

Universidad de las Ciencias Informáticas  
Facultad 10



**Interfaz Web para el sistema de análisis Airesweb**

Trabajo de Diploma para optar por el Título de Ingeniería en Ciencias Informáticas

**Autores:** Adrian Gallardo Palay  
Oscar Andrés Casas Machado

**Tutor:** Alain Guerrero Enamorado

**Ciudad de la Habana, 7 de junio de 2009**

## **Declaración de autoría**

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

---

Adrian Gallardo Palay

Autor

---

Oscar Andrés Casas Machado

Autor

---

Alain Guerrero Enamorado

Tutor

## **Agradecimientos**

Agradecemos a todas aquellas personas que de una forma u otra contribuyeron al desarrollo de este trabajo, en especial:

A nuestro tutor Alain por dedicarnos parte de su escaso tiempo y por sus valiosos consejos.

A nuestros compañeros de trabajo del proyecto Aires, sin los que el desarrollo de este trabajo hubiera sido imposible, Carlos, León, Yordan, Jorgito, Adalberto, Denis y Dunia, gracias por su contribución.

A nuestros compañeros, Dovie, Kiuver, Jose y Luis, gracias por su acertada guía.

A nuestros compañeros del 10502 por los años compartidos.

A Dayani y Katia, por el tiempo dedicado.

A Yilian y Yari por sus horas gastadas en tablas y ortografía.

A nuestra Madre Merci por el apoyo profesional y por impulsarnos a trabajar e investigar.

A mi compañero de tesis.

A nuestros profesores de toda la vida, a esos que quedan olvidados en el tiempo y a los que no.

A la Revolución Cubana por ayudarnos a pensar como pensamos.

A la UCI, ese maravilloso lugar donde todo es posible e imposible.

A nuestra Facultad X, a sus profes.

## **Dedicatoria**

Dedico esta tesis a la persona que es mi guía, que me educó durante toda mi vida, abuela, para tí son mis logros.

A mi hermanito Adniel, por haber aprendido a caminar solo por la vida, por su humildad y sencillez.

A mi mamá María Elena y mi papá Faures, por el apoyo incondicional.

A mis otras madres, Mercedes y Taidé, gracias por existir.

A mi tío Pocho y mi tío Maleta, por mostrarme el camino y por estar ahí para todo.

A mis hermanos Carlitos, Ale, Coty y Marcel, en las buenas y malas seguiremos juntos.

***Adrián Gallardo Palay.***

Este trabajo cierra una etapa de más de 20 años, son muchas las personas que pusieron su granito de arena para tratar de convertirme en un hombre de bien, a ellas mis pensamientos.

A mis padres por la herencia de amor dejada en mí, por ser los adoquines del camino, esas personas que en la mayoría de sus sueños estoy.

A mi hermana por tantos combates celebrados sin contar una victoria en mi palmarés.

A mis abuela Lury y Delia por su infinita ternura.

A mis abuelos Andrés y Oscar por sus nombres, sus peloteros y sus herramientas.

A mis primos Laura, Andresito y Cristian, los nuevos.

A mis otras madres Marlene y Adita, por enseñarme a caminar.

A mis hermanos de toda la vida Adrián, Carlos, Alejandro, Marcel y David, la distancia no influirá.

***Oscar Andrés Casas Machado.***

*“Hoy en día la mayoría del software existe no para resolver un problema, sino para actuar de interfaz con otro software”.*

**I. O. Angell**

## Resumen

La toma de decisiones más eficiente, mejorar la estrategia de posicionamiento en Internet , optimizar el diseño de las páginas web, permitir la evaluación del entorno del sitio, hacer un adecuado seguimiento de los clientes, averiguar la procedencia y destino de los visitantes, aumentar la fidelidad de los clientes, entre otras utilidades, son aspectos que el análisis de registros de servidores web brinda. Analizador Inteligente de Registros de Servidores Web (Airesweb) es un software de este tipo, desarrollado en la Universidad de las Ciencias Informáticas (UCI), diseñado para instalarse en uno o varios servidores y cuenta con una base de datos donde se almacena la información generada por el análisis de los archivos de registros. Para mostrar y gestionar la información almacenada, este trabajo tiene como propósito el desarrollo de una Interfaz Web (IW) para el sistema de análisis Airesweb, aprovechando el resultado de fusionar las ventajas que brindan las Interfaces Gráficas de Usuario (GUIs) con el uso de la web como medio de aplicación. Para lograr dicho objetivo se caracterizaron las aplicaciones web y se analizó su historia, se realizó un estudio de las Interfaces Web de los principales softwares de análisis de registro de servidores, se especificaron las pautas de diseño gráfico para la IW, se utilizó el framework Symfony para la elaboración de la aplicación y se eligió RUP para guiar a través de sus fases y flujos el proceso de desarrollo. Los artefactos generados junto a las facilidades que brindó el framework escogido, hicieron posible obtener una aplicación web funcional y eficiente, que permite gestionar el contenido del análisis de la información procesada.

**Palabras claves:** aplicación web, interfaz web, análisis, analítica web, estadísticas, archivo de registro de servidores web, Symfony, Airesweb, Internet.

# Índice general

---

<b>Introducción</b>	<b>1</b>
<b>1. Fundamento teórico</b>	<b>7</b>
1.1. Trabajos similares . . . . .	7
1.1.1. Ámbito internacional . . . . .	7
1.1.2. Ámbito nacional . . . . .	10
1.2. Interfaces Gráficas de Usuario . . . . .	11
1.3. Aplicaciones web . . . . .	11
1.3.1. Concepto de aplicación web . . . . .	12
1.3.2. Evolución de las aplicaciones web . . . . .	13
1.3.3. Generaciones . . . . .	14
1.3.4. Características . . . . .	19
1.3.5. Categorías o tipos . . . . .	20
1.3.6. Arquitectura . . . . .	21
1.4. Tecnologías a utilizar . . . . .	23
1.4.1. Lenguaje de programación . . . . .	23
1.4.2. Lenguaje de modelado . . . . .	26
1.4.3. Framework . . . . .	26
1.4.4. Herramientas . . . . .	29
1.5. Metodología de desarrollo . . . . .	31

1.5.1. RUP . . . . .	31
1.6. Conclusiones . . . . .	33
<b>2. Características del sistema</b>	<b>34</b>
2.1. Descripción del problema . . . . .	34
2.2. Modelo de Dominio . . . . .	37
2.3. Solución propuesta . . . . .	38
2.4. Requisitos funcionales . . . . .	39
2.5. Requisitos no funcionales . . . . .	40
2.5.1. Usabilidad . . . . .	40
2.5.2. Precisión . . . . .	41
2.5.3. Rapidez . . . . .	41
2.5.4. Flexibilidad . . . . .	41
2.5.5. Seguridad . . . . .	41
2.5.6. Software . . . . .	41
2.5.7. Ayuda y documentación . . . . .	42
2.6. Definición de los casos de uso . . . . .	42
2.6.1. Definición de los actores . . . . .	42
2.6.2. Listado de casos de uso . . . . .	42
2.6.3. Diagramas de casos de uso . . . . .	47
2.6.4. Casos de uso expandidos . . . . .	47
2.7. Conclusiones . . . . .	65
<b>3. Análisis y diseño del sistema</b>	<b>66</b>
3.1. Análisis . . . . .	66
3.1.1. Modelo de Clases del Análisis . . . . .	66
3.2. Diseño . . . . .	69
3.2.1. Symfony . . . . .	69
3.2.2. Patrones de diseño usados . . . . .	72

3.2.3. Diagramas de Clases del Diseño . . . . .	75
3.2.4. Diagramas de Interacción . . . . .	80
3.2.5. Descripción de las clases . . . . .	85
3.2.6. Diseño de la Base de Datos . . . . .	85
3.2.7. Interfaz . . . . .	85
3.2.8. Concepción de la ayuda . . . . .	85
3.3. Conclusiones . . . . .	85
<b>4. Implementación y prueba</b>	<b>87</b>
4.1. Implementación . . . . .	87
4.1.1. Diagrama de Despliegue . . . . .	87
4.1.2. Diagramas de Componentes . . . . .	88
4.2. Modelo de prueba . . . . .	90
4.2.1. Pruebas de caja negra . . . . .	91
4.3. Conclusiones . . . . .	92
<b>Conclusiones</b>	<b>93</b>
<b>Recomendaciones</b>	<b>94</b>
<b>Bibliografía</b>	<b>98</b>
<b>Referencias bibliográficas</b>	<b>102</b>
<b>A. Figuras relacionadas</b>	<b>103</b>
<b>B. Diagramas de clases del Análisis</b>	<b>107</b>
<b>C. Casos de Uso expandidos</b>	<b>109</b>
<b>D. Diagramas de clases del Diseño</b>	<b>123</b>
<b>E. Diagramas de Interacción</b>	<b>137</b>

<b>F. Diagramas de Componentes</b>	<b>158</b>
<b>Glosario de términos</b>	<b>162</b>

# Índice de figuras

---

1.1. Componentes genéricos de una aplicación web. . . . .	23
2.1. Diagrama Entidad-Relación de la BD. . . . .	36
2.2. Modelo de Dominio del problema. . . . .	38
2.3. Diagrama de Casos de Uso. . . . .	47
3.1. DCA Iniciar sesión . . . . .	66
3.2. DCA Listar sitios monitorizados . . . . .	67
3.3. DCA Seleccionar fecha a monitorizar . . . . .	67
3.4. DCA Ver reportes . . . . .	68
3.5. Esquema del MVC . . . . .	70
3.6. Flujo de trabajo del MVC en Symfony . . . . .	71
3.7. Organización de la aplicación . . . . .	72
3.8. DCD Iniciar sesión . . . . .	75
3.9. DCD Listar sitios monitorizados . . . . .	76
3.10.DCD Seleccionar sitio a monitorizar . . . . .	77
3.11.DCD Seleccionar fecha a monitorizar . . . . .	77
3.12.DCD Mostrar reportes sección mostrar visitantes únicos . . . . .	78
3.13.DCD Mostrar reportes sección mostrar visitantes por países . . . . .	79
3.14.DCD Mostrar reportes sección mostrar visitantes por mes . . . . .	79
3.15.Iniciar sesión . . . . .	80

3.16. Listar sitios monitorizados . . . . .	81
3.17. Seleccionar fecha a monitorizar . . . . .	82
3.18. Mostrar visitantes únicos . . . . .	83
3.19. Mostrar visitantes por países . . . . .	84
3.20. Mostrar visitantes por mes . . . . .	84
4.1. Diagrama de Despliegue de la aplicación. . . . .	88
4.2. Diagrama de componentes general. . . . .	89
4.3. Diagrama de componentes de Symfony. . . . .	90
A.1. Interfaz Web de Awstats. . . . .	103
A.2. Interfaz Web de Sawmill. . . . .	104
A.3. Interfaz Web de WebLog Expert. . . . .	105
A.4. Interfaz Web de Webtrends. . . . .	106
B.1. DCA Administrador general Gestionar usuario y Cambiar contraseña del usuario . . . . .	107
B.2. DCA Editar información personal y Cambiar contraseña personal . . . . .	108
B.3. DCA Gestionar usuario y Cambiar contraseña del usuario . . . . .	108
D.1. DCD Cambiar Contraseña . . . . .	123
D.2. DCD Editar Información Personal . . . . .	124
D.3. DCD Finalizar Sesión . . . . .	124
D.4. DCD Gestionar Administrador Escenario Adicionar Administrador . . . . .	125
D.5. DCD Gestionar Administrador Escenario Cambiar Contraseña del Administrador . . . . .	125
D.6. DCD Gestionar Administrador Escenario Editar Información Personal de Administrador . . . . .	126
D.7. DCD Gestionar Administrador Escenario Eliminar Administrador . . . . .	126
D.8. DCD Gestionar Usiario Escenario Adicionar Usuario . . . . .	127
D.9. DCD Gestionar Usuario Escenario Editar Información Personal de Usuario . . . . .	127
D.10.DCD Gestionar Usuario Escenario Eliminar Usuario . . . . .	128
D.11.DCD Gestionar Usuario Escenario Cambiar Contraseña de Usuario . . . . .	128

D.12.DCD Mostrar Reportes Sección Mostrar Visitantes por horas . . . . .	129
D.13.DCD Mostrar Reportes Sección Mostrar Visitantes por días . . . . .	130
D.14.DCD Mostrar Reportes Sección Mostrar robots . . . . .	130
D.15.DCD Mostrar Reportes Sección Mostrar Referencias . . . . .	131
D.16.DCD Mostrar Reportes Sección Mostrar Principales navegadores y sistemas operativos utilizados . . . . .	132
D.17.DCD Mostrar Reportes Sección Mostrar principales buscadores por donde acceden los visitantes	133
D.18.DCD Mostrar Reportes Sección Mostrar Páginas Vistas por visitas(cambiar a visitas) . . . . .	134
D.19.DCD Mostrar Reportes Sección Mostrar Páginas Vistas . . . . .	135
D.20.DCD Mostrar Reportes Sección Mostrar errores de apache . . . . .	136
D.21.DCD Mostrar Reportes Sección Mostrar bytes transferidos . . . . .	136
E.1. Cambiar contraseña . . . . .	138
E.2. Cerrar Sesión . . . . .	139
E.3. Gestionar Administrador Escenario Adicionar Administrador . . . . .	140
E.4. Gestionar Administrador Escenario Cambiar Contraseña del Administrador . . . . .	141
E.5. Gestionar Administrador Escenario Editar Información Personal de Administrador . . . . .	142
E.6. Gestionar Administrador Escenario Eliminar Administrador . . . . .	143
E.7. Gestionar Usuario Escenario Adicionar Usuario . . . . .	144
E.8. Gestionar Usuario Escenario Cambiar Contraseña de Usuario . . . . .	145
E.9. Gestionar Usuario Escenario Editar Información Personal . . . . .	146
E.10.Gestionar Usuario Escenario Editar Información Personal de Usuario . . . . .	147
E.11.Gestionar Usuario Escenario Eliminar Usuario . . . . .	148
E.12.Mostrar Visitantes por Horas . . . . .	149
E.13.Mostrar Visitantes por Días . . . . .	150
E.14.Mostrar Robots . . . . .	150
E.15.Mostrar Referencias . . . . .	151
E.16.Mostrar Principales Navegadores y Sistemas Operativos utilizados . . . . .	152
E.17.Mostrar Principales Buscadores por donde acceden los visitantes . . . . .	153

E.18.Mostrar las visitas . . . . . 154

E.19.Mostrar Páginas Vistas . . . . . 155

E.20.Mostrar Errores de Apache . . . . . 156

E.21.Mostrar Bytes Transferidos . . . . . 157

F.1. Diagrama de Componente Acceso a Datos. . . . . 159

F.2. Diagrama de Componente Acciones. . . . . 160

F.3. Diagrama de Componente Plantillas. . . . . 161

# Introducción

---

Internet es una enorme red que conecta computadoras distribuidas por todo el mundo, permitiendo comunicación, búsqueda y transferencia de información sin grandes requerimientos tecnológicos ni económicos relativos para el individuo. Una forma moderna y rápida de estos intercambios de datos se realiza mediante los sitios web, que no son más que un conjunto organizado y coherente de páginas web conectadas entre ellas por vínculos y que muestran de forma sencilla textos, imágenes, videos, animaciones, sonidos, y otras formas de representar información.

