49
15 . Ejemplo de un diagrama de componentes de una biblioteca.....	50
16 . Ejemplo de un diagrama de despliegue. ....	53
17 . Ejemplo de un diagrama de secuencia.....	57
18 . Ejemplo de un diagrama de secuencia para una biblioteca. ....	58
19 . Ejemplo de un diagrama de paquetes ....	61
20 . Diagrama de caso de uso, actualizar informacion de cliente.....	63
21 . Realizacion del caso de uso, actualizar informacion de cliente.....	65
22 . Diagrama de caso de uso, agregar preferencias.....	66
23 . Realizacion de caso de uso, agregar preferencias.....	68

24 . Diagrama de caso de uso, agregar premio. ....	68
25 . Realizacion de caso de uso, agregar premio.....	70
26 . Diagrama de caso de uso, agregar mensaje publicidad. ....	70
27 . Realizacion de caso de uso, agregar mensaje publicidad.....	72
28 . Diagrama de caso de uso, cargar backup.....	72
29 . Realizacion de caso de uso, cargar backup. ....	74
30 . Diagrama de caso de uso, crear usuario.....	75
31 . Diagrama de realización de caso de uso, crear usuario. ....	76
32 . Diagrama de caso de uso, enviar informacion a clientes. ....	77
33 . Diagrama de realización de caso de uso, enviar información a clientes.....	79
34 . Diagrama de caso de uso, enviar resultado de encuesta. ....	79
35 . Diagrama de realización de caso de uso, enviar resultado de encuesta.....	81
36 . Diagrama de caso de uso, logueo de usuario.....	82
37 . Diagrama de realización de caso de uso, logueo de usuario. ....	83
38 . Diagrama de caso de uso, modificar mensaje de publicidad. ....	84
39 . Diagrama de realización de caso de uso, modificar mensaje de publicidad. ....	86
40 . Diagrama de caso de uso, modificar preferencias.....	86
41 . Diagrama de realización de caso de uso, modificar preferencias. ....	88
42 . Diagrama de caso de uso, modificar premio. ....	88
43 . Diagrama de realización de caso de uso, modificar premio.....	90
44 . Diagrama de caso de uso, modificar usuarios. ....	91
45 . Diagrama de realización de caso de uso, modificar usuarios.....	92
46 . Diagrama de caso de uso, realizar backup. ....	93
48 . Diagrama de caso de uso, llenar encuesta. ....	95
49 . Diagrama de realización de caso de uso, realizar encuesta.....	96
50 . Diagrama de caso de uso, realizar sorteo.....	96



51 . Diagrama de realización de caso de uso, realizar sorteo. ....	98
52 . Diagrama de caso de uso, recibir comentario/queja.....	99
53 . Diagrama de realización de caso de uso, recibir comentario/queja. ....	100
54 . Diagrama de caso de uso, registrar datos de cliente.....	101
55 . Diagrama de realización de caso de uso, registrar datos de cliente..	102
56 . Diagrama de caso de uso, ver comentarios de cliente. ....	103
57. Diagrama de realización de caso de uso, ver comentarios de cliente.....	104
58 . Diagrama de caso de uso, ver mensajes de publicidad.....	104
59 . Diagrama de realización de caso de uso, ver mensajes publicidad. ....	105
60 . Diagrama de caso de uso, analizar preferencias.....	106
61 . Diagrama de realización de caso de uso, analizar preferencias. ....	107
62. Vistas para representar una arquitectura.....	110
63 . Caso de uso, realización de sorteo.....	114
64 . Diseño de paquetes significativos de la arquitectura. ....	118
65. Diagrama con las clases del paquete aplicacion.....	119
66 . Diagrama con las clases del paquete ajax. ....	122
67 .Diagrama con las clases del paquete reglas del negocio.....	123
68 . Diagrama con las clases del paquete datos.....	124
69 . Diagrama de realización de casos de uso, realizar sorteo. ....	125
70 . Diagrama realización de casos de uso, enviar información.....	126
71 . Diagrama realización de casos de uso, enviar backup.....	127
72 . Diagrama realización de casos de uso, cargar backup. ....	128
73 . Diagrama de procesos, realizar sorteo.....	129
74 . Diagrama de procesos, enviar información. ....	130
75 . Diagrama de procesos, realizar backup. ....	131
76 . Diagrama de procesos, cargar backup.....	132
77 . Diagrama de comonentes, realizar sorteo. ....	132
78 . Diagrama de componentes, enviar información.....	133

<b>79 . Diagrama de componentes, realizar backup.....</b>	<b>134</b>
<b>80 . Diagrama de componentes, cargar backup. ....</b>	<b>134</b>
<b>81 . Diagrama de despliegue sistema. ....</b>	<b>135</b>

## GLOSARIO

<b>PYMES</b>	Abreviatura de pequeñas y medianas empresas, es una clasificación dada un determinado grupo de empresas que se caracteriza por el tamaño en número de trabajadores y ventas anuales, estas varían de país en país.
<b>TICs</b>	Son aquellos dispositivos que capturan, transmiten y despliegan datos e información electrónica y que apoyan el crecimiento y desarrollo económico de la industria manufacturera y de servicios.
<b>PASSWORD</b>	Una <b>contraseña o password</b> es una serie secreta de caracteres que permite a un usuario tener acceso a un archivo, a un ordenador, o a un programa. En sistemas multiusos, cada usuario debe incorporar su contraseña antes que el ordenador responda a los comandos.
<b>SPAM</b>	El <b>spamming</b> es el hecho de enviar mensajes electrónicos ( <b>spam</b> ) (habitualmente de tipo comercial) no solicitados y en cantidades masivas.
<b>SOFTWARE</b>	La parte "que no se puede tocar" de un ordenador: los programas y los datos.
<b>LENGUAJES DE PROGRAMACIÓN</b>	Lenguaje artificial que puede ser usado para controlar el comportamiento de una máquina, especialmente una

	<p>computadora. Estos se componen de un conjunto de reglas sintácticas y semánticas que permiten expresar instrucciones que luego serán interpretadas. Debe distinguirse de "lenguaje informático", que es una definición más amplia, puesto estos incluyen otros lenguajes como son el HTML o PDF que dan formato a un texto y no es programación en sí misma.</p>
<b>ESTÁNDARES</b>	<p>Es cualquier tecnología aprobada por un comité formalizado.</p>
<b>BASE DE DATOS</b>	<p>Aplicación informática para manejar información en forma de "fichas": clientes, artículos, películas, etc. La mayoría de las bases de datos actuales permiten hacer listados, consultas, crear pantallas de visualización de datos, controlar el acceso de los usuarios, etc. También es cada vez más frecuente que las consultas se puedan hacer en un lenguaje estándar conocido como SQL.</p>
<b>PRECONDICIONES</b>	<p>Condiciones que se deben dar antes de la realización de un evento.</p>
<b>POSTCONDICIONES</b>	<p>Condiciones que se dan después de que se realice un evento.</p>
<b>CLASE</b>	<p>Abstracciones de objetos. Una clase es una definición de un objeto. Un objeto es una instancia de una clase.</p>
<b>OBJETO</b>	<p>En programación orientado a objetos, representación detallada y particular de algo de la realidad. Todo objeto tiene un identidad o nombre, estado</p>

(características definidas generalmente en variables) y comportamiento (sus funciones o procedimientos).

**HARDWARE**

La parte "que se puede tocar" de un ordenador: caja (y todo su contenido), teclado, pantalla, etc.

**UML (UNIFIED  
MODELING  
LANGUAJE)**

Es un conjunto de herramientas, que permite modelar (analizar y diseñar) sistemas orientados a objetos.

**PHP**

PHP es un lenguaje de programación muy potente que, junto con html, permite crear sitios web dinámicos. Php se instala en el servidor y funciona con versiones de Apache, Microsoft IIs, Netscape Enterprise Server y otros.

La forma de usar php es insertando código php dentro del código html de un sitio web. Cuando un cliente (cualquier persona en la web) visita la página web que contiene éste código, el servidor lo ejecuta y el cliente sólo recibe el resultado. Su ejecución es por tanto en el servidor, a diferencia de otros lenguajes de programación que se ejecutan en el navegador.

**MYSQL**

Es un sistema de gestión de bases de datos (SGBD) multiusuario, multiplataforma y de código abierto.

**MULTIUSUARIO**

Cualquier hardware o software que tiene la capacidad de soportar múltiples usuarios. Su antónimo es mono usuario.

**MULTIPLATAFORMA**

Cross-platform en inglés. Que tiene la capacidad de soportar múltiples plataformas.

<b>PLATAFORMAS</b>	(platform). En informática, determinado software y/o hardware con el cual una aplicación es compatible y permite ejecutarla.
<b>CÓDIGO ABIERTO</b>	(Open source). Denominación para aquellas aplicaciones que tienen su código fuente liberado. En general, los programa de código abierto suele ser libres. Aunque existen aplicaciones de código abierto que no son libres.
<b>CÓDIGO FUENTE</b>	(Source code, code base). Texto escrito en un lenguaje de programación específico y que puede ser leído por un programador. Debe traducirse a lenguaje máquina para que pueda ser ejecutado por la computadora o a bytecode para que pueda ser ejecutado por un intérprete.
<b>SERVIDOR WEB APACHE</b>	(Acrónimo de "a patchy server"). Servidor web de distribución libre y de código abierto, siendo el más popular del mundo desde abril de 1996, con una penetración actual del 50% del total de servidores web del mundo (agosto de 2007).
<b>NAVEGADOR WEB</b>	Los buscadores exploradores o navegadores (diferentes de los motores de búsqueda) son programas (software), generalmente gratuitos, que instalados en el ordenador permiten ver documentos almacenados en el disco duro, disquete, etc, o a través de Internet, acceder a documentos alojados en servidores web.

**SISTEMA OPERATIVO LINUX** Versión de libre distribución (gratis) del sistema operativo Desarrollada por Linus Torvalds, con contribuciones de programadores de todo el mundo.

**BACKUP** (Copia de seguridad) Es la copia total o parcial de información importante del disco duro, CDs, bases de datos u otro medio de almacenamiento. Esta copia de respaldo debe ser guardada en algún otro sistema de almacenamiento masivo, como ser discos duros, CDs, DVDs o cintas magnéticas.

**AJAX** Siglas de Asynchronous JavaScript and XML, es un término que describe un nuevo acercamiento a usar un conjunto de tecnologías existentes juntas, incluyendo las siguientes: HTML o XHTML, hojas de estilo (Cascading Style Sheets o css), Javascript, el DOM (Document Object Model), XML, XSLT, y el objeto XMLHttpRequest.





## RESUMEN

Un paso muy importante al crear un sistema es crear la documentación adecuada antes del proceso de desarrollo para que este se pueda realizar de una manera adecuada y el sistema desarrollado represente lo que realmente se desea y esto evite la realización de sistemas inadecuados lo que nos conlleva la pérdida de tiempo lo cual es pérdida de recursos que podrían ser aprovechados para otras actividades.

El presente trabajo nos presenta la información necesaria para poder llegar a comprender la fase de análisis y diseño de un sistema definiéndonos el significado de los diagramas utilizados y mostrando ejemplos de estos diagramas, nos describe porque es importante realizar la fase de análisis y diseño que puntos son importantes a tomar en cuenta y que es lo que se debe diseñar. Se presenta el análisis y diseño de un sistema que ayuda a las empresas a personalizar su publicidad y poder tener una mejor comunicación con sus clientes y este se presenta por medio de la utilización de diagramas de uso de las procesos que se realizan, y se define la arquitectura del sistema por medio de las 5 vistas las cuales son: Casos de uso, lógica, procesos, producción y implementación.



## **OBJETIVOS**

### **GENERAL**

Contar con un documento el cual pueda definir un sistema que pueda ser implementado para ayuda de la pequeñas y medianas empresas, y que a la vez pueda llegar a ser comprendido por personas que no tengan conocimiento sobre el análisis y diseño de sistemas de información y puedan comprender las fases que se realizan antes del desarrollo de un sistema.

### **ESPECÍFICOS:**

1. Utilizar un lenguaje de modelado como lo es UML para poder definir un sistema por medio de los diferentes diagramas, los cuales representan diferentes vistas del sistema.
2. Contar con una base teórica sobre el significado de los diagramas utilizados, para que puedan ser comprendidos por personas que no conozcan su funcionamiento.
3. Presentar una idea de cómo puede ser un sistema que pueda ser utilizado por pequeñas y medianas empresas para mejorar su marketing para que otras personas puedan tomarlas como base y continuar con el trabajo y así se pueda llevar a cabo el desarrollo e implementación de un sistema funcional.



## INTRODUCCIÓN

Ocho de cada 10 pequeñas y medianas empresas (Pymes) en Guatemala tienen por lo menos una computadora para hacer mejor sus negocios, según un estudio del Banco Interamericano de Desarrollo (BID) [1].

Actualmente, el uso de tecnología en los procesos internos de las empresas se hace cada vez más necesario para manejar la información de una manera más rápida y eficiente para poder tenerla disponible rápidamente cuando se considere necesario para la toma de decisiones basados en datos reales y actuales. La toma de decisiones basadas en información confiable es clave para una empresa porque las Pymes no cuentan con los recursos para realizar operaciones riesgosas en las que puedan perder capital o recursos.

En Guatemala el número de pequeñas y medianas empresas es muy alto, debido a las condiciones económicas del país. Se estima que en Guatemala hay entre 20 mil y 30 mil Pymes, según estadísticas oficiales [1], El uso de tecnología es importante porque ayuda a poder alcanzar mercados que antes no se podrían alcanzar por eso el uso de sistemas que les permitan poder conectarse al mundo por medio de internet es necesario para que se den a conocer y por esto es necesario que se diseñen sistemas para que ellos puedan utilizarlos. La habilidad de las TICs para poner en contacto a compradores y vendedores en forma más eficiente (combinada con el almacenamiento y procesamiento de información), representa una manera de obtener importantes ganancias potenciales, ya que reduce los costos de

búsqueda, mejora la identificación (matching) entre vendedores y compradores, e inclusive, facilita la creación de nuevos mercados[2].

Una pequeña o mediana empresa puede tener un excelente producto o servicio que cuente con ventajas comparativas envidiables, pero si no desarrolla una estrategia de búsqueda de mercado y comercialización ordenada y profesional sus posibilidades de desarrollar operaciones internacionales se verán reducidas [3]. Por eso es importante no solo tener procesos internos organizados para una óptima producción, una parte muy importante es cómo comercializamos los productos, otra es conocer lo que el cliente desea sus preferencias para presentarle los productos que más necesite.

# 1. EVOLUCIÓN DEL MARKETING

El marketing tal como lo conocemos hoy, comenzó en la década de 1970 con el nacimiento de la "orientación al marketing". Durante la primera etapa del capitalismo, la empresa tenía una orientación a la producción. La empresa se preocupaba de asuntos relativos a la producción, la fabricación, y la eficacia. A mediados de la década de 1950, emergió una segunda etapa, la etapa de la orientación a las ventas. La primera preocupación de la empresa se convirtió en vender lo que producía. A principios de la década de 1970, emergió una tercera etapa, la de orientación al marketing cuando las empresas se dieron cuenta de que los deseos y necesidades del consumidor conducían todo el proceso. La investigación de mercado llegó entonces a ser importante. Los negocios se dieron cuenta que era inútil invertir un gran esfuerzo en la producción y venta de productos que la gente no deseaba. Algunos especialistas dicen que ahora estamos al borde de una cuarta etapa, la de orientación al marketing personal. Creen que hoy en día existe la tecnología disponible para comercializar productos sobre una base individual (el marketing personalizado, el marketing de permiso, y la customización). Sienten que no es necesario pensar más en términos agregados como segmentos de mercado o mercados objetivos. [4]

## 1.1. Comunicaciones de Marketing

Comunicaciones de marketing es parte de un área de investigación de marketing más amplia, especializada en la investigación de las actividades promocionales y de promoción, **especialmente publicidad, promoción de ventas y relaciones públicas.**

Está genéricamente aceptado por los prácticos de la industria que las Comunicaciones Integradas de Marketing son el objetivo ideal de la mayoría de las organizaciones.

En el Reino Unido, las Universidades de Bournemouth y de Middlesex ofrecen cursos especializados en Comunicaciones de Marketing en los niveles de grado y postgrado. [5]

## 1.2. Marketing directo

El marketing directo es un sistema interactivo que utiliza uno o más medios de comunicación para obtener una respuesta medible en un público objetivo.

La forma más común de marketing directo es el mailing por el que los responsables de marketing envían sus mensajes a los consumidores de una determinada área, generalmente, extraídos de una base de datos. Una variedad del mailing lo constituiría el buzoneo que consiste en la introducción de folletos y otros elementos publicitarios directamente en los buzones de los edificios. El segundo método más común de marketing directo es el telemarketing por el que las compañías llaman a números de teléfono que han sido previamente



seleccionados o bien al azar. También es posible realizar envíos de marketing directo a través de Internet -e-mailings-, que, cuando se desarrolla de manera maliciosa, es comúnmente conocido como spam. Por último, existe un cuarto método que consistiría en el envío masivo de faxes, si bien es menos común.

El marketing directo difiere de los métodos habituales de publicidad en que no utiliza un medio de comunicación intermedio o se expone en público, como por ejemplo, en el punto de venta. Por el contrario, se envía directamente al consumidor. Algunas de las ventajas que se predicen del marketing directo son:

- Es un método rápido y económico de llegar al consumidor.
- Teóricamente, se dirige directamente a los potenciales clientes de un producto o servicio por lo que su efectividad es mayor que otros medios masivos.

La mayor parte del marketing directo es realizado por compañías cuya única función es diseñar y ejecutar este tipo de publicidad. Generalmente, utilizan bases de datos de consumidores y muy a menudo manejan criterios muy sofisticados para incluirlos o eliminarlos de sus listas de marketing.

Al tratarse de una comunicación interactiva entre la empresa y el cliente, su uso como herramienta de marketing y comunicación se ve fuertemente condicionado por las legislaciones existentes en cada país sobre el uso de datos de carácter personal. De ahí que en sociedades no excesivamente restrictivas con el uso de la información de carácter personal (EE.UU) tenga más eficacia que en otras economías más celosas del uso de este tipo de datos (Alemania, España, etc.)

En estos países tan celosos de la intimidad de sus ciudadanos sólo se puede producir una comunicación interactiva con un cliente si sus datos personales (nombre, dirección postal, teléfono) se encuentran en un directorio público (guía de teléfonos u otros directorios públicos) o se ha recabado de ellos un consentimiento expreso para ese tipo de comunicación. [6]

### 1.3. Marketing electrónico

El marketing electrónico es el paradigma que muestra que las compañías pueden utilizar medios electrónicos para conducirse exitosamente, es decir por el Internet y cuyas ventajas son que se pueden comprar las 24 horas del día, se evitan los viajes locales o supermercados, se pueden comparar artículos y precios competitivos más cómodamente y se puede dialogar con los vendedores más fácilmente. **También pueden dirigirse al mercado global y no sólo al local, realizar encuestas, enviar mensajes y recibir sugerencias.**

Es uno de los cuatro paradigmas de marketing, según Phillip Kotler, que una empresa debe elegir como base para la aplicación de una estrategia. [7]

### 1.4. Las 4 F's del Marketing Online

Como comenta Paul Fleming en "Hablemos de la Mercadotecnia Interactiva", las 4 F's de la mercadotecnia en internet serían:

- Flujo: Según Fleming, flujo es “estado mental en que entra un usuario de Internet al sumergirse en una web que le ofrece una experiencia llena de interactividad y valor añadido”
- Funcionalidad: Si el cliente ha entrado en estado de flujo, está en camino de ser captado, pero para que el flujo de la relación no se rompa, queda dotar a la presencia on-line de funcionalidad, es decir, construir páginas teniendo en cuenta las limitaciones de la tecnología. Se refiere a una home page atractiva, navegación clara y útil para el usuario.
- Feedback: La relación se ha comenzado a construir. El usuario está en estado de flujo y además no se exaspera en su navegación. Ha llegado el momento de seguir dialogando y sacar partido de la información a través del conocimiento del usuario. Internet da la oportunidad de preguntar al cliente qué le gusta, qué mejorar. En definitiva, dialogar con el cliente para conocerlo mejor y construir una relación basada en sus necesidades, personalizar después de cada contacto.
- Fidelización: Internet ofrece la creación de comunidades de usuarios que aporten contenidos de manera que se establezca un diálogo personalizado con los clientes, quienes podrán ser así más fieles. [8]

## **1.5. Relaciones públicas**

Se llama relaciones públicas al arte, técnica y ciencia de gestionar la comunicación entre una organización y público clave para construir, administrar y mantener su imagen positiva. Es una disciplina planificada y deliberada que

se lleva a cabo de modo estratégico. Tiene la característica de ser una forma de comunicación bidireccional, puesto que no sólo se dirige a su público (tanto interno como externo) sino que también lo escucha y atiende sus necesidades, favoreciendo así la mutua comprensión entre la organización y su público.[9]

## **2. OBJETIVOS AL DISEÑAR UN SISTEMA DE INFORMACIÓN**

Los requerimientos se trasladan en especificaciones de diseño. ¿Qué pretende alcanzar el analista cuando diseña un sistema de información? Esta sección describe los objetivos del diseño de sistemas y explora la naturaleza del diseño así como los resultados explícitos esperados de esta actividad.

### **2.1. Especificación de los elementos lógicos del diseño**

El diseño de sistemas tiene dos etapas: el diseño lógico y la construcción física del sistema. Cuando el analista formula el diseño lógico, escribe las especificaciones detalladas del nuevo sistema, es decir aquellas que describen sus características: salidas, entradas, archivos bases de datos y los procedimientos, todo en una forma que satisfaga los requerimientos del proyecto. El conjunto formado por todas estas características recibe el nombre de especificaciones del diseño del sistema.

El diseño lógico de un sistema de información es similar al proyecto de ingeniería de un automóvil: muestra las características más sobresalientes (como el motor, la transmisión y el espacio para los pasajeros) y la relación que guardan entre si (donde se conectan los componentes unos a otros o cual es la separación que existe entre las puertas). Los reportes y salidas generadas por el analista son similares a los componentes de diseño del ingeniero. Los procedimientos y datos se enlazan entre si para producir un sistema que trabaja.

Al diseñar un sistema de inventarios, por ejemplo, las especificaciones del mismo incluyen definiciones de reportes y pantallas de presentación que describen las existencias disponibles, el abastecimiento y retiro de artículos, y el resumen de transacciones realizadas durante, por ejemplo, un mes de operación. El diseño lógico también especifica los formatos de entrada y la distribución de la salida en pantalla por todas las transacciones y archivos que son necesarios para dar mantenimiento a los datos de los proveedores. Las especificaciones de procedimientos describen los métodos utilizados para ingresar datos en el sistema, copiar archivos y detectar problemas, si estos se presentaran.

La construcción física, que es la siguiente actividad después del diseño lógico, produce el software, los archivos y un sistema que funciona. Las especificaciones de diseño indican a los programadores lo que el sistema debe hacer. A su vez, los programadores escriben programas que aceptan la entrada proporcionada por los usuarios, procesan los datos, producen los reportes y guardan los datos en los archivos.

El diseño físico para el sistema de inventarios ya mencionado, esta formado por instrucciones de programa, escritas en un lenguaje de programación. Estos pasos revisan los registros de mercancía en existencia utilizando para ello los datos asentados en la transacción, imprimen los reportes y guardan datos. El analista especifica los algoritmos necesarios para cambiar las cantidades de mercancía en existencia. Durante la construcción física, los programadores escriben las instrucciones necesarias del programa para calcular los cambios y producir los resultados.

## **2.2. Apoyo para las actividades de la empresa**

Un objetivo fundamental en el diseño de un sistema de información es asegurar que este brinde apoyo a la actividad de la empresa para la que fue desarrollado. En otras palabras, la tecnología de cómputo y comunicaciones especificada en el diseño siempre debe tener un papel secundario en relación con los resultados que se pretende que el sistema proporcione.

Por ejemplo, si es esencial para una organización mover la información con mucha rapidez para mantener su competitividad, entonces las especificaciones de diseño del sistema de información deben girar alrededor de este objetivo esencial de la empresa. En este ambiente, un sistema que procesa la información con lentitud causara daños a la organización. Imagine el lector un sistema de reservación de vuelos que no dice con prontitud a sus usuarios que asientos quedan disponibles en un determinado vuelo o un sistema de cajero automático en un banco que no permite a los clientes conocer con rapidez e saldo de sus cuentas. Para estas organizaciones, dichas actividades son esenciales y los sistemas de información deben apoyarlas.

De manera similar, el diseño tiene que ajustarse a la forma en que atrabaja la empresa. Si el sistema de ventas de una compañía es diseñado para trabajar mejor con pedidos que son pagados en efectivo, cuando de hecho la empresa tiene una política de crédito para las ventas, entonces la gerencia no estará muy feliz con el sistema, y lo mismo sucederá con los clientes. Aun a pesar de que el sistema de información trabaje bien desde el punto de vista técnico, no se ajusta a las actividades de la empresa.

Los ejemplos anteriores ilustran la importancia de adecuar el sistema a las necesidades de la organización, un objetivo que debe servir virtualmente de guía a todas las decisiones de diseño.

### **2.3. Asegurar que las características del sistema cumplan con los requerimientos del usuario**

Durante el diseño, los requerimientos del usuario se trasladan en características del sistema. Se dice que un sistema de información satisface las necesidades de los usuarios si:

- Realiza en forma apropiada los procedimientos correctos
- Presenta información e instrucciones en una forma aceptable y efectiva
- Produce resultados exactos
- Proporciona una interface y métodos de interacción aceptables
- Es percibido por los usuarios como un sistema confiable

El principal interés del analista es determinar el sistema correcto y proporcionar el sistema correcto. El objetivo del diseño de sistemas es alcanzar estos dos objetivos.



## **2.4. Proporcionar un sistema que sea fácil de utilizar**

Los beneficios que se obtienen por NO utilizar un sistema de información, ¿son mayores que los que se pueden alcanzar con su uso? Esta parece ser una pregunta ilógica, ya que siempre se prefiere la información correcta que la errónea. Pero los analistas con experiencia saben que muchas de las características técnicas de un sistema de información --como su confiabilidad, exactitud y velocidad de procesamiento—son secundarias en relación con los aspectos humanos del diseño de sistemas. Por consiguiente, los analistas procuran diseñar el sistema con ingeniería dirigida hacia las personas y, en consecuencia, incluyen características ergonómicas.

### **2.4.1. Ingeniería humana**

Después que el sistema ha sido instalado y de que los analistas se han ido, los gerentes y sus empleados comienzan a interactuar con el sistema. A medida que el entusiasmo (y el miedo) por la nueva aplicación toma su cauce normal y su uso se convierten en algo rutinario, los usuarios comienzan a examinar y probar sus características. Es en este contexto donde las características de ingeniería humana a menudo tienen mayor importancia de las técnicas. Si los sistemas de información no son diseñados para las personas entonces no tendrán éxito.

En general, sin embargo, el analista debe procurar formular el diseño de sistemas en forma que:

- Incorpore características del sistema que sean fáciles de comprender y utilizar

- Desaliente los errores cometidos por los usuarios a la falta de cuidado por parte de ellos
- Evite fallas o procedimientos inapropiados que generen perjuicios o complicaciones para los usuarios o vayan en detrimento de la organización
- Tenga suficiente flexibilidad para adaptarse a las necesidades de cada usuario
- Se adapte a la creciente familiaridad del usuario con el sistema
- Funcione, en general, en una forma que parezca natural al usuario

A menudo se emplea el termino orientado hacia el usuario para designar aquellos sistemas que exhiben excelentes características de ingeniería humana (desafortunadamente, el empleo frecuente de este termino en la publicidad lo ha hecho tan popular que existe la tendencia a pasar por alto su significado).

## **2.5. Diseño ergonómico**

En el contexto de los sistemas de información, la ergonomía estudia los factores físicos que afectan el rendimiento, la comodidad y la satisfacción de los usuarios directos. El diseño de terminales, sillas y otro tipo de equipos, influye en la fatiga y tensión derivadas del uso de dichos objetos. Estos factores afectan a su vez la tasa de errores cometidos por los usuarios cuando ingresan datos en el sistema, la eficiencia de los usuarios y hasta el ausentismo.

La ergonomía tiene relación con la selección de equipo y el diseño de las áreas de trabajo. Sin embargo, como se vera mas adelante, se deben considerar factores ergonómicos cuando se escogen los colores para la

presentación de información, la ubicación de las teclas de mandatos, o los métodos de interacción con el sistema. En los sistemas más eficientes, los analistas consideran con cuidado y desde diferentes perspectivas, los aspectos ergonómicos del diseño de sistemas.

## **2.6. Proporcionar especificaciones detalladas para el desarrollo de software**

Al igual que con las características de un sistema de información, el software también debe ser diseñado con cuidado. El diseño de sistemas incluye la formulación de las especificaciones de software.

Estas especificaciones establecen las funciones de entrada, salida y procesamiento de software junto con las rutinas, se enfocan sobre lo que cada función realiza; asimismo, se especifican los procedimientos necesarios para llevar a cabo dichas funciones. La selección de lenguajes de programación, paquetes de software y utilerías se efectúa durante el proceso de diseño lógico y las recomendaciones se incluyen como parte de las especificaciones del software.

## **2.7. Ajustarse a los estándares de diseño**

Como puede observarse, los objetivos del diseño de sistemas son muy amplios y afectan aspectos tanto de la aplicación como de la organización en la que será utilizado el sistema. Por consecuencia, no debe sorprender el hecho de que los grupos de sistemas de información mejor manejados, también mantengan estándares para el desarrollo de sistemas. Las especificaciones de diseño se establecen dentro del marco fijado por los estándares.

Los siguientes son ejemplos de áreas incluidas en el diseño de estándares:

- Estándares para datos

Lineamientos para asignar nombre a los datos y especificar su longitud y tipo. Estos lineamientos son utilizados por todas las aplicaciones desarrolladas por el grupo de sistemas de información. Con frecuencia están contenidas en el diccionario de datos.

- Estándares de codificación

Abreviaturas y designaciones formales para describir actividades y entidades dentro de la organización (por ejemplo, clasificación de clientes y tipos de transacciones).

- Estándares estructurales

Lineamientos sobre como estructurar el software y el sistema. Políticas para dividir el software en módulos, para la codificación estructurada y la relación existente entre los componentes del sistema. Pueden incluir estándares sobre la longitud del programa y lineamientos para volver a utilizar los módulos de software.

- Estándares de documentación

Descripciones de las características del diseño de sistemas, de la relación entre componentes y de las características de operación que pueden ser revisadas para conocer los detalles de la aplicación.

Con el objeto de garantizar que la aplicación cumpla con los estándares, muchas organizaciones cuentan con un grupo de control de calidad que tiene la responsabilidad de revisar todas las especificaciones de diseño de sistemas de información, así como el propio sistema una vez terminado.



### **3. CARACTERÍSTICAS QUE SE DEBEN DISEÑAR**

Las especificaciones de diseño describen las características del sistema, los componentes o elementos del sistema y la forma en que estos aparecen ante los usuarios. Para muchos usuarios, el éxito de un sistema está relacionado con la creencia que tengan sobre si el sistema tiene las características adecuadas.

Esta sección describe las características que debe diseñar el analista de sistemas. Pero antes de considerarlas, es conveniente primero aclarar que elementos tienen que tomarse en cuenta en las especificaciones formales del diseño.

#### **3.1. Elementos de diseño**

Los componentes de un sistema de información descritos durante el análisis de requerimientos, son el punto focal del diseño de sistemas. Los analistas deben diseñar los siguientes elementos.

- Flujos de datos

Movimientos de datos hacia, alrededor y desde el sistema.

- Almacenes de datos

Conjuntos temporales o permanentes de datos.

- Procesos

Actividades para aceptar, manejar y suministrar datos e información. Pueden ser manuales o basadas en computadora.

- Procedimientos

Métodos y rutinas para utilizar el sistema de información y lograr con ello los resultados esperados.

- Controles

Estándares y lineamientos para determinar si las actividades están ocurriendo en la forma anticipada o aceptada, es decir si se encuentran “bajo control”. Asimismo, debe especificar las acciones que tienen que emprenderse cuando ocurren problemas o se presentan circunstancias inesperadas. Puede incluirse un reporte sobre las excepciones o procedimientos para la corrección de los problemas.

- Funciones del personal

Las responsabilidades de todas las personas que tienen que ver con el nuevo sistema, incluyendo los usuarios, operadores de computadora y personal de apoyo. Abarca todo el espectro de componentes del sistema, incluso desde la entrada de datos hasta la distribución de salidas o resultados. A menudo, las funciones del personal se establecen en forma de procedimientos.



### **3.2. Diseño de salida**

El término salida, como es probable que el lector lo conozca, se refiere a los resultados e información generados por el sistema. Para muchos usuarios finales, la salida es la única razón para el desarrollo del sistema y la base sobre la que ellos evaluarán la utilidad de la aplicación. En la realidad, muchos usuarios no operan el sistema de información y tampoco ingresan datos en él, pero utilizan la salida generada por el sistema.

Cuando diseñan la salida, los analistas deben realizar lo siguiente:

- Determinar que información presentar
- Decidir si la información será presentada en forma visual, verbal o impresa y seleccionar el medio de salida
- Disponer la presentación de la información en un formato aceptable
- Decidir como distribuir la salida entre los posibles destinatarios

La disposición de la información sobre una pantalla o documento impreso se denomina distribución.

Para llevar a cabo las actividades antes mencionadas, se requieren decisiones específicas tales como el empleo de formatos ya impresos cuando se preparan reportes, cuantas líneas planear sobre una página impresa o si se deben emplear graficas y colores.

El diseño de la salida esta especificado en los formularios de distribución, que son hojas que describen la ubicación, características (como longitud y tipo) y formato de los encabezados de columnas y la paginación. Tal como ya se indicó al inicio de la presente discusión, estos elementos son análogos al bosquejo donde el arquitecto indica la localización de cada componente.

### **3.3. Diseño de archivos**

El diseño de archivos incluye decisiones con respecto a la naturaleza y contenido del propio archivo, como si se fuera a emplear para guardar detalles de las transacciones, datos de tipo histórico o información de referencia. Entre las decisiones que se toman durante el diseño de archivos, se encuentran las siguientes:

- Los datos que deben incluirse en el formato de los registros contenidos en el archivo
- La longitud de cada registro, con base en las características de los datos que contiene
- La secuencia a disposición de los registros dentro del archivo (la estructura de almacenamiento que puede ser secuencial, indexada o relativa)

No todos los nuevos sistemas de información requieren del diseño de todos los archivos utilizados por la aplicación. Por ejemplo, es probable que ya existan archivos maestros porque estos son utilizados por otras aplicaciones existentes. Tal vez la nueva aplicación necesite hacer referencia solo al archivo maestro. En este caso, los detalles del archivo se incluyen en las especificaciones de diseño de la aplicación, pero el archivo no vuelve a diseñarse.

### **3.4. Diseño de interacciones con la base de datos**

Muchos sistemas de información, y a sea implantados en sistemas de computo grandes o pequeños, interactúan con las bases de datos que abarcan varias aplicaciones. Dada la importancia que tienen las bases de datos en muchos sistemas, su diseño es establecido y vigilado por un administrador de base de datos, que es una persona (o grupo de personas) que tiene la responsabilidad de desarrollar y mantener la base de datos. En estos casos, el analista de sistemas no efectúa el diseño de la base de datos sino que consulta al administrador de la base para determinar las interacciones mas apropiadas con lavase de datos. El analista proporciona al administrador la descripción de 1) los datos que son necesarios de la base de datos, y 2) las acciones que tendrán efecto sobre la propia base (por ejemplo, la recuperación de datos, cambios en los valores de los datos o el ingreso de nuevos datos en la base).

A su vez, el papel del administrador de bases de datos incluye las siguientes responsabilidades:

- Evaluar la conveniencia de la solicitud del analista
- Describir los métodos para interactuar con la base de datos

- Asegurar que la aplicación no pueda dañar lavase de datos o que la afecte de manera adversa a las necesidades de otros sistemas de información

### **3.5. Diseño de la entrada**

Los analistas de sistemas deciden los siguientes detalles del diseño de entradas:

1. Qué datos ingresan al sistema
2. Qué medios utilizar
3. La forma en que se deben disponer o codificar los datos
4. El diálogo que servirá de guía a los usuarios para dar entrada a los datos
5. Validación necesaria de datos y transacciones para detectar errores
6. Métodos para llevar a cabo la validación de las entradas y los pasos a seguir cuando se presentan errores

Las decisiones de diseño para el manejo de entradas, especifican la forma en que serán aceptados los datos para su procesamiento por computadora. Los analistas deciden si los datos serán proporcionados directamente, quizá a través de una estación de trabajo, o por el uso de documentos, como talones de ventas, cheques bancarios o facturas, donde los datos a su vez son transferidos hacia la computadora para su procesamiento.

El diseño de la entrada también incluye la especificación de los medios por los que tanto los usuarios finales como los operadores darán instrucciones al sistema sobre las acciones que debe emprender. Por ejemplo, un usuario que interactúa con el sistema por medio de una estación de trabajo, tiene que ser capaz de indicarle al sistema ya sea que acepte una entrada, genere un reporte o termine el procesamiento.

Los sistemas en línea incluyen un diálogo o conversación entre el usuario y el sistema. Por medio del diálogo, el usuario solicita servicios al sistema y le indica cuando realizar cierta función. A menudo la naturaleza de la conversación en línea hace la diferencia entre un diseño exitoso y otro inaceptable. Un diseño inapropiado, que deja la pantalla en blanco, confunde al usuario con respecto a que acción debe emprender. Ya que esta idea es muy importante y dado que los sistemas en línea son bastante comunes.

La disposición de mensajes y comentarios en las conversaciones en línea, así como la ubicación de los datos, encabezados y títulos sobre las pantallas o documentos fuentes, también forman parte del diseño de entradas. En general, se preparan bosquejos para comunicar la disposición a los usuarios, para que ellos la revisen, y a los programadores y otros miembros del equipo de diseño de sistemas.

### **3.6. Diseño de controles**

Los analistas de sistemas también deben anticipar los errores que se cometerán al ingresar los datos en el sistema o al solicitar la ejecución de ciertas funciones. Algunos errores no tienen importancia ni consecuencias, pero otros pueden ser tan serios ocasionarían el borrado de datos o el uso inapropiado del sistema.

Aunque exista solo la más mínima probabilidad de cometer un error serio, un buen diseño de sistema de información ofrecerá los medios para detectar y manejar el error.

Los controles de entrada proporcionan medios para 1)asegurar que solo los usuarios autorizados tengan acceso al sistema, 2)garantizar que las transacciones sean aceptables, 3)validar los datos para comprobar su exactitud y 4)determinar si se han omitido datos que son necesarios.

### **3.7. Diseño de procedimientos**

Los procedimientos especifican que tareas deben efectuarse al utilizar el sistema y quienes son los responsables de llevarlas a cabo. Entre los procedimientos importantes se encuentran:

- Procedimientos para entrada de datos

Métodos para la captura de datos de las transacciones y su ingreso en el sistema de información (ejemplo: secuencia para dar entrada a los datos registrados en los documentos fuentes)

- Procedimientos durante la ejecución

Pasos y acciones emprendidos por los operadores del sistema y, en ciertos casos, por los usuarios finales que interactúan con el sistema para alcanzar los resultados deseados (ejemplo: montar paquetes de discos o colocar en las impresoras formas pre impresas)

- Procedimientos para el manejo de errores

Acciones a seguir cuando se presentan resultados inesperados (ejemplo: ocurre un error cuando el sistema intenta leer datos de un archivo o la impresora se atasca durante la impresión de una gran cantidad de hojas)

- Procedimientos de seguridad y respaldo

Acciones para proteger al sistema y sus recursos contra posibles daños (ejemplo: ¿Cuándo y cómo hacer copias de los archivos maestros o de partes de la base de datos?)

Estos procedimientos deben formularse por escrito y formar parte de la documentación del sistema.

### **3.8. Diseño de especificaciones para programas**

Las especificaciones para programas son por si mismas un diseño. Ellas describen como transformar las especificaciones de diseño del sistema – salidas, entradas, archivos, procesamiento y otras—en software de computadora.

El diseño del software de computadora es importante para asegurar que:

- Los programas producidos lleven a cabo todas las tareas y los hagan en la forma establecida
- La estructuración del software en módulos permita su prueba y validación para determinar si los procedimientos son correctos

- Las modificaciones futuras se puedan realizar en forma eficiente y con un mínimo de interrupción en el diseño del sistema

Un sistema de software en particular será diseñado solo una vez, pero será usado repetidamente y es muy probable que evolucione en la medida que cambien las necesidades de los usuarios. Estas observaciones añaden más importancia al diseño de software.

En algunas organizaciones, existe una separación entre las responsabilidades del programador y las que tienen el analista. En otras, tanto los programadores como analistas comparten las responsabilidades facilita la comunicación del diseño a ciertos programadores que trabajan en el proyecto, esta no elimina los aspectos mencionados hasta este momento.

Los métodos para desarrollar el diseño o para especificar los detalles varían de acuerdo con las prácticas establecidas en cada organización. También serán diferentes como consecuencia de los lenguajes utilizados para escribir el software. Los lenguajes de tercera generación requieren una atención mucho mayor en lo que respecta a los detalles de los procedimientos en comparación con la dedicada cuando se emplean lenguajes de cuarta generación. [10]



## **4. PROCESO DE DESARROLLO**

En el proceso de desarrollo de sistemas de información se incluyen, tanto las técnicas propias de un desarrollo orientado a objetos como estructurado, ya que las actividades de ambas aproximaciones están integradas en una estructura común.

Para cada una de las técnicas referenciadas en el documento se explica brevemente el objetivo que se persigue al utilizarlas. Se describen: los elementos básicos asociados y los principios fundamentales de elaboración; la notación utilizada, en el caso de técnicas gráficas, para la representación de cada uno de los elementos implicados.

### **4.1. Técnicas de desarrollo**

Las técnicas de desarrollo son un conjunto de procedimientos que se basan en reglas y notaciones específicas en términos de sintaxis, semántica y gráficos, orientadas a la obtención de productos en el desarrollo de un sistema de información. En desarrollos del tipo estructurado o de orientación a objetos merecen especial atención las técnicas gráficas, que proponen símbolos y notaciones estándares para una mejor comprensión de los sistemas o sus componentes.

## **4.2. Casos de uso**

Los objetivos de los casos de uso son los siguientes:

- Capturar los requisitos funcionales del sistema y expresarlos desde el punto de vista del usuario.
- Guiar todo el proceso de desarrollo del sistema de información.

Los casos de uso proporcionan, por tanto, un modo claro y preciso de comunicación entre cliente y desarrollador. Desde el punto de vista del cliente proporcionan una visión de “caja negra” del sistema, esto es, cómo aparece el sistema desde el exterior sin necesidad de entrar en los detalles de su construcción. Para los desarrolladores, suponen el punto de partida y el eje sobre el que se apoya todo el desarrollo del sistema en sus procesos de análisis y diseño.

### **4.2.1. Descripción**

Un caso de uso es una secuencia de acciones realizadas por el sistema, que producen un resultado observable y valioso para un usuario en particular, es decir, representa el comportamiento del sistema con el fin de dar respuestas a los usuarios.

Aquellos casos de uso que resulten demasiados complejos se pueden descomponer en un segundo nivel, en el que los nuevos casos de uso que intervengan resulten más sencillos y manejables.

Para especificar este comportamiento existen una serie de recomendaciones o técnicas que se aplican dependiendo del momento del desarrollo que se esté y de la complejidad del caso de uso. Puede ser desde una simple descripción textual que recoja un requisito funcional a una especificación del caso de uso, e incluso un conjunto de diagramas:

#### **4.2.1.1 Especificación de un caso de uso**

Un caso de uso recoge, en un primer momento, una descripción general. Esta descripción reflejará posiblemente uno o varios requisitos funcionales del sistema o formará parte de algún requisito.

Se puede completar la descripción definiendo cuáles son las precondiciones y pos condiciones del sistema, es decir, qué condiciones deben cumplirse para que se realice un caso de uso y cuáles son aquellas condiciones que se cumplen posteriormente al caso de uso.

También se pueden enumerar los diferentes escenarios del caso de uso si los tuviese y dar una breve descripción de ellos. Los escenarios son los distintos caminos por los que puede evolucionar un caso de uso, dependiendo de las condiciones que se van dando en su realización.

#### 4.2.1.2 Diagrama de casos de uso

Estos diagramas presentan dos tipos de elementos fundamentales:

– Actores. Un actor es algo o alguien que se encuentra fuera del sistema y que interactúa con él. En general, los actores serán los usuarios del sistema y los sistemas externos al que se esté desarrollando. Si se habla de usuarios, un actor es el papel que puede llevar a cabo en cuanto a su forma de interactuar con el sistema, es decir, un único actor puede representar a muchos usuarios diferentes y de la misma forma, un usuario puede actuar como actores diferentes.

– Casos de uso. Un caso de uso representa el comportamiento que ofrece el sistema de información desde el punto de vista del usuario. Típicamente será un conjunto de transacciones ejecutadas entre el sistema y los actores. Para facilitar la comprensión de los casos de uso del sistema de información en el análisis, es posible agruparlos en paquetes según funcionalidades semejantes o relacionadas.

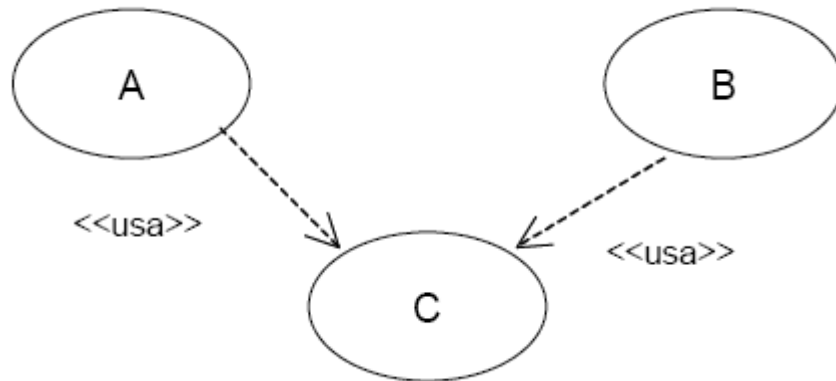
Además de estos elementos, un diagrama de casos de uso presenta relaciones. Las relaciones pueden tener lugar entre actores y casos de uso o entre casos de uso.

La relación entre un actor y un caso de uso es una relación de comunicación, que indica que un actor interviene en el caso de uso. Normalmente, el actor aporta información para la realización de un caso de uso o recibe información como resultado de la realización del mismo, por ello, esta relación puede ser unidireccional o bidireccional, aunque generalmente se muestra como bidireccional, ya que no es necesario especificar en detalle estas relaciones.

La relación entre casos de uso es una relación unidireccional. Esta relación puede presentar uno de los dos siguientes tipos: “usa” y “extiende”.

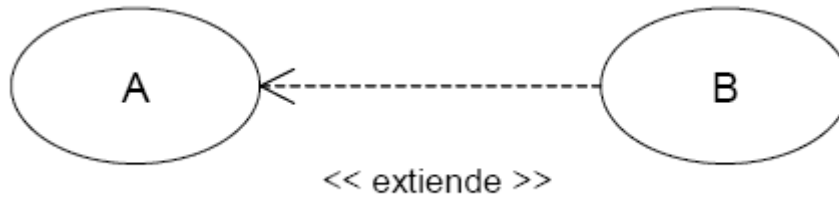
- La relación “usa” se utiliza cuando se quiere reflejar un comportamiento común en varios casos de uso. Es decir, si los casos de uso A y B presentan una parte común, ésta se puede sacar a un tercer caso de uso C. Entonces, habrá una relación “usa” del caso de uso A al C y otra del B al C.

### 1. Ejemplo diagrama de casos de uso.



- La relación “extiende” se utiliza cuando se quiere reflejar un comportamiento opcional de un caso de uso. Por ejemplo, se tiene el caso de uso A que representa un comportamiento habitual del sistema. Sin embargo, dependiendo de algún factor, este caso de uso puede presentar un comportamiento adicional o ligeramente diferente, que se podría reflejar en un caso de uso B. En este caso, habrá una relación “extiende” del caso de uso B al A.

## 2 . Ejemplo de una relación extiende.



### 4.2.2. Notación

El diagrama de casos de uso es un grafo de actores, casos de uso y las relaciones entre estos elementos.

Opcionalmente, los casos de uso se pueden enmarcar en un cuadrado que representa los límites del sistema.

#### 4.2.2.1 Caso de uso

Un caso de uso se representa mediante una elipse con el nombre del caso de uso dentro o debajo.

## 3 . Ejemplo de cómo se representa un caso de uso.



#### 4.2.2.2 Actor

Un actor se representa con una figura de 'hombre de palo' con el nombre del actor debajo de la figura.

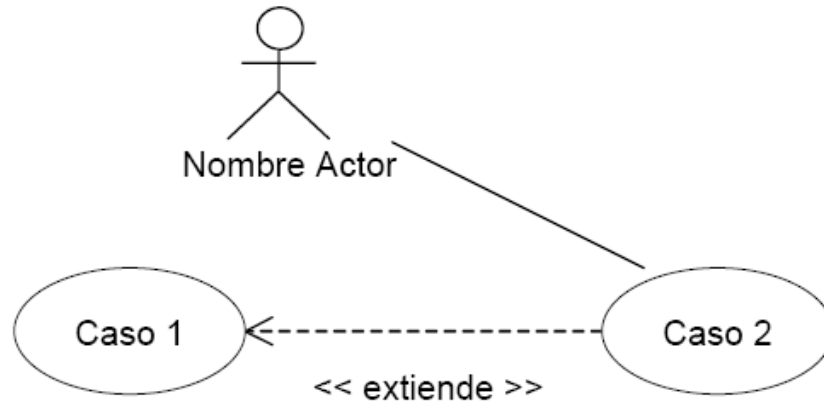
#### 4 . Ejemplo de cómo se representa un actor.



#### 4.2.2.3 Relación

Dependiendo del tipo de relación, la representación en los diagramas será distinta. Así pues, las relaciones entre un actor y un caso de uso se representan mediante una línea continua entre ellos. Las relaciones entre casos de uso se representan con una flecha discontinua con el nombre del tipo de relación como etiqueta. En las relaciones “extensión” la flecha parte del caso de uso con el comportamiento adicional hacia aquel que recoge el comportamiento básico y en las relaciones “usa” desde el caso de uso básico hacia el que representa el comportamiento común.

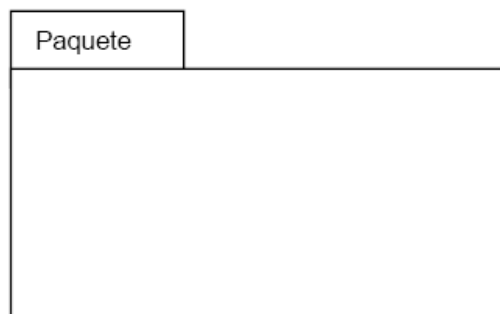
### 5 . Ejemplo de un diagrama de casos de uso con relaciones.



#### 4.2.2.4 Paquete

Un paquete se representa con un icono con forma de carpeta y con el nombre colocado en la 'pestaña'. Los paquetes también pueden formar diagramas que complementen al diagrama de casos de uso (ver Diagrama de paquetes).

### 6 . Ejemplo de cómo se representa un paquete.

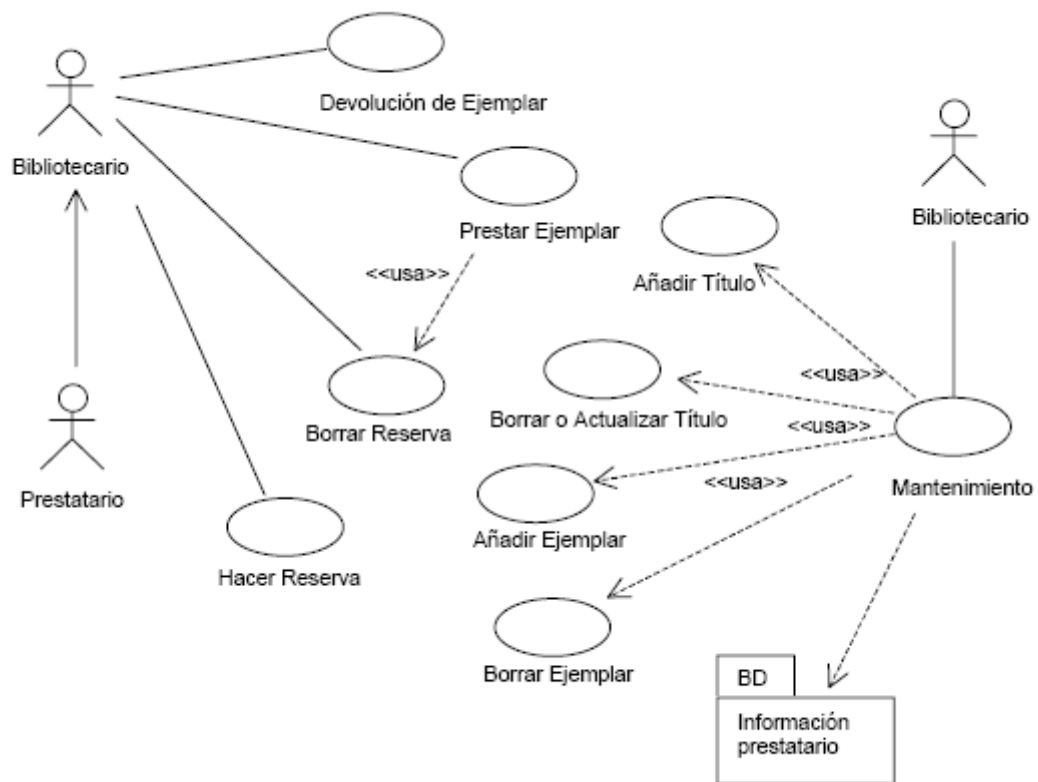




#### 4.2.2.5 Ejemplo

Estudio de una aplicación que se encarga de la gestión de los préstamos y reservas de libros y revistas en una biblioteca.

### 7 . Ejemplo de un caso de uso de una biblioteca.



### 4.3. Diagrama de clases

El objetivo principal de este modelo es la representación de los aspectos estáticos del sistema, utilizando diversos mecanismos de abstracción (clasificación, generalización, agregación).

### **4.3.1. Descripción**

El diagrama de clases recoge las clases de objetos y sus asociaciones. En este diagrama se representa la estructura y el comportamiento de cada uno de los objetos del sistema y sus relaciones con los demás objetos, pero no muestra información temporal.

Con el fin de facilitar la comprensión del diagrama, se pueden incluir paquetes como elementos del mismo, donde cada uno de ellos agrupa un conjunto de clases.

Este diagrama no refleja los comportamientos temporales de las clases, aunque para mostrarlos se puede utilizar un diagrama de transición de estados.

Los elementos básicos del diagrama son:

#### **4.3.1.1 Clases**

Una clase describe un conjunto de objetos con propiedades (atributos) similares y un comportamiento común. Los objetos son instancias de las clases.

No existe un procedimiento inmediato que permita localizar las clases del diagrama de clases. Éstas suelen corresponderse con sustantivos que hacen referencia al ámbito del sistema de información y que se encuentran en los documentos de las especificaciones de requisitos y los casos de uso.

Dentro de la estructura de una clase se definen los atributos y las operaciones o métodos:

- Los atributos de una clase representan los datos asociados a los objetos instanciados por esa clase.
- Las operaciones o métodos representan las funciones o procesos propios de los objetos de una clase, caracterizando a dichos objetos.

El diagrama de clases permite representar clases abstractas. Una Clase abstracta es una clase que no puede existir en la realidad, pero que es útil conceptualmente para el diseño del modelo orientado a objetos. Las clases abstractas no son instanciables directamente sino en sus descendientes. Una clase abstracta suele ser situada en la jerarquía de clases en una posición que le permita ser un depósito de métodos y atributos para ser compartidos o heredados por las subclases de nivel inferior.

Las clases y en general todos los elementos de los diagramas, pueden estar clasificados de acuerdo a varios criterios, como por ejemplo su objetivo dentro de un programa. Esta clasificación adicional se expresa mediante un Estereotipo. Algunos de los autores de métodos OO, establecen una clasificación de todos los objetos que pueden aparecer en un modelo. Los tipos son:

- Objetos Entidad.
- Objetos límite o interfaz.
- Objetos de control.

Éstos son estereotipos de clases. Un estereotipo representa una la meta-clasificación de un elemento.

Dependiendo de la herramienta utilizada, también se puede añadir información adicional a las clases para mostrar otras propiedades de las mismas, como son las reglas de negocio, responsabilidades, manejo de eventos, excepciones, etc.

#### 4.3.1.2 Relaciones

Los tipos más importantes de relaciones estáticas entre clases son los siguientes:

- **Asociación.** Las relaciones de asociación representan un conjunto de enlaces entre objetos o instancias de clases. Es el tipo de relación más general, y denota básicamente una dependencia semántica. Por ejemplo, una Persona trabaja para una Empresa.

Cada asociación puede presentar elementos adicionales que doten de mayor detalle al tipo de relación:

- Rol, o nombre de la asociación, que describe la semántica de la relación en el sentido indicado. Por ejemplo, la asociación entre persona y empresa recibe el nombre de trabaja para, como rol en ese sentido.

- Multiplicidad, que describe la cardinalidad de la relación, es decir, especifica cuántas instancias de una clase están asociadas a una instancia de la otra clase. Los tipos de multiplicidad son: Uno a uno, uno a muchos y muchos a muchos.

- **Herencia.** Las jerarquías de generalización/especialización se conocen como herencia.

Herencia es el mecanismo que permite a una clase de objetos incorporar atributos y métodos de otra clase, añadiéndolos a los que ya posee. Con la herencia se refleja una relación “es\_un” entre clases. La clase de la cual se hereda se denomina superclase, y la que hereda subclase.

La generalización define una superclase a partir de otras. Por ejemplo, de las clases profesor y estudiante se obtiene la superclase persona. La especialización o especificación es la operación inversa, y en ella una clase se descompone en una o varias subclases. Por ejemplo, de la clase empleado se pueden obtener las subclases secretaria, técnico e ingeniero.

– **Agregación.** La agregación es un tipo de relación jerárquica entre un objeto que representa la totalidad de ese objeto y las partes que lo componen. Permite el agrupamiento físico de estructuras relacionadas lógicamente. Los objetos “son-parte-de” otro objeto completo. Por ejemplo, motor, ruedas, carrocería son parte de automóvil.

– **Composición.** La composición es una forma de agregación donde la relación de propiedad es más fuerte, e incluso coinciden los tiempos de vida del objeto completo y las partes que lo componen. Por ejemplo, en un sistema de Máquina de café, las relaciones entre la clase máquina y producto, o entre máquina y depósito de monedas, son de composición.

– **Dependencia.** Una relación de dependencia se utiliza entre dos clases o entre una clase y una interfaz, e indica que una clase requiere de otra para proporcionar alguno de sus servicios.

#### **4.3.1.3 Interfaces**

Una interfaz es una especificación de la semántica de un conjunto de operaciones de una clase o paquete que son visibles desde otras clases o paquetes. Normalmente, se corresponde con una parte del comportamiento del elemento que la proporciona.

#### **4.3.1.4 Paquetes**

Los paquetes se usan para dividir el modelo de clases del sistema de información, agrupando clases u otros paquetes según los criterios que sean oportunos. Las dependencias entre ellos se definen a partir de las relaciones establecidas entre los distintos elementos que se agrupan en estos paquetes (ver Diagrama de paquetes).