En los últimos años el uso de Internet ha crecido vertiginosamente por parte de cualquier tipo de individuo, ampliándose los mercados y las áreas de interés tradicionales al mundo virtual. Esta circunstancia ha propiciado un incremento de la competitividad entre las empresas y una búsqueda de métodos más eficaces para llegar al usuario final. Estas empresas e instituciones se enfrentan a nuevos comportamientos de acceso a la información cada vez más dinámicos e inciertos, caracterizados por generar grandes volúmenes de datos, que una vez analizados pueden ser útiles para la organización.

Para cualquier entidad es imprescindible poder llegar a todos los usuarios sin importar limitaciones físicas ni psíquicas de estos, Internet, aunque es la forma ideal para llegar a todo el mundo, no está exenta de obstáculos, muchos derivados de malos diseños tanto técnicos como gráficos de las páginas web, es en este punto donde el análisis de los registros de servidores web desempeña un rol importante.

El análisis de registros de servidores web es una solución al problema que se plantea a la hora de la toma de decisiones sobre un sitio web, tanto técnicas como de diseño, y es utilizado para analizar los registros que generan los servidores de estos sitios sobre su tráfico y funcionamiento. Algunos de los sistemas de

análisis de registros de servidores web como WebTrends<sup>1</sup>, Awstats<sup>2</sup> y Sawmill<sup>3</sup> poseen una interfaz web para la interacción con el usuario.

Una interfaz web es un caso particular de las interfaces de manipulación directa también llamadas GUIs, estas son prácticamente universales. Las interfaces que utilizan ventanas, íconos y menús se han convertido en estándar en los materiales computacionales[1].

Las GUIs son también una especialización de las interfaces de usuario. Una interfaz de usuario es el vínculo entre el usuario y el programa de computadora. Una interfaz es un conjunto de comandos o menús a través de los cuales el usuario se comunica con el programa[2].

Una interfaz de usuario es una de las partes más importantes de cualquier programa ya que determina que tan fácilmente el usuario le dice al programa que desea hacer. Un programa muy poderoso con una interfaz pobremente elaborada tiene poco valor para un usuario no experto[3].

La *World Wide Web*(WWW) es otro éxito espectacular de la investigación sobre las interfaces de usuario[4], las páginas web dieron paso a la aparición de las IW, este tipo de interfaces deben servir de mediadoras entre usuarios con pocos conocimientos informáticos y sistemas que gestionen información. A pesar de que las aplicaciones web tienen un menor grado de interactividad que las aplicaciones de escritorio con los usuarios finales, han ido creando un alto grado de homogeneidad y el diseño de las mismas ha evolucionado hacia un esquema general definido, con un conjunto de componentes gráficos y funcionales similares que hacen posible que sea cual sea el usuario que acceda al sitio web la comunicación será simple y efectiva, además de poseer ventajas indiscutibles. Necesitan bajos requerimientos tanto de hardware como de software para el usuario final debido al poco consumo de recursos del navegador web; no lleva un proceso de instalación en todos los clientes, sino que consumen los recursos del servidor donde esté instalado el sistema informático; su mantenimiento es fácil, sin gastos asociados a la actualización de los clientes; son multiplataforma pues cualquier cliente desde cualquier sistema operativo puede acceder a ello mediante un navegador web; siempre que se siguen buenas prácticas, su desarrollo es económico, ágil y sencillo.

Las IW brindan acceso de forma sencilla a las funcionalidades de gestión de la información de un servicio o recurso determinado, según el grado de privilegio que tenga cada usuario dentro del sistema. En el marco

---

<sup>1</sup><http://www.webtrends.com/>

<sup>2</sup><http://awstats.sourceforge.net/>

<sup>3</sup><http://www.sawmill.net/>

de un software de análisis de registros de servidores web es una herramienta útil para su configuración y gestión.

Desde febrero del 2008, en la UCI a solicitud del Centro de Información para la Prensa (CIPRE), el cual es un centro de información especializada con alcance nacional sobre el periodismo en todas sus vertientes que se dedica al tratamiento de la información generada por fuentes internas y externas a la organización y ponen a disposición de sus usuarios productos y servicios tecnológicos e informativos, se desarrolla un software de análisis de registros de servidores web que lleva por nombre Airesweb, sistema que realiza un análisis de los archivos de registros de servidores web, está diseñado para instalarse en un servidor Debian GNU/Linux que no tendrá necesariamente un entorno gráfico de usuario. Además cuenta con una base de datos donde se almacena la información generada por el análisis de los registros de los servidores web, que es preciso gestionar para su funcionamiento, y aunque esta tarea puede realizarse utilizando algunas herramientas disponibles, no todos los usuarios cuentan con las habilidades necesarias para utilizarlas y el modo de presentarla no es el mejor estructurado.

A partir de los planteamientos anteriores, se identificó por parte del grupo de proyectos Analizadores Inteligentes de Registros de Servidores (AIRES), perteneciente al polo productivo *Soluciones Informáticas para Internet*, la necesidad de que los usuarios puedan interactuar con el sistema sin tener conocimientos avanzados, dando lugar al siguiente **problema a resolver**: ¿Cómo gestionar la información procesada por el sistema Airesweb?

Dando lugar a la siguiente **idea a defender**: El desarrollo de una Interfaz Web permitirá gestionar la información procesada por el sistema Airesweb.

La aceptación final de este producto, estará ligada en gran medida a la percepción que tengan los usuarios de dicha interfaz, teniendo en cuenta que será el único modo de interacción con el sistema y que el éxito o fracaso de un software está determinado por quienes lo usan, deben considerarse entonces, elementos de calidad y utilidad.

En contraste con el problema y la idea a defender antes planteada, se define como **objeto de estudio** las aplicaciones web y el **campo de acción** será las interfaces web de los sistemas de análisis de registro de servidores web.

El **objetivo general** puede definirse como: Desarrollar una IW para el sistema de análisis Airesweb que per-

mita gestionar los reportes estadísticos del sistema. Del mismo derivan los siguientes objetivos específicos:

- Caracterizar las aplicaciones web a partir del estado del arte asociado a ellas.
- Analizar y diseñar las funcionalidades de la IW del sistema Airesweb.
- Implementar las funcionalidades de la IW del sistema Airesweb.
- Probar las funcionalidades de la IW del sistema Airesweb en un entorno real.

Para lograr los objetivos propuestos se precisan las siguientes **tareas investigativas**:

- Investigar la existencia de sistemas similares relacionados con el que se va a desarrollar y la importancia que tienen las GUIs en los mismos.
- Definir que se entiende por aplicación web y analizar su historia, características, categorías y arquitectura.
- Seleccionar las tecnologías y herramientas adecuadas para el desarrollo de la IW teniendo en cuenta lenguaje de programación, framework a utilizar, Entorno de Desarrollo Integrado (IDE) y metodología de desarrollo.
- Realizar una descripción del problema a resolver y representarlo a través de un diagrama.
- Proponer una estructura para la IW e identificar, describir y modelar las funcionalidades que esta debe tener.
- Especificar las pautas de diseño gráfico para la IW del sistema Airesweb y diseñar la interfaz a partir de las mismas.

Del trabajo se esperan los siguientes **aportes prácticos**:

- Que facilite la gestión de la información procesada por el sistema Airesweb.
- Agilización y eficiencia en el uso de Airesweb para contribuir a la toma de decisiones sobre un sitio web.

- Que el usuario reciba la información mediante gráficos interactivos que le faciliten la comprensión de los datos.
- Esta aplicación informática contribuirá a la utilización de las herramientas de análisis de registros de servidores web y a fomentar el uso de la analítica web y del software libre en Cuba.

En el cumplimiento de las tareas se usarán los siguientes **métodos**:

- El **Analítico-Sintético** se aplicará para entender las aplicaciones web a partir del análisis de las características, categorías y arquitectura que presentan, y para formular conclusiones a través de la síntesis de los conocimientos y resultados obtenidos.
- El **Histórico-Lógico** permitirá una mayor comprensión del estado actual de las aplicaciones web a partir del análisis de su evolución y las etapas principales por las que han transitado.
- La **Entrevista** al cliente es esencial a la hora de definir las funcionalidades del sistema, identificando a la vez las funcionalidades particulares de cada cliente y las restricciones que se imponen.
- La **Modelación** mediante el lenguaje de modelado Unified Modeling Language (UML) se utilizará para reflejar la estructura, relaciones internas y características de la solución a través de diagramas.

El contenido del presente trabajo está estructurado en cuatro capítulos distribuidos de la siguiente manera: El primer capítulo está centrado en el estado del arte relacionado con el objeto de estudio. En el se analiza la existencia de soluciones a nivel nacional e internacional que puedan reutilizarse y se abordan temas relacionados con las GUIs y las aplicaciones web. Respecto a esta última se profundiza en aspectos tales como su historia o evolución, características, categorías existentes y arquitectura que poseen. Además se identifican las tecnologías y herramientas, así como la metodología para el desarrollo de la IW.

En el segundo capítulo se profundiza en el problema a resolver a través de su descripción y se muestra el modelo de dominio generado. Se realiza una propuesta de solución y se identifican los requisitos funcionales y no funcionales que deben tenerse en cuenta. Por último se hace una descripción de los Casos de Uso obtenidos a partir de los requisitos y se presentan los diagramas relacionados con estos.

En el tercer capítulo se presentan los artefactos generados en el flujo de análisis y diseño. Además se exponen las ideas relacionadas con la validación de los datos, el tratamiento de errores y la seguridad,

basándose en los mecanismos que el framework Symfony provee para dichas cuestiones y se describen una serie de aspectos relativos a la base de datos, el diseño gráfico de la presentación y la ayuda con que debe contar la aplicación.

El cuarto y último capítulo se enfoca en la implementación y prueba de la solución propuesta tomando como base los resultados que se presentan en los capítulos anteriores. También ilustra la relación de los principales componentes de la aplicación y como estará desplegada.

---

## Capítulo 1

# Fundamento teórico

---

Las GUIs permiten la interacción entre los sistemas informáticos y los usuarios finales de los mismos. Aunque siguen ligadas estrechamente a las aplicaciones de escritorio, han ido ganado un espacio en la Web, con la estandarización de esta. Una Interfaz de Usuario (UI) de un sistema de análisis de registros de servidores es un caso especial dentro de las aplicaciones web, siendo necesario para su comprensión un grupo de aspectos que van desde su surgimiento, evolución, características, categorías, arquitectura y tendencias actuales. En el marco de los sistemas de análisis de registros de servidores web, permite gestionar de forma eficiente los datos generados por el sistema Airesweb, precisándose para su desarrollo la selección de tecnologías y herramientas adecuadas.

### 1.1. Trabajos similares

Un analizador de registros de servidores web es un software que extrae información valiosa de los registros de los servidores web. Airesweb es un software que persigue este objetivo, y ante la necesidad de una IW para el mismo, se precisa un estudio de soluciones similares a nivel mundial.

#### 1.1.1. Ámbito internacional

En el caso específico de los sistemas de análisis de registros de servidores web las Interfaces Web para el proceso de muestra de los datos, son ampliamente utilizadas por ser una de las formas más baratas y sencillas de llevar esta información a varios usuarios. En las figuras A.1, A.2, A.3 y A.4 se muestran las interfaces web de los sistemas de análisis *AwStats*, *Sawmill*, *WebLog Expert* y *Webtrends* respectivamente.

A continuación una referencia de cada uno.

### **Awstats[5]**

<http://awstats.sourceforge.net/>

AWStats, abreviatura de Advanced Web Statistics, es una poderosa y completa herramienta que genera estadísticas gráficas avanzadas para sitios web. Este analizador de archivos de registros trabaja como un CGI o desde línea de comandos, y muestra, en pocas páginas, toda la información que los archivos de registros contienen. Puede analizar archivos de registro desde IIS (formato de archivo de registro *World Wide Web Consortium* W3C), archivos de registros de Apache/Linux, etc.), Windows, Macintosh y otras.XQX(formato archivo de registro NCSA combinado/XLF/ELF o formato de archivo de registro común/CLF).

AWStats muestra la siguiente información:

- Número de visitas y número de visitantes únicos.
- Duración de las visitas y últimas visitas.
- Usuarios autenticados y últimos usuarios autenticados.
- Días de la semana y horas de mayor tráfico (páginas, hits, KB por cada hora y día de la semana).
- Dominios/países de Host visitantes.
- Lista de Hosts, últimas visitas y lista de IP no resueltas.
- Páginas más vistas, páginas de entrada y salida.
- Tipos de archivo.
- Estadísticas de compresión web (para mod\_gzip o mod\_deflate).
- Navegadores utilizados (páginas, hits, KB por cada usuario, versión, etc.).
- Sistemas Operativos usados.
- Visitas de robots (307 robots detectados).

- Buscadores, palabras clave, frases clave usadas para encontrar tu sitio.
- Errores HTTP.
- Otros reportes personalizados sobre URL, parámetros de URL, referidos.
- Cantidad de veces que tu sitio es añadido a Favoritos.
- Ratio de navegadores con soporte Java, Flash, RealG2 reader, Quicktime reader, WMA reader, PDF reader.

Escrito en Perl, AWStats puede ser utilizado en la mayoría de los sistemas operativos. Es una aplicación muy popular como herramienta de administración de servidores. AWStats utiliza la GNU General Public License (GPL)[6].

### **Webtrends[7]**

<http://www.webtrends.com/>

Webtrends es privativo, provocando que su código no pueda ser reutilizado por terceras personas. Como segundo elemento, debe tenerse en cuenta que su IW constituye una solución a la medida que responde a las necesidades particulares de dicho sistema. Webtrends es la solución líder en el mercado de la analítica web que mide y obtiene información sobre las necesidades de sus clientes, respuestas y perfiles de navegación que serán muy útiles para tomar decisiones estratégicas.

### **Sawmill[8]**

<http://www.sawmill.net/>

Sawmill es un paquete de software para el análisis y presentación de informes estadísticos de archivos de registro, presenta filtrado dinámico de los datos a analizar, acercamiento a datos específicos y personalización de la interfaz de usuario. Sawmill también incorpora el análisis y alertas en tiempo real. Disponible desde 1997 soporta más de 850 formatos de archivos de registros de servidores. Sawmill también incluye un análisis con javascript del lado del cliente que proporciona una visión total del tráfico de visitantes y del comportamiento sobre el sitio web en si. Sawmill Analytics es ofrecido como paquete completo el cual

incluye análisis para servidores web, firewalls, servidores proxy, servidores de correo, dispositivos de red (enrutadores y conmutadores etc), servidores de bases de datos, o como paquete separado(uno solo de los anteriores).

### **WebLog Expert[9]**

<http://www.weblogexpert.com/>

WebLog Expert es un rápido y potente analizador de registro de servidores. Brinda información sobre los visitantes de un sitio web: estadísticas de actividad, acceso a los archivos, las rutas seguidas a través del sitio, información acerca de páginas de referencia, los motores de búsqueda, navegadores, sistemas operativos y otros. Produce reportes fáciles de leer en formato HTML que incluyen información en texto y gráficos. WebLog Expert puede analizar los registros de Apache y los servidores web Internet Information Server (IIS). Su interfaz es intuitiva y presenta un asistente para mejorar la interacción con el sitio.

#### **1.1.2. Ámbito nacional**

Aunque el desarrollo informático en Cuba ha tenido un auge en los últimos tiempos, gracias entre otras cosas al surgimiento y desarrollo de la UCI, no se encontraron referencias por el grupo de proyecto AIRES de ninguna solución de este tipo. Las interfaces de los sistemas mencionados, no constituyen una solución que pueda adaptarse a Airesweb ya que Sawmill, Webtrends y WebLog Expert como se expuso anteriormente son privativos, esto impide cualquier reproducción total o parcial, en cambio AwStats aunque si es libre y pudiera aprovecharse no es lo totalmente modular para que su código sea reutilizable. Se llega a la conclusión del desarrollo de una interfaz de usuario, para suplir la carencia de esta, la Web como medio para propiciar la interacción requerida, converge con el uso que hacen de esta los usuarios al navegar por Internet de modo que una IW sería la solución adecuada para realizar las tareas asociadas a la gestión de de la información generada por Airesweb.

Una IW es el resultado de la unión de los conceptos de las GUIs y las aplicaciones web, esto hace necesario ahondar en ambos elementos para poder establecer las bases teóricas que permitan llevar a cabo el desarrollo de la misma.

## **1.2. Interfaces Gráficas de Usuario**

Una UI es la parte de una aplicación informática que interactúa con el usuario. La UI más simple, conocida como consola, utiliza solo un monitor y un teclado conectado al sistema y un programa que interpreta los comandos tecleados por el usuario en la línea de comandos para ejecutar acciones específicas. En la actualidad, son comunes GUIs, que usan además dispositivos apuntadores como el ratón o mouse e incluyen múltiples elementos gráficos intuitivos como iconos o botones.

En los inicios solo se pensaba que fueran funcionales, pero con el pasar del tiempo fueron cambiando y haciéndolas más manejables por el usuario, necesitándose así, nuevos elementos gráficos como botones, menús, y adecuadas combinaciones de colores, en este momento entró a jugar un papel fundamental el diseño gráfico en el mundo de las GUIs.

Las GUIs constituyen una de las partes más importantes de casi todos los programas informáticos de la actualidad, considerándolo así ya que a través de ella determina cuán fácil es que el programa haga lo que el usuario desee. Lograr un buen diseño exige una gran dedicación, pues generalmente son complejas, difíciles de implementar, depurar y modificar, pero sus características y facilidades han logrado que se conviertan en un estándar en la actualidad.

## **1.3. Aplicaciones web**

En nuestro mundo actual, las tecnologías de la información son parte de la vida de la mayoría de los seres humanos, la comunicación ha llegado a un alto punto y las fronteras se desdibujan cada vez más. Un componente de este nuevo espacio, es la web, que como mayor plataforma de intercambio juega un papel fundamental en la vida de las personas.

Actualmente el uso que se le da a la Web es bastante diferente del propósito para el cual fue originalmente creada: un medio para compartir información científica entre algunos científicos. El alcance y la complejidad de las aplicaciones web actuales varían notablemente: desde servicios efímeros a pequeña escala hasta aplicaciones de empresa a gran escala distribuidas a través de la Internet y de las Intranets/Extranets corporativas[10].

### 1.3.1. Concepto de aplicación web

¿Qué es una aplicación web? Por definición, es algo más que un "sitio web". Es una aplicación cliente/servidor que utiliza un navegador web como su programa cliente, y por eso constituye un servicio interactivo mediante la conexión con servidores a través de Internet (o Intranet). Un sitio web ofrece simplemente el contenido de los archivos. Una aplicación web debe adaptar dinámicamente su contenido sobre la base de los parámetros de una solicitud, seguir el comportamiento de los usuarios, y tener consideraciones de seguridad"[11].

Estas han ganado en popularidad debido a la facilidad para actualizarlas y mantenerlas, sin tener que entrar al engorroso sistema de distribuir e instalar software en cada uno de los clientes. Aplicaciones como los webmails, tiendas online(en donde se realiza la mayor parte del comercio electrónico hoy día), pagos de servicios online, accesos a cuentas bancarias, etc, son ejemplos conocidos de su viabilidad.

En su libro Web Engineering, Kappel define de un forma magistral las aplicaciones web: "Una aplicación web es un software basado en tecnologías y estándares de W3C que provee recursos específicos tales como contenidos y servicios a través de una UI (User Interface por sus siglas en inglés) a la que puede accederse utilizando un navegador web"[12].

#### **Ventajas y desventajas**

Ventajas[13]:

- Las aplicaciones web son fáciles y baratas de ofrecer. Con aplicaciones web, una empresa puede reducir los costos del departamento que se encarga de la instalación del software en las máquinas de los usuarios. Con aplicaciones web, los usuarios todo lo que necesitamos es un equipo de trabajo con un navegador web y una conexión de Internet o Intranet.
- Los gastos de mantenimiento del software siempre han sido significativos debido a que la mejora de un por ciento de el software es similar a la instalación de uno nuevo. Sin embargo con las aplicaciones web, tan pronto como la aplicación en el servidor está actualizada, todos los usuarios reciben la nueva versión.

- Las aplicaciones web tienen requisitos flexibles para los usuarios finales. La aplicación solo estará instalada en un servidor y se utilizará a través de Internet/Intranet desde cualquier sistema operativo moderno como Mac, Windows o Linux. Si la solicitud está bien construida, funcionará igual de bien en cualquier navegador web moderno, como Internet Explorer, Mozilla Firefox, Opera o Safari.
- Las aplicaciones web pueden disponer de un almacén de datos central. Cuando haya varios lugares que necesitan tener acceso a los mismos datos, teniendo todos los datos almacenados en un lugar es mucho más fácil que tener bases de datos independientes en cada lugar. De esta manera se evitan posibles errores de desincronización de datos y menores riesgos de seguridad.

Desventajas:

- Acceso limitado, la necesidad de conexión permanente y rápida a Internet/Intranet hacen que el acceso a estas aplicaciones no esté al alcance de todos.
- La interactividad no se produce en tiempo real, en las aplicaciones web cada acción del usuario conlleva un tiempo de espera hasta que se obtiene la respuesta del sistema.
- Elementos de interacción muy limitados. En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones web (mediante formularios principalmente) son muy escasas, aunque se ha avanzado mucho con la introducción del Ajax en la construcción de páginas web.

### **1.3.2. Evolución de las aplicaciones web**

Al principio del concepto de Internet, la web era una colección de páginas estáticas, documentos, etc., para su consulta o descarga[14]. Se hacía necesario dar un nuevo paso en la interacción cliente/página web, es donde surge el concepto de páginas dinámicas, que permitieran que su contenido se generara a partir de lo que un usuario introduce en un formulario, este contenido puede leerse desde una BD o actualizarse en esta. Una de las primeras formas de programación dinámica que surgen es *Common Gateway Interface*(CGI). El funcionamiento de los CGI tenía un punto débil: cada vez que se recibía una petición, el servidor debía lanzar un proceso para ejecutar el programa CGI. Como la mayoría de CGI estaban escritos en lenguajes

interpretados, como Perl o Python, o en lenguajes que requerían run-time environment”, como Java o Visual-Basic, el servidor se veía sometido a una gran carga. La concurrencia de múltiples accesos al CGI podía presentar problemas graves[14].

Por eso se empiezan a desarrollar alternativas a los CGI que solucionarán el problema del rendimiento. Las soluciones llegan básicamente por 2 vías: 1) se diseñan sistemas de ejecución de módulos mejor integrados con el servidor, que evitan la instanciación y ejecución de varios programas, y 2) se dota a los servidores de un intérprete de algún lenguaje de programación que permita incluir el código en las páginas de forma que lo ejecute el servidor, reduciendo el intervalo de respuesta[14].

### 1.3.3. Generaciones

A continuación se muestran las generaciones en que se ha dividido la evolución de las aplicaciones web[15].  
Primera generación: Esta generación abarca desde el 1992 (nacimiento de la Web) hasta mediados de 1994. La creación de páginas web durante esta generación se ve limitada por diversas razones tecnológicas: ancho de banda limitado (módems de 2.4 Kbps), navegadores poco desarrollados, monitores monocromos, etc.

- Tiempo de carga rápido. Uso de textos, pocas imágenes y ningún recurso multimedia.
- Navegación poco estructurada.
- Páginas largas que parece que nunca se acaban. Poco uso de enlaces entre páginas.
- Texto escrito como si fuera una hoja de papel: de lado a lado de la página y desde el principio hasta el final.
- Empleo de líneas horizontales para separar secciones en una misma página.
- Páginas que poseen un contenido educativo o científico. Pocas empresas poseen un sitio web.
- Empleo de listas para organizar la información.
- Se pueden visualizar prácticamente en cualquier navegador.

En definitiva, durante este primer periodo, se emplea la Web como si fuera uno de los medios de comunicación tradicionales (libros, revistas, etc.). Aún no se sabe como aprovechar todas las posibilidades que ofrece

la Web. Respecto a la generación de las páginas, no existe generación: las páginas son estáticas. A finales de este primer período aparece la tecnología CGI, que permite la generación dinámica de páginas web.

Segunda generación: La segunda se extiende desde 1995 hasta la actualidad. La diferencia principal con las páginas web de la generación anterior es la masiva incorporación de elementos gráficos en las páginas web:

- Los *iconos* sustituyen las palabras.
- El color de fondo se sustituye por una imagen.
- Los *banners* sustituyen los encabezados de las páginas.
- Tiempo de carga lento: se emplean imágenes con multitud de colores y animaciones en exceso, debido a la novedad de su uso. No se comprueba el rendimiento de las páginas con conexiones lentas: no se tiene en cuenta al usuario final.
- Las páginas todavía poseen una estructura de arriba a abajo.
- La navegación suele ser jerárquica, a partir de una página principal. Sin embargo, no existe una filosofía de planificación de la navegación.
- Aparecen tecnologías multimedia propietarias, que necesitan la instalación de un plug-in para su visualización. Prima el uso de tecnologías (imágenes y sonidos), aunque luego el público no pueda visualizar correctamente las páginas.

En definitiva, las páginas web de esta generación se caracterizan porque prima el uso de la tecnología, sin tener en cuenta el propósito del sitio web. Además, no se tiene en cuenta la legibilidad o claridad de la presentación de la información.

Respecto a la generación de las páginas, la mayoría siguen siendo estáticas, aunque cada vez se emplea más la tecnología CGI. El uso de esta tecnología abre un abanico de posibilidades enorme: la creación de aplicaciones web que acceden a bases de datos.

Las primeras aplicaciones que se desarrollan son pequeñas y sencillas: libro de visitas, formulario de más información, etc. Si se necesita almacenar información de forma persistente, se emplean ficheros en vez de bases de datos.

Tercera generación: Esta generación aparece a mediados de 1996. Las páginas pertenecientes a esta generación son las más comunes en la actualidad. Se caracterizan por:

- Tiempo de carga rápido. Uso de *Cascading Style Sheets* (CSS) y optimización del código *HyperText Markup Language* (HTML).
- Las páginas se limitan para que se puedan visualizar completamente en una pantalla sin tener que realizar desplazamiento (*scroll*).
- Se crean pensando en el usuario final, teniendo como objetivos ofrecer servicios, informar, vender, etc.
- Se tienen en cuenta principios tipográficos y de organización visual de la información.
- Se incorporan los principios de usabilidad y accesibilidad.
- Se comprueba con usuarios reales su funcionamiento.
- Se emplean de forma coherente los colores, las imágenes, tipos de letras, los símbolos e *iconos*, etc.

La característica principal de las páginas web pertenecientes a la tercera generación es la planificación: los diseñadores invierten tiempo en analizar los posibles caminos que los visitantes tomarán al visitar un sitio web, y en función de ello diseñan los sitios web.

Durante este período la estructura del sitio web cobra una gran importancia además de que hay una explosión en el número de herramientas informáticas relacionadas con la Web.

En cuanto a la generación de las páginas, este periodo supone la consolidación de la generación de páginas web dinámicas. El uso de CGI está muy extendido, pero debido a sus limitaciones aparecen nuevas tecnologías. Las primeras soluciones relevantes provienen de Microsoft, primero con *Internet Database Connector* IDC y luego con *Active Server Pages* ASP, que supone una verdadera revolución en la creación de páginas web dinámicas. A partir de ahí aparecen nuevas tecnologías ColdFusion, PHP o *Java Server Page* JSP basada en Java.

Cuarta generación: Esta es considerada última generación hasta el momento, empieza a desarrollarse plenamente en 1999 y discurre hasta la actualidad. Sus características principales son:

- Se emplean nuevamente los recursos gráficos.

- HTML evoluciona: se extiende el uso de tecnologías poco empleadas hasta ese momento, como CSS y aparecen nuevas como *Dynamic HTML* (DHTML). Estas tecnologías generan incompatibilidad entre distintos navegadores, a pesar de introducir considerables ventajas.
- Uso de nuevas tecnologías multimedia: Se puede crear un sitio web sin emplear HTML.
- Crean a los usuarios una experiencia agradable desde que visita la primera página hasta que abandona el sitio.
- El aumento del ancho de banda permite streaming de video y audio en tiempo real.
- Respecto a la generación de páginas web dinámicas supone la consolidación de las tecnologías de generación dinámica. La mayoría de las páginas web pertenecientes a esta generación se crean a partir de información almacenada en bases de datos.

Esta generación constituye un salto en la forma de ver Internet con la aparición del concepto Web 2.0. Las aplicaciones web alcanzan su máximo esplendor y son utilizadas para cualquier tipo de gestión de la información.

### **La Web 2.0[16]**

La Web 2.0 es la representación de la evolución de las aplicaciones tradicionales hacia aplicaciones web enfocadas al usuario final. La Web 2.0 es una actitud y no precisamente una tecnología.

La Web 2.0 es la transición que se ha dado de aplicaciones tradicionales hacia aplicaciones que funcionan a través del web enfocadas al usuario final. Se trata de aplicaciones que generen colaboración y de servicios que reemplacen las aplicaciones de escritorio.

Es una etapa que ha definido nuevos proyectos en Internet y está preocupándose por brindar mejores soluciones para el usuario final.

Principios de las aplicaciones web 2.0:

- La web es la plataforma.
- La información es lo que mueve Internet.

- Efectos de la red movidos por una arquitectura de participación.
- La innovación surge de características distribuidas por desarrolladores independientes.
- El fin del círculo de adopción de software pues existen servicios en beta perpetuo.
- Se simula el software de escritorio pero sin la necesidad de instalarlo en la computadora.
- Sitios fáciles, usables y ágiles para localizar la información.
- Colaboración en línea a través de los distintos recursos disponibles.
- Nuevos procedimientos para trabajar, comunicarse y participar en la Web.

### Ejemplos de Web 2.0

Entender la evolución que ha llegado con la Web 2.0 puede realizarse con ejemplos, con proyectos. Podemos comparar servicios web que marcan claramente la evolución hacia la Web 2.0 con una nueva forma de hacer las cosas:

Web 1.0	Web 2.0	Propósito
DoubleClick	Google AdSense	Servicios Publicidad
Ofoto	Flickr	Comunidades fotográficas
Akamai	BitTorrent	Distribución de contenidos
mp3.com	Napster	Descargas de música
Britannica Online	Wikipedia	Enciclopedias
Sitios personales	Blogs	Páginas personales

Cuadro 1.1: Evolución de aplicaciones hacia la Web 2.0

Tecnologías que dan vida a un proyecto Web 2.0:

- Transformar software de escritorio hacia la plataforma web.
- Respeto a los estándares como el XHTML.