### **4.3.2. Notación**

#### **4.3.2.1 Clases**

Una clase se representa como una caja, separada en tres zonas por líneas horizontales.

En la zona superior se muestra el nombre de la clase y propiedades generales como el estereotipo. El nombre de la clase aparece centrado y si la clase es abstracta se representa en cursiva. El estereotipo, si se muestra, se sitúa sobre el nombre y entre el símbolo: << .... >>.

La zona central contiene una lista de atributos, uno en cada línea. La notación utilizada para representarlos incluye, dependiendo del detalle, el nombre del atributo, su tipo y su valor por defecto, con el formato

Visibilidad nombre: tipo = valor-inicial {propiedades}

La visibilidad será en general pública (+), privada (-) o protegida (#), aunque puede haber otros tipos de visibilidad dependiendo del lenguaje de programación empleado.

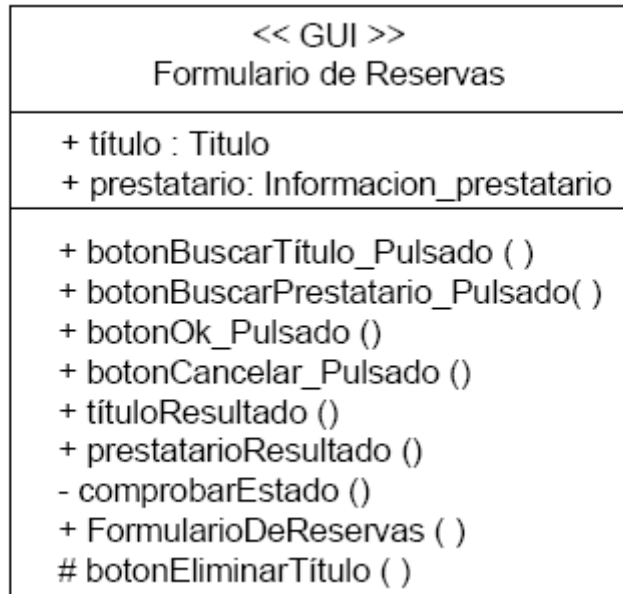
En la zona inferior se incluye una lista con las operaciones que proporciona la clase. Cada operación aparece en una línea con formato:

Visibilidad nombre (lista-de-parámetros): tipo-devuelto {propiedad}

La visibilidad será en general pública (+), privada (-) o protegida (#), aunque como con los atributos, puede haber otros tipos de visibilidad dependiendo del lenguaje de programación. La lista de parámetros es una lista con los parámetros recibidos en la operación separados por comas. El formato de un parámetro es:

Nombre: tipo = valor-por-defecto

## 8 . Ejemplo de cómo se representa una clase.



La notación especificada se puede simplificar según el nivel de detalle con el que se quiera trabajar en un momento dado.

### 4.3.2.2 Relaciones

Una relación de asociación se representa como una línea continua entre las clases asociadas. En una relación de asociación, ambos extremos de la línea pueden conectar con la misma clase, indicando que una instancia de una clase, está asociada a otras instancias de la misma clase, lo que se conoce como asociación reflexiva.

La relación puede tener un nombre y un estereotipo, que se colocan junto a la línea. El nombre suele corresponderse con expresiones verbales presentes en las especificaciones, y define la semántica de la asociación.

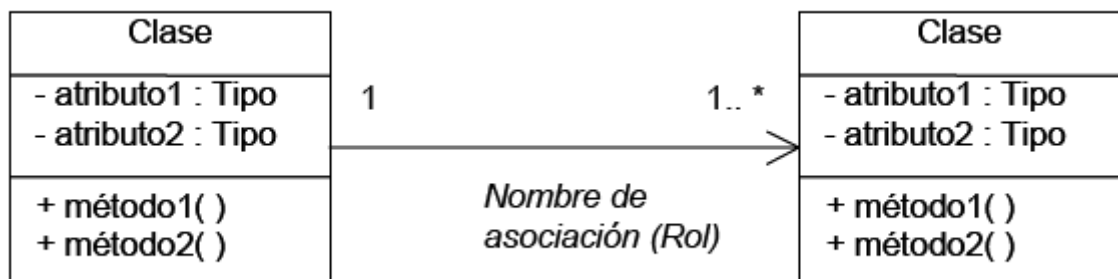


Los estereotipos permiten clasificar las relaciones en familias y se escribirán entre el símbolo: << ... >>.

Las diferentes propiedades de la relación se pueden representar con la siguiente notación:

- Multiplicidad: La multiplicidad puede ser un número concreto, un rango o una colección de números. La letra 'n' y el símbolo '\*' representan cualquier número.
- Orden: Se puede especificar si las instancias guardan un orden con la palabra clave '{ordered}'. Si el modelo es suficientemente detallado, se puede incluir una restricción que indique el criterio de ordenación.
- Navegabilidad: La navegación desde una clase a la otra se representa poniendo una flecha sin relleno en el extremo de la línea, indicando el sentido de la navegación.
- Rol o nombre de la asociación: Este nombre se coloca junto al extremo de la línea que esta unida a una clase, para expresar cómo esa clase hace uso de la otra clase con la que mantiene la asociación.

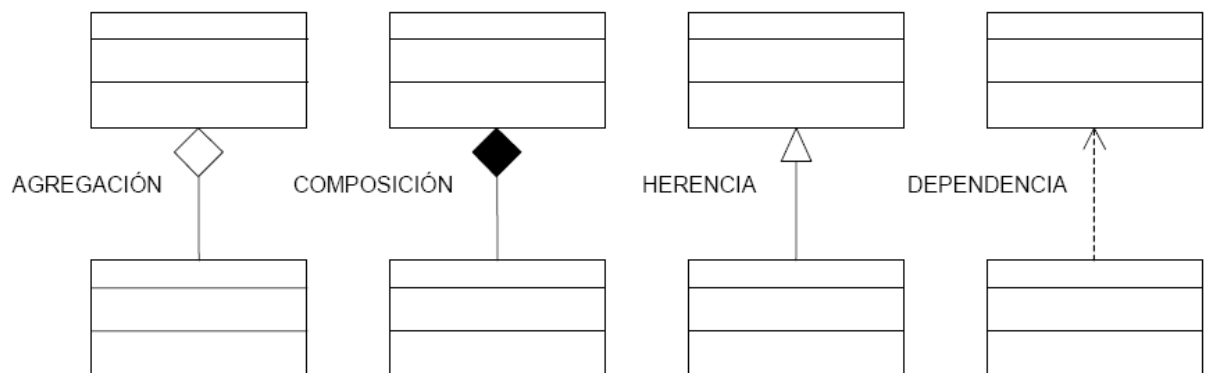
### 9 . Ejemplo de una relación entre clases.



Además, existen notaciones específicas para los otros tipos de relación, como son:

- Agregación: Se representa con un rombo hueco en la clase cuya instancia es una agregación de las instancias de la otra.
- Composición: Se representa con un rombo lleno en la clase cuya instancia contiene las instancias de la otra clase.
- Dependencia: Una línea discontinua con una flecha apuntando a la clase cliente. La relación puede tener un estereotipo que se coloca junto a la línea, y entre el símbolo: << ... >>.
- Herencia: Esta relación se representa como una línea continua con una flecha hueca en el extremo que apunta a la superclase.

### 10 . Ejemplo de tipos de relaciones entre clases.



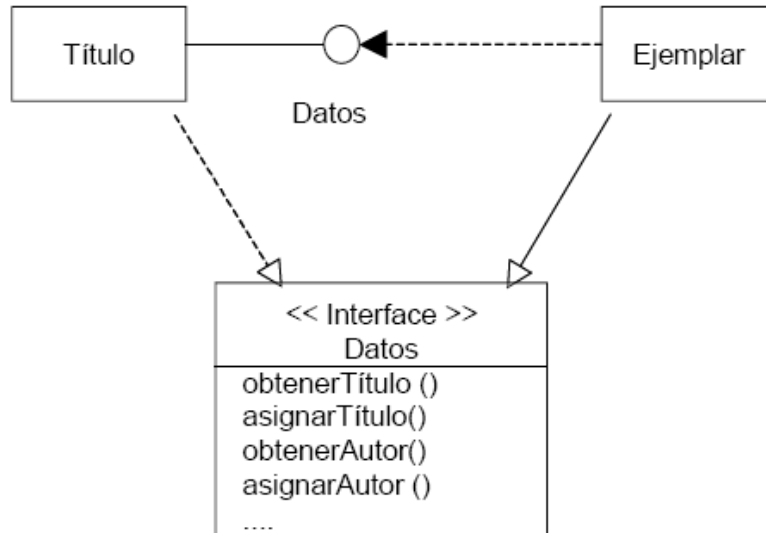
### 4.3.2.3 Interfaces

Una interfaz se representa como una caja con compartimentos, igual que las clases. En la zona superior se incluye el nombre y el estereotipo <<Interface>>. La lista de operaciones se coloca en la zona inferior, igual que en las representaciones de clases. La zona en la que se listan los atributos estará vacía o puede omitirse.

Existe una representación más simple para la interfaz: un círculo pequeño asociado a una clase con el nombre de la interfaz debajo. Las operaciones de la interfaz no aparecen en esta representación; si se quiere que aparezcan, debe usarse la primera notación.

Entre una clase que implementa las operaciones que una interfaz ofrece y esa interfaz se establece una relación de realización que, dependiendo de la notación elegida, se representará con una línea continua entre ellas cuando la interfaz se representa como un círculo y con una flecha hueca discontinua apuntando a la interfaz cuando se represente como una clase.

### 11 . Ejemplo de cómo se representa una interfaz.



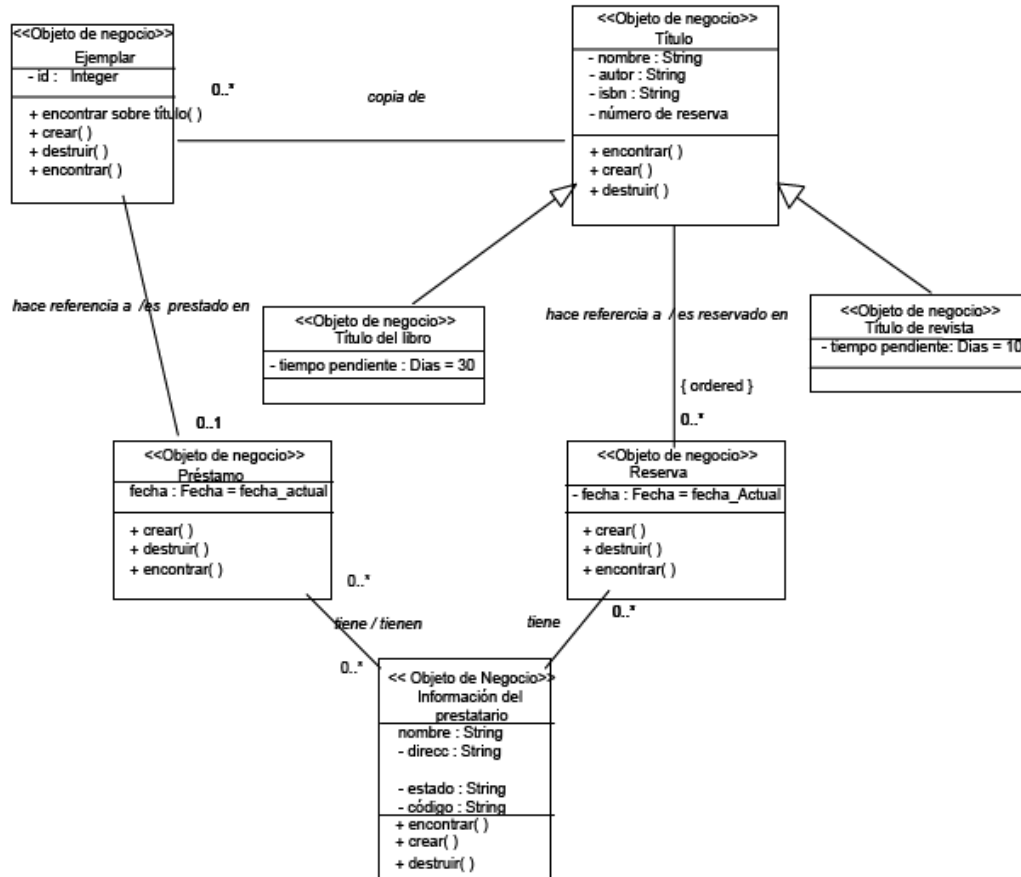
#### 4.3.2.4 Paquetes

Los paquetes se representan mediante un icono con forma de carpeta y las dependencias con flechas discontinuas entre los paquetes dependientes (ver Diagrama de paquetes).

#### 4.3.2.5 Ejemplo

Estudio del sistema encargado de la gestión de préstamos y reservas de libros y revistas de una biblioteca. Dependiendo del momento del desarrollo el diagrama estará más o menos detallado. Así, el diagrama tendría la siguiente estructura en el proceso de análisis:

## 12 . Ejemplo de un diagrama de clases de una biblioteca.



### 4.4. Diagrama de componentes

El diagrama de componentes proporciona una visión física de la construcción del sistema de información. Muestra la organización de los componentes software, sus interfaces y las dependencias entre ellos.

#### 4.4.1. Descripción

Como ya se ha indicado, los elementos de estos diagramas son los componentes software y las dependencias entre ellos.

Un componente es un módulo de software que puede ser código fuente, código binario, un ejecutable, o una librería con una interfaz definida. Una interfaz establece las operaciones externas de un componente, las cuales determinan una parte del comportamiento del mismo.

Además se representan las dependencias entre componentes o entre un componente y la interfaz de otro, es decir uno de ellos usa los servicios o facilidades del otro.

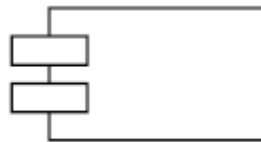
Estos diagramas pueden incluir paquetes que permiten organizar la construcción del sistema de información en subsistemas y que recogen aspectos prácticos relacionados con la secuencia de compilación entre componentes, la agrupación de elementos en librerías, etc.

#### **4.4.2. Notación**

##### **4.4.2.1 Componente**

Un componente se representa como un rectángulo, con dos pequeños rectángulos superpuestos perpendicularmente en el lado izquierdo.

#### **13 . Ejemplo de cómo se representa un componente.**



Para distinguir distintos tipos de componentes se les puede asignar un estereotipo, cuyo nombre estará dentro del símbolo: << ... >>

#### 4.4.2.2 Interfaz

Se representa como un pequeño círculo situado junto al componente que lo implementa y unido a él por una línea continua. La interfaz puede tener un nombre que se escribe junto al círculo. Un componente puede proporcionar más de una interfaz.

#### 14 . Ejemplo de cómo se presersenta una interfaz de un componente.



#### 4.4.2.3 Paquete

Un paquete se representa con un icono de carpeta (ver Diagrama de Paquetes).

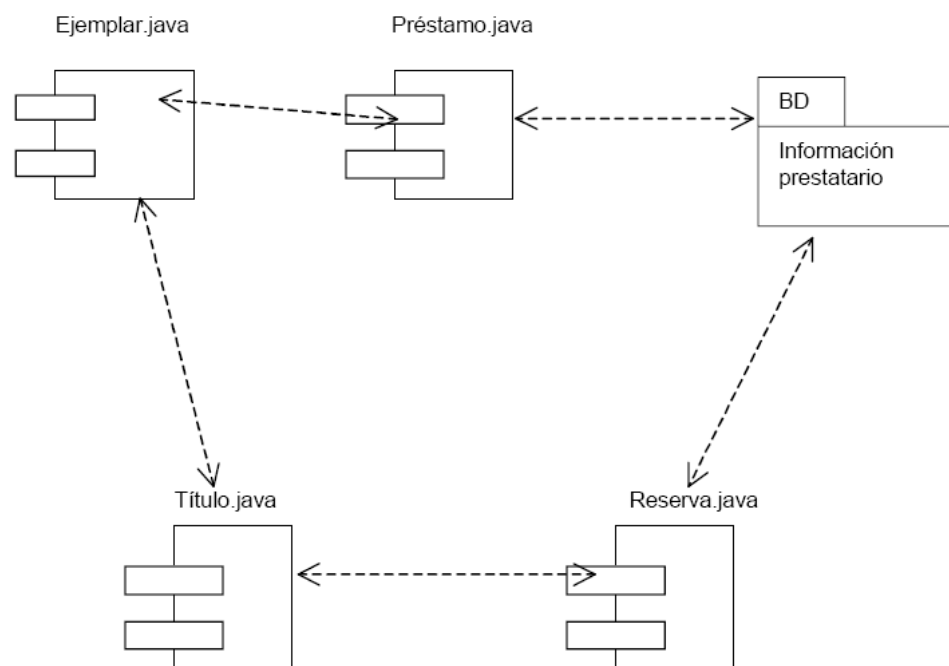
#### 4.4.2.4 Relación de dependencia

Una relación de dependencia se representa mediante una línea discontinua con una flecha que apunta al componente o interfaz que provee del servicio o facilidad al otro. La relación puede tener un estereotipo que se coloca junto a la línea, entre el símbolo: <<...>>.

#### 4.4.2.5 Ejemplo

Sistema encargado de la gestión de los préstamos y reservas de libros y revistas en una biblioteca. El lenguaje de desarrollo será Java, y los accesos a la información del prestatario serán mediante un paquete de Base de Datos.

#### 15 . Ejemplo de un diagrama de componentes de una biblioteca.



#### 4.5. Diagrama de despliegue

El objetivo de estos diagramas es mostrar la disposición de las particiones físicas del sistema de información y la asignación de los componentes software a estas particiones. Es decir, las relaciones físicas entre los componentes software y hardware en el sistema a entregar.



#### **4.5.1. Descripción**

En estos diagramas se representan dos tipos de elementos, nodos y conexiones, así como la distribución de componentes del sistema de información con respecto a la partición física del sistema.

En MÉTRICA Versión 3 se propone una definición concreta de nodo, prescindiendo de determinados detalles, pero permitiendo una continuidad tanto en el diseño como en la construcción del sistema de información. Con este fin, se utiliza el nodo como partición física o funcional real, pero sin descender a detalles de infraestructura o dimensionamiento; por ejemplo, interesa si el nodo procesador es arquitectura Intel, pero no tanto si tiene dos o cuatro procesadores.

Las conexiones representan las formas de comunicación entre nodos.

Además, a cada nodo se le asocia un subsistema de construcción que agrupa componentes software, permitiendo de este modo, determinar la distribución de estos componentes. Por lo tanto, un diagrama de despliegue puede incluir, dependiendo del nivel de detalle, todos los elementos descritos en la técnica de diagrama de componentes, además los nodos y las conexiones propios de esta técnica.

## **4.5.2. Notación**

### **4.5.2.1 Nodo**

Se representa con la figura de un cubo. El nodo se etiqueta con un nombre representativo de la partición física que simboliza. Se pueden asociar a los nodos subsistemas de construcción.

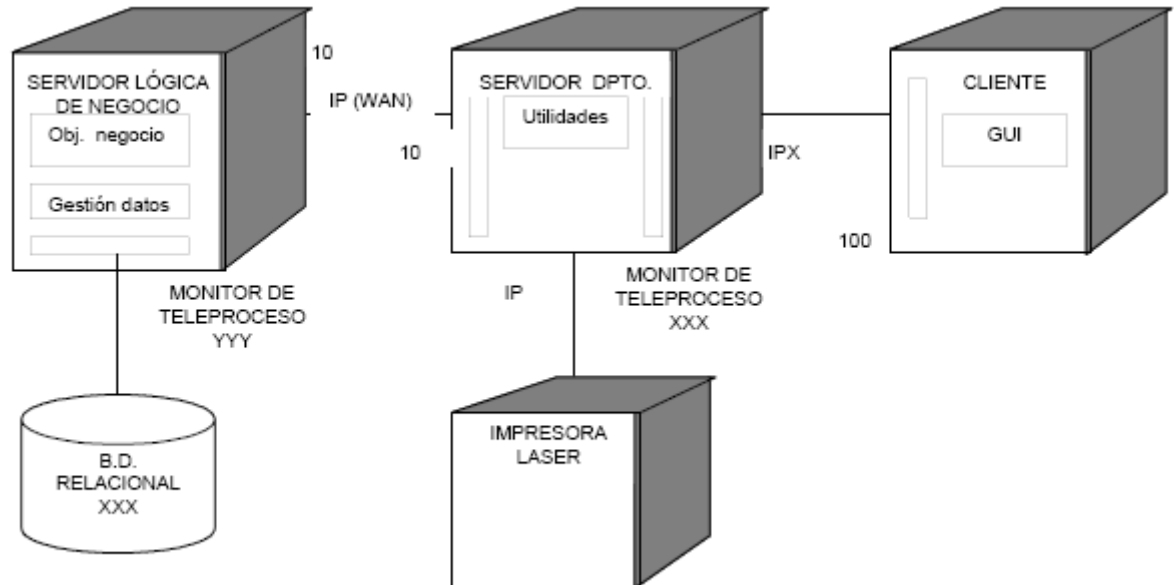
### **4.5.2.2 Conexión**

Las conexiones se representan con una línea continua que une ambos nodos y pueden tener una etiqueta que indique el tipo de conexión. (Ejemplo: canal, red, protocolo, etc.)

### **4.5.2.3 Ejemplo**

El diagrama representa una arquitectura compuesta por un servidor central de lógica de negocio y acceso a datos, en un monitor de teleproceso de tipo XXX, al cual hay conectados 10 servidores departamentales, con clientes (100) e impresora conectados a cada uno de ellos. No interesa tanto recoger en el diagrama la infraestructura real (la exactitud de la configuración, número de procesadores que pueden cambiar con el tiempo y en principio no afecta ni al diseño ni a la construcción), como el tipo “genérico” de los servidores, los volúmenes en el caso de que sean significativos (por ejemplo: 100 puestos por departamento).

## 16 . Ejemplo de un diagrama de despliegue.



### 4.6. Diagrama de interacción

El objetivo de esta técnica es describir el comportamiento dinámico del sistema de información mediante el paso de mensajes entre los objetos del mismo. Además representa un medio para verificar la coherencia del sistema mediante la validación con el modelo de clases.

#### 4.6.1. Descripción

Un diagrama de interacción describe en detalle un determinado escenario de un caso de uso. En él se muestra la interacción entre el conjunto de objetos que cooperan en la realización de dicho escenario. Suele ser conveniente especificar en la parte izquierda del diagrama el caso de uso que se está representando para que resulte más sencilla su validación.

Los elementos que componen los diagramas de interacción son los objetos y los mensajes:

- Un objeto es una entidad que tiene un estado, un comportamiento e identidad. La estructura y el comportamiento común de diferentes objetos se recoge en una clase. En un diagrama de interacción, los objetos serán al final instancias de una determinada clase o de un actor.
- Un mensaje es una comunicación entre dos objetos. El envío de un mensaje por parte de un objeto (emisor) a otro (receptor), puede provocar que se ejecute una operación, se produzca un evento o se cree o destruya un objeto.

Hay dos tipos de diagramas de interacción: diagramas de secuencia y diagramas de colaboración. Ambos tipos de diagramas tratan la misma información pero cada uno hace énfasis en un aspecto particular en cuanto a la forma de mostrarla.

Los diagramas de secuencia muestran de forma explícita la secuencia de los mensajes intercambiados por los objetos, mientras que los diagramas de colaboración muestran de forma más clara cómo colaboran los objetos, es decir, con qué otros objetos tiene vínculos o intercambia mensajes un determinado objeto.

A continuación se detallan las particularidades de uno de ellos.

## **4.7. Diagrama de secuencia**

El diagrama de secuencia es un tipo de diagrama de interacción cuyo objetivo es describir el comportamiento dinámico del sistema de información haciendo énfasis en la secuencia de los mensajes intercambiados por los objetos.

### **4.7.1. Descripción**

Un diagrama de secuencia tiene dos dimensiones, el eje vertical representa el tiempo y el eje horizontal los diferentes objetos. El tiempo avanza desde la parte superior del diagrama hacia la inferior. Normalmente, en relación al tiempo sólo es importante la secuencia de los mensajes, sin embargo, en aplicaciones de tiempo real se podría introducir una escala en el eje vertical.

Respecto a los objetos, es irrelevante el orden en que se representan, aunque su colocación debería poseer la mayor claridad posible.

Cada objeto tiene asociados una línea de vida y focos de control. La línea de vida indica el intervalo de tiempo durante el que existe ese objeto. Un foco de control o activación muestra el periodo de tiempo en el cual el objeto se encuentra ejecutando alguna operación, ya sea directamente o mediante un procedimiento concurrente.

## **4.7.2. Notación**

### **4.7.2.1 Objeto y línea de vida**

Un objeto se representa como una línea vertical discontinua, llamada línea de vida, con un rectángulo de encabezado con el nombre del objeto en su interior. También se puede incluir a continuación el nombre de la clase, separando ambos por dos puntos.

Si el objeto es creado en el intervalo de tiempo representado en el diagrama, la línea comienza en el punto que representa ese instante y encima se coloca el objeto. Si el objeto es destruido durante la interacción que muestra el diagrama, la línea de vida termina en ese punto y se señala con un aspa de ancho equivalente al del foco de control.

En el caso de que un objeto existiese al principio de la interacción representada en el diagrama, dicho objeto se situará en la parte superior del diagrama, por encima del primer mensaje. Si un objeto no es eliminado en el tiempo que dura la interacción, su línea de vida se prolonga hasta la parte inferior del diagrama.

La línea de vida de un objeto puede desplegarse en dos o más líneas para mostrar los diferentes flujos de mensajes que puede intercambiar un objeto, dependiendo de alguna condición.

### **4.7.2.2 Foco de control o activación**

Se representa como un rectángulo delgado superpuesto a la línea de vida del objeto. Su largo dependerá de la duración de la acción.

La parte superior del rectángulo indica el inicio de una acción ejecutada por el objeto y la parte inferior su finalización.

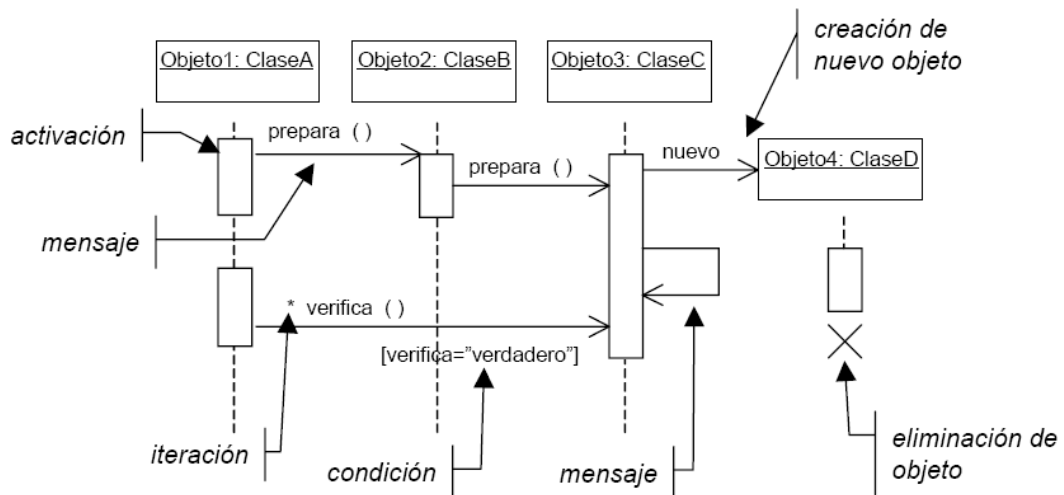
### 4.7.2.3 Mensaje

Un mensaje se representa como una flecha horizontal entre las líneas de vida de los objetos que intercambian el mensaje. La flecha va desde el objeto que envía el mensaje al que lo recibe.

Además, un objeto puede mandarse un mensaje a sí mismo, en este caso la flecha comienza y termina en su línea de vida.

La flecha tiene asociada una etiqueta con el nombre del mensaje y los argumentos. También pueden ser etiquetados los mensajes con un número de secuencia, sin embargo, este número no es necesario porque la localización física de las flechas que representan a los mensajes ya indica el orden de los mismos.

### 17 . Ejemplo de un diagrama de secuencia.

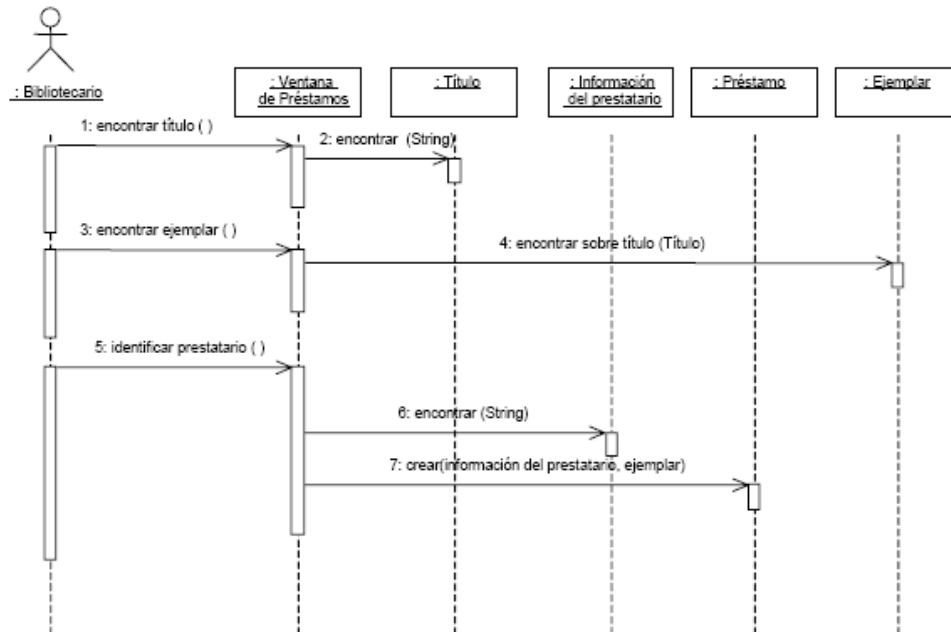


Los mensajes pueden presentar también condiciones e iteraciones. Una condición se representa mediante una expresión booleana encerrada entre corchetes junto a un mensaje, e indica que ese mensaje sólo es enviado en caso de ser cierta la condición. Una iteración se representa con un asterisco y una expresión entre corchetes, que indica el número de veces que se produce.