- Separación de contenido del diseño con uso de hojas de estilo.
- Sindicación de contenidos.
- Ajax (javascript asíncrono y XML).
- Uso de Flash, Flex o Lazlo.
- Uso de Ruby on Rails para programar páginas dinámicas.
- Utilización de redes sociales al manejar usuarios y comunidades.
- Dar control total a los usuarios en el manejo de su información.
- Proveer APis o XML para que las aplicaciones puedan ser manipuladas por otros.
- Facilitar el posicionamiento con URL sencillos.

#### 1.3.4. Características

Las aplicaciones web tienen diversas aplicaciones y por tanto, diversas formas de infraestructura, sin embargo, tienen los siguientes atributos comunes[17]:

- **Intensivas en la red:** Está disponible en la red: Intranet, Extranet o Internet.
- **Controladas por el contenido:** Uso de hipermedia.
- **Evolución continua:** Actualizaciones frecuentes.
- **Inmediatez:** Tiempo de distribución muy corto.
- **Seguridad:** Infraestructura de seguridad.
- **Estética:** Atractivo del sitio.

A continuación se exponen las características de las aplicaciones web simples (más tempranas) y de las aplicaciones web avanzadas(más recientes)[18].

<b>SISTEMAS BASADOS EN LA WEB SIMPLES</b>	<b>SISTEMAS BASADOS EN LA WEB AVANZADOS</b>
Páginas web simples presentando principalmente información textual.	Páginas web complejas.
El contenido de la información no cambia, bastante estático.	La información es dinámica, cambia con el tiempo y las necesidades de los usuarios.
Navegación simple.	Dificultad para navegar y encontrar la información.
Sistemas automáticos	Sistemas integrados con bases de datos y otros sistemas de planificación, programación y localización/rastreo.
No es un requerimiento mayor el alto rendimiento.	Requieren alto rendimiento y disponibilidad
Desarrollados por un individuo o un equipo pequeño.	Requieren un gran equipo de desarrollo con experiencia en diversas áreas.
Usados para la diseminación de información en aplicaciones no vitales.	Despligue en misión, aplicaciones críticas.

Cuadro 1.2: Características de los sistemas basados en la Web - simples / avanzados

### 1.3.5. Categorías o tipos

De acuerdo a la funcionalidad, los sitios web se clasifican en[17]:

- Informativa.
- Descarga.
- Personalizable.
- Interacción.
- Entrada de usuario.
- Orientación a transacciones.

- Orientado a servicios.
- Portal.
- Acceso a Base de datos.
- Almacenes de datos.

También otras clasificaciones, como lo que propone Ginige[18].

<b>CATEGORÍA</b>	<b>EJEMPLOS</b>
De información.	Diarios 'OnLine', catálogos de productos, boletines, manuales de servicio, clasificados 'OnLine', libros electrónicos 'OnLine'.
Interactivo ( Usa información provista por el usuario o personalizada al acceso).	Formularios de registro, presentación de información, juegos 'OnLine'.
De transacción.	Comercio electrónico, pedido de servicios y banca 'OnLine'.
Flujo de trabajo.	Sistemas de programación y planificación 'OnLine', gestión de inventarios, monitoreo de estado.
Entornos de trabajo de colaboración.	Sistemas de autoridad distribuidos, herramientas de diseño de colaboración.
Mercados, comunidades 'OnLine'.	Grupos de 'Chat', sistemas de recomendaciones que aconsejan productos y servicios, mercados 'OnLine', subastas 'OnLine'.
Portales web.	Centros de compras electrónicos, intermediarios 'OnLine'.

Cuadro 1.3: Categorías de las aplicaciones web

### 1.3.6. Arquitectura

La calidad de una aplicación web es considerablemente influenciada por su arquitectura subyacente. Aspectos arquitectónicos incompletos o perdidos hacen que sea difícil darse cuenta de los requisitos de la calidad

de las aplicaciones web; bajo rendimiento, insuficiente mantenimiento, y la baja disponibilidad de una aplicación web son a menudo causadas por una inadecuada arquitectura. Además de las limitaciones técnicas, la arquitectura de las aplicaciones web también debe considerar el marco organizativo en el que están incorporados. El uso flexible de las arquitecturas multicapa, el examen de contenidos multimedia, y la integración de los repositorios de datos y aplicaciones son desafíos en el desarrollo exitoso de las arquitecturas de las aplicaciones web[19].

### **Modelo Cliente Servidor**

En el modelo cliente-servidor, actúan un grupo de programas distribuidos que están corriendo en la red de computadoras, e interactúan a través de protocolos de comunicación conocidos, en lugar de funcionar todos en un único sistema, las aplicaciones cliente-servidor distribuyen el procesamiento entre el servidor y los equipos cliente[20].

Una aplicación web es una aplicación cliente-servidor(generalmente), utiliza los navegadores web como su cliente. Los navegadores envían las solicitudes a los servidores, los servidores generan una respuestas y se las regresan a los navegadores[20].

La arquitectura cliente-servidor llamado modelo cliente-servidor es una forma de compartimentar y centrar los programas y las computadoras para ganar en eficiencia, simplificación, y objetividad.

Las aplicaciones web están basadas en varios componentes que se comunican entre sí mediante el principio petición-respuesta. Un componente envía una petición a otro componente y este envía una respuesta a dicha solicitud a través de un canal de comunicación sincrónico. A continuación se muestra una breve descripción de los componentes principales:

- **Cliente:** generalmente es un navegador web que es utilizado por un usuario para acceder a la aplicación.
- **Servidor web:** es un software específico que soporta varios protocolos entre los cuales están HTTP y *HyperText Transfer Protocol Security* (HTTPS). Permite además procesar y responder a las peticiones de los clientes.
- **Servidor de base de datos:** es utilizado para almacenar información persistente.

La figura 1.1 muestra la relación entre ellos.

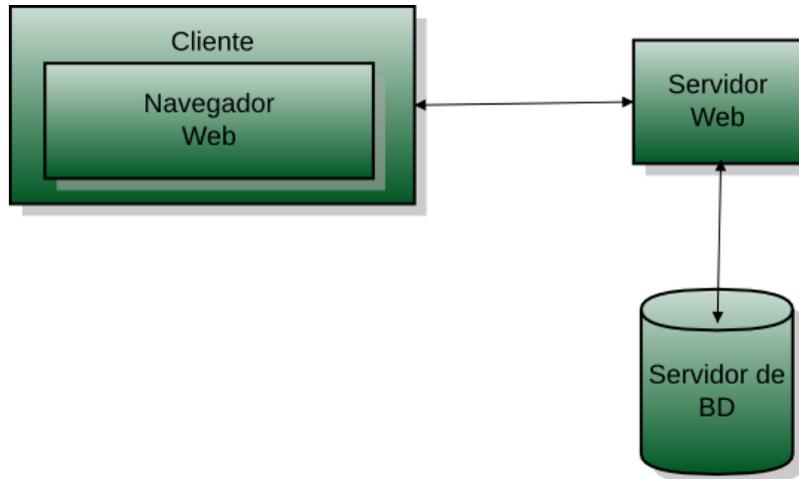


Figura 1.1: Componentes genéricos de una aplicación web.

## 1.4. Tecnologías a utilizar

Airesweb es un software que será implantado en un entorno real, por lo que la selección de las tecnologías debe ser un proceso cuidadoso, se debe tener en cuenta todos los factores relacionados con el despliegue final del software así como la disponibilidad de las herramientas, sus licencias y su grado de dificultad son puntos claves a tratar.

### 1.4.1. Lenguaje de programación

Las tecnologías que están disponibles para la web pueden ser agrupadas en dos grandes categorías, según la propia arquitectura de una aplicación web. **Tecnologías del lado del cliente:** disponibles para los clientes web(en su mayoría navegadores web), para hacer más interesante la presentación de los documentos. Usualmente estas tecnologías son extensiones de HTML. **Tecnologías del lado del servidor:** son las que están disponibles en el servidor, que es el encargado de almacenar la lógica para construir las páginas web dinámicas[13].

En el momento de escoger un lenguaje de programación, debe tenerse en cuenta específicamente lo perse-

guido. La versatilidad de un lenguaje está estrechamente relacionada con la complejidad que posea. Según la complejidad en el aprendizaje de cierto lenguaje, así será el espectro de tareas que puede resolver, mientras más complejo más amplio[21].

En el dominio de las redes, los lenguajes comúnmente más utilizados son: PHP, ASP, Java, Ruby on Rail Python y PERL. Satisfaciendo las necesidades del cliente y futuro usuario. La elección de uno de ellos debe sustentarse principalmente en las particularidades de la solución que se proponga y los conocimientos de los desarrolladores de los mismos. A partir de esto se eligió PHP, teniendo en cuenta sus características, ventajas y desventajas.

## PHP

*PHP* es el acrónimo de *PHP Hipertext Preprocesor*[22]. Es un lenguaje de programación del lado del servidor gratuito e independiente de la plataforma, rápido, con una gran librería de funciones y mucha documentación. PHP es un lenguaje híbrido entre algunos de sus predecesores como C, shell, Perl, tomando lo mejor de estos lenguajes y creando un fácil lenguaje script para la programación de aplicaciones web[22].

PHP se escribe dentro del código HTML, lo que lo hace realmente fácil de utilizar. Cualquiera puede descargar a través de la página principal de PHP (<http://www.php.net/>) y de manera gratuita, un módulo que hace que el servidor web comprenda los scripts realizados en este lenguaje. Es independiente de plataforma, puesto que existe un módulo de PHP para casi cualquier servidor web. Esto hace que cualquier sistema pueda ser compatible con el lenguaje y significa una ventaja importante, ya que permite portar el sitio desarrollado en PHP de un sistema a otro sin mucho esfuerzo[23].

### **Características.** [24]

- Simplicidad: usuarios con experiencia en Perl, C ó C++ podrán utilizarlo rápidamente.
- Velocidad: alta velocidad de ejecución sin introducir demoras en la máquina, bajo consumo de recursos y muy buena integración con Apache.
- Estabilidad: ninguna aplicación está completamente libre de errores (*bugs*), pero con una amplia comunidad de programadores y usuarios corregirlos es mucho más fácil. Posee un sofisticado manejo de variables que lo hacen muy robusto y estable.

- Seguridad: permite la protección contra diversos ataques a través de diferentes niveles de seguridad que pueden ser configurados desde el archivo *.ini*.

### **Ventajas.** [24]

- PHP es *Open Source* o código abierto, lo cual significa que el usuario no depende de una compañía específica para arreglar cosas que no funcionan, además no está forzado a pagar actualizaciones anuales para tener una versión que funcione.
- Gran número de funciones predefinidas. A diferencia de otros lenguajes de programación, PHP fue diseñado especialmente para el desarrollo de páginas web dinámicas. Por ello, está dotado de un gran número de funciones que simplificarán enormemente tareas habituales como descargar documentos, enviar correos, trabajar con cookies y sesiones.
- Puede interactuar con muchos motores de bases de datos tales como MySQL, Oracle, Informix, PostgreSQL o cualquier otra base de datos compatible con ODBC<sup>1</sup>.
- Actualmente se puede ejecutar bajo Apache, IIS, AOLServer, Roxen y THTTPD.
- PHP corre en “casi” cualquier plataforma utilizando el mismo código fuente, pudiendo ser compilado y ejecutado en aproximadamente 25 plataformas, incluyendo diferentes versiones de Unix, Windows (95,98,NT,ME,2000,XP,etc) y Macs.

En una entrevista, el creador de PHP Rasmus Lerdorf<sup>2</sup> resumió de forma brillante porque usar PHP, “PHP como lenguaje de programación no es mejor, probablemente es peor que otros lenguajes, pero lo que tiene PHP es que está orientado a un objetivo: la programación web en Internet.”

### **PHP5**

En los últimos años, PHP ha crecido hasta ser el más generalizado en Internet, funcionando en más de un tercio de los servidores web a través de el mundo. PHP ha tenido un crecimiento no solo cuantitativo sino también cualitativo. Cada vez más empresas, se basan en PHP para administrar sus aplicaciones críticas[22].

---

<sup>1</sup>Open Database Connectivity Standard

<sup>2</sup>Arquitecto de infraestructuras y creador del lenguaje PHP. Lerdorf pertenece al selecto grupo de personas sin las cuales no hubiera sido posible la Web tal y como la conocemos hoy

*PHP5* ha mantenido el liderazgo en la web. *PHP5* no solo revoluciona el modelo orientado a objetos sino que también contiene muchas nuevas características que lo convierten en un lenguaje mucho más maduro[22]. La llegada de *PHP5* vino emparejada de una reestructuración del Core de PHP, lo que los creadores de PHP llaman Zend Engine<sup>3</sup>. Así, como el lejano PHP3 incluye su Zend Engine 0.5, y PHP4 el Zend Engine 1.0, tenemos Zend Engine 2.0 en PHP5. El cambio de versión no fue trivial; incluye la reescritura casi total del modelo de objetos, entre sus cambios más sustanciales[25].

La programación orientada a objetos y todos sus estándares está siendo completada, ya están añadidos los modificadores de acceso a los métodos, *public/private/protected*, nombre unificado para el constructor de la clase, soporte de un destructor de la clase con un nombre genérico, soporte de *Interface*, que le da la posibilidad a las clases de tener varias relaciones, nuevo operador para conocer el tipo de una variable, el paso de parámetros por referencia de forma implícita, declaración de métodos abstractos, entre otras[22]. Además incluye mejoras importantes en el rendimiento y la seguridad.

Por lo antes expuesto se decide utilizar PHP en su versión actual 5.2.6.

## 1.4.2. Lenguaje de modelado

### UML

Lenguaje Unificado de Modelado UML, por sus siglas en inglés, es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el Object Management Group (OMG). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables[26].

## 1.4.3. Framework

En general, con el término framework, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos

---

<sup>3</sup><http://www.zend.com/php5/zend-engine2.php>

añadirle las últimas piezas para construir una aplicación concreta[27].

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. Un framework web, por tanto, podemos definirlo como un conjunto de componentes que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas web[27].

La mayoría de frameworks web se encargan de ofrecer una capa de controladores de acuerdo con el patrón MVC, ofreciendo mecanismos para facilitar la integración con otras herramientas para la implementación de las capas de negocio y presentación[27].

A continuación se enuncian una serie de características que podemos encontrar en prácticamente todos los frameworks existentes[27].

- No es necesario manipular directamente las URLs ni las sesiones, el framework ya se encarga de hacerlo.
- Incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en BBDD, XML, etc.
- La mayoría de frameworks implementa una serie de controladores para gestionar eventos, como una introducción de datos mediante un formulario o el acceso a una página. Estos controladores suelen ser fácilmente adaptables a las necesidades de un proyecto concreto.
- Incluyen mecanismos para la identificación de usuarios mediante login y password y permiten restringir el acceso a determinadas páginas a determinados usuarios.

### **Framework Symfony**

Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web[28].

Symfony está desarrollado completamente con PHP 5. Ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel. Symfony es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Es multiplataforma. Se seleccionó a Symfony como framework a utilizar por algunas de las características que a continuación se mencionan[28]:

- Fácil de instalar y configurar en la mayoría de plataformas.
- Independiente del sistema gestor de bases de datos.
- Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador solo debe configurar aquello que no es convencional.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- Preparado para aplicaciones empresariales y adaptable a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros.
- Está escrito en PHP5 basándose en el patrón MVC.
- Cuenta con la versión estable 1.0.
- Numerosos *plugins* son creados y actualizados constantemente ante las necesidades de los desarrolladores.

Fabien Potencier<sup>4</sup> resume de forma precisa, “... si desarrollas aplicaciones web complejas con mucha lógica de negocio, no es recomendable utilizar solo PHP. Para asegurar el mantenimiento y las ampliaciones futuras

---

<sup>4</sup>Fundador del proyecto Symfony, presidente de Sensio, empresa francesa de desarrollo de aplicaciones web

de la aplicación, es necesario que el código sea ligero, legible y efectivo. Si quieres incorporar los últimos avances en interacción con usuarios(...), puedes acabar escribiendo cientos de líneas de JavaScript. Si quieres desarrollar aplicaciones de forma divertida y muy rápida, no es aconsejable utilizar solo PHP. En todos estos casos, deberías utilizar Symfony.”

#### 1.4.4. Herramientas

##### Herramienta de programación

En la actualidad un IDE de programación es una herramienta indispensable para el desarrollo de un proyecto, aunque son varios los que están disponibles para el desarrollo de un proyecto utilizando PHP como Zend Studio<sup>5</sup>, Open Komodo Project<sup>6</sup>, PhpEdit<sup>7</sup>, Php Designer<sup>8</sup>, Netbeans<sup>9</sup>, Eclipse PDT<sup>10</sup>. En la prestigiosa revista Smashing Magazine<sup>11</sup> nos muestran un estupendo artículo “The Big PHP IDE Test: Why Use One And Which To Choose” con una tabla comparativa de IDE de PHP incluida, abordando las ventajas y desventajas de cada IDE. La tabla se encuentra en el documento adjunto **PHP\_IDE.pdf**.

Basándonos en esto y las características que a continuación se exponen se escogió Eclipse PDT como IDE de programación.

Eclipse es una plataforma de desarrollo de código abierto basada en Java. Es un desarrollo de IBM cuyo código fuente fue puesto a disposición de los usuarios. En si mismo Eclipse es un marco y un conjunto de servicios para construir un entorno de desarrollo a partir de componentes conectados (plugins). Cuenta con un editor de texto donde puede verse el contenido de los archivos que se están trabajando, una lista de tareas, y otros módulos similares. Si bien las funciones de Eclipse son más bien de carácter general, las características del programa se pueden ampliar y mejorar mediante el uso de plugins[29].

Uno de los plug-in básicos para el trabajo con PHP es el PTD.

Eclipse PDT es un conjunto de herramientas y frameworks que mejoran la productividad de los desarrolla-

<sup>5</sup>[http://www.zend.com/products/zend\\_studio](http://www.zend.com/products/zend_studio)

<sup>6</sup><http://www.activestate.com/openkomodo/>

<sup>7</sup><http://www.waterproof.fr/products/PHPEdit/>

<sup>8</sup><http://www.mpsoftware.dk/phpdesigner.php>

<sup>9</sup><http://www.netbeans.org/>

<sup>10</sup><http://www.eclipse.org/pdt/>

<sup>11</sup><http://www.smashingmagazine.com/>

dores que usan PHP por las siguientes características[30]:

- Editor sensible al contexto, el cual resalta el código, asistente de código y autocompletado de código.
- Integración con el modelo del proyecto Eclipse y la vista PHP Explorer.
- Soporte para el debug incremental del código de PHP.
- Extensos frameworks y Interfaz de aplicación (APIs) que permiten a los desarrolladores extender PDT para crear nuevas herramientas orientadas al desarrollo de PHP.

Principios del proyecto Eclipse PDT[30]:

- Fácil de usar e intuitivo.
- Se adhiere a los estándares de Eclipse.
- Extensibilidad y soporte continuo para el desarrollo de PHP.

Otra característica importante es la integración con subversión como herramienta para el control de versiones.

### **Herramienta de modelado**

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML. Presenta un entorno *todo-en-uno* para la especificación de los detalles de los casos de uso, incluyendo la especificación del modelo general y de las descripciones de los casos de casos de usos[31].

## 1.5. Metodología de desarrollo

La Web e Internet al principio crecía de manera poco organizada. Luego, la gente especializada entendió que debía llevarse el asunto a un trato formal, es decir el uso de ingeniería. De esta manera se establecieron principios que llevan al desarrollo disciplinado de la Web. Al igual que los grandes sistemas, la Web necesita la aplicación de conceptos de ingeniería a fin de llevar al éxito el desarrollo de enormes sistemas web que estarán al servicio de la sociedad y que no se pueden dar el lujo de fallar porque ocasionaría grandes pérdidas y la desconfianza de la gente[32].

“La ingeniería web está relacionada con el establecimiento y utilización de principios científicos, de ingeniería y gestión, y con enfoques sistemáticos y disciplinados del éxito y desarrollo, empleo y mantenimiento de sistemas y aplicaciones basados en la Web de alta calidad”[17].

Sin embargo, esta disciplina es relativamente joven y no posee sus propios modelos de procesos para desarrollar aplicaciones web, por tanto muchas personas tratan de adaptar los existentes para el desarrollo de software convencional, proponiendo nuevas metodologías [33]. Algunas de ellas son:

- WAE-Process Conallen (Web Application Extension for UML – Process Conallen).
- OO-Method, con su adaptación a interfaces hipertextuales.
- SOHDM (Scenario-based Object-oriented Hypermedia Design Methodology).
- WebML (Web Modelling Language).
- UML Based Web Engineering (UWE).
- OOWS (Object Oriented Web Solutions).

### 1.5.1. RUP

Teniendo en cuenta la necesidad e importancia de utilizar una metodología, en el caso particular de la IW de Airesweb se decidió utilizar Rational Unified Process (RUP). La decisión final quedó sustentada por los siguientes elementos:

- Es una de las metodologías más usadas y mayormente probadas a nivel internacional para el desarrollo de software e incluye una extensión para UML que permite el modelado de aplicaciones web.
- Para mantener un proceso homogéneo entre el desarrollo de la IW y Airesweb, ya que este usa RUP para la gestión de sus procesos.
- La experiencia del equipo de desarrollo.

Por otra parte el hecho de RUP utilizar como lenguaje de modelado UML, que es orientado a objetos, favorece y se encuentra en concordancia con el desarrollo que se pretende realizar, aunque es válido destacar que esta metodología aún posee algunas desventajas en el desarrollo de aplicaciones web, relacionadas con requerimientos de navegación y diseño gráfico [34]. Un elemento importante que se debe tener presente es el uso de una herramienta CASE<sup>12</sup> de modo que se seleccionó Visual Paradigm<sup>13</sup> por ser la mejor alternativa en entornos libres.

Entre las características fundamentales de esta metodología están las siguientes [35]:

- Dirigido por Casos de Uso: Tiene a los Casos de Uso como el hilo conductor que orienta las actividades de desarrollo. Se centra en la funcionalidad que el sistema debe poseer para satisfacer las necesidades de un usuario (persona, sistema externo, dispositivo) que interactúa con él.
- Centrado en la arquitectura: Propone arquitectura de forma similar a la de un edificio. Es necesario tener varios planos con diferentes aspectos, para tener una imagen completa del edificio antes que comience su construcción, aquí entra a jugar el término Arquitectura de Software, que abarca las diferentes vistas que se mencionaron anteriormente.
- Iterativo e incremental: Propone la descomposición de proyectos grandes en mini-proyectos, cada mini-proyecto es una iteración, y cada iteración debe estar controlada y tratar un determinado grupo de casos de uso.

---

<sup>12</sup>Computer Aided Software Engineering (Ingeniería de Software Asistida por Ordenador)

<sup>13</sup><http://www.visual-paradigm.com/>

## **1.6. Conclusiones**

En este capítulo se abordaron los principales conceptos relacionados con el objeto de estudio, se realizó un breve recorrido por los principales analizadores de registros que existen en el mundo. Luego se profundizó en los conceptos de Interfaz Gráfica de Usuario (GUI) y aplicaciones web. Tras delimitar sus fronteras, características y ventajas se expusieron los lenguajes de programación o técnicas utilizadas, herramientas, metodologías en el diseño y desarrollo del software, mostrando algunas de sus ventajas más relevantes para el desarrollo de la IW en particular, por lo que se eligió PHP5 como lenguaje de programación y Symfony como framework de desarrollo, entre las herramientas se escogió el Visual Paradigm como herramienta de modelado, y Eclipse PDT como IDE. Optamos por RUP como metodología por su características fundamentales de ser iterativo, incremental centrado en la arquitectura y orientado a los casos de uso, además de ser la metodología más extendida en el mundo.

---

## Capítulo 2

# Características del sistema

---

El desarrollo de un software parte de entender la estructura y la dinámica de la organización para la cual el sistema va a ser desarrollado, comprender el problema y asegurar que todas las partes involucradas tengan un entendimiento común de los objetivos. RUP, permite lograr la comprensión del problema en su fase de inicio, a través del levantamiento de requisitos se identifican los principales casos de usos y riesgos, y con la realización del Modelo de Dominio, que forma parte del Modelo de Negocio se logra un mayor entendimiento por todos los implicados. En el caso de la IW de Airesweb, nos basaremos en la descripción del problema, el levantamiento de requisitos llevado a cabo y los conocimientos que posee el equipo de desarrollo, del problema en particular para proponer la estructura de la IW, además se especifican los requisitos funcionales y no funcionales que debe cumplir el sistema, lo cual permite mediante un diagrama de casos de uso analizar con más profundidad el mismo. Por último se hará una descripción de los casos de usos y se presentarán los diagramas asociados a estos.

### 2.1. Descripción del problema

En el capítulo anterior, donde fue analizado el problema planteado se expuso la necesidad de contar con una IW para gestionar la información procesada por el sistema Airesweb, y se arribó a la conclusión de desarrollarla como una aplicación web. El objetivo de la misma es gestionar esta información de forma eficiente y rápida, con lo cual arribamos a la siguiente incógnita. ¿Cuál es la información generada por Airesweb?, para dar respuesta a esta interrogante debemos abordar en el funcionamiento de dicho sistema. Airesweb es un sistema que va a analizar los registros de los servidores web, generando estadísticas a partir

de ellos, el sistema va a leer estos ficheros y va a guardar en una Base de Datos (BD) el sitio al cual se le hace el análisis, el identificador del reporte y los diferentes reportes estadísticos realizados, ejemplos de ellos pueden ser, los visitantes que tuvo el sitio, navegadores por los que arribaron, demografía, etc. Esto queda guardado en la BD de forma que un sitio puede tener varios reportes, y cada reporte puede tener varios reportes estadísticos, que no son más que los reportes desglosados en diferentes tablas, una para cada reporte estadístico. En la BD también se almacena toda la información referente a los usuarios que interactúan con la IW. La figura 2.1 muestra el diagrama *entidad-relación* correspondiente a la BD.

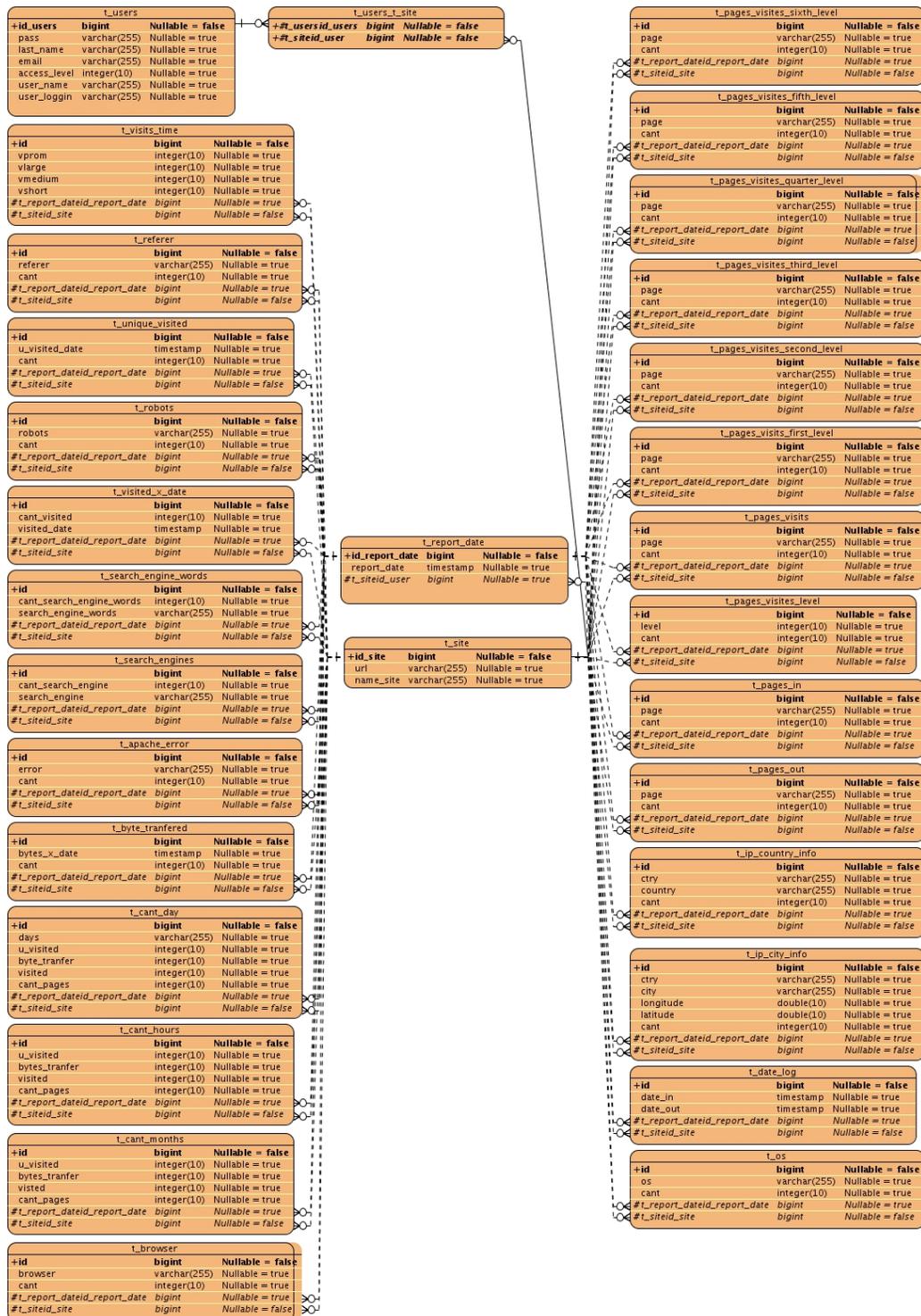


Figura 2.1: Diagrama Entidad-Relación de la BD.

Para que un usuario pueda acceder a la IW, es necesario que sus datos estén almacenados en la tabla *'t\_user'* de la BD. Cuando un usuario después de autenticarse entra al sitio puede escoger el sitio al cual va a monitorizar de la tabla *'t\_site'*, siempre y cuando tenga los permisos suficientes. De ahí pasa a la próxima página donde puede escoger el día que va a revisar o el rango de días a revisar, esto lo selecciona de la tabla *'t\_report\_date'*, cuando tiene el día seleccionado, el sistema busca en la tabla *'t\_date\_log'*, y muestra en todos los reportes estadísticos los datos correspondiente a la fecha escogida.

La interrogante es fácil de responder ahora, los reportes generados por el sistema Airesweb son los descritos anteriormente, están en una BD, los cuales podrían ser consultados desde una herramienta como el PgAdminIII, pero habría que contar con conocimientos avanzados de Structured Query Language (SQL), BD y manejo de esta herramienta, con la IW estos conocimientos pasan a un segundo plano y cualquier tipo de usuario puede consultar los reportes realizados.