#### 4.7.2.4 Ejemplo

Diagrama de secuencia para el caso de uso: Prestar un ejemplar de una aplicación encargada de los préstamos y reservas de una biblioteca:

#### 18 . Ejemplo de un diagrama de secuencia para una biblioteca.





## **4.8. Diagrama de paquetes**

El objetivo de estos diagramas es obtener una visión más clara del sistema de información orientado a objetos, organizándolo en subsistemas, agrupando los elementos del análisis, diseño o construcción y detallando las relaciones de dependencia entre ellos. El mecanismo de agrupación se denomina Paquete.

Estrictamente hablando, los paquetes y sus dependencias son elementos de los diagramas de casos de uso, de clases y de componentes, por lo que se podría decir que el diagrama de paquetes es una extensión de éstos.

### **4.8.1. Descripción**

Estos diagramas contienen dos tipos de elementos:

- Paquetes: Un paquete es una agrupación de elementos, bien sea casos de uso, clases o componentes. Los paquetes pueden contener a su vez otros paquetes anidados que en última instancia contendrán alguno de los elementos anteriores.
- Dependencias entre paquetes: Existe una dependencia cuando un elemento de un paquete requiere de otro que pertenece a un paquete distinto. Es importante resaltar que las dependencias no son transitivas.

Se pueden optimizar estos diagramas teniendo en cuenta cuestiones como: la generalización de paquetes, el evitar ciclos en la estructura del diagrama, la minimización de las dependencias entre paquetes, etc.

## **4.8.2. Notación**

### **4.8.2.1 Paquete**

Un paquete se representa mediante un símbolo con forma de 'carpeta' en el que se coloca el nombre en la pestaña y el contenido del paquete dentro de la 'carpeta'. En los casos en que no sea visible el contenido del paquete se podrá colocar en su lugar el nombre.

Si el paquete tiene definido un estereotipo, éste se representa encima del nombre entre el símbolo << ... >>, y si se definen propiedades, se representan debajo del nombre y entre llaves.

La visibilidad de los elementos que forman el paquete se debe indicar anteponiendo a su nombre los símbolos: '+' para los públicos, '-' para los privados y '#' para los protegidos.

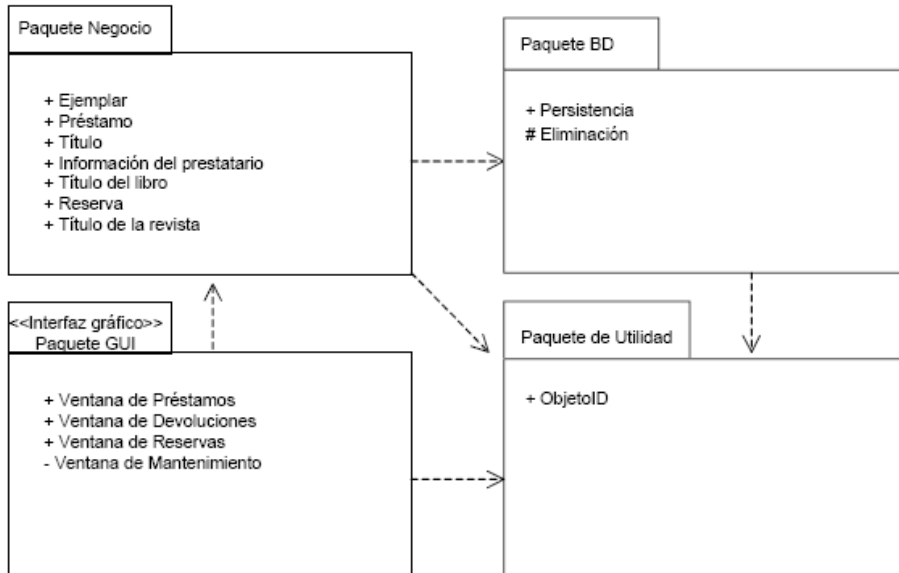
### **4.8.2.2 Dependencia**

Las dependencias se representan con una flecha discontinua con inicio en el paquete que depende del otro.

### **4.8.2.3 Ejemplo**

Sistema encargado de la gestión de los préstamos y reservas de libros y revistas en una biblioteca.

## 19 . Ejemplo de un diagrama de paquetes.



[11]



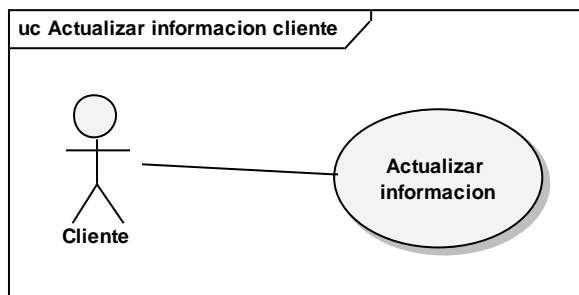
## 5. ESPECIFICACIÓN DE CASOS DE USO

### 5.1. Propósito

En esta sección del documento se describen brevemente los flujos básicos y alternativos de los casos de uso del sistema.

### 5.2. Actualizar información de cliente

#### 20 . Diagrama de caso de uso, actualizar informacion de cliente.



#### 5.2.1. Descripción

Es cuando el usuario desea ingresar a nuestro sitio para realizar un cambio en sus datos que se encuentran almacenados.

## **5.2.2. Flujo básico**

- Ingresar cambios en información
- Recibe página con información modificada

## **5.2.3. Flujo alternativo**

### **5.2.3.1 Tipo de datos erróneo**

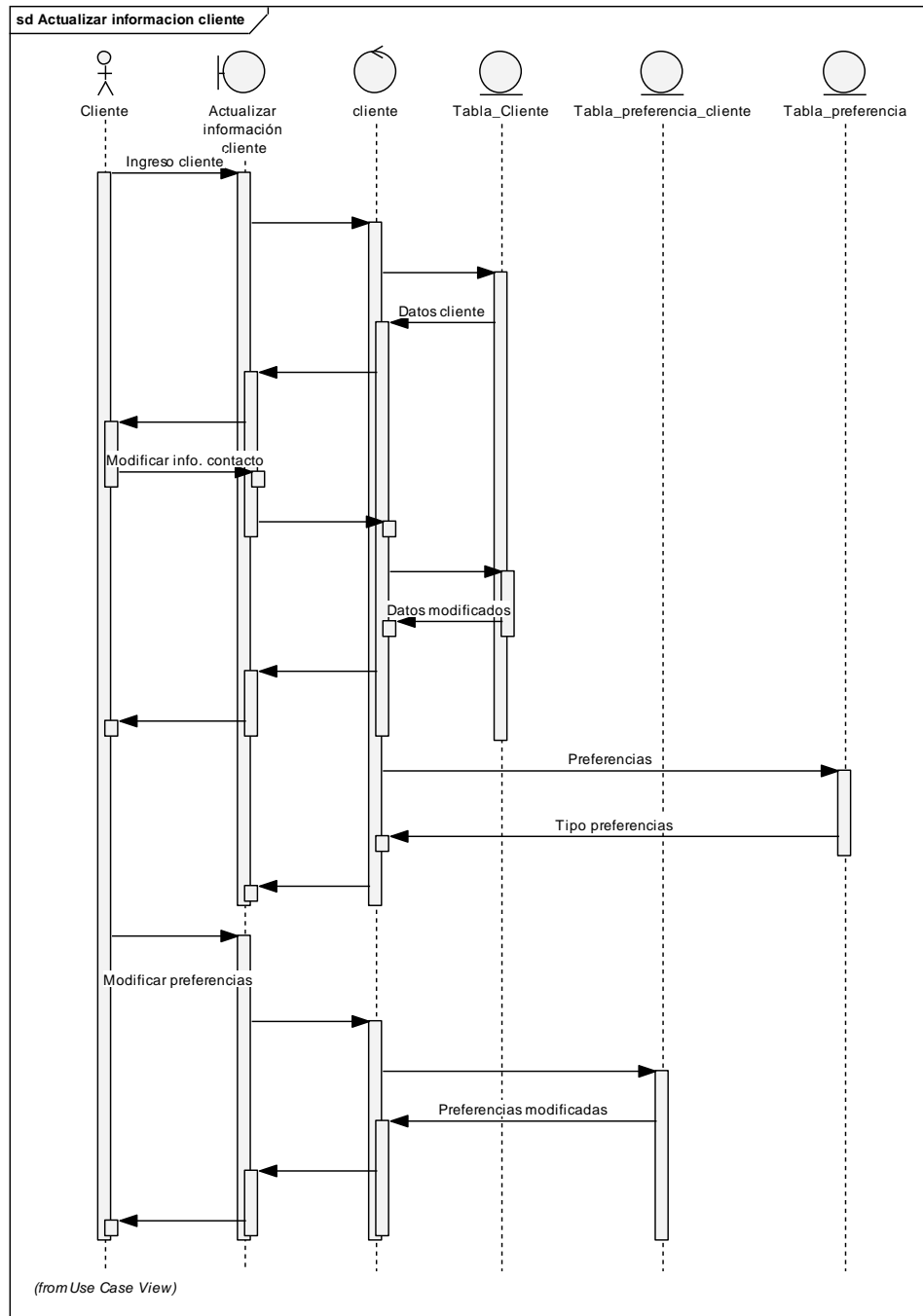
Es cuando un usuario en el paso 1 ingresa valores no permitidos en los campos de la información.

## **5.2.4. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

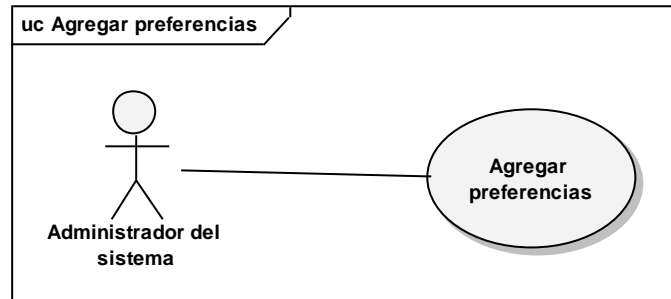
## 5.2.5. Realización del caso de uso

### 21 . Realizacion del caso de uso, actualizar informacion de cliente.



### 5.3. Agregar preferencias

#### 22 . Diagrama de caso de uso, agregar preferencias.



#### 5.3.1. Descripción

Es cuando el administrador del sistema desea ingresar nuevas preferencias para que los usuarios puedan seleccionar y guardarlas como sus preferencias.

#### 5.3.2. Flujo básico

- Ingresar nueva preferencias
- Recibe una página informando que las preferencias se almacenaron



### **5.3.3. Flujo alternativo**

#### **5.3.3.1 Preferencia existe**

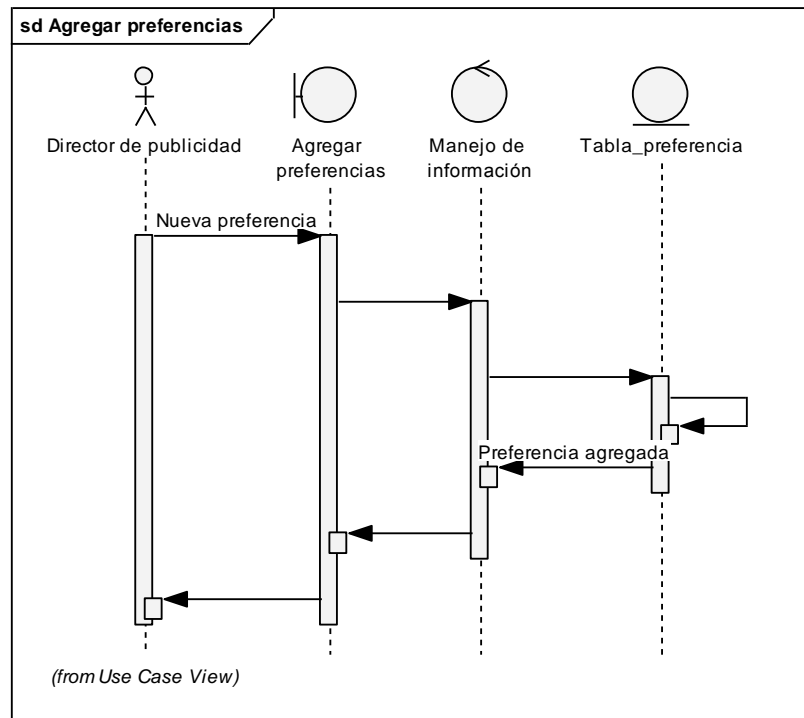
Es cuando en el paso 1 ingresa una preferencia que ya existe en el sistema.

### **5.3.4. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

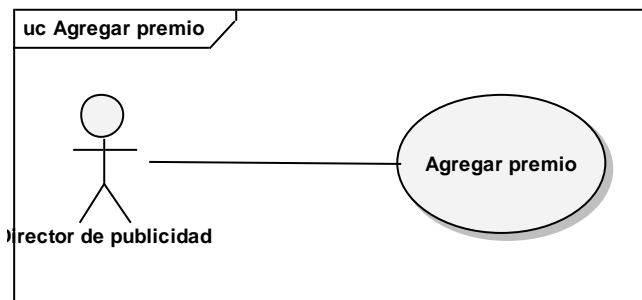
### 5.3.5. Realización del caso de uso

#### 23 . Realizacion de caso de uso, agregar preferencias.



### 5.4. Agregar premio

#### 24 . Diagrama de caso de uso, agregar premio.



#### **5.4.1. Descripción**

Es cuando el Director de publicidad desea ingresar un nuevo tipo de premio que es utilizado cuando se crea un sorteo.

#### **5.4.2. Flujo básico**

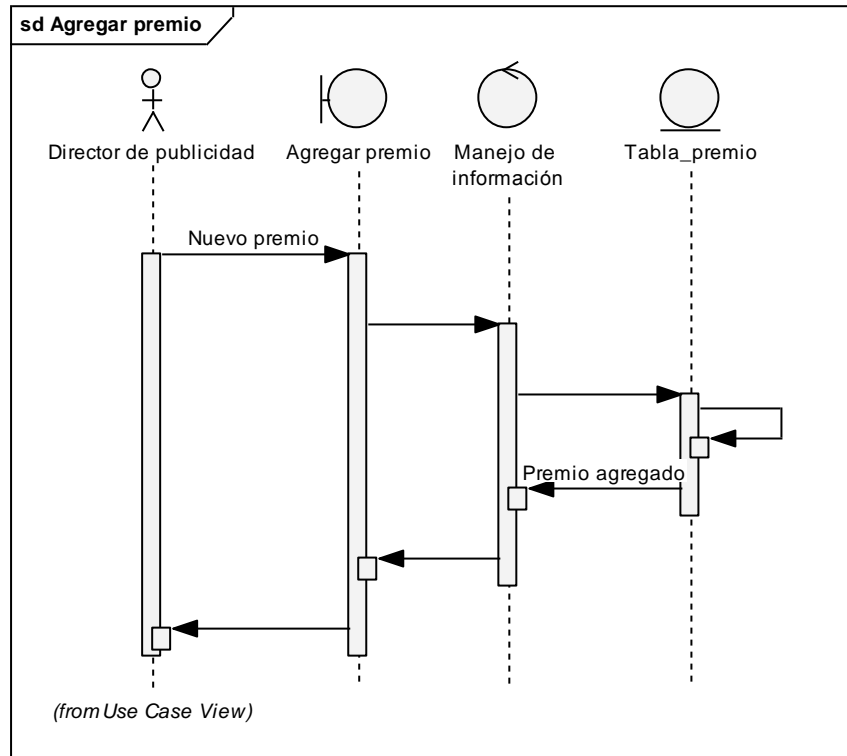
- Ingresar nuevo premio
- Recibe una página informando que se almaceno el premio

#### **5.4.3. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

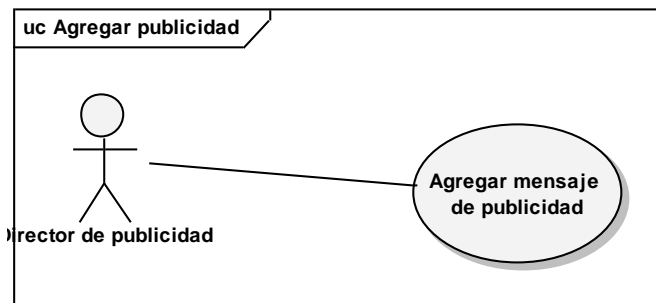
#### 5.4.4. Realización del caso de uso

#### 25 . Realización de caso de uso, agregar premio.



#### 5.5. Agregar mensaje de publicidad

#### 26 . Diagrama de caso de uso, agregar mensaje publicidad.



### **5.5.1. Descripción**

Es cuando se desea crear un nuevo mensaje de publicidad el cual puede ser enviado a los clientes por medio de mensajes a celular, e-mail, etc.

### **5.5.2. Flujo básico**

- Ingresar nuevo mensaje
- Recibe página informando que mensaje fue almacenado

### **5.5.3. Flujo alternativo**

#### **5.5.3.1 Mensaje excede tamaño mínimo**

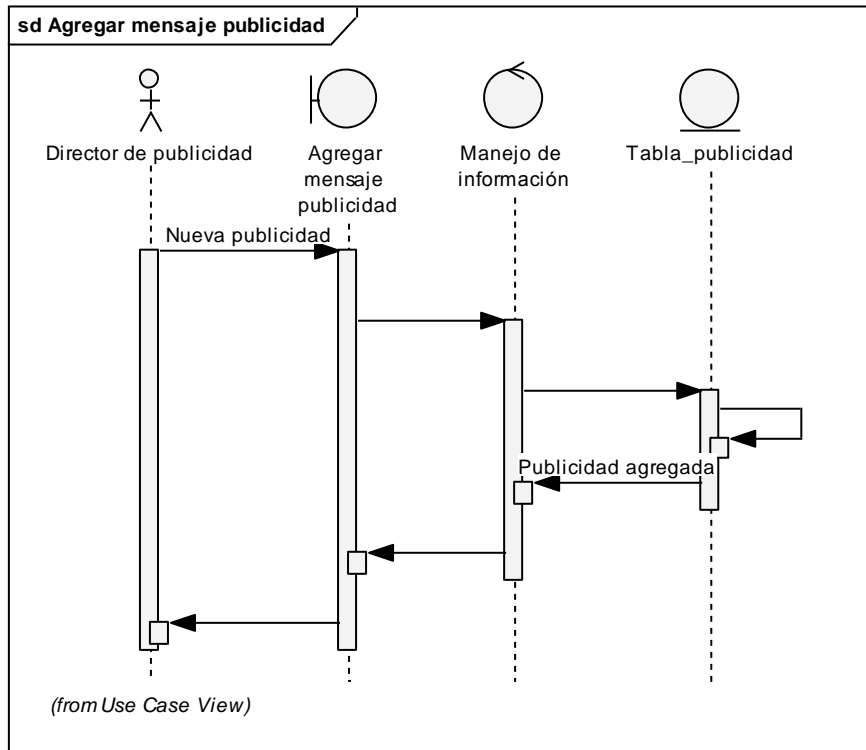
Es cuando en el paso 1 se ingresa un mensaje de mayor tamaño del que se tiene capacidad de almacenar.

### **5.5.4. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

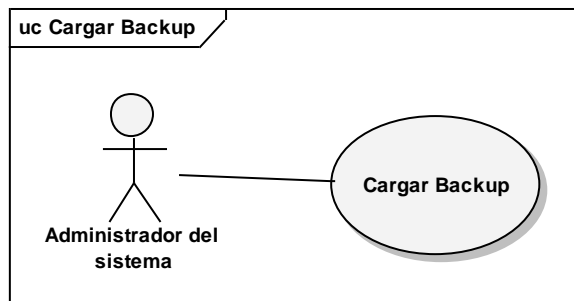
5.5.5. Realización del caso de uso

27 . Realización de caso de uso, agregar mensaje publicidad.



5.6. Cargar backup

28 . Diagrama de caso de uso, cargar backup.



### **5.6.1. Descripción**

Es cuando el administrador del sistema desea cargar los datos del sistema de un backup realizado previamente por un fallo en la capa de datos y hubo una perdida de datos.

### **5.6.2. Flujo básico**

- Ingresar nombre de archivo de entrada
- Recibe página de carga de datos terminada

### **5.6.3. Flujo alternativo**

#### **5.6.3.1 Archivo no existe**

Es cuando en el paso 1 se ingresa un archivo que no existe en el sistema.

#### **5.6.3.2 Archivo corrupto**

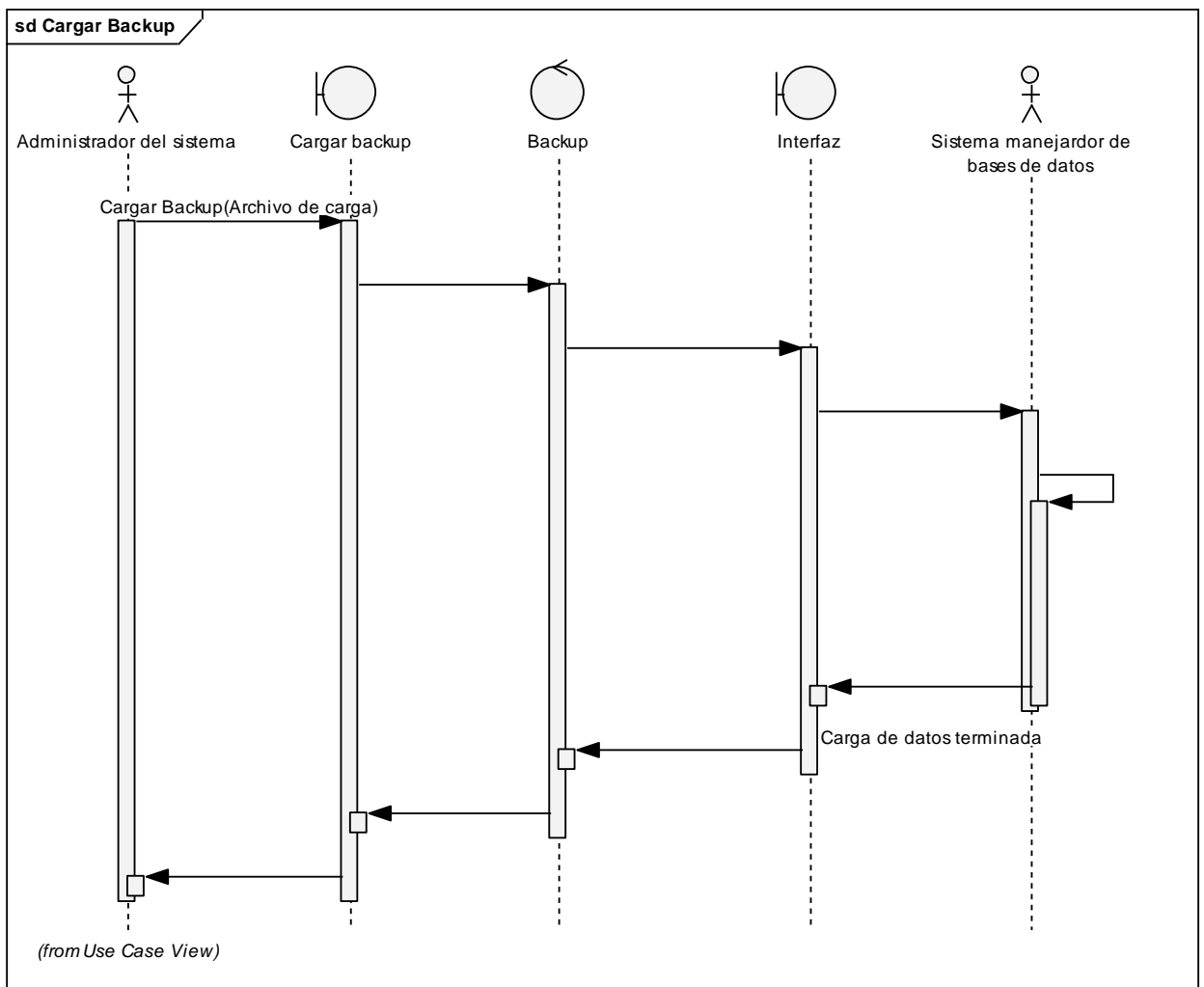
Es cuando en el paso 1 se ingresa un archivo que existe en el sistema pero que su contenido esta corrupto y no se puede realizar la carga normalmente.

#### 5.6.4. Precondiciones

- El usuario debe estar logueado.
- El sistema debe estar fuera de línea para los usuarios hasta que termine la carga de datos.

#### 5.6.5. Realización del caso de uso

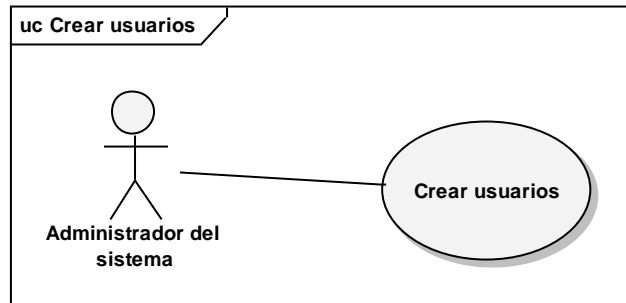
##### 29 . Realizacion de caso de uso, cargar backup.





## 5.7. Crear usuarios

### 30 . Diagrama de caso de uso, crear usuario.



#### 5.7.1. Descripción

Es cuando el administrador del sistema crea un usuario nuevo ingresando todos los datos necesarios.

#### 5.7.2. Flujo básico

- Ingresar información de usuario
- Recibe página informando que se creo usuario

#### 5.7.3. Flujo alternativo

##### 5.7.3.1 Usuario, password ya existe

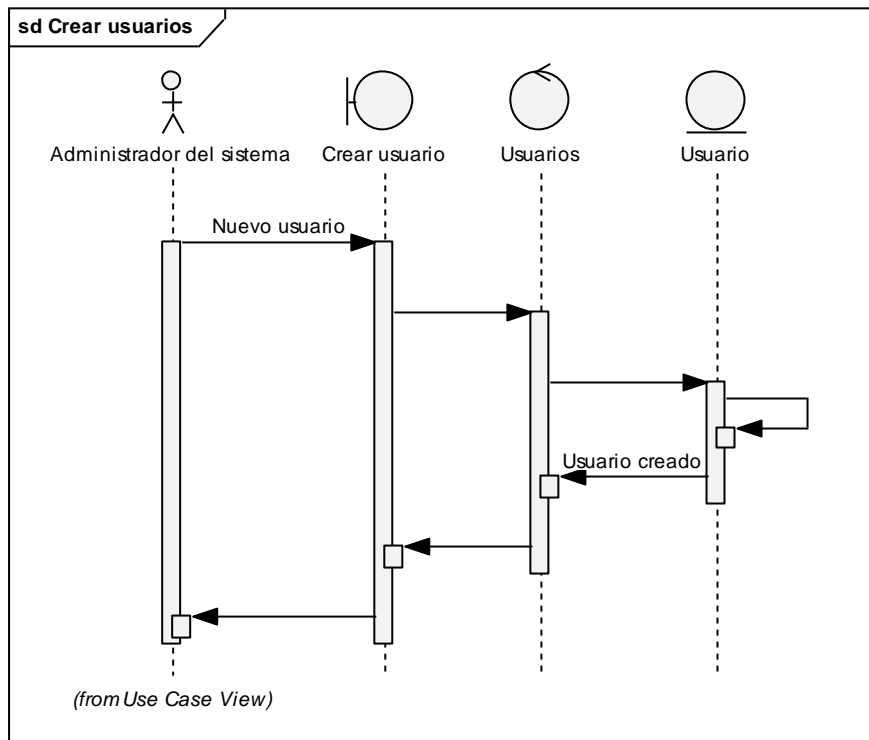
Es cuando en el paso 1 se ingresa un usuario o password que ya existe en el sistema.

#### 5.7.4. Precondiciones

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

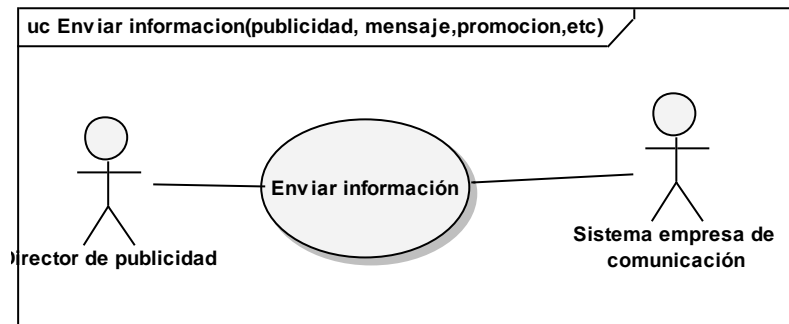
#### 5.7.5. Realización del caso de uso

### 31 . Diagrama de realización de caso de uso, crear usuario.



## 5.8. Enviar información a clientes

### 32 . Diagrama de caso de uso, enviar informacion a clientes.



#### 5.8.1. Descripción

Es cuando el director de publicidad envía información (publicidad, mensaje, promoción, etc.) a los clientes por medio de mensaje a celular, e-mail, etc.