## **2.2. Modelo de Dominio**

Debido a la relativa simplicidad del entorno donde está enmarcado el sistema y el conocimiento que se posee acerca de su funcionamiento, no es necesario realizar un Modelo de Negocio completo para comprender la problemática que a de resolverse, siendo suficiente el Modelo de Dominio. Este último es una representación visual de los conceptos u objetos del mundo real significativos para un problema o área de interés. Representa clases conceptuales del dominio del problema, conceptos del mundo real, no de componentes de software[36]. La figura 2.2 muestra el Modelo de Dominio realizado teniendo en cuenta lo planteado en la descripción del problema.

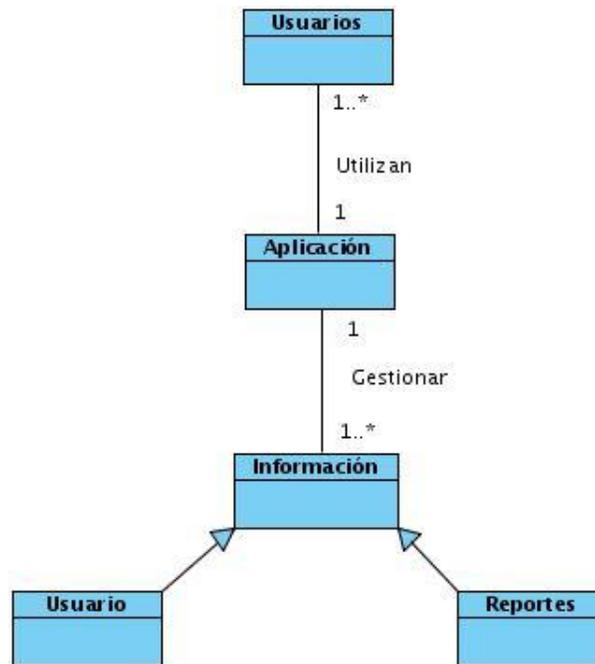


Figura 2.2: Modelo de Dominio del problema.

### 2.3. Solución propuesta

La propuesta para dar solución a la problemática planteada anteriormente consiste en estructurar la aplicación web de modo que cuente con dos módulos, uno ellos relacionado con las tareas de la gestión de usuario, y el otro módulo agrupará las funcionalidades que permiten obtener los reportes a partir de la información almacenada en la BD.

El sistema debe contar con tres roles, un Usuario con los privilegios de ver los reportes y de gestionar su información personal, un rol de Administrador de Sitio con los privilegios de gestionar su información personal y de los demás usuarios que administra, además de gestionar los reportes de los sitios que administra, y por último, el rol de Administrador General, con la autoridad para gestionar la información personal de todos los usuarios y administradores de la aplicación, y ver los reportes de todos los sitios monitorizados por la aplicación.

Los requisitos funcionales y no funcionales que se presentan a continuación brindarán más detalles de la

solución propuesta.

## **2.4. Requisitos funcionales**

- R1.** El sistema debe permitir iniciar sesión de usuario.
- R2.** El sistema debe permitir finalizar la sesión de usuario.
- R3.** El sistema debe permitir que el usuario edite su información personal.
- R4.** El sistema debe permitir que el usuario cambie su contraseña personal.
- R5.** El sistema debe permitir la gestión de los usuarios.
- R6.** El sistema debe permitir cambiar la contraseña de los usuarios.
- R7.** El sistema debe permitir la gestión de los administradores.
- R8.** El sistema debe permitir cambiar la contraseña del administrador.
- R9.** El sistema debe permitir listar los sitios monitorizados por la aplicación.
- R10.** El sistema debe permitir seleccionar sitio a monitorizar por la aplicación.
- R11.** El sistema debe permitir seleccionar la fecha a monitorizar.
- R12.** El sistema debe brindar reportes sobre las páginas vistas.
- R13.** El sistema debe brindar reportes sobre las visitas.
- R14.** El sistema debe brindar reportes sobre los visitantes únicos.
- R15.** El sistema debe brindar reportes sobre los visitantes por países.
- R16.** El sistema debe brindar reportes sobre los principales buscadores por donde acceden los visitantes.
- R17.** El sistema debe brindar reportes sobre las palabras claves por las cuales acceden los visitantes.
- R18.** El sistema debe brindar reportes sobre los visitantes distribuidos mensualmente.

- R19.** El sistema debe brindar reportes sobre los visitantes distribuidos diariamente.
- R20.** El sistema debe brindar reportes sobre los visitantes distribuido por horas.
- R21.** El sistema debe brindar reportes sobre los principales navegadores y sistemas Operativos utilizados.
- R22.** El sistema debe brindar reportes sobre los bytes transferidos.
- R23.** El sistema debe brindar reportes sobre las páginas desde las que accedieron al sitio(Referencia).
- R24.** El sistema debe brindar reportes sobre los robots que accedieron al sitio.
- R25.** El sistema debe brindar reportes sobre los errores de Apache.

## **2.5. Requisitos no funcionales**

### **2.5.1. Usabilidad**

1. Debe poseer una funcionabilidad adecuada, o sea, satisfacer los requisitos funcionales declarados e implícitos. Una subcaracterística importante de este requisito es la seguridad.
2. Debe contar con una confiabilidad adecuada, de modo que mantenga el mismo nivel de ejecución a lo largo del tiempo.
3. Debe tener buena usabilidad, de modo que el esfuerzo para usarlo sea mínimo, además debe ser atractivo, útil e intuitivo para los usuarios.
4. La eficiencia debe ser alta, contando con tiempos de respuesta no mayores de 8 segundos. Además el consumo de recursos debe ser mínimo.
5. La mantenibilidad debe ser alta, de modo que pueda adaptarse a condiciones cambiantes del entorno en que se ejecute y las modificaciones puedan hacerse fácilmente.
6. Debe contar con la portabilidad necesaria para poder ser transferido de un ambiente a otro, reemplazado por nuevas versiones y además fácil de instalar y adaptar.

### **2.5.2. Precisión**

Para las estadísticas el sistema deberá tener una efectividad por encima del 90 % en cuanto a registrar los accesos correctamente.

### **2.5.3. Rapidez**

Parte de lo relacionado con la rapidez de respuesta dependerá de las características del hardware donde se instale el sistema, pero también dependerá de optimizar el código lo más posible para su desarrollo.

### **2.5.4. Flexibilidad**

- La IW debe poder gestionar la información de varios sitios.
- El sistema completo debe ser diseñado para ser instalado en arquitecturas i386 y compatibles, bajo cualquier configuración de hardware soportada por Debian GNU/Linux 5.0 (lenny).

### **2.5.5. Seguridad**

- Contar con mecanismos de manejo de sesiones y roles.
- Posibilitar el acceso a la Interfaz de Administración Web mediante el uso de protocolos seguros (HTTPS).

### **2.5.6. Software**

- El sistema se desarrollará con PHP5(5.2.6).
- Para el desarrollo de la IW se utilizará el Framework Symfony 1.0.
- El sistema debe estar soportado sobre el Sistema Operativo Debian GNU/Linux 5.0 (Lenny).
- El sistema de utilizar una BD gestionada con cualquier Sistema Gestor de Base de Datos (SGBD) compatible con ODBC.

### 2.5.7. Ayuda y documentación

- El sistema debe ir acompañado de un Manual de Usuario para facilitar su trabajo de uso.

## 2.6. Definición de los casos de uso

Los Casos de Uso son la base para la implementación de las fases y disciplinas del RUP. Un Caso de Uso es una secuencia de pasos a seguir para la realización de un fin o propósito, y se relaciona directamente con los requerimientos, ya que un Caso de Uso es la secuencia de pasos que conlleva la realización e implementación de un requerimiento planteado.

### 2.6.1. Definición de los actores

Actores	Descripción
Usuario	Persona que tendrá acceso al sistema como un usuario básico.
Administrador de sitio	Encargado de realizar cualquier modificación en los sitios que administra.
Administrador General	Encargado de realizar cualquier modificación en el sistema, o sea, tendrá control total sobre el mismo.

Tabla 2.1: Definición de los actores

### 2.6.2. Listado de casos de uso

#### Módulo Usuario

Caso de Uso:	Iniciar sesión de usuario
Actores:	Administrador de sitio, Administrador General, Usuario.
Resumen:	El caso de uso se inicia cuando el actor introduce los datos para acceder a la aplicación, estos se verifican y finalizan dándole los permisos que tiene y habilitándole o denegándole la entrada al sistema.
Referencia:	R1

Tabla 2.2: CU Iniciar sesión de usuario

Caso de Uso:	Finalizar sesión de usuario
Actores:	Administrador de sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide salir del sistema.
Referencia:	R2

Tabla 2.3: CU Finalizar sesión de usuario

Caso de Uso:	Editar Información personal.
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide editar su información personal en el sitio.
Referencia:	R3

Tabla 2.4: CU Editar Información personal

Caso de Uso:	Cambiar contraseña.
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide cambiar su contraseña.
Referencia:	R4

Tabla 2.5: CU Cambiar contraseña.

Caso de Uso:	Gestionar usuario.
Actores:	Administrador de Sitio, Administrador General.
Resumen:	El Caso de Uso se inicia cuando el actor decide eliminar, adicionar o modificar la información personal de algún usuario que administra.
Referencia:	R5

Tabla 2.6: CU Gestionar usuario

Caso de Uso:	Cambiar contraseña del usuario.
Actores:	Administrador de Sitio, Administrador General.
Resumen:	El Caso de Uso se inicia cuando el administrador decide cambiar la contraseña del usuario.
Referencia:	R6

Tabla 2.7: CU Cambiar contraseña del usuario

Caso de Uso:	Gestionar administrador.
Actores:	Administrador General
Resumen:	El Caso de Uso se inicia cuando el actor decide eliminar, adicionar o modificar la información personal de algún Administrador del sistema.
Referencia:	R7

Tabla 2.8: CU Gestionar administrador

Caso de Uso:	Cambiar contraseña del administrador.
Actores:	Administrador
Resumen:	El Caso de Uso se inicia cuando el actor decide cambiar la contraseña de algún administrador.
Referencia:	R8

Tabla 2.9: CU Cambiar contraseña del administrador

Caso de Uso:	Listar sitios monitorizados
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide mostrar los sitios monitorizados.
Referencia:	R9

Tabla 2.10: CU Listar sitios monitorizados

Caso de Uso:	Seleccionar sitio a monitorizar.
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide seleccionar un sitio para ver sus reportes.
Referencia:	R10

Tabla 2.11: CU Seleccionar sitio a monitorizar

Caso de Uso:	Seleccionar fecha a monitorizar
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide seleccionar una fecha o un rango de fechas para ver los reportes correspondientes a un sitio monitorizado.
Referencia:	R11

Tabla 2.12: CU Seleccionar fecha a monitorizar

## Módulo Reporte

Caso de Uso:	Ver reportes
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide ver los reportes que genera el sistema.
Referencia:	R12,R13,R14,R15,R16,R17,R18,R19,R20,R21,R22,R23,R24,R25
Continúa en la próxima página	

Tabla 2.13: CU Ver reportes

### 2.6.3. Diagramas de casos de uso

El diagrama de Casos de Uso para el sistema se muestra en la figura 2.3.

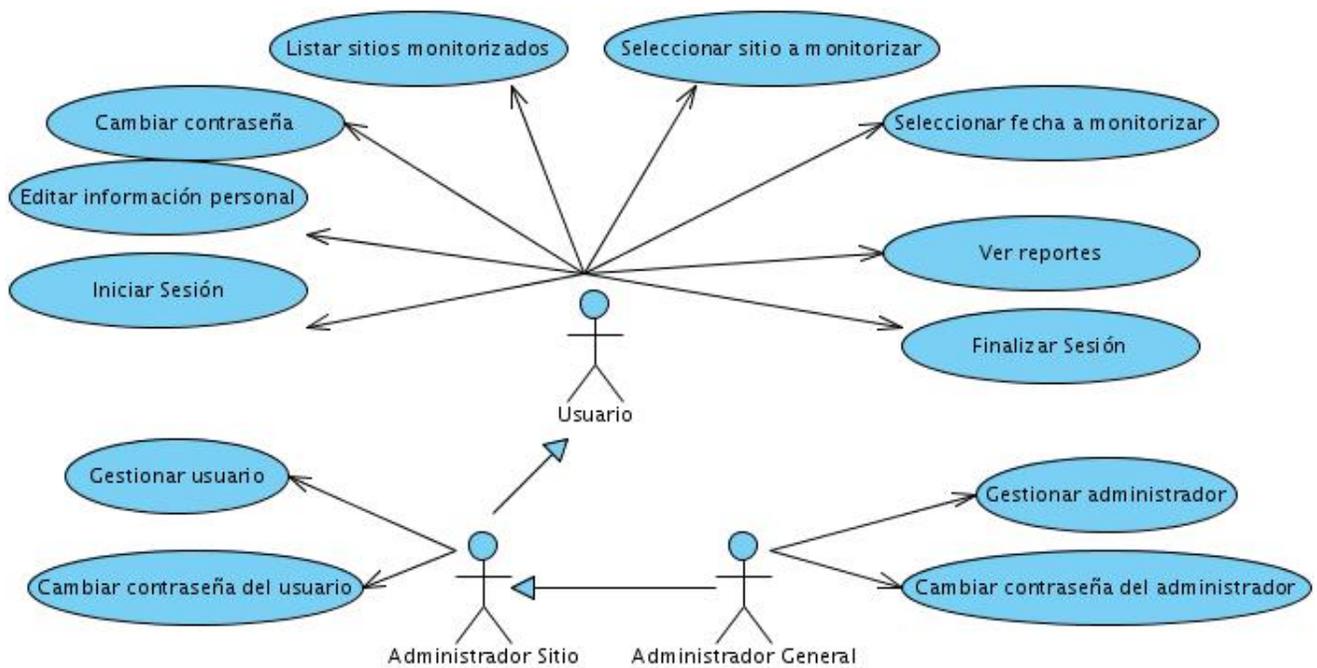


Figura 2.3: Diagrama de Casos de Uso.