#### 5.8.2. Flujo básico

- Ingresar información a enviar
- Ingresar el medio por el cual se desea enviar (celular, e-mail, etc.)
- Recibe página informando que la información fue enviada

### **5.8.3. Flujo alternativo**

#### **5.8.3.1 El sistema de la empresa de comunicación esta fuera de línea**

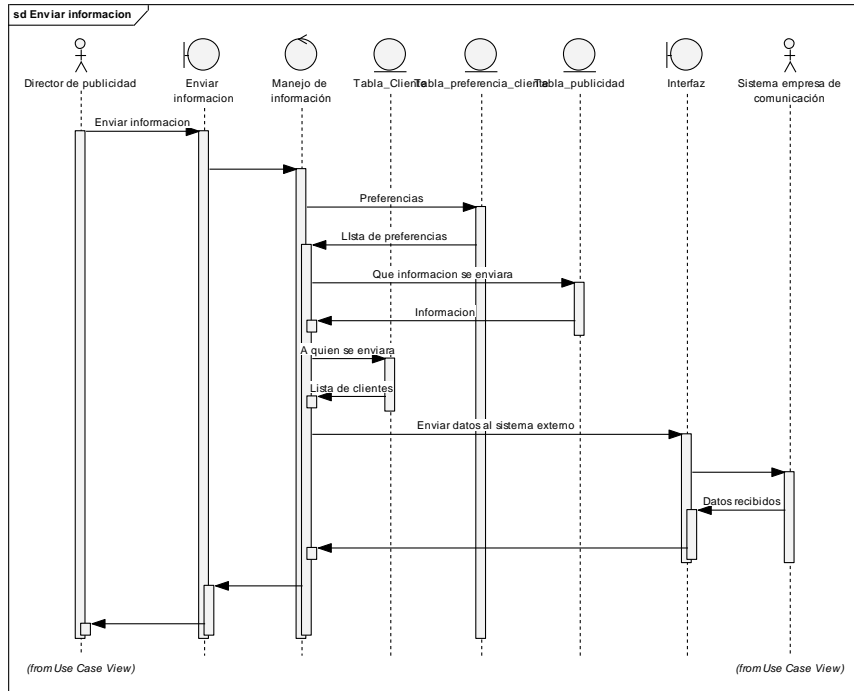
Es cuando el sistema de la empresa de comunicación esta fuera de línea y no se puede enviar el mensaje.

### **5.8.4. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

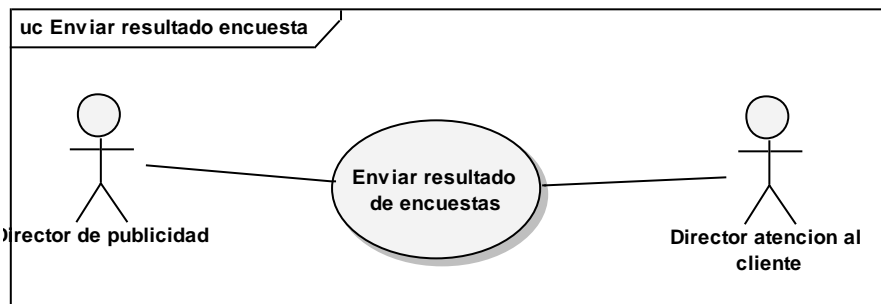
5.8.5. Realización del caso de uso

33 . Diagrama de realización de caso de uso, enviar información a clientes.



5.9. Enviar resultado de encuestas

34 . Diagrama de caso de uso, enviar resultado de encuesta.



### **5.9.1. Descripción**

Es cuando el director de publicidad desea hacer llegar la información tabulada de las encuestas que fueron llenadas por los clientes al director de atención al cliente.

### **5.9.2. Flujo básico**

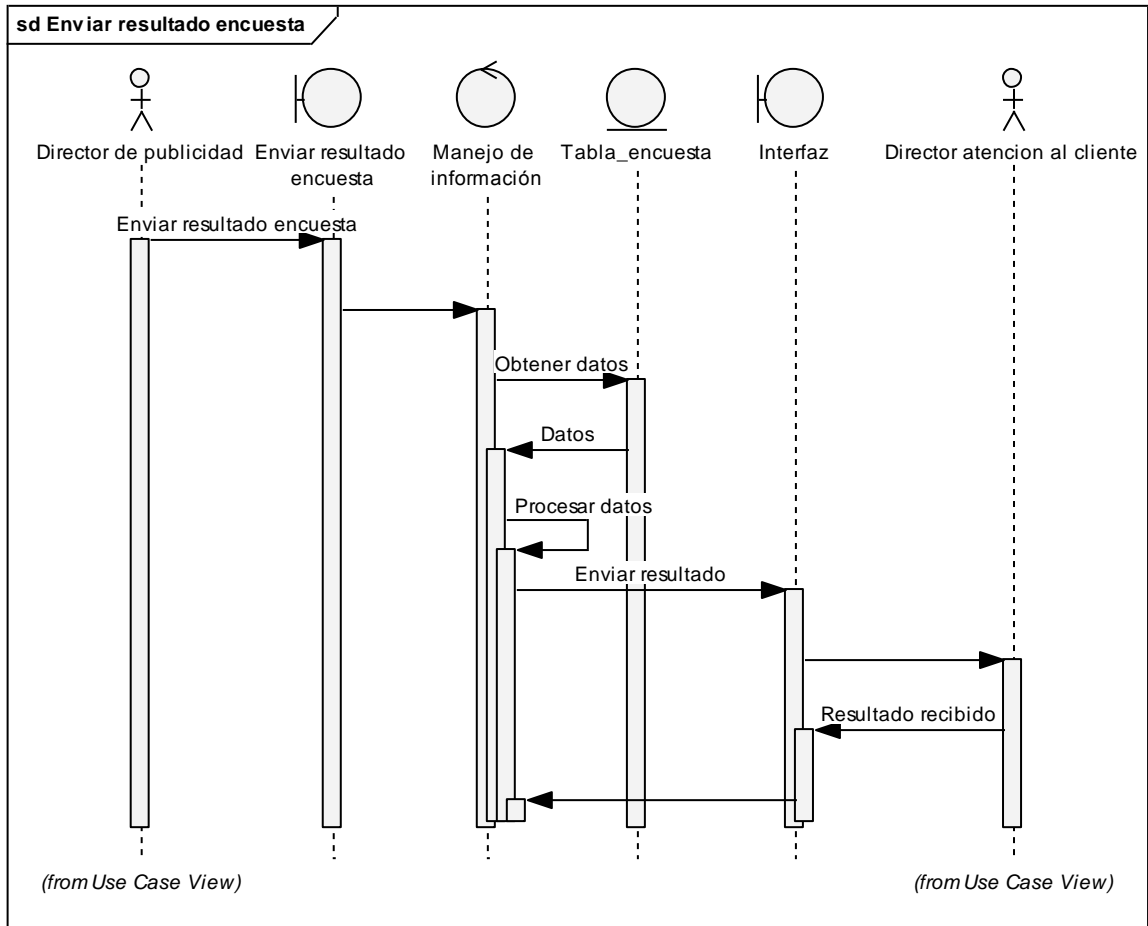
- Ingresar nombre de archivo de salida
- Recibe página informando que fue enviada la información

### **5.9.3. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

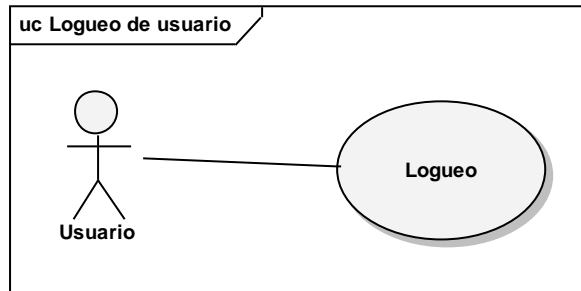
5.9.4. Realización del caso de uso

35 . Diagrama de realización de caso de uso, enviar resultado de encuesta.



## 5.10. Logueo de usuario

### 36 . Diagrama de caso de uso, logueo de usuario.



#### 5.10.1. Descripción

Es cuando el usuario desea ingresar a nuestro sitio para realizar un proceso como realizar una modificar preferencias, crear usuario, etc.

#### 5.10.2. Flujo básico

- Ingresar nombre de usuario y password
- Recibe pagina de bienvenida

#### 5.10.3. Flujo alternativo

##### 5.10.3.1 Usuario no existe

Es cuando un usuario en el paso 1 no esta registrado en la base de datos del sistema y debe registrarse.

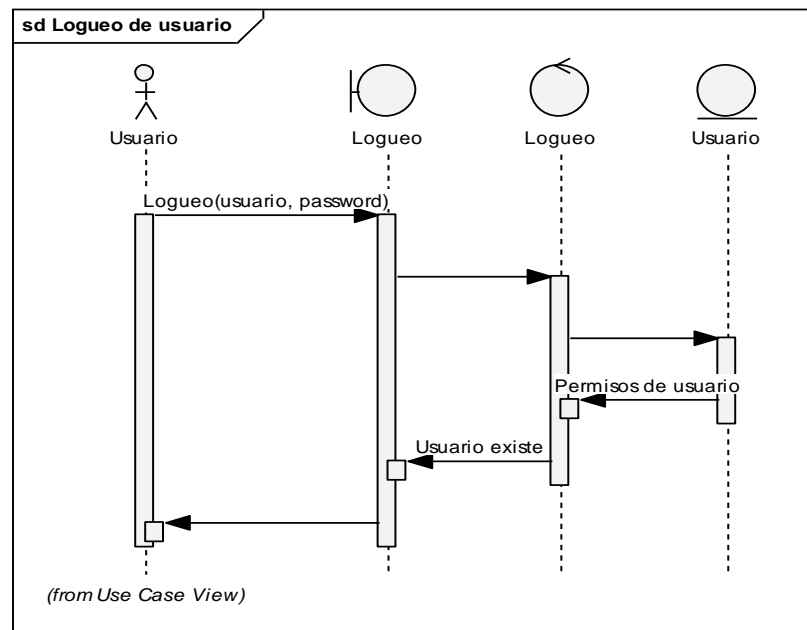


#### 5.10.4. Precondiciones

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

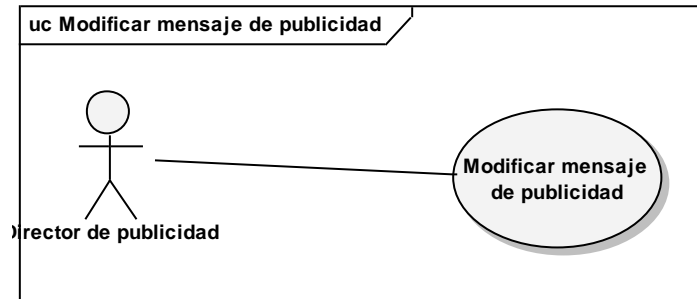
#### 5.10.5. Realización del caso de uso

### 37 . Diagrama de realización de caso de uso, logueo de usuario.



## 5.11. Modificar mensaje de publicidad

### 38 . Diagrama de caso de uso, modificar mensaje de publicidad.



#### 5.11.1. Descripción

Es cuando el director de publicidad desea modificar un mensaje de publicidad existente en el sistema.

#### 5.11.2. Flujo básico

- Se le muestran los mensajes
- Ingresar mensaje a modificar
- Recibe página informando que fue modificado el mensaje

### **5.11.3. Flujo alternativo**

#### **5.11.3.1 Mensaje excede tamaño mínimo**

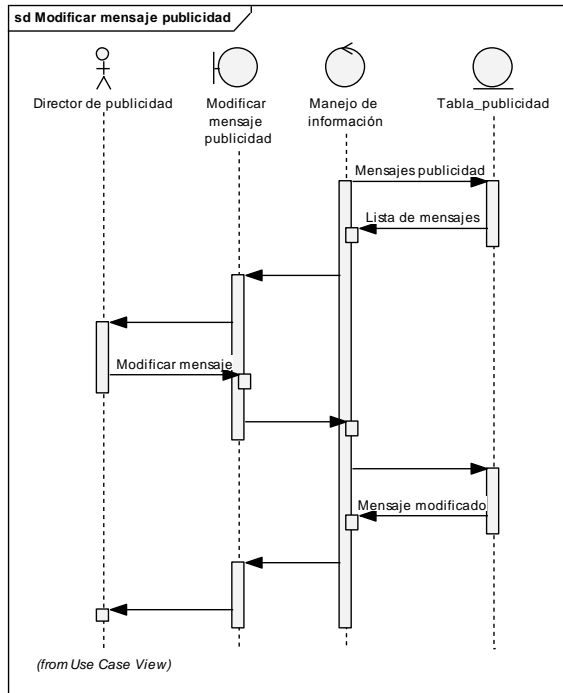
Es cuando en el paso 1 se ingresa un mensaje de mayor tamaño del que se tiene capacidad de almacenar.

#### **5.11.4. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

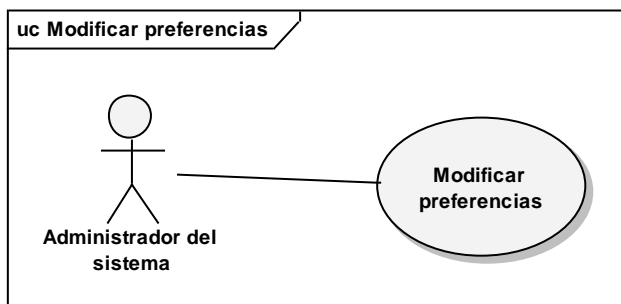
### 5.11.5. Realización del caso de uso

#### 39 . Diagrama de realización de caso de uso, modificar mensaje de publicidad.



### 5.12. Modificar preferencias

#### 40 . Diagrama de caso de uso, modificar preferencias.



### **5.12.1. Descripción**

Es cuando el administrador del sistema ingresa al sitio para modificar una preferencia almacenada en el sistema.

### **5.12.2. Flujo básico**

- Se le muestran las preferencias
- Ingresar preferencia a modificar
- Recibe la pagina informando que fue almacenado el cambio

### **5.12.3. Flujo alternativo**

### **5.12.4. Preferencia existe**

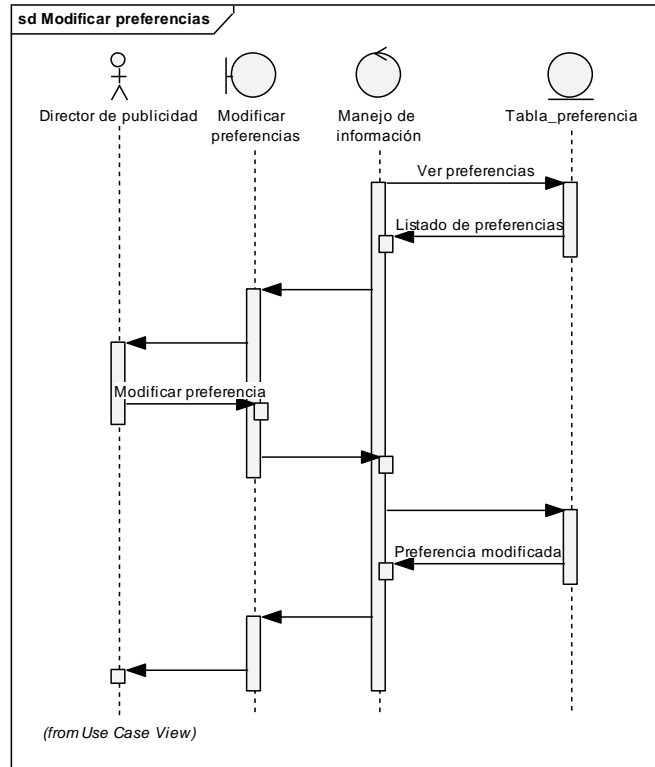
Es cuando en el paso 1 ingresa una preferencia que ya existe en el sistema.

### **5.12.5. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

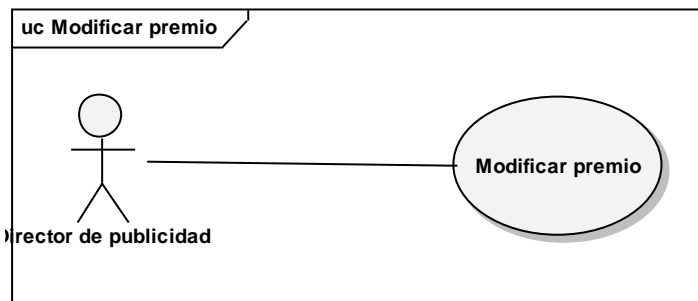
### **5.12.6. Realización del caso de uso**

**41 . Diagrama de realización de caso de uso, modificar preferencias.**



**5.13. Modificar premio**

**42 . Diagrama de caso de uso, modificar premio.**



### **5.13.1. Descripción**

Es cuando el director de publicidad ingresa al sitio para modificar un premio que se encuentra almacenado.

### **5.13.2. Flujo básico**

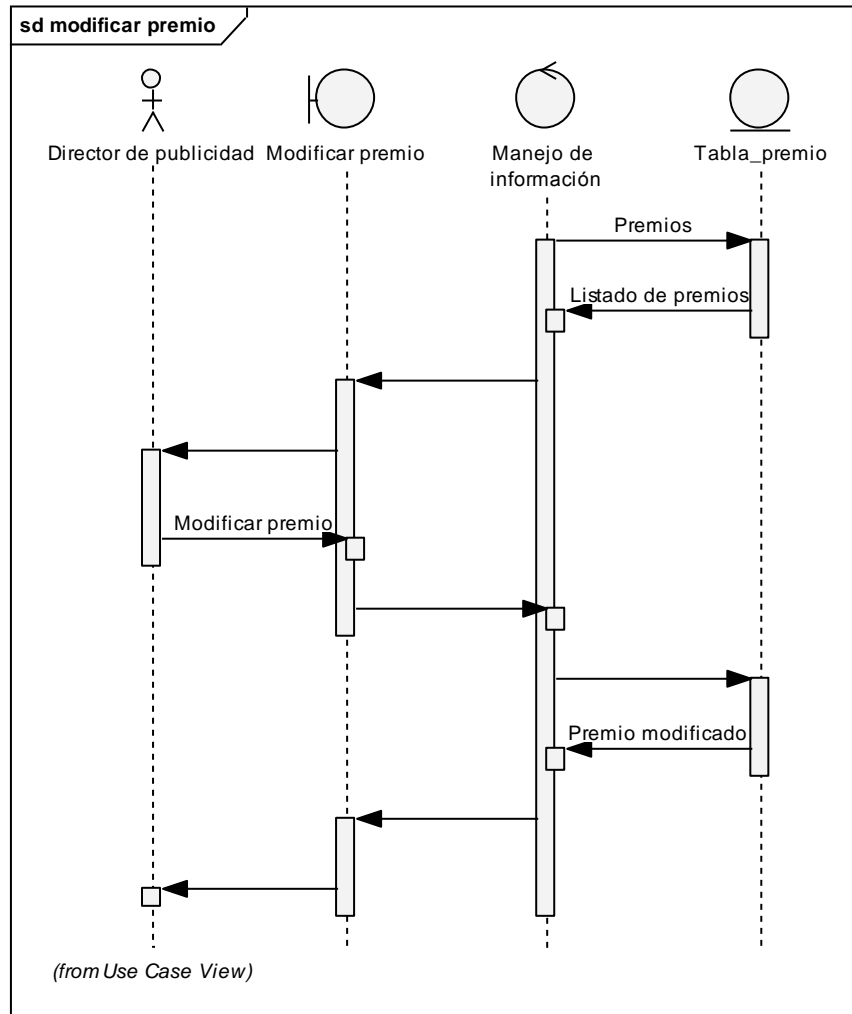
- Se le muestran los premios almacenados
- Ingresar premio a modificar
- Recibe página informando que premio fue modificado

### **5.13.3. Precondiciones**

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

### 5.13.4. Realización del caso de uso

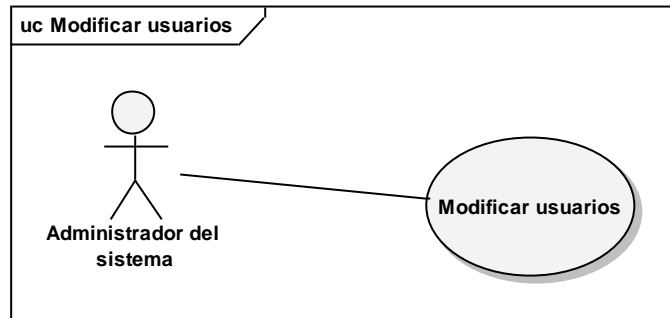
#### 43 . Diagrama de realización de caso de uso, modificar premio.





## 5.14. Modificar usuarios

### 44 . Diagrama de caso de uso, modificar usuarios.



#### 5.14.1. Descripción

Es cuando el administrador del sistema desea modificar los usuarios existentes en sistema.

#### 5.14.2. Flujo básico

- Se le muestran los usuarios
- Ingresar usuario a modificar
- Recibe página informando que fue modificado el cliente

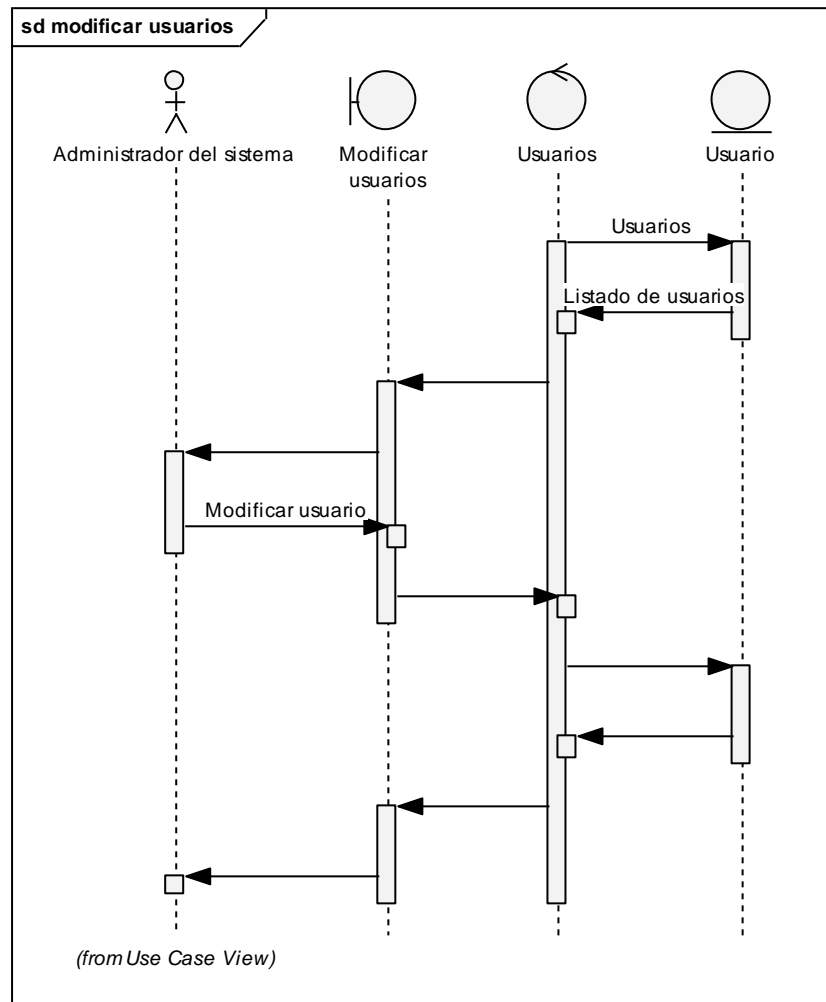
#### 5.14.3. Precondiciones

- El sistema debe de estar en línea.

- El usuario debe estar logueado.

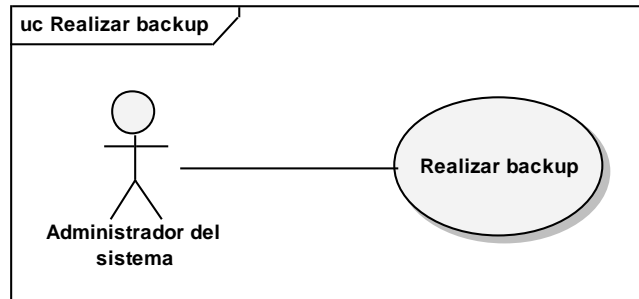
#### 5.14.4. Realización del caso de uso

#### 45 . Diagrama de realización de caso de uso, modificar usuarios.



## 5.15. Realizar backup

### 46 . Diagrama de caso de uso, realizar backup.



#### 5.15.1. Descripción

Es cuando el administrador del sistema desea crear un backup de los datos que se encuentran en el sistema para poder evitar la perdida de datos por un fallo en la capa de datos.

#### 5.15.2. Flujo básico

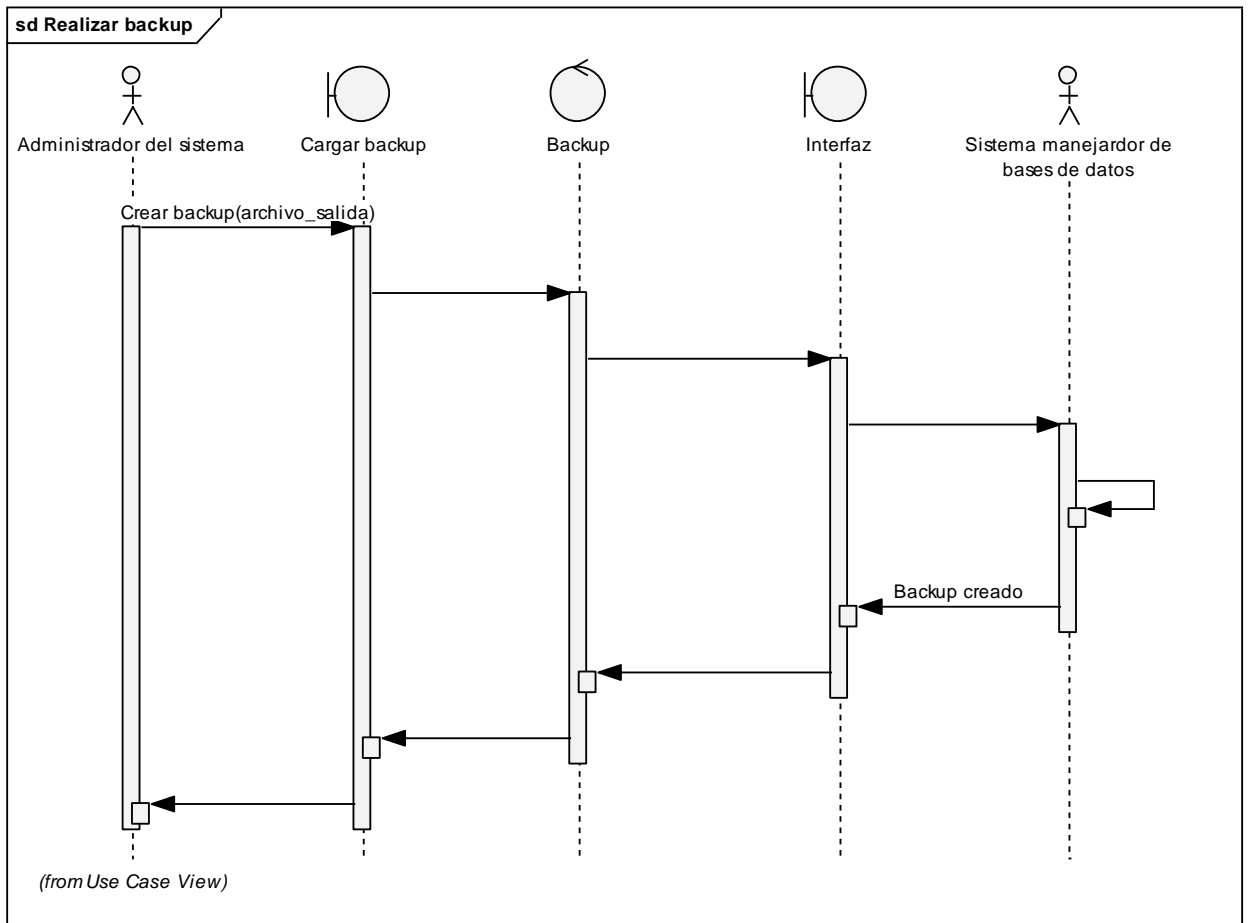
- Ingresar nombre de archivo de salida
- Recibe página informando que fue creado el backup

#### 5.15.3. Precondiciones

- El sistema debe estar fuera de línea.
- El usuario debe estar logueado.

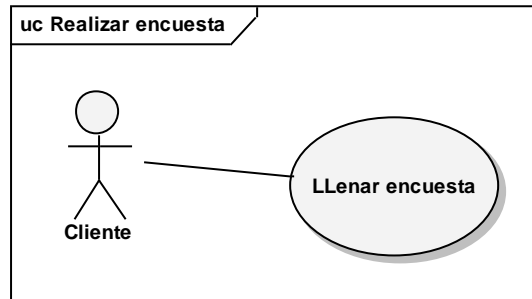
#### 5.15.4. Realización del caso de uso

47 . Diagrama de realización de caso de uso, realizar backup.



## 5.16. Llenar encuesta

### 48 . Diagrama de caso de uso, llenar encuesta.



#### 5.16.1. Descripción

Es cuando el cliente ingresa a nuestro sitio para llenar una encuesta.

#### 5.16.2. Flujo básico

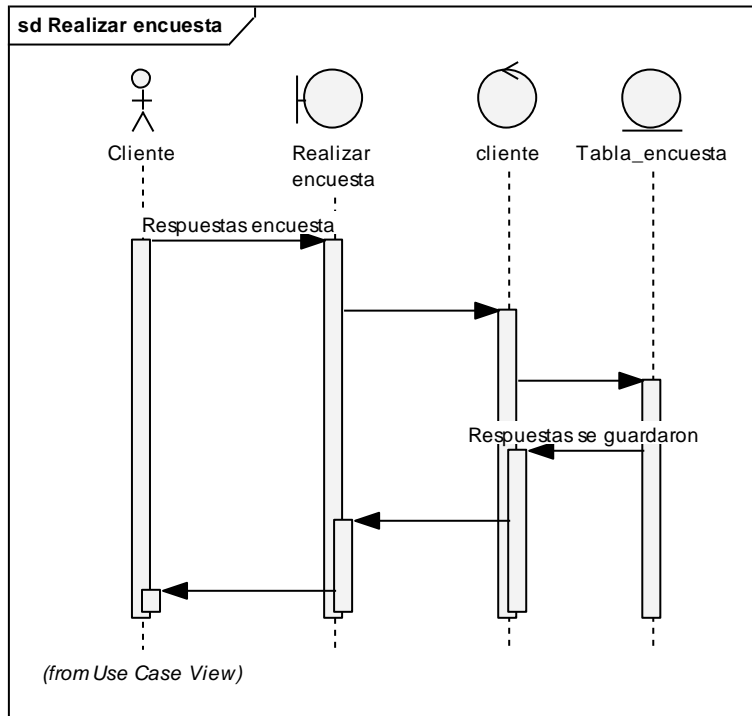
- Ingresar repuestas en encuesta
- Recibe pagina agradeciendo su colaboración

#### 5.16.3. Precondiciones

- El sistema debe de estar en línea.

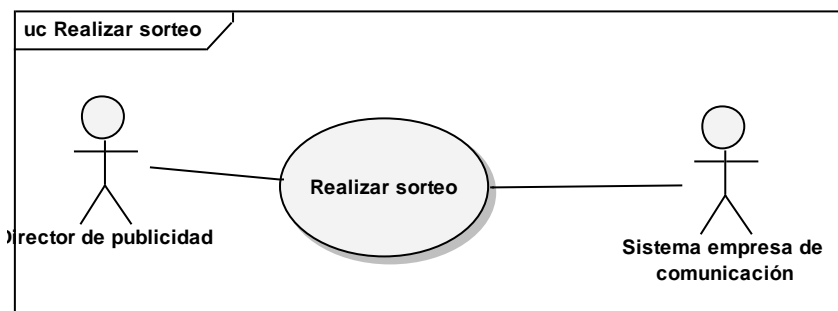
#### 5.16.4. Realización del caso de uso

#### 49 . Diagrama de realización de caso de uso, realizar encuesta.



#### 5.17. Realizar sorteo

#### 50 . Diagrama de caso de uso, realizar sorteo.



### **5.17.1. Descripción**

Es cuando el director ingresa al sitio para crear un sorteo y se le comunica al usuario que ganó el sorteo la información necesaria para recoger el premio.

### **5.17.2. Flujo básico**

- Ingresar premio de sorteo
- Ingresar medio de comunicación (celular, e-mail, etc.)
- Recibe página informando al ganador

### **5.17.3. Flujo Alternativo**

#### **5.17.3.1 Premio no existe**

Es cuando en el paso 1 se ingresa un premio que no existe.

#### **5.17.3.2 El sistema de la empresa de comunicación esta fuera de línea**

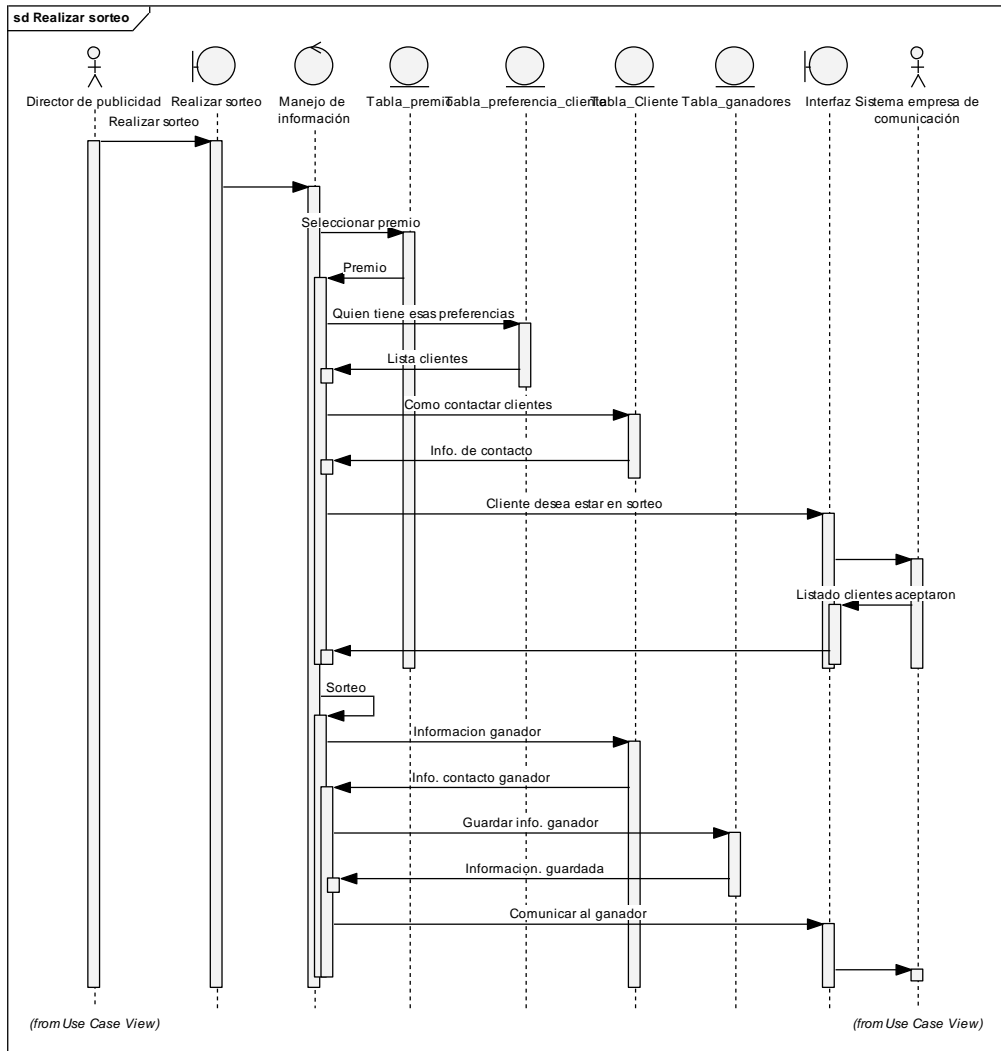
Es cuando el sistema de la empresa de comunicación esta fuera de línea y no se puede enviar el mensaje.

### 5.17.4. Precondiciones

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

### 5.17.5. Realización del caso de uso

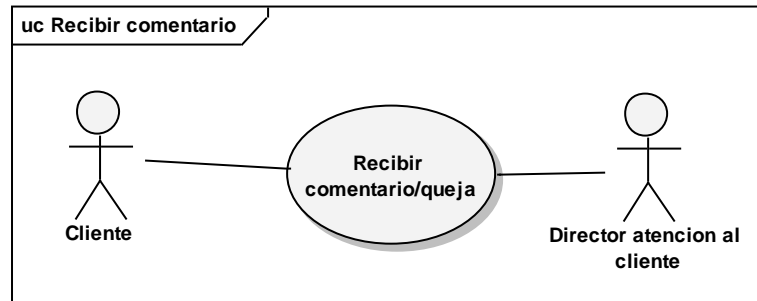
51 . Diagrama de realización de caso de uso, realizar sorteo.





## 5.18. Recibir comentario/queja

### 52 . Diagrama de caso de uso, recibir comentario/queja.



#### 5.18.1. Descripción

Es cuando el cliente ingresa al sitio para enviar un comentario/queja sobre un mensaje publicitario, producto, etc.

#### 5.18.2. Flujo básico

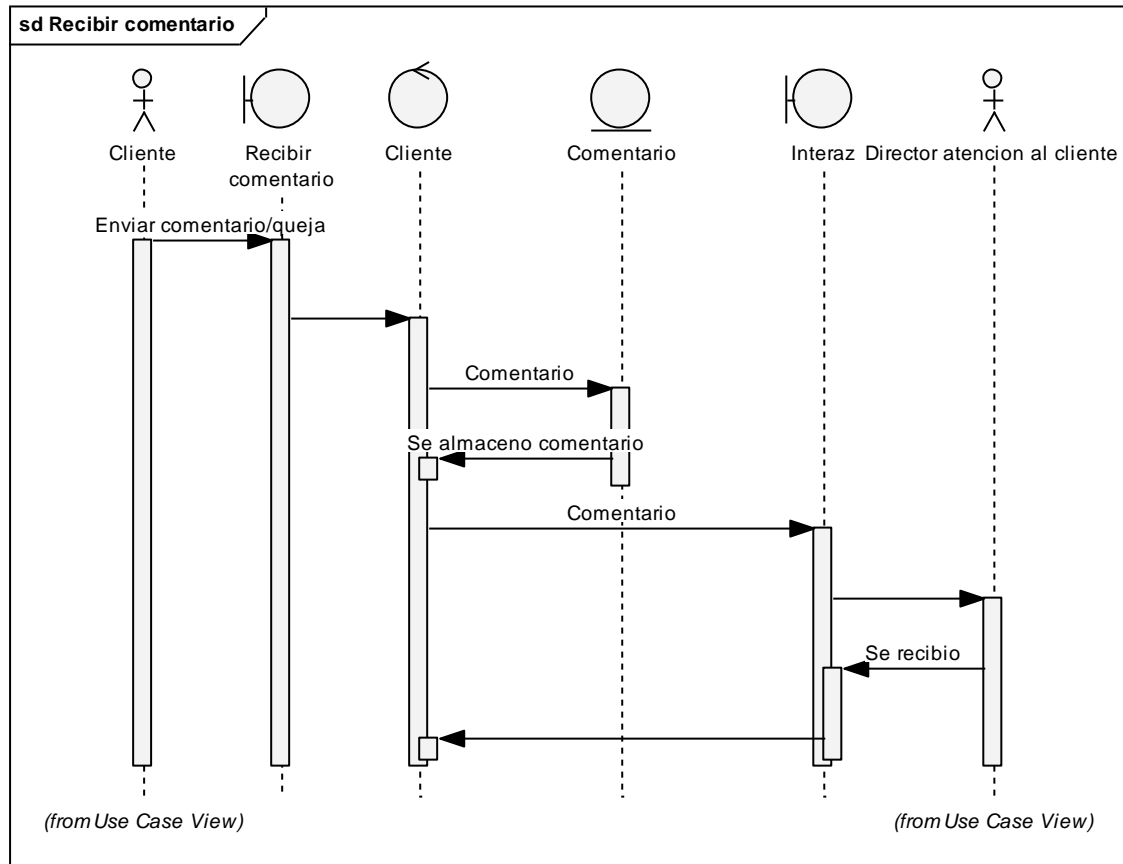
- Ingresar comentario
- Recibe página informando que comentario fue enviado

#### 5.18.3. Precondiciones

- El sistema debe de estar en línea.

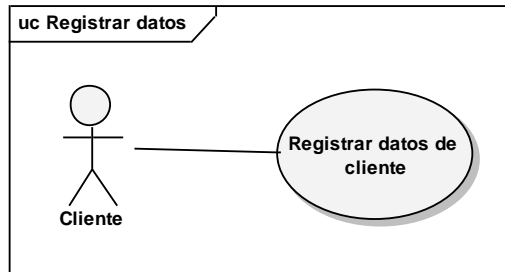
#### 5.18.4. Realización del caso de uso

### 53 . Diagrama de realización de caso de uso, recibir comentario/queja.



## 5.19. Registrar datos de cliente

### 54 . Diagrama de caso de uso, registrar datos de cliente.



#### 5.19.1. Descripción

Es cuando un usuario ingresa al sitio para ingresar sus datos e información de contacto para poder estar comunicado.

#### 5.19.2. Flujo básico

- Ingresar información de usuario
- Recibe página informando que su usuario fue creado

#### 5.19.3. Flujo alternativo

##### 5.19.3.1 Usuario, password ya existe

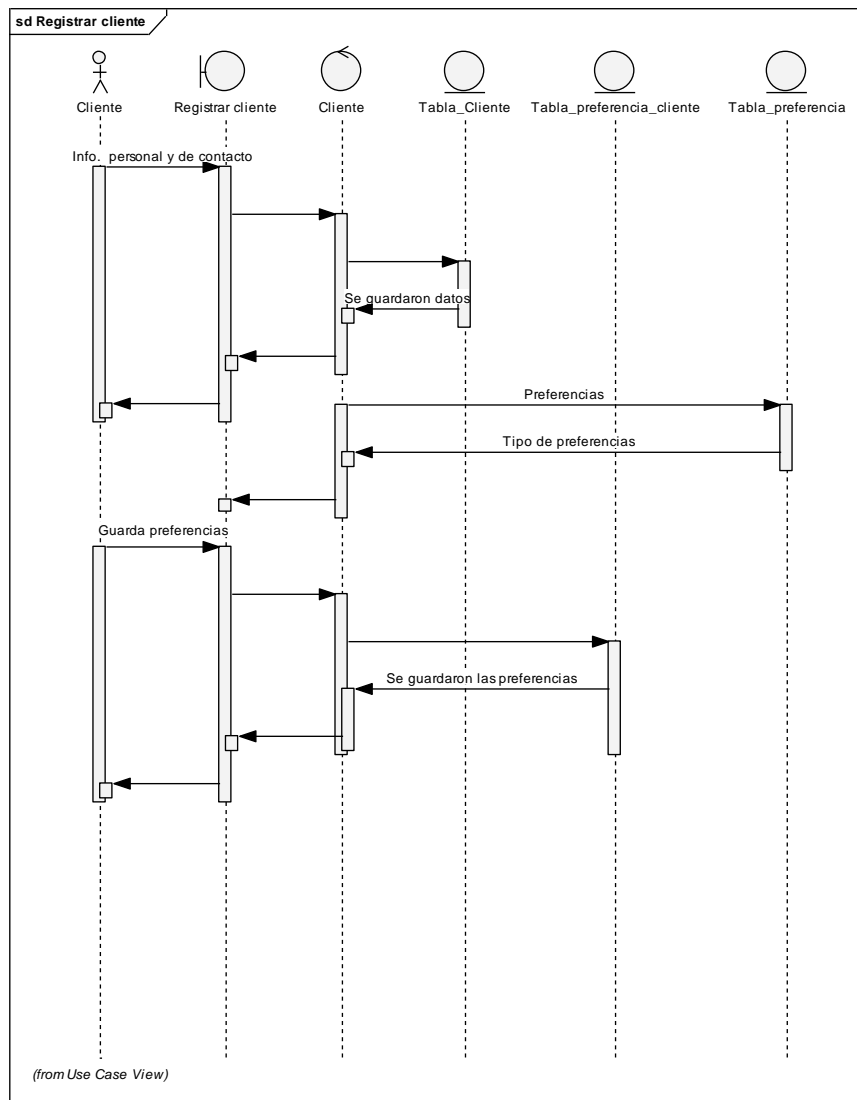
Es cuando en el paso 1 se ingresa un usuario o password que ya existe en el sistema.

#### 5.19.4. Precondiciones

- El sistema debe de estar en línea.

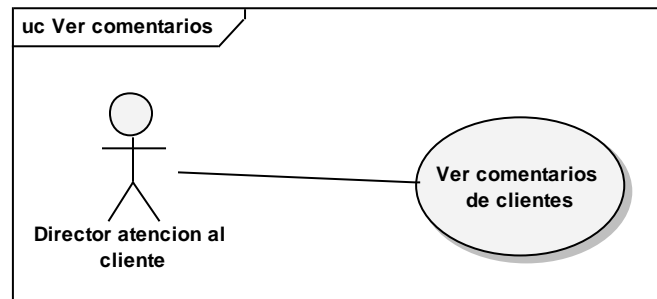
#### 5.19.5. Realización del caso de uso

### 55 . Diagrama de realización de caso de uso, registrar datos de cliente.



## 5.20. Ver comentarios de clientes

### 56 . Diagrama de caso de uso, ver comentarios de cliente.



#### 5.20.1. Descripción

Es cuando el director de atención al cliente se le muestran todos los mensajes enviados por los clientes (No es necesario que estén registrados en el sistema).

#### 5.20.2. Flujo básico

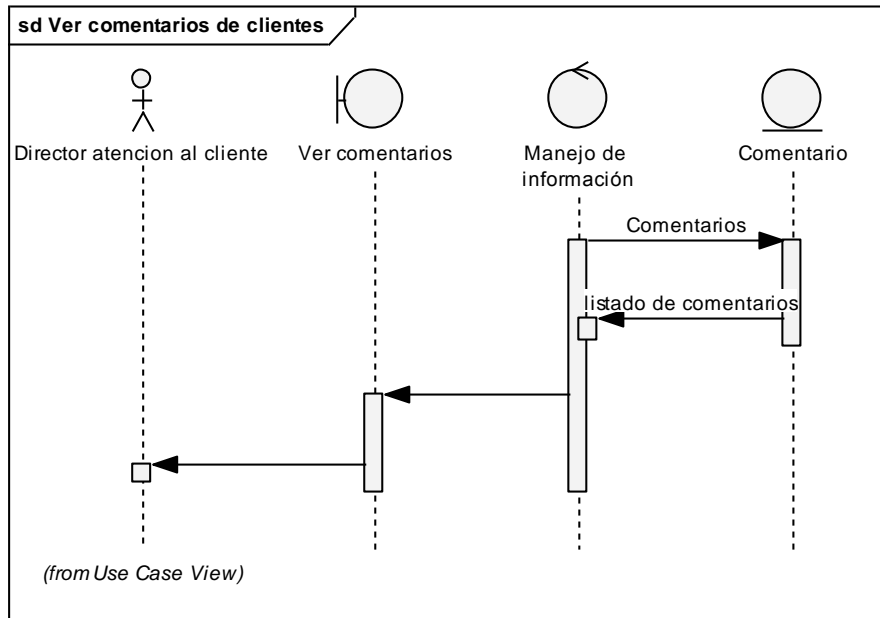
- Recibe pagina con comentarios

#### 5.20.3. Precondiciones

- El sistema debe de estar en línea.
- El usuario debe estar logueado.

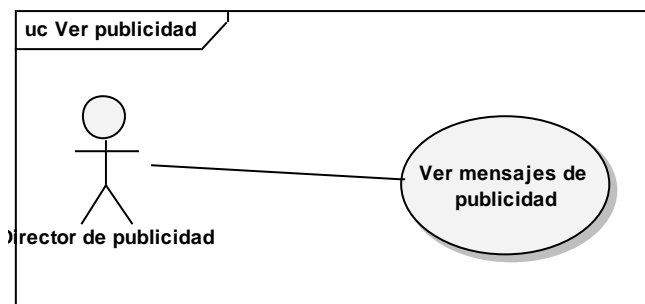
#### 5.20.4. Realización del caso de uso

#### 57. Diagrama de realización de caso de uso, ver comentarios de cliente.



#### 5.21. Ver mensajes de publicidad

#### 58 . Diagrama de caso de uso, ver mensajes de publicidad.



### 5.21.1. Descripción

Es cuando el director de publicidad ingresa al sitio para poder ver todos los mensajes de publicidad que están almacenados en el sistema.

### 5.21.2. Flujo básico

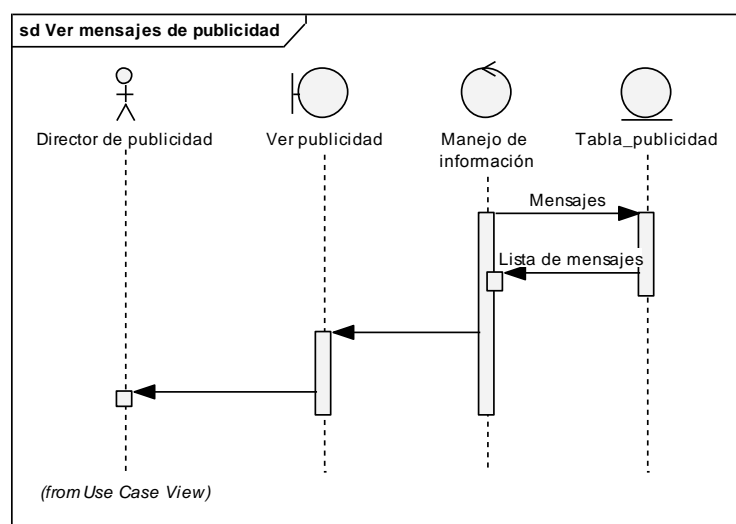
- Recibe página con mensajes de publicidad

### 5.21.3. Precondiciones

- El sistema debe de estar en línea.

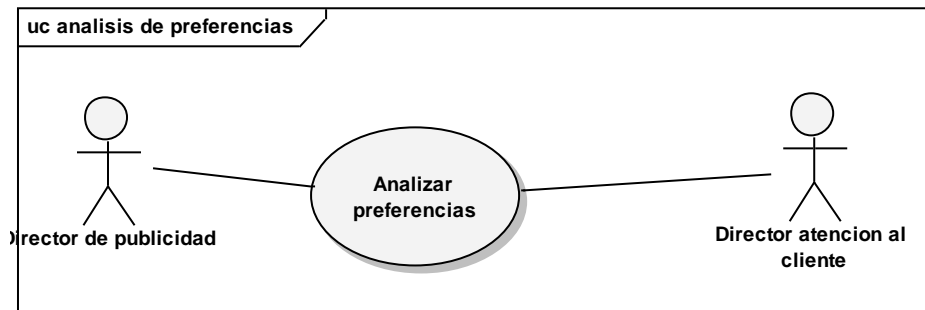
### 5.21.4. Realización del caso de uso

## 59 . Diagrama de realización de caso de uso, ver mensajes publicidad.



## 5.22. Analizar preferencias

### 60 . Diagrama de caso de uso, analizar preferencias.



#### 5.22.1. Descripción

Es cuando el director de publicidad ingresa al sitio para poder hacer un análisis de las preferencias de los clientes almacenadas para poder realizar un estudio.

#### 5.22.2. Flujo básico

- Ingresar tipo análisis
- Recibe pagina con resultado de análisis

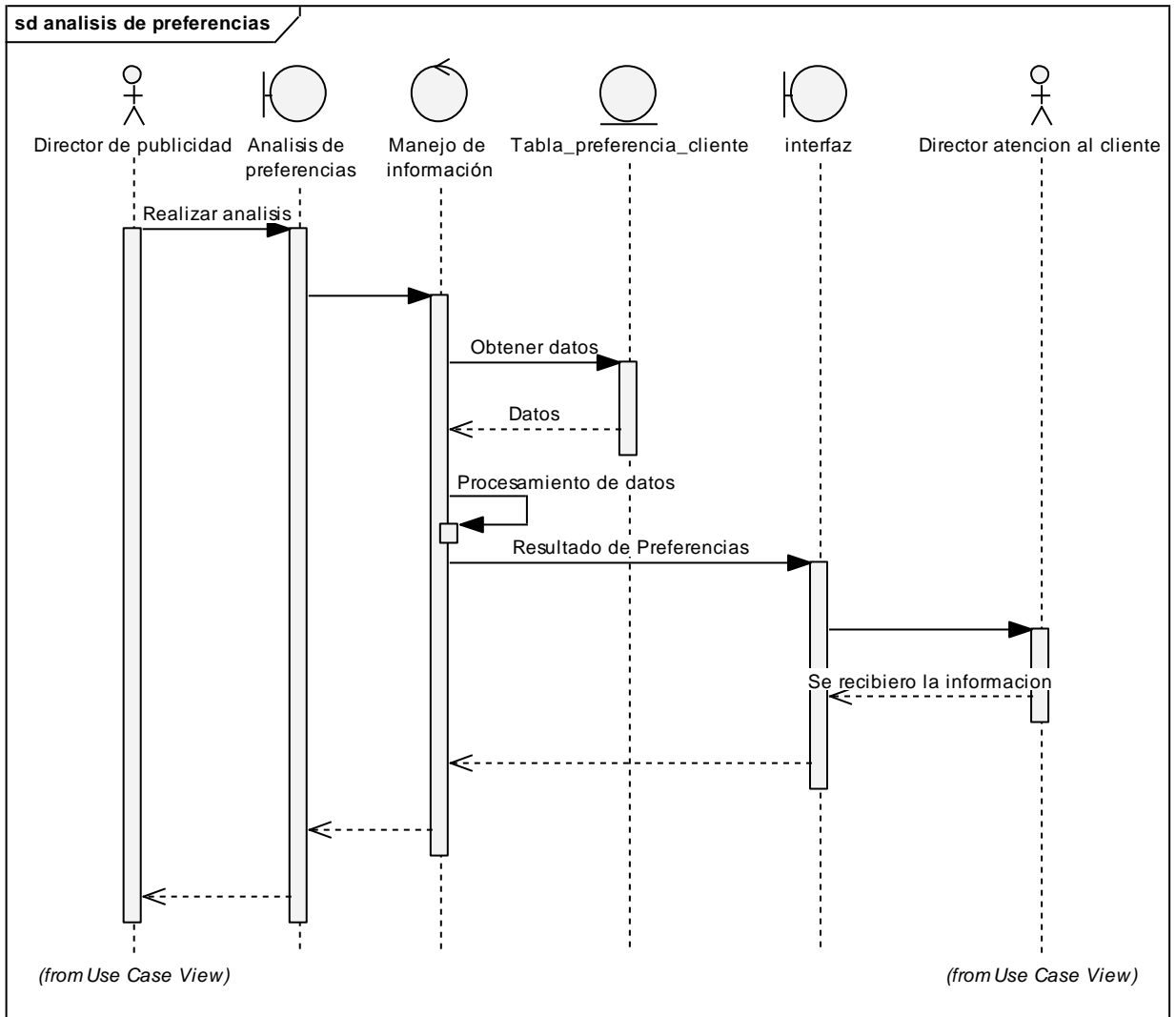
#### 5.22.3. Precondiciones

- El sistema debe de estar en línea.
- El usuario debe estar logueado.



#### 5.22.4. Realización del caso de uso

61 . Diagrama de realización de caso de uso, analizar preferencias.





## 6. ARQUITECTURA DE SOFTWARE

### 6.1. Propósito

Este documento provee un resumen de la arquitectura del sistema, usando un número de diferentes vistas que representan diferentes aspectos del sistema y modelos usando un lenguaje de modelado visual como es UML (Describe una arquitectura de software). Este documento es un intento de capturar y presentar las decisiones de arquitectura que serán hechas en el sistema.

La arquitectura de software le concierne no solo la estructura y comportamiento sino también su uso, funcionalidad, desempeño, reutilización, economía y limitaciones tecnológicas.

#### 6.1.1. Alcance

- Tener un modelo de software provisto de mucha información vital que incremente la calidad del sistema de software.

#### 6.1.2. Definiciones, acrónimos y abreviaturas

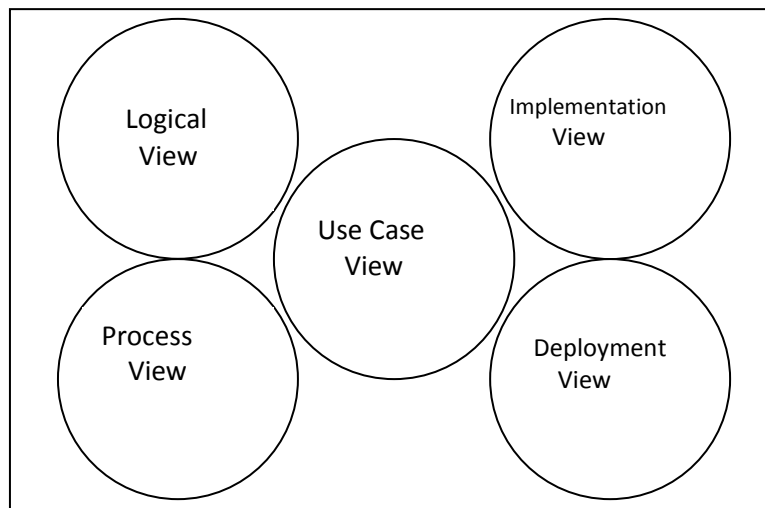
- Arquitectura de Software: Estructura de los componentes de un programa/sistema sus relaciones y principios y guías que gobiernan su diseño y evolución en el tiempo.
- UML (Unified Modeling Language)

- Modelo: Grupo de diversos elementos que tienen vértices conectados por lados en una grafica representando la semántica y estructura de los elementos del modelo.
- Diagrama: Forma grafica de representar un modelo de UML.

## 6.2. Representación de la arquitectura

La arquitectura de software contiene 5 vistas de un sistema, como se muestra en la siguiente figura.

### 62. Vistas para representar una arquitectura.



Las 5 vistas usadas para capturar la arquitectura de software de un sistema.

Las 5 vistas de arquitectura de software:

**Use case** (Casos de Uso): Ilustra las funcionalidades clave del sistema como es percibido por actores externos (personas de otros sistemas) como la comunicación que da a lugar entre los actores y el sistema (Diagramas de casos de uso). Esta vista guía el desarrollo de las otras vistas porque las metas del sistema se muestran en los casos de uso, que son usados para validar otras vistas.

**Logical** (Lógica): Muestra la realización de los requerimientos funcionales como el grupo de clases en paquetes (Diagramas de clases con paquetes), las clases en la arquitectura y sus relaciones (Diagramas de clases), su estado y comportamiento (Diagramas Statechart), y la colaboración (Diagramas de secuencia, Diagrama de colaboración, o Diagramas de actividades).

**Deployment** (Producción): Describe el sistema en términos de nodos físicos (i.e., computadoras y devices) y coloca los componentes y procesos en esos nodos (Diagramas de deployment). Los componentes que implementan las clases en la vista lógica. La vista deployment muestra la topología del sistema.

**Implementation** (Implementación): Estructura el desarrollo de el sistema en términos de módulos de código, en código y en módulos binarios, como son librerías y ejecutables (Diagramas de componentes). La vista de implementación también maneja la configuración y administración de versiones y la estrategia de release del sistema.

**Process** (Procesos): Estructura el sistema en procesos e hilos (e.g., objetos activos que se ejecutan concurrentemente) y la interacción y mecanismos de sincronización entre los objetos activos (Diagramas de clases, Diagramas de secuencia, Diagramas de colaboración, Diagramas de actividades).