### 2.6.4. Casos de uso expandidos

#### Módulo Usuario

Caso de Uso:	Iniciar sesión
Continúa en la próxima página	

Actores:	Administrador de sitio, Administrador General, Usuario.
Resumen:	El caso de uso se inicia cuando el actor introduce los datos para acceder a la aplicación, estos se verifican y finalizan dándole los permisos que tiene y habilitándole o denegándole la entrada al sistema.
Referencia:	R1
CU asociados:	
Precondiciones:	Que al actor previamente se le haya creado una cuenta en el sistema.
<b>Flujo Normal de Eventos</b>	
<b>Sección “nombre del escenario”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor entra su identificador y contraseña.	1.1 El sistema chequea si existe el identificador en la base de datos. 1.2 El sistema encripta la contraseña. 1.3 El sistema verifica los permisos del usuario. 1.4 El sistema crea la credencial (token). 1.5 El sistema asigna los niveles de acceso al usuario. 1.6 El sistema carga el módulo en dependencia de los niveles de acceso.
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

	1.1a Si no existe el identificador del actor, volver al paso 1.
Poscondiciones:	La sesión finalmente es iniciada.
Prioridad:	Crítico
Especificaciones:	
Complementaria:	

Caso de Uso:	Finalizar sesión.
Actores:	Administrador de sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide salir del sistema.
Referencia:	R2
CU asociados:	
Precondiciones:	Actor autenticado en el sistema.
<b>Flujo Normal de Eventos</b>	
<b>Sección “nombre del escenario”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor solicita la opción de terminar sesión. 2. El actor confirma la solicitud.	1.1 El sistema pide confirmación por la opción solicitada. 2.1 El sistema destruye la sesión del actor. 2.2 El sistema retorna a la página principal.
Continúa en la próxima página	

<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Poscondiciones:	La sesión finalmente es terminada y se sale del sistema.
Prioridad:	Crítico
Especificaciones:	
Complementaria:	

Caso de Uso:	Editar Información personal.
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide editar su información personal en el sitio.
Referencia:	R3
CU asociados:	
Precondiciones:	Actor autenticado en el sistema.
<b>Flujo Normal de Eventos</b>	
<b>Sección “nombre del escenario”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor solicita la opción de editar información personal.</p> <p>2. 1.El actor introduce los datos a ser modificados.</p>	<p>1.1 El sistema muestra un formulario con los datos actuales del perfil y da la posibilidad de modificar los datos mostrados.</p> <p>2.1 El sistema valida los datos introducidos.</p> <p>2.3 El sistema inserta en la base de datos la nueva información del perfil del actor.</p>
<b>Flujo alternativo</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>2.1a En caso de que la la validación de los datos sea incorrecta, se muestra un mensaje de advertencia y regresa al paso 1.1.</p>
Poscondiciones:	La información finalmente es modificada en la Base de Datos del sistema.
Prioridad:	Crítico
Especificaciones:	
Complementaria:	

Caso de Uso:	Cambiar contraseña.
Actores:	Administrador de sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide cambiar su contraseña.
Referencia:	R4
Continúa en la próxima página	

CU asociados:	
Precondiciones:	Actor autenticado en el sistema.
<b>Flujo Normal de Eventos</b>	
<b>Sección “nombre del escenario”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. El actor solicita cambiar la contraseña.</p> <p>1. El actor inserta y confirma su nueva contraseña.</p>	<p>1.1 El sistema le muestra un formulario para que inserte y confirme la nueva contraseña.</p> <p>2.1 El sistema valida la información.</p> <p>2.2 El sistema actualiza la contraseña del actor en la Base de Datos.</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>2.1a El sistema envía un mensaje en caso de tener problema la contraseña entrada por el actor y retorna al paso 2.</p>
Poscondiciones:	La contraseña del actor finalmente es modificada y guardada en la Base de Datos del sistema.
Prioridad:	Crítico
Especificaciones:	
Complementaria:	

**Módulo Reporte**

Caso de Uso:	Ver reportes.
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide ver los reportes que genera el sistema.
Referencia:	R12,R13,R14,R15,R16,R17,R18,R19,R20,R21,R22,R23,R24,R25
CU asociados:	
Precondiciones:	Actor autenticado en el sistema. Que el actor haya seleccionado un sitio para monitorizar. Que el actor haya seleccionado una fecha o un rango de fechas para ver los reportes.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Principal”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor necesita algunos de los reportes:</p> <ul style="list-style-type: none"> <li>- Páginas vistas.</li> <li>- Visitas.</li> <li>- Visitantes únicos.</li> <li>- Visitantes por países.</li> <li>- Principales buscadores por donde acceden los visitantes.</li> <li>- Palabras claves por las cuales acceden los visitantes.</li> <li>- Visitantes por mes.</li> <li>- Visitantes por días.</li> </ul>	<p>1. El sistema ejecuta alguna de las siguientes acciones:</p> <ul style="list-style-type: none"> <li>1.1 Si decide ver las páginas vistas, ir a la sección “Mostrar páginas vistas”.</li> <li>1.2 Si decide ver las visitas, ir a la sección “Mostrar las visitas”.</li> <li>1.3 Si decide ver los visitantes únicos, ir a la sección “Mostrar visitantes únicos”.</li> <li>1.4 Si decide ver los visitantes por países, ir a la sección “Mostrar visitantes por países”.</li> <li>1.5 Si decide ver los principales buscadores por donde acceden los visitantes, ir a la sección “Mostrar principales buscadores por donde acceden los visitantes”.</li> <li>1.6 Si decide ver las palabras claves por las cuales acceden los visitantes, ir a la sección “Mostrar palabras claves por las cuales acceden los visitantes”.</li> <li>1.7 Si decide ver los visitantes por mes, ir a la sección “Mostrar visitantes por mes”.</li> </ul>
<p>Continúa en la próxima página</p>	

<ul style="list-style-type: none"> <li>- Visitantes por horas.</li> <li>- Principales navegadores y sistemas Operativos utilizados.</li> <li>- Bytes transferidos.</li> <li>- Páginas desde las que accedieron al sitio(Referencia).</li> <li>- Robots que accedieron al sitio.</li> <li>- Errores de Apache.</li> </ul>	<p>1.8 Si decide ver los visitantes por días, ir a la sección “Mostrar visitantes por días”.</p> <p>1.9 Si decide ver los visitantes por horas, ir a la sección “Mostrar visitantes por horas”.</p> <p>1.10 Si decide ver los principales navegadores y Sistemas Operativos utilizados, ir a la sección “Mostrar principales navegadores y Sistemas Operativos utilizados”.</p> <p>1.11 Si decide ver los bytes transferidos, ir a la sección “Mostrar bytes transferidos”.</p> <p>1.12 Si decide ver las páginas desde las que accedieron al sitio(Referencia), ir a la sección “Mostrar páginas desde las que accedieron al sitio”.</p> <p>1.13 Si decide ver los robots que accedieron al sitio, ir a la sección “Mostrar robots que accedieron al sitio”.</p> <p>1.14 Si decide ver los errores de Apache, ir a la sección “Mostrar errores de Apache”.</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar páginas vistas”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor solicita la opción de ver las páginas vistas	<p>1.1 El sistema realiza la búsqueda en la Base de Datos.</p> <p>1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar las visitas”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

1. El actor solicita la opción de ver las visitas.	<p>1.1 El sistema realiza la búsqueda en la Base de Datos.</p> <p>1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar visitantes únicos”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor solicita la opción de ver los visitantes únicos.	<p>1.1 El sistema realiza la búsqueda en la Base de Datos.</p> <p>1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar visitantes por países”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor solicita la opción de ver los visitantes por países.	<p>1.1 El sistema realiza la búsqueda en la Base de Datos.</p> <p>1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar principales buscadores por donde acceden los visitantes”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor solicita la opción de ver los principales buscadores por donde acceden los visitantes al sitio monitorizado.</p>	<p>1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.</p>
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar palabras claves por las cuales acceden los visitantes”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. El actor solicita la opción de ver las palabras claves por las cuales acceden los visitantes al sitio monitorizado.</p>	<p>1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar visitantes por mes”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor solicita la opción de ver los visitantes distribuidos por mes.	<p>1.1 El sistema realiza la búsqueda en la Base de Datos.</p> <p>1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar visitantes por días”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor solicita la opción de ver los visitantes distribuidos por días.</p>	<p>1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.</p>
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar visitantes por horas”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. El actor solicita la opción de ver los visitantes distribuidos por horas.</p>	<p>1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>Continúa en la próxima página</p>	

	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar Principales navegadores y Sistemas Operativos utilizados”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor solicita la opción de ver los principales navegadores y Sistemas Operativos utilizados.	<p>1.1 El sistema realiza la búsqueda en la Base de Datos.</p> <p>1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar bytes transferidos”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor solicita la opción de ver los bytes transferidos desde el sitio monitorizado.</p>	<p>1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.</p>
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar páginas desde las que accedieron al sitio”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. El actor solicita la opción de ver las páginas desde las que accedieron al sitio monitorizado.</p>	<p>1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos .</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar robots que accedieron al sitio”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor solicita la opción de ver los robots que accedieron al sitio.	1.1 El sistema realiza la búsqueda en la Base de Datos.  1.2 El sistema muestra los resultados obtenidos .
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Mostrar errores de Apache”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor solicita la opción de ver los errores de Apache.</p>	<p>1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos .</p>
<p><b>Flujo alternativo</b></p>	
<p><b>Acción del Actor</b></p>	<p><b>Respuesta del Sistema</b></p>
	<p>1.1.a Se emite un mensaje informado en caso de que no existan datos relacionados al reporte seleccionado.</p>
<p>Poscondiciones:</p>	
<p>Prioridad:</p>	<p>Crítico</p>
<p>Especificaciones:</p>	
<p>Complementaria:</p>	

La descripción de los restantes Casos de Uso se encuentran en el anexo C.

## 2.7. Conclusiones

A partir de la descripción del problema y del desarrollo del Modelo de Dominio se hizo posible elaborar una propuesta de solución al problema e identificar los requisitos funcionales y los no funcionales. Además puede concluirse que la descripción de los Casos de Uso elaborada será un aspecto fundamental y de gran ayuda para las fases posteriores. Hasta el momento se le ha dado cumplimiento parcialmente al objetivo trazado que se refiere a analizar y diseñar las funcionalidades de la IW del sistema Airesweb.

---

## Capítulo 3

# Análisis y diseño del sistema

---

En este capítulo se describe de manera general la solución que se presenta. Los elementos que se tendrán en cuenta para el análisis y diseño de la aplicación y los elementos necesarios para su comprensión serán explicados; se hablará sobre el framework empleado en la realización de los prototipos, los patrones más significativos que este utiliza, y los que se utilizan en el diseño. Se hace una breve descripción de la arquitectura y se presentan los diagramas de clases de diseño, de secuencia del diseño, de los casos de usos más importantes, así como el modelo de datos.

## 3.1. Análisis

### 3.1.1. Modelo de Clases del Análisis

El análisis consiste en obtener una visión del sistema que se preocupa de ver qué hace, de modo que solo se interesa por los requisitos funcionales.

#### Módulo Usuario



Figura 3.1: DCA Iniciar sesión

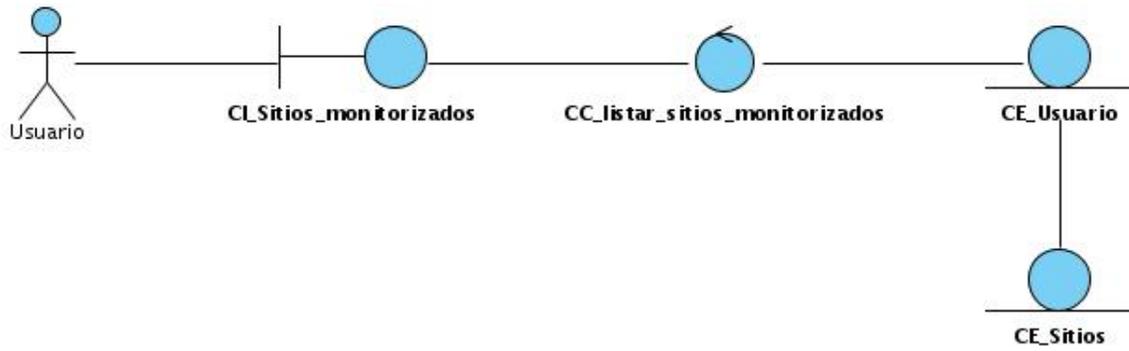


Figura 3.2: DCA Listar sitios monitorizados



Figura 3.3: DCA Seleccionar fecha a monitorizar

Módulo Reporte

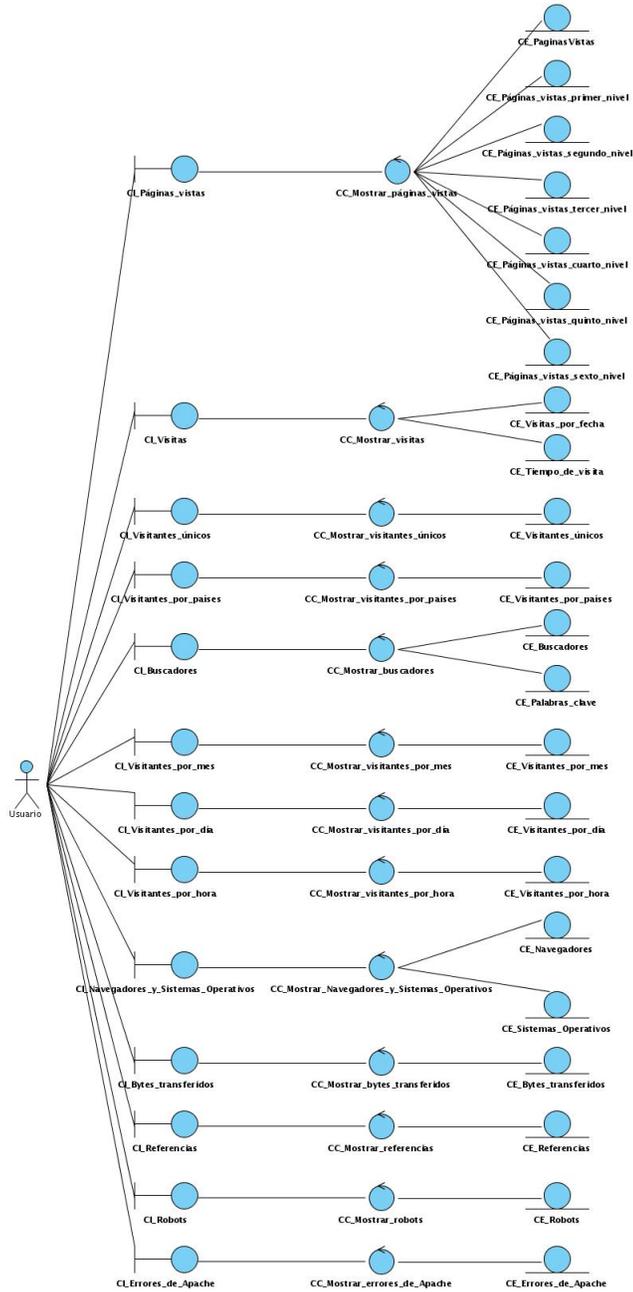


Figura 3.4: DCA Ver reportes

Los restantes diagramas de clases del análisis se encuentran en el anexo B.

## **3.2. Diseño**

El Modelo de Diseño, es un modelo de objetos que describe la realización física de los Casos de Uso, centrándose en como los requisitos funcionales y no funcionales tienen impacto en el sistema a desarrollar. El diseño es el centro de atención al final de la fase de elaboración y el comienzo de las iteraciones de construcción.

### **3.2.1. Symfony**

Anteriormente se exponía que Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja[28].

#### **Implementación del MVC por Symfony**

Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC, que está formado por tres niveles[28]:

- El modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La vista transforma el modelo en una página web que permite al usuario interactuar con ella.
- El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

En la figura 3.5 se muestra el funcionamiento del patrón MVC.