La vista de procesos también describe específicamente cosas como arrancar o detener el sistema. [12]

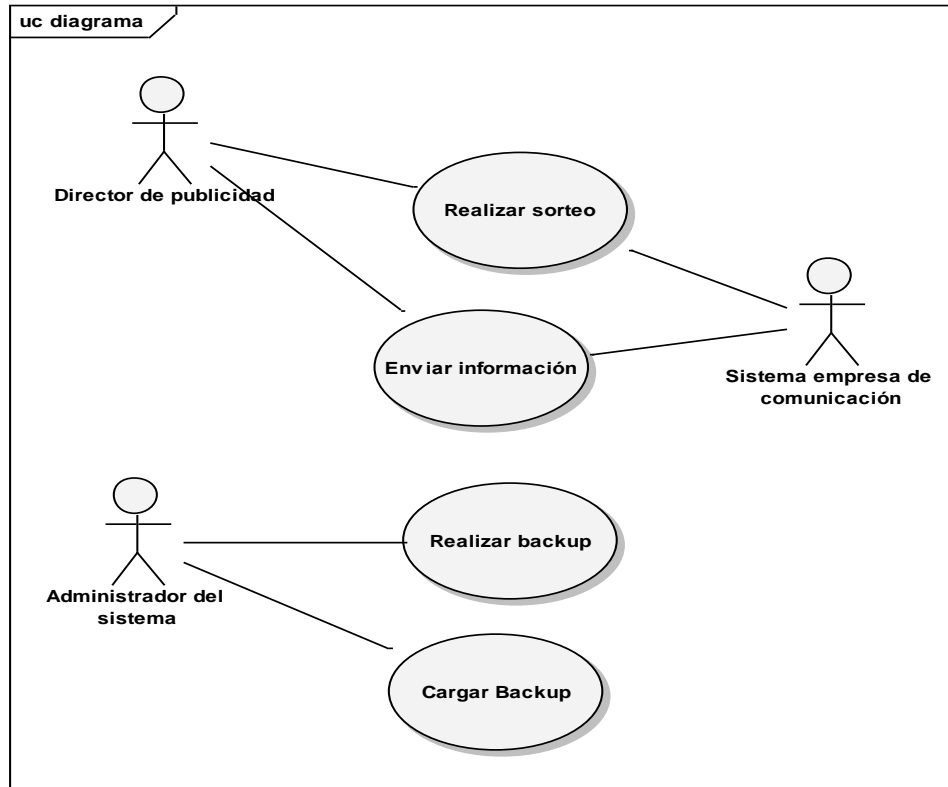
### **6.2.1. Vista de casos de uso**

Los casos del uso en este sistema se enumeran abajo. Los casos del uso en **negrilla** son significativos a la arquitectura. Una descripción de estos casos del uso se puede encontrar más adelante en esta sección.

- **Realizar sorteo**
- **Enviar información**
- **Realizar backup**
- **Cargar backup**
- Logueo de usuarios
- Actualizar información de cliente
- Enviar resultado encuesta
- Realizar encuesta
- Recibir comentario/queja
- Registrar cliente
- Análisis de preferencias

- Crear usuario
- Modificar usuario
- Agregar preferencia
- Modificar preferencia
- Ver comentarios
- Agregar premio
- Modificar premio
- Agregar mensaje de publicidad
- Modificar mensaje de publicidad
- Ver mensajes de publicidad

### 63 . Caso de uso, realización de sorteo.



#### 6.2.1.1 Realizar sorteo

##### Breve Descripción

Este caso de uso ocurre cuando el director de publicidad realiza un sorteo y se le avisa al cliente ganador por medio de mensajes a celular, e-mail, etc.

Actores	Objetivos
Director de	Realizar un sorteo y comunicar al cliente



publicidad	ganador.
Sistema empresa de comunicación	Comunicar al cliente ganador por medio de mensajes a celular, e-mail, etc.

### 6.2.1.2 Enviar información

#### Breve Descripción

Ocurre cuando se desea enviar publicidad a los clientes por medio de mensajes de texto, e-mail, etc.

<b>Actores</b>	<b>Objetivos</b>
Director de publicidad	Dar a conocer a los clientes publicidad o una promoción nueva.
Sistema empresa de comunicación	Hacer llegar la información al cliente

### 6.2.1.3 Realizar backup

#### Breve descripción

Este caso de uso ocurre cuando el administrador del sistema desea realizar un backup de la base de datos para prevenir pérdida de datos en una posible falla del manejador de base de datos.

<b>Actores</b>	<b>Objetivos</b>
Administrador del sistema	Poder tener un backup de la información importante.

### 6.2.1.4 Cargar backup

#### Breve descripción

Este caso de uso ocurre cuando el administrador del sistema ante un fallo del manejador de la base de datos carga los datos del último backup realizado.

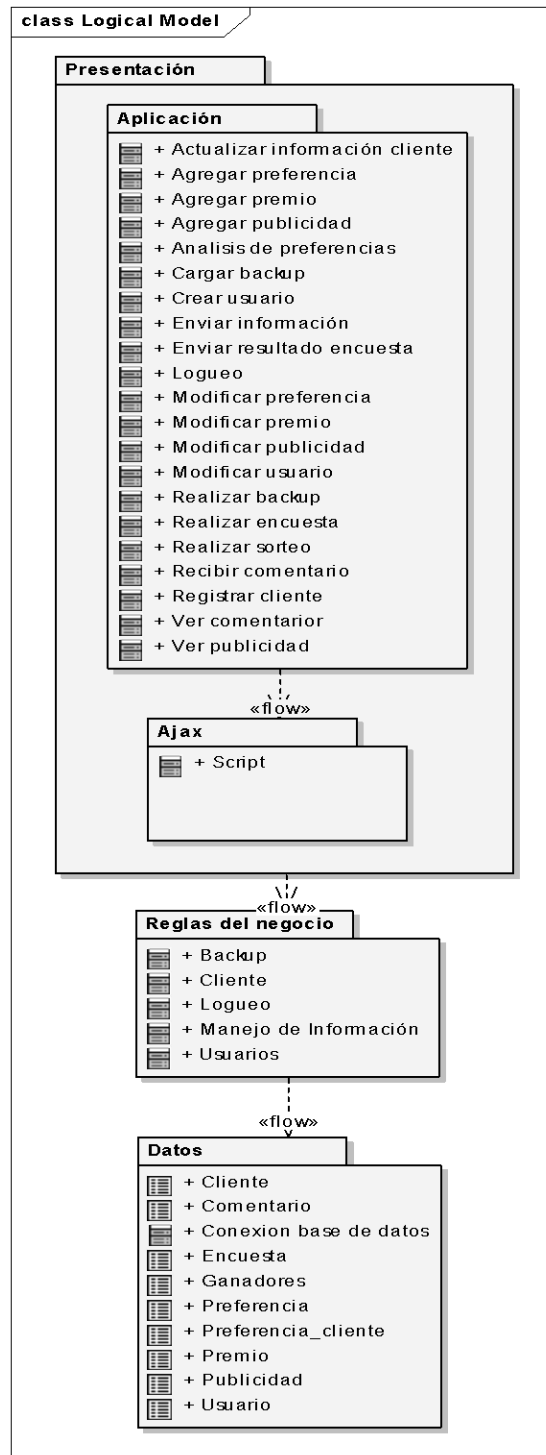
<b>Actores</b>	<b>Objetivos</b>
Administrador del sistema	Activar el sistema ante perdida de información de la base de datos.

## **6.2.2. Vista lógica**

### **6.2.2.1 Vista global**

El modelo se presenta en una arquitectura de 3 capas la primera capa esta subdividida en 2 capas internas una descripción se hallara mas adelante en esta sección, para lo cual se presenta un paquete por cada capa y las respectivas clases de cada paquete.

## 64 . Diseño de paquetes significativos de la arquitectura.



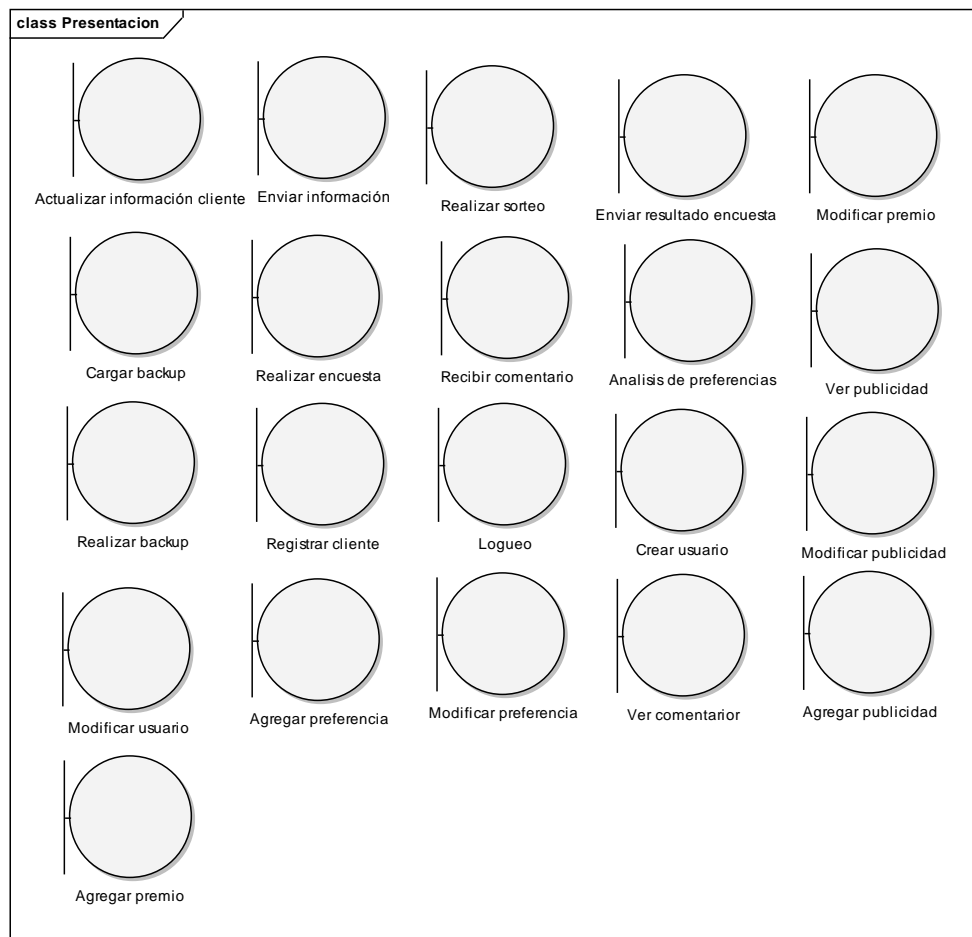
### 6.2.2.1.1 Capa de presentación

Contiene las clases expuestas al usuario final, es decir toda la interfaz.

#### 6.2.2.1.1.1 Subcapa aplicación

Contiene todas las clases que se le presentan al usuario final que en nuestro caso seria el código HTML que se le presentara al usuario en su navegador.

**65. Diagrama con las clases del paquete aplicacion.**



**Logueo:** Muestra un formulario para ingreso de usuario y password de los usuarios.

**Actualizar información cliente:** Ingreso, modificación y eliminación de preferencias.

**Cargar backup:** Muestra un formulario para el ingreso del archivo del cual se obtendrá el backup para su ingreso.

**Realizar encuesta:** Muestra un formulario con las preguntas que el cliente deberá contestar.

**Enviar información:** Muestra un formulario donde se deberá seleccionar que información se desea enviar a los clientes.

**Análisis de preferencias:** Muestra un formulario para el envío de el resultado al Director de atención al cliente junto con un comentario.

**Realizar sorteo:** Muestra un formulario donde se podrá ingresar información sobre el sorteo que se realizara.

**Enviar resultado encuesta:** Muestra un formulario donde se le enviara los resultados de la encuesta al Director de atención al cliente junto con un comentario.

**Recibir comentario:** Presenta un formulario para ingreso de comentarios/quejas.

**Registrar cliente:** Presenta un formulario para poder registrar los datos del usuario.

**Realizar backup:** Presenta un formulario para ingresar el nombre de archivo de salida del backup a realizar.

**Crear usuario:** Presenta un formulario para ingresar los datos del usuario.

**Modificar usuario:** Presenta los usuarios existentes y se debe seleccionar el deseado y modificar los datos que se quieran o eliminarlo.

**Agregar preferencia:** Presenta un formulario donde se puede ingresar la información de la preferencia.

**Modificar preferencia:** Presenta las preferencias existentes y se debe seleccionar la deseada y modificarla o eliminarla.

**Ver comentarios:** Muestra todos los comentarios enviados por los clientes que se encuentran almacenados.

**Agregar premio:** Presenta un formulario donde se ingresa la información del premio.

**Modificar premio:** Muestra todos los premios existentes y se debe seleccionar el que se desea modificar y modificarlo o eliminarlo.

**Agregar publicidad:** Presenta un formulario donde se puede ingresar el mensaje que se desea guardar.

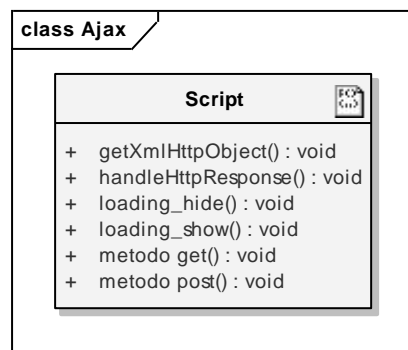
**Modificar publicidad:** Muestra todos los mensajes de publicidad existente y luego de seleccionar el deseado se pueden modificar los datos o eliminarlo.

**Ver publicidad:** Muestra todos los mensajes de publicidad que se encuentran almacenados.

### 6.2.2.1.1.2 Subcapa ajax

Contiene la clase con que contiene código script el cual será ejecutado en el lado del cliente para poder permitir la comunicación entre el navegador y el servidor sin la necesidad de cargar una nueva pagina solo actualizando la página actual.

### 66 . Diagrama con las clases del paquete ajax.

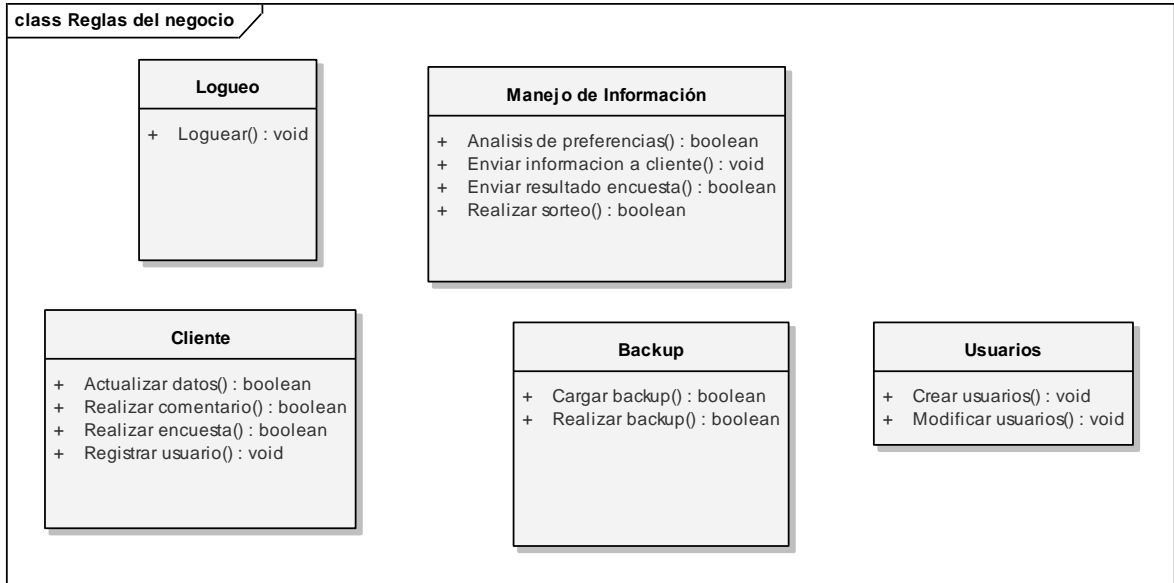


### 6.2.2.1.2 Reglas del negocio

Contiene las clases que definen la interoperabilidad entre la capa de presentación y la capa de datos, implementa las reglas a definir del negocio.



## 67 .Diagrama con las clases del paquete reglas del negocio.



**Cliente:** Lleva el control del registro, actualización de datos, presentación de encuesta y envió de comentarios de los clientes del sistema.

**Logueo:** Lleva el control del logueo de los usuarios del sistema para luego presentarle sus opciones correspondientes.

**Backup:** Clase utilizada para llevar el control del manejo de los backups en la base de datos.

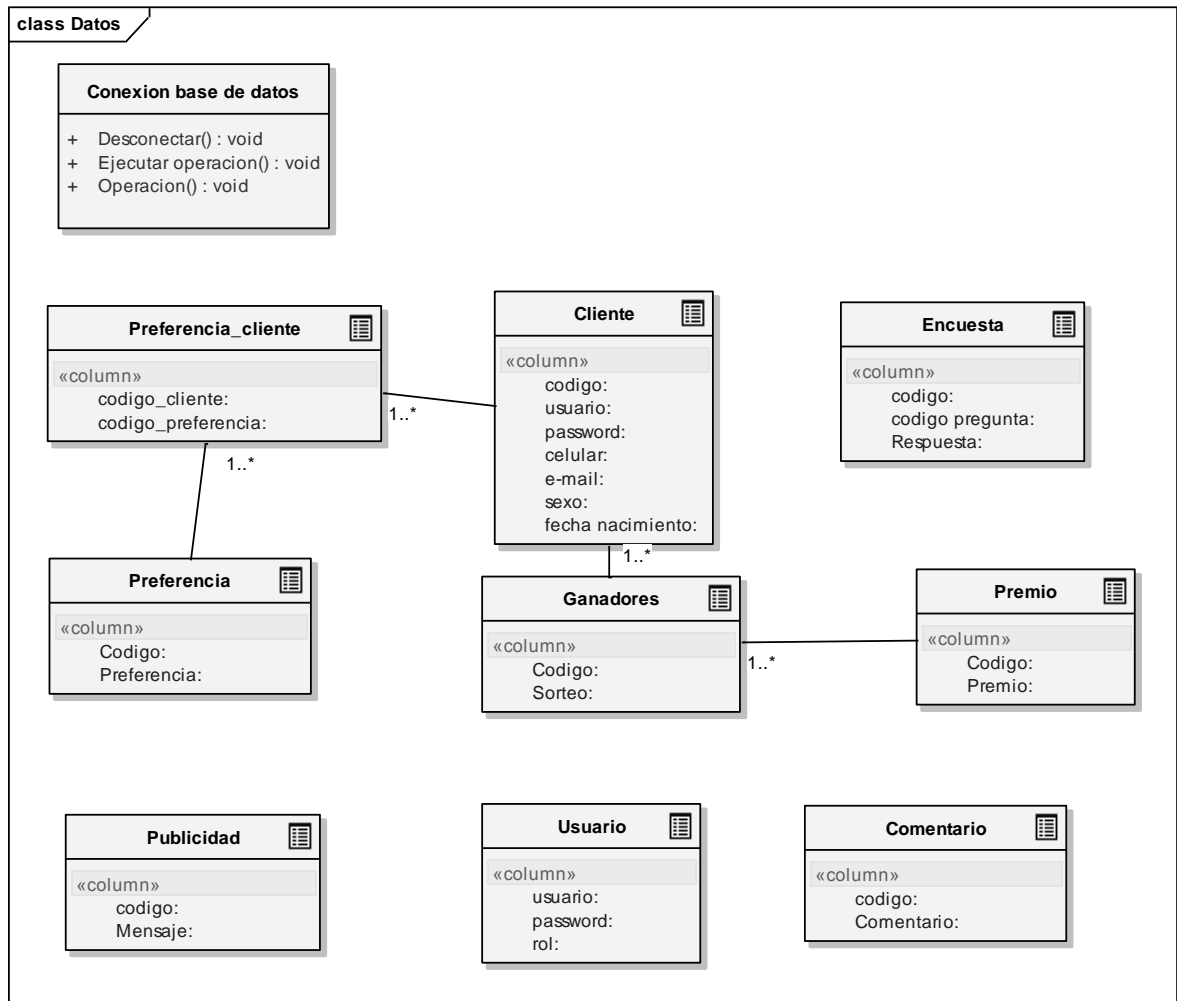
**Manejo de información:** Realiza los cálculos y procesos estadísticos para el análisis de la información y para el envió de la información a las personas que se requiera.

**Usuarios:** Permite la creación y modificación de usuarios del sistema.

### 6.2.2.1.3 Capa de datos

Contiene las clases que se encargan de la comunicación con la base de datos.

68 . Diagrama con las clases del paquete datos.



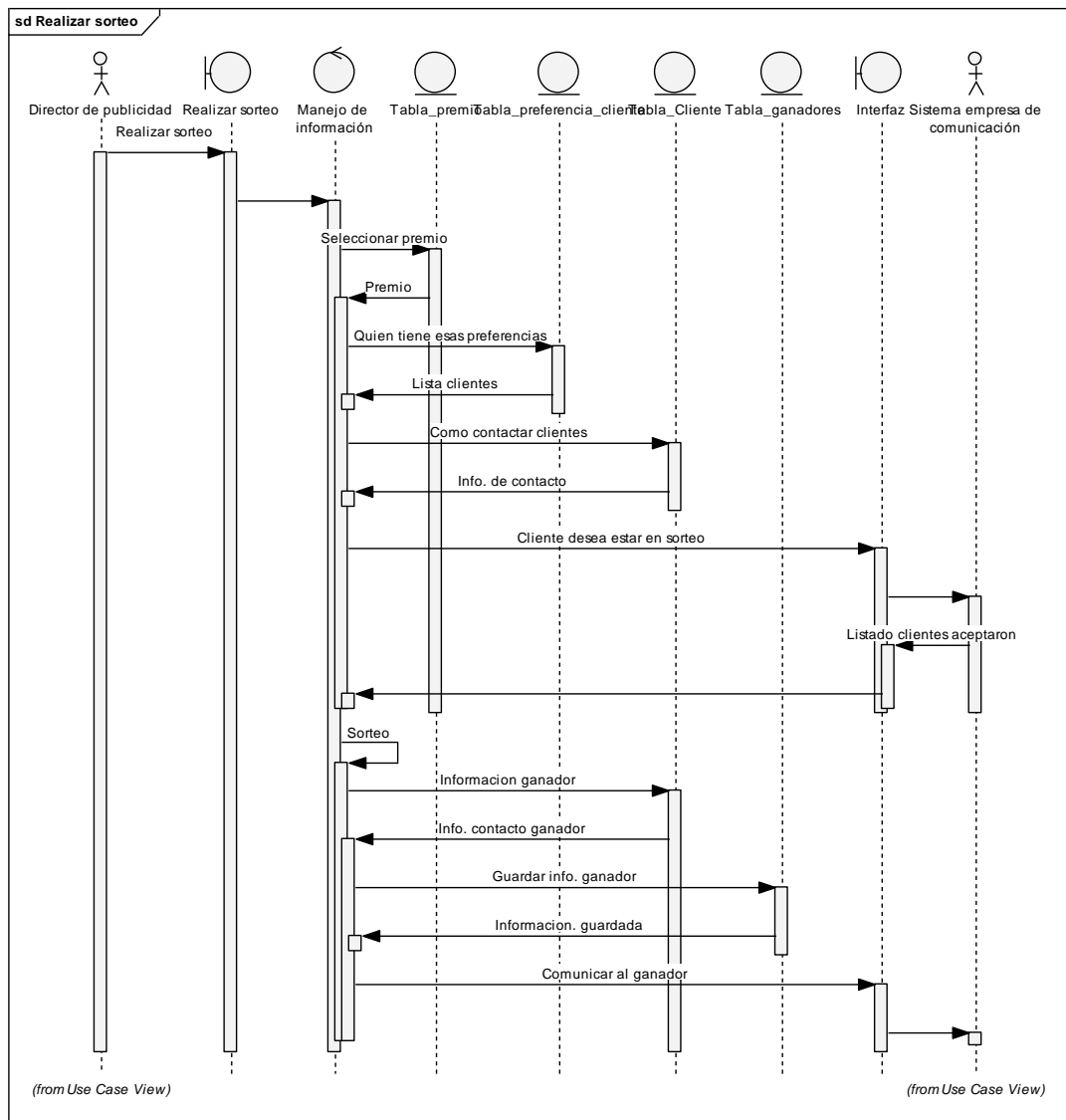
La clase conexión con la base de datos es la que funciona como interfaz entre las tablas de la base de datos y la capa de reglas del negocio, por lo tanto las otras son tablas que contienen los atributos que se almacenan en la base de datos.

## 6.2.2.2 Realización de casos de uso

Estos casos de uso ocurren en el sistema en condiciones normales.

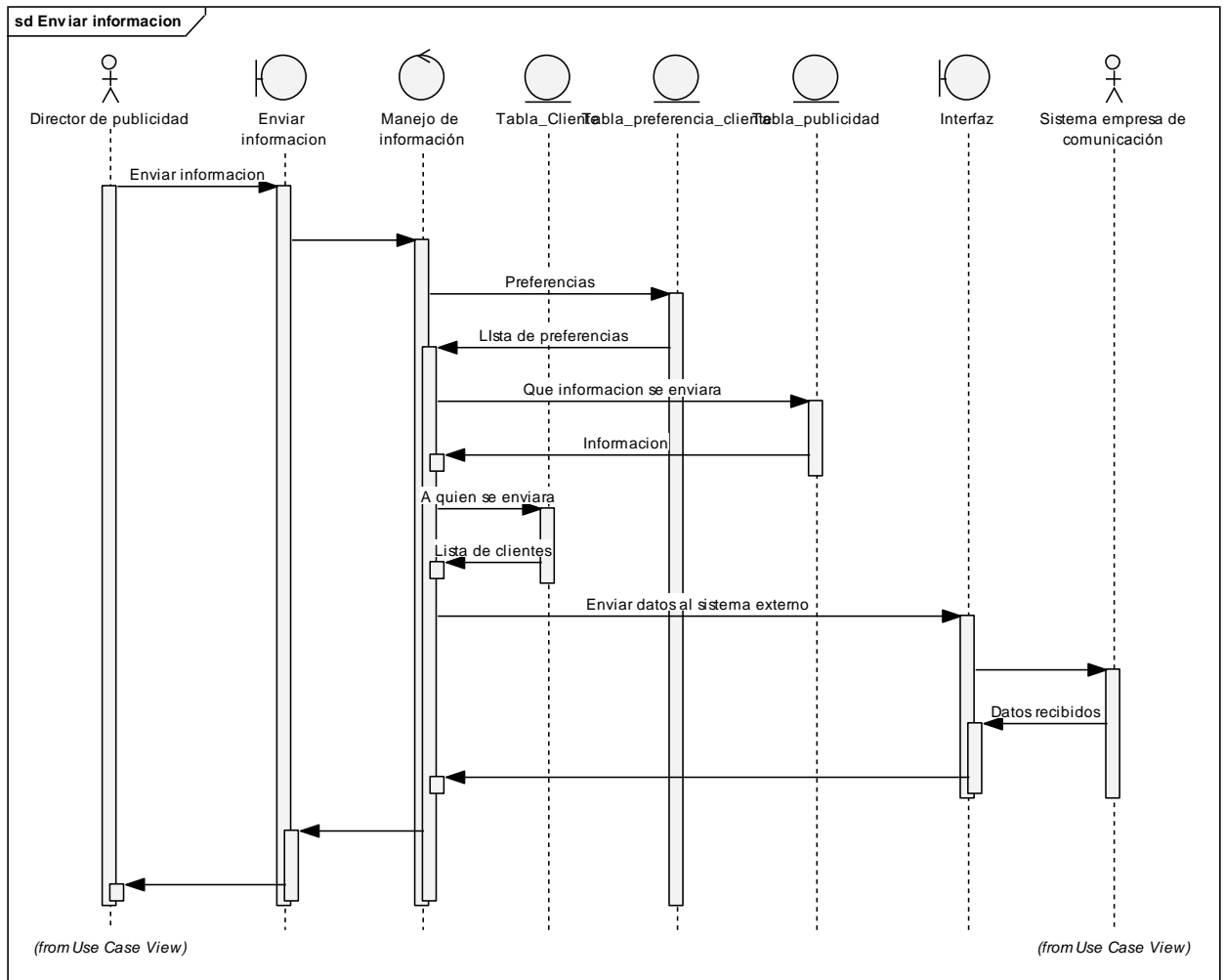
### 6.2.2.2.1 Realizar sorteo

#### 69 . Diagrama de realización de casos de uso, realizar sorteo.



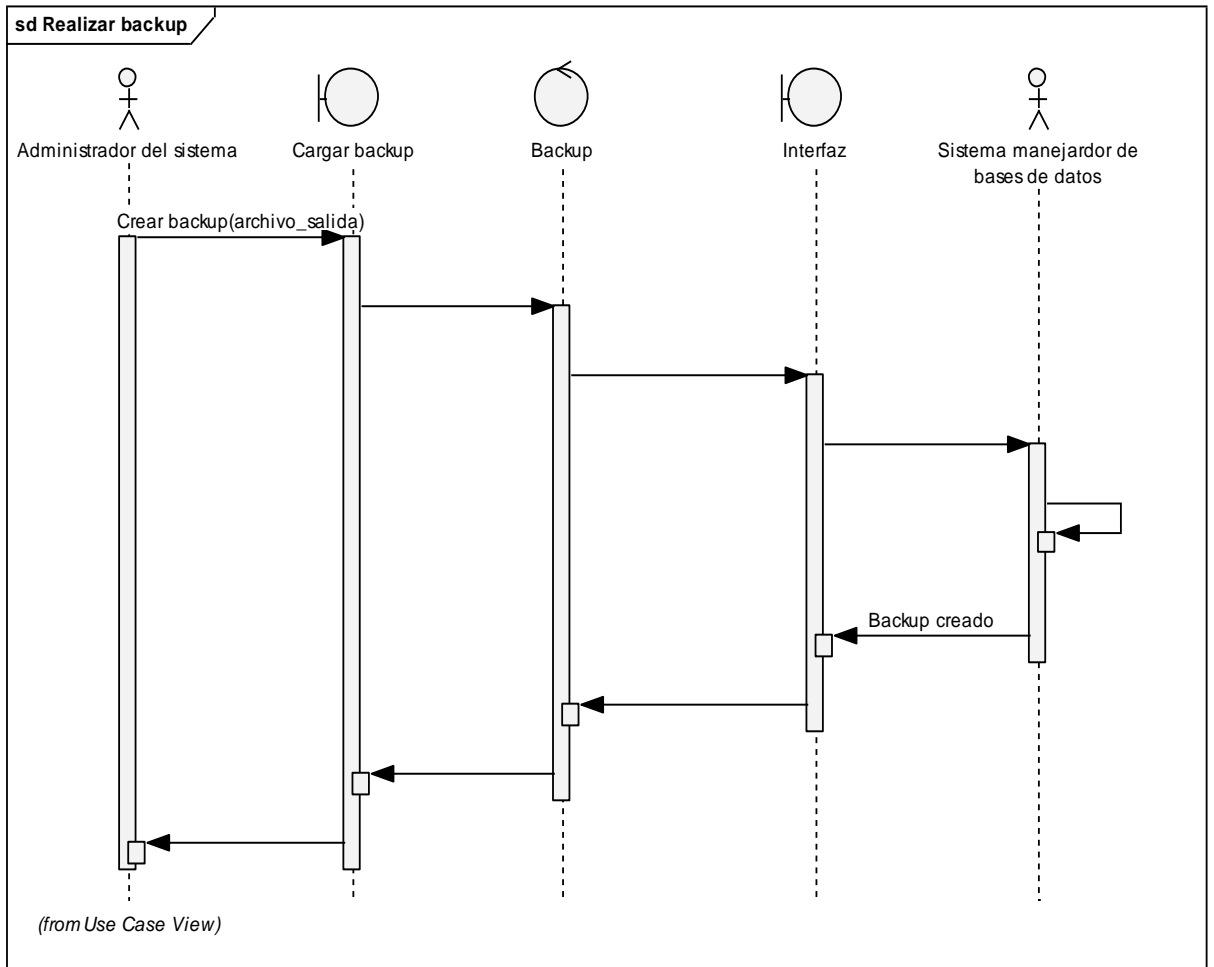
## 6.2.2.2 Enviar información

### 70 . Diagrama realización de casos de uso, enviar información.



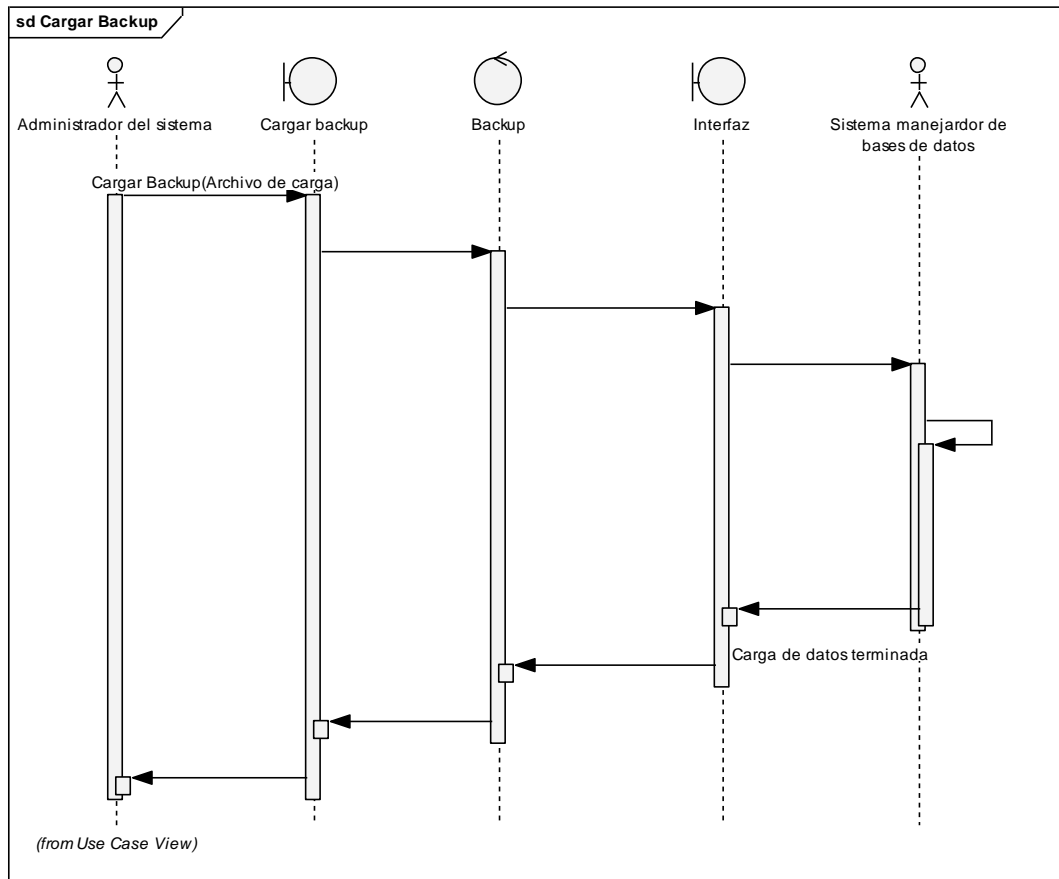
### 6.2.2.2.3 Realizar backup

71 . Diagrama realización de casos de uso, enviar backup.



#### 6.2.2.2.4 Cargar backup

### 72 . Diagrama realización de casos de uso, cargar backup.

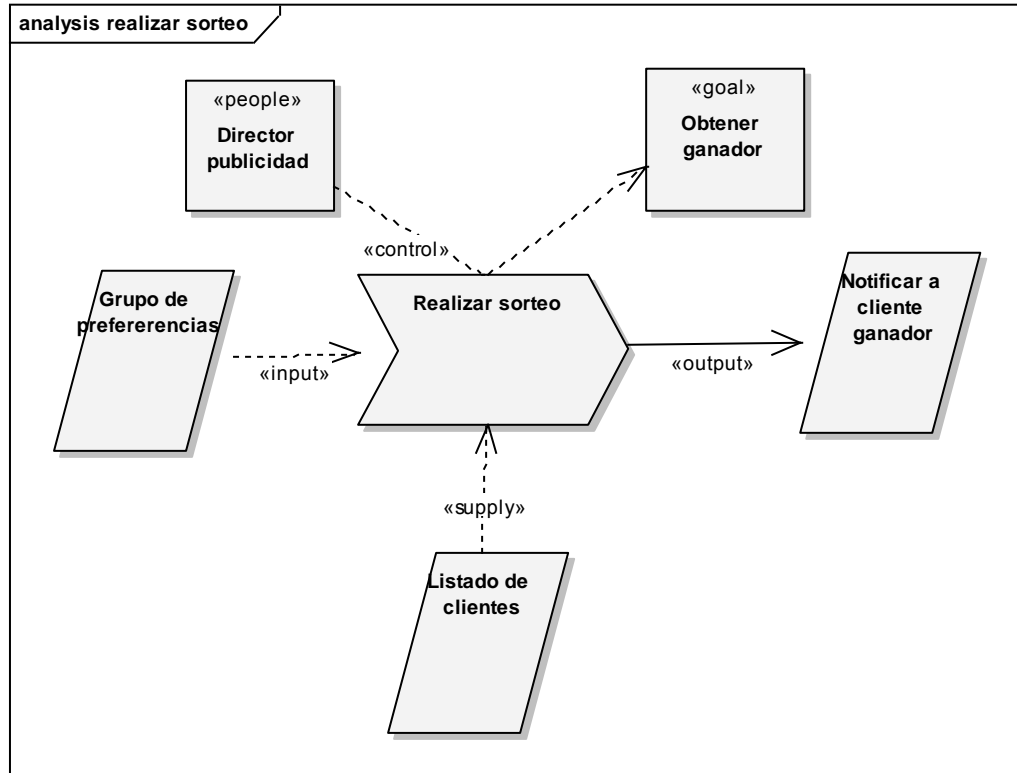


#### 6.2.3. Vista de procesos

##### 6.2.3.1 Realizar sorteo

Realizar sorteo es el proceso por el cual el director de publicidad realiza un sorteo para un grupo de clientes con preferencias en específico y al obtener al ganador se le comunica que ah ganado por medio de una empresa de comunicación.

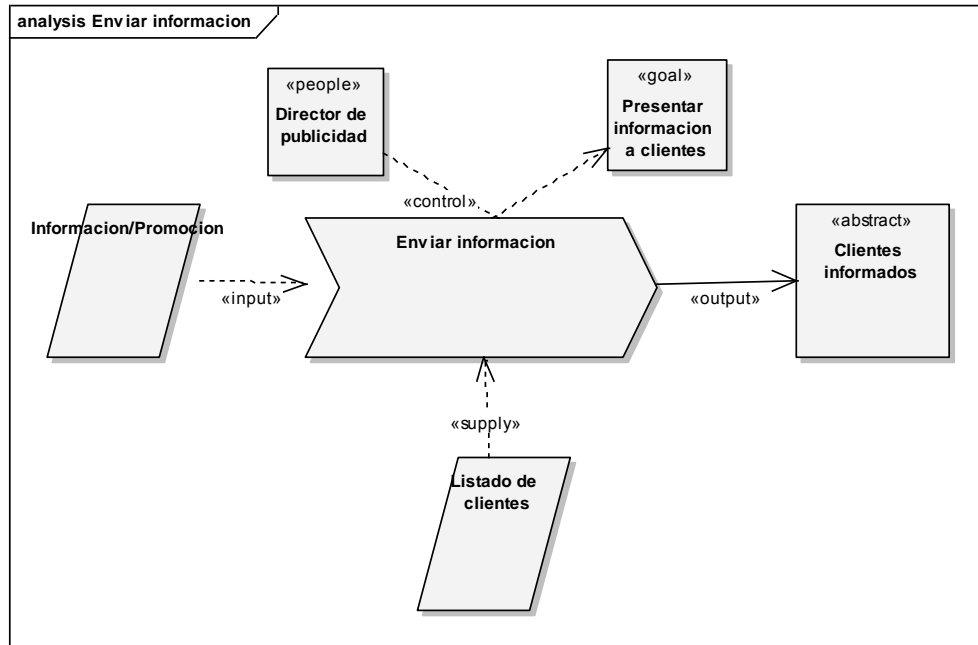
### 73 . Diagrama de procesos, realizar sorteo.



#### 6.2.3.2 Enviar información

Enviar información es el proceso por el cual el director de publicidad desea enviar información, promociones, etc. A los clientes para que ellos estén informados.

## 74 . Diagrama de procesos, enviar información.

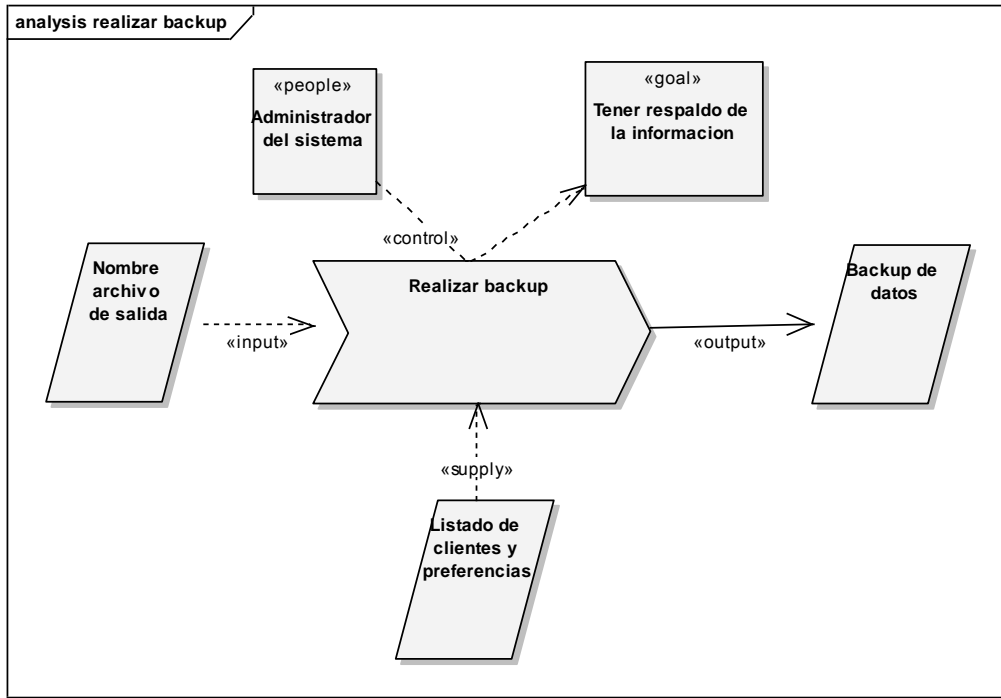


### 6.2.3.3 Realizar backup

Cargar backup es el proceso por el cual el administrador del sistema desea tener una copia de respaldo de la información contenida en la capa de datos por un posible fallo de esta.



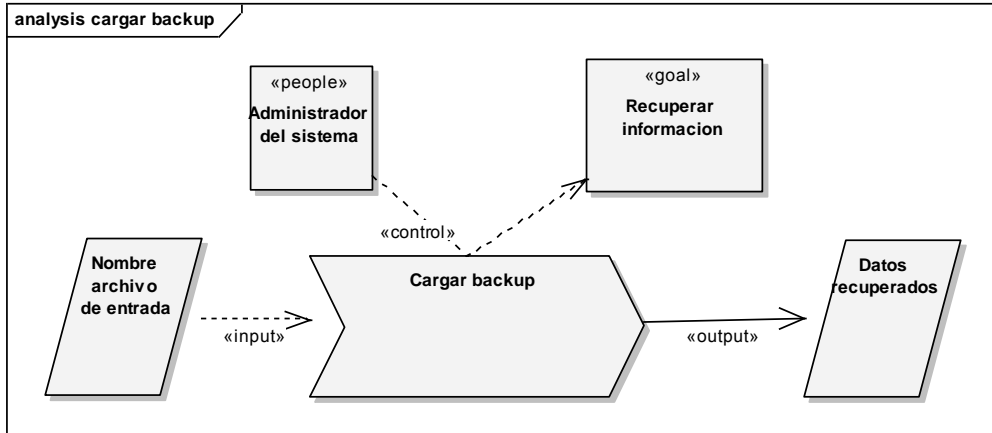
## 75 . Diagrama de procesos, realizar backup.



### 6.2.3.4 Cargar backup

Cargar backup es el proceso por el cual el administrador del sistema desea recuperar información cuando se ha perdido esta por un fallo en la capa de datos.

## 76 . Diagrama de procesos, cargar backup.

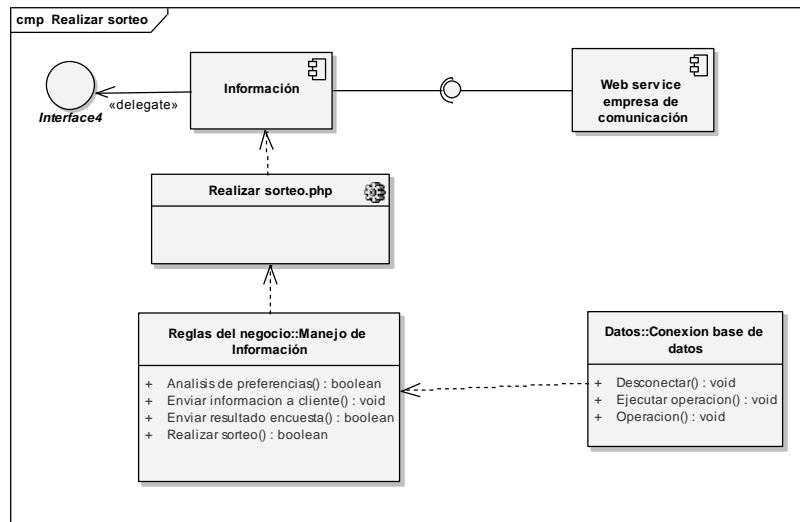


### 6.2.4. Vista de componentes

Es la manera en que están organizadas las clases para poder realizar las operaciones.

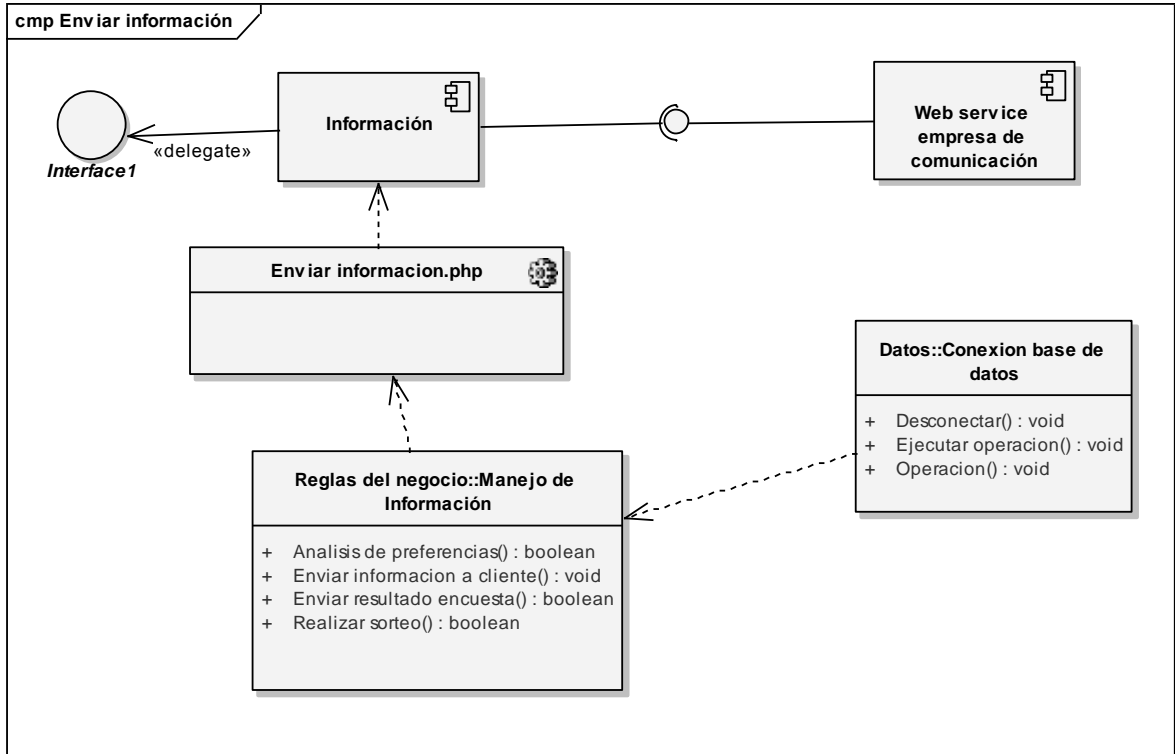
#### 6.2.4.1 Realizar sorteo

## 77 . Diagrama de comonentnes, realizar sorteo.



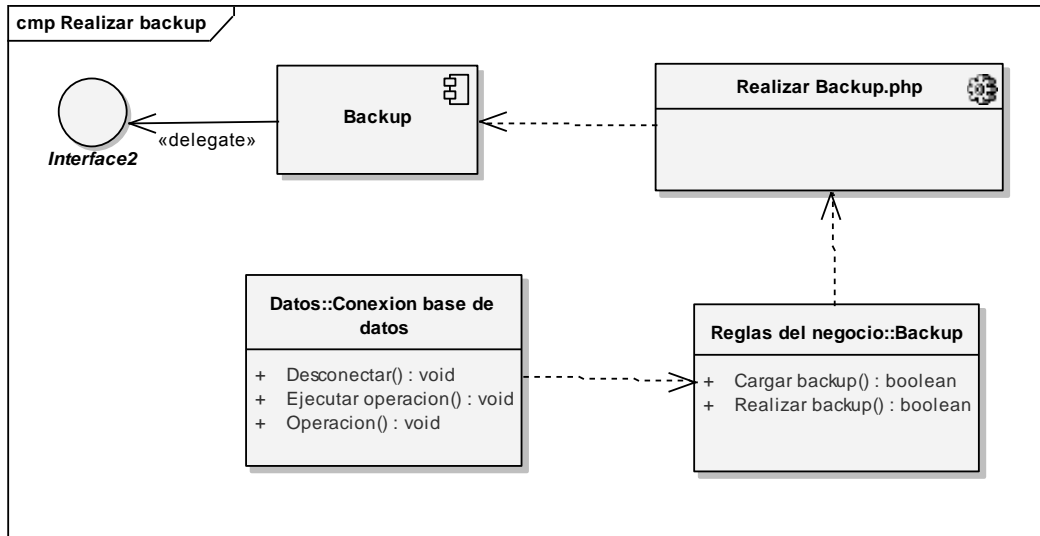
### 6.2.4.2 Enviar información

78 . Diagrama de componentes, enviar información.



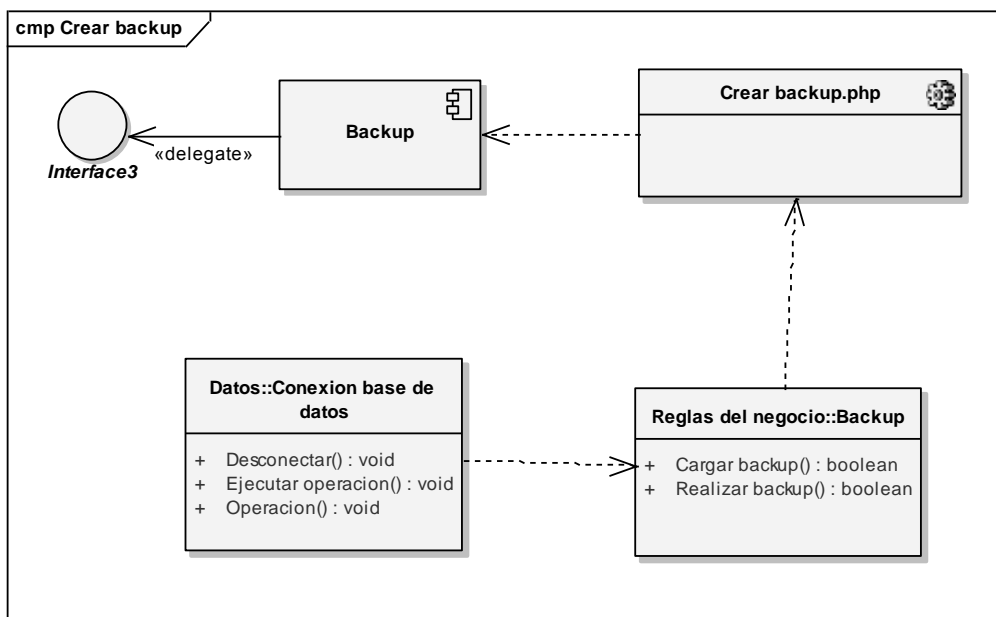
### 6.2.4.3 Realizar backup

79 . Diagrama de componentes, realizar backup.



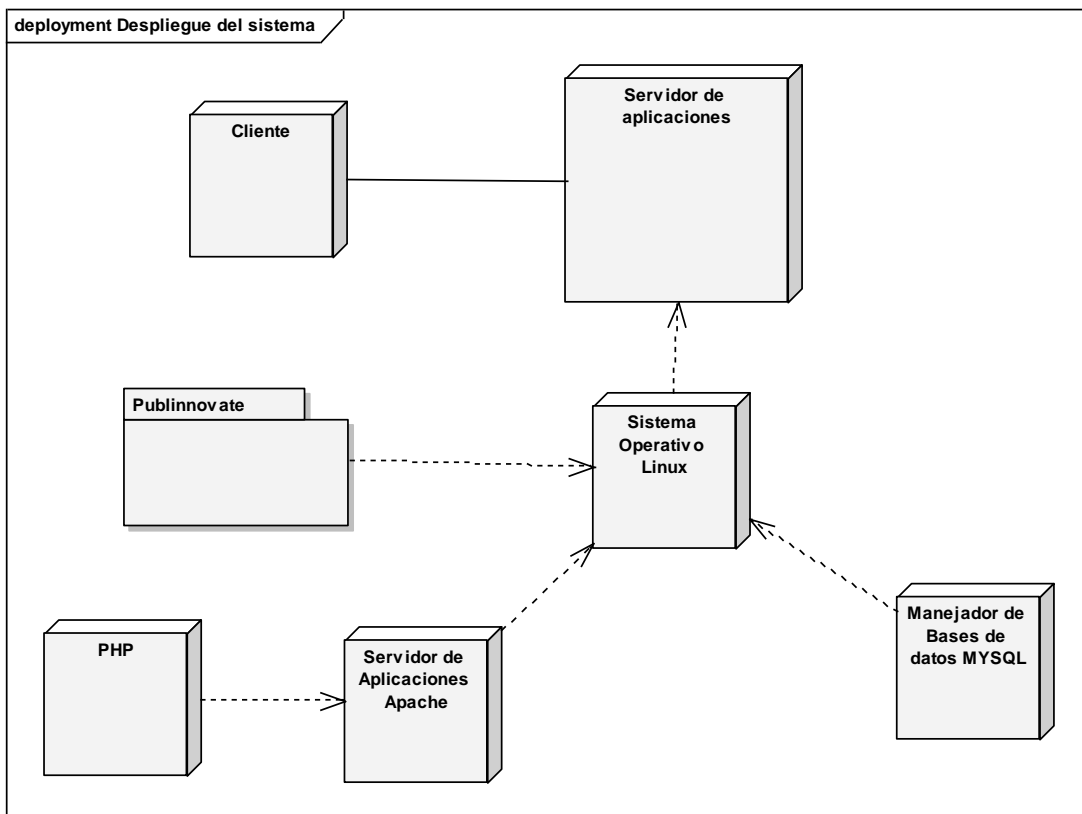
### 6.2.4.4 Cargar backup

80 . Diagrama de componentes, cargar backup.



## 6.2.5. Vista de despliegue

### 81 . Diagrama de despliegue sistema.



### 6.2.5.1 Servidor de aplicaciones

Este servidor contiene la aplicación Publinnovate, el servidor Apache, y los componentes de PHP necesarios para la ejecución de las páginas dinámicas PHP.

#### 6.2.5.1.1 Publinnovate

Es el conjunto de clases de la aplicación del manejo de la publicidad.

#### **6.2.5.1.2          Manejador de base de datos MySql**

Es el manejador de base de datos donde se encuentra almacenada la información en forma de tablas las cuales están definidas anteriormente en este documento.

#### **6.2.5.2          Cliente**

Es cualquier PC que sirve para acceder a las páginas del sistema para realizar cambio de preferencias, sorteos, encuestas, etc. a través de un navegador web.

## CONCLUSIONES

1. Al utilizar un lenguaje de modelado como lo es UML se pueden crear documentos que puedan ser comprendidos por personas que tengan los conceptos básicos de diagramas UML y así poder compartir el conocimiento con muchas personas usando estos estándares.
2. La representación grafica en las documentación para el desarrollo de sistemas es de gran ayuda porque podemos llegar a conocer cuáles son las ideas presentadas más rápidamente que de una manera descriptiva.





## RECOMENDACIONES

1. Para poder compartir el conocimiento mediante documentación es necesario la utilización de estándares establecidos para que más personas puedan llegar a comprenderlo.
2. Llevar una documentación adecuada sobre el proceso de desarrollo de sistemas es muy importante para poder tener un historial sobre el trabajo y los cambios realizados.
3. La utilización de diagramas en la fase de documentación es útil para poder presentar las ideas de una manera grafica para comprender más rápidamente la idea presentada.



## REFERENCIAS BIBLIOGRÁFICAS

1. Smith, Eduardo. **Crece uso de computadora en Pymes**, Artículo Electrónico Prensa Libre, 14 de Junio 2007.
2. Monge González, Ricardo, Alfaro Azofeifa, Cyndi y Alfaro Chamberlain, José. **Tics en las Pymes de Centro América**, Primera edición copublicada por: Editorial Tecnológica de Costa Rica y el Centro Internacional de Investigaciones para el Desarrollo, 2005, 251 pp.
3. Martínez Villavicencio, José. **Factores Críticos para el Desempeño Exportador de las Pymes**.
4. **Evolución del Marketing**, <http://www.wikipedia.com>, visitado el 20 de noviembre de 2006.
5. **Comunicaciones de Marketing**, <http://www.wikipedia.com>, visitado el 20 de noviembre de 2006.
6. **Marketing Directo**, <http://www.wikipedia.com>, visitado el 20 de noviembre de 2006.
7. **Marketing electrónico**, <http://www.wikipedia.com>, visitado el 20 de noviembre de 2006.
8. **Marketing en Internet**, <http://www.wikipedia.com>, visitado el 20 de noviembre de 2006.
9. **Relaciones publicas**, <http://www.wikipedia.com>, visitado el 20 de noviembre de 2006.
10. A. Senn, James. **Análisis y Diseño de Sistemas de Información**, McGraw – Hill Interamericana 1992, Georgia State University, 942 pp.

11. **MÉTRICA. Versión 3**, Sección Técnicas Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información, Ministerio de Administraciones Públicas España.
12. Philippe Kruchten , **Planos Arquitectónicos: El Modelo de “4+1” Vistas de la Arquitectura del Software**, Artículo publicado Noviembre 1995

## BIBLIOGRAFÍA

1. Cabanes, Nacho. **Diccionario básico de informática**. Versión 0.40. Disponible: <http://usuarios.lycos.es/Resve/diccioninform.htm>, visitado el 15 de julio 2007.
2. Gracia, Joaquin. **Diagramas UML**. 7 de mayo 2005. Disponible: <http://www.ingenierosoftware.com/analisisydiseno/uml.php>, visitado el 15 de julio 2007.
3. **Preguntas Frecuentes, Conceptos**. masadelante.com. Disponible: <http://www.masadelante.com>, visitado el 20 de agosto 2008.
4. OCED, Reviewing the ICT Sector Definition: Issues for Discussion. Working Party on Indicators for the Information Society. Stockholm: OCED, 2002.
5. Pymeonline.net. Disponible: <http://www.pymeonline.net>, visitado el 15 de julio 2007.