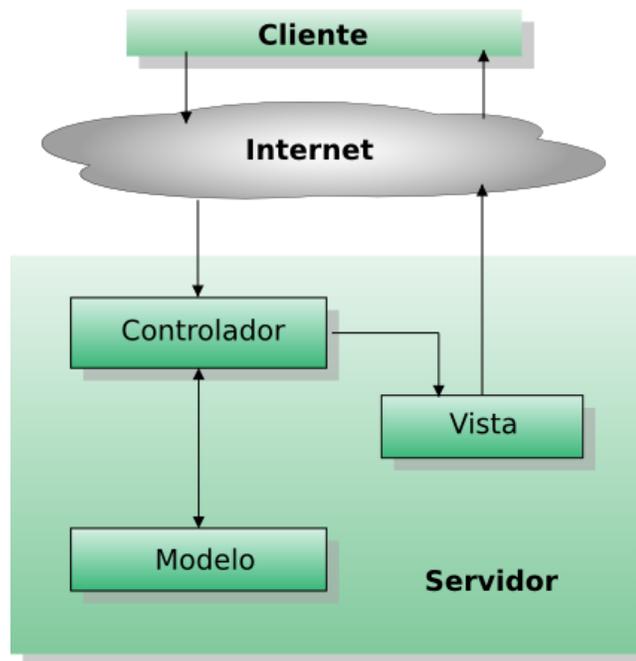


Figura 3.5: Esquema del MVC

La arquitectura del sistema estará compuesta en capas. Los niveles estarán compuestos de la siguiente forma de acuerdo con la estructura del framework y como se muestra en la 3.6.

- **La capa de la Vista:** Aprovecha la separación de código porque se descompone en:
  - Layout: El Layout es global en toda la aplicación.
  - Lógica de la vista: se puede transformar en un archivo de configuración sencillo, sin necesidad de programarla.
  - Plantilla: Solo se encarga de visualizar las variables definidas en el controlador.
- **La capa del Controlador:** Se encarga del manejo de las peticiones del usuario, de la seguridad, carga la configuración de la aplicación.
  - Controlador frontal: Este es único por cada aplicación y es el único punto de entrada a la misma.
  - Acciones: Incluyen el código específico del controlador en cada página.

- **La capa del Modelo:** Esta capa se genera automáticamente, en función de la estructura de datos de la aplicación.
  - Acceso a los datos.
  - Abstracción de la base de datos.

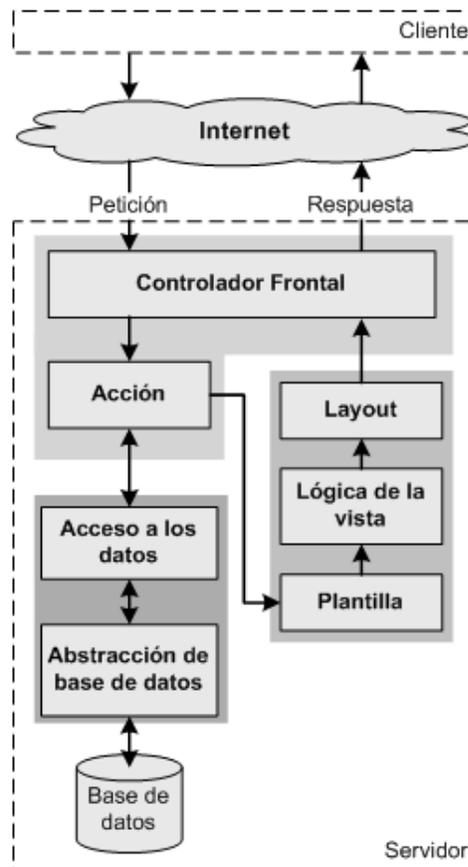


Figura 3.6: Flujo de trabajo del MVC en Symfony

### Organización de la aplicación

Symfony organiza el código fuente en una estructura de tipo proyecto y almacena los archivos del proyecto en una estructura estandarizada de tipo árbol.

En Symfony un proyecto se considera como “*un conjunto de servicios y operaciones disponibles bajo un determinado nombre de dominio y que comparten el mismo modelo de objetos*”[28]. Dentro de un proyecto, las operaciones se agrupan de forma lógica en aplicaciones y cada aplicación está formada por uno o más módulos. Un módulo normalmente representa a una página Web o un grupo de páginas con un propósito relacionado y almacenan las acciones, que representan cada una de las operaciones que se pueden realizar en un módulo. La IW seguirá una estructura aproximada a la que se muestra en la figura3.7.

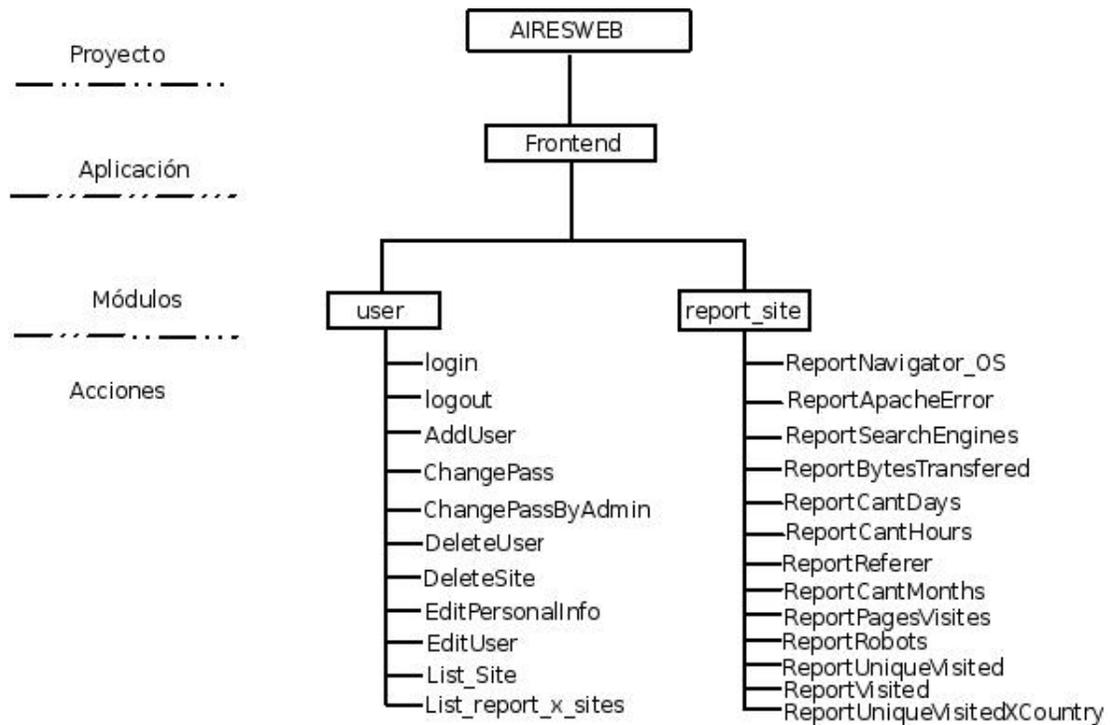


Figura 3.7: Organización de la aplicación

### 3.2.2. Patrones de diseño usados

Symfony sigue las mejores prácticas y patrones de diseño para el desarrollo de aplicaciones Web. Seguidamente se mencionan cuales son lo patrones de diseño[37] más significativos que se utilizan.

### **Active record**

Este patrón propone una tabla para cada clase entidad y clases intermedias que se encargan de llevar de objetos a tablas y de tablas a objetos, en este caso el framework abstrae al usuario trabajo con base de datos, esto se logra mediante las librerías de Propel y Creole, Creole como abstracción de base de datos y Propel implementando el patrón active Record. Propel es un framework que el Symfony agregó a su arquitectura para el trabajo con base de datos, el mismo contiene la clase Creole.

### **Singleton:**

Este patrón propone una sola instancia de la clase controladora para acceder a los datos y modificarlos, al igual que el framework usa una sola instancia para manejar las sesiones, los Layouts y cualquier otra información que se requiera.

### **Decorador:**

Permite añadir funcionalidades dinámicamente a una clase con la creación de otra clase sin necesidad de la herencia. En el Symfony, los layout decoran a las vistas, pues en un momento determinado les confieren propiedades dinámicas que no poseían.

### **Experto:**

Consiste en asignarle una responsabilidad al experto en información, cuando surge el problema de a quién asignar cierta responsabilidad si se aplica este patrón la solución será, al que conoce los datos necesarios, al experto en la información.

### **Chain of Responsibility:**

La ejecución de la secuencia de filtros se basa en este patrón el cual permite que más de un objeto pueda manejar una petición.

## **Validación y tratamiento de errores**

La validación y el tratamiento de errores en la aplicación se han diseñado en torno a los mecanismos que Symfony provee.

Posee un mecanismo de validación en el servidor a través del uso de ficheros de validación y una serie de validadores que permite hacer este proceso más fácil y extensible[28], sin la consecuente promoción de errores que puede traer consigo la programación relacionada con este aspecto.

Para el tratamiento de excepciones se utilizará el mecanismo provisto por PHP y las clases que para este fin Symfony posee.

## **Seguridad**

### **Autenticación y autorización**

Cuando se diseña una aplicación Web es importante definir los aspectos relacionados con la seguridad de la misma. En este contexto es importante prestar atención a la autenticación y gestión de credenciales. Symfony simplifica esta gestión, mejorando el mecanismo de sesiones que utiliza PHP y haciéndolo más configurable y fácil de usar. Una vez que un usuario se autentica, se le asignan los privilegios necesarios (*credenciales*) para que pueda utilizar las funcionalidades que brinda la aplicación según el rol que desempeñe[28].

A partir de los requisitos funcionales y no funcionales, y tal como se había planteado, la IW poseerá tres tipos de usuarios diferentes, un administrador general, un administrador de sitio y un usuario básico. De este modo solo será necesario definir la seguridad y la *credencial* o *credenciales* que se requieran para la ejecución de cada una de las acciones, pues la responsabilidad de chequear que un usuario esté correctamente autenticado y posea los privilegios necesarios será de Symfony.

### 3.2.3. Diagramas de Clases del Diseño

#### Módulo Usuario

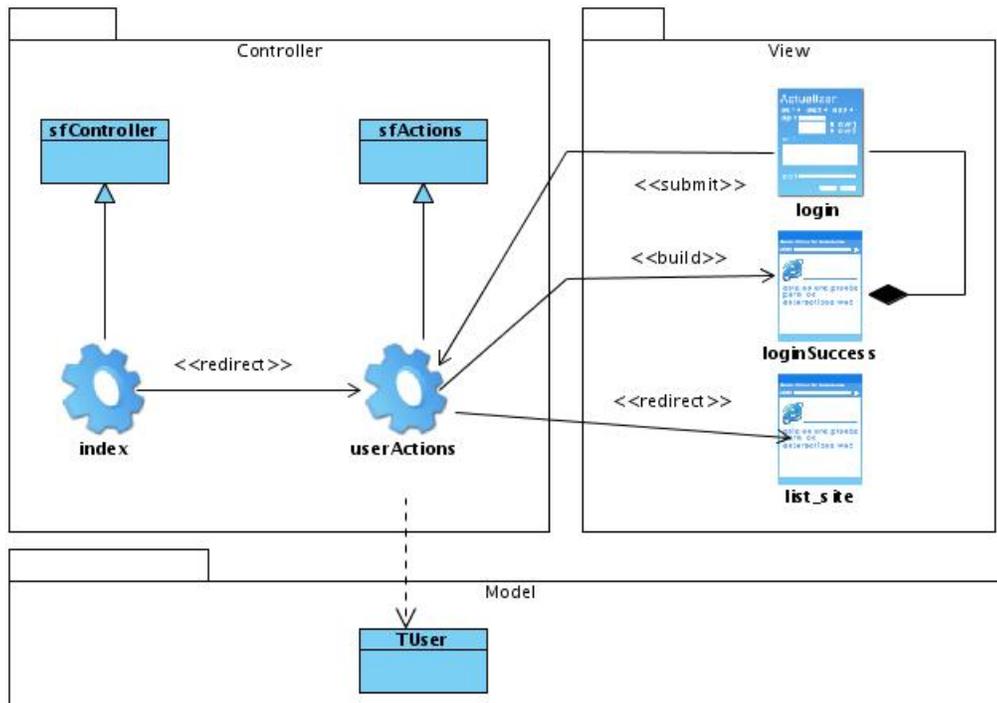


Figura 3.8: DCD Iniciar sesión

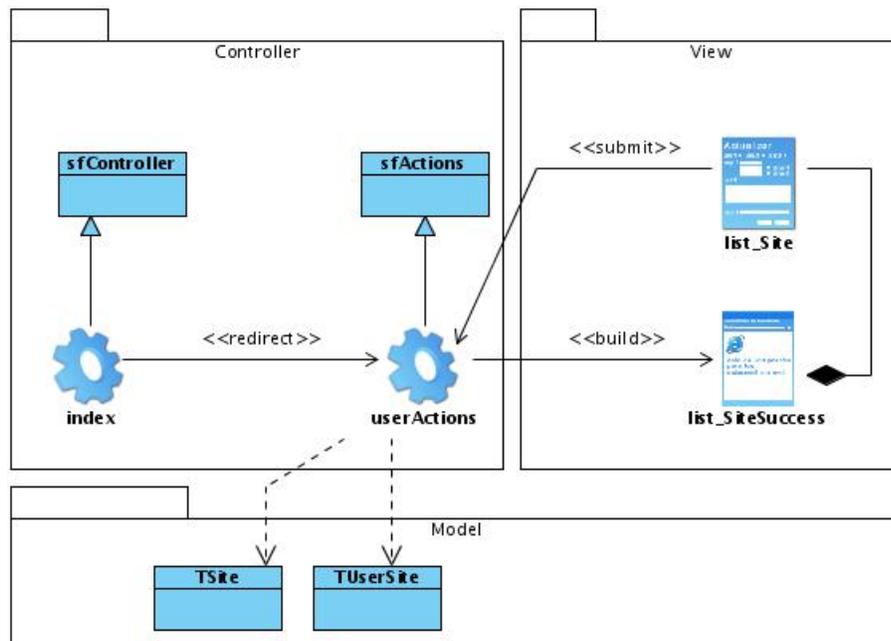


Figura 3.9: DCD Listar sitios monitorizados

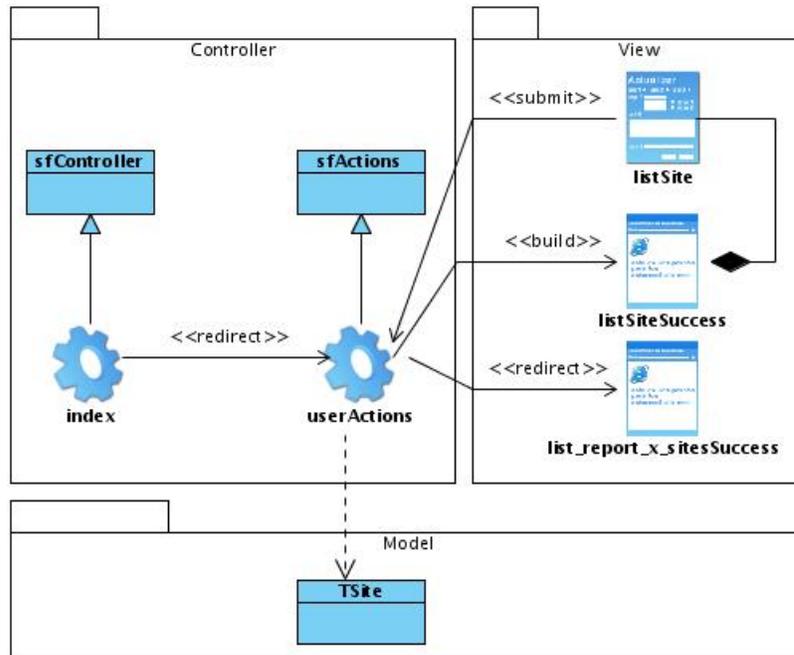


Figura 3.10: DCD Seleccionar sitio a monitorizar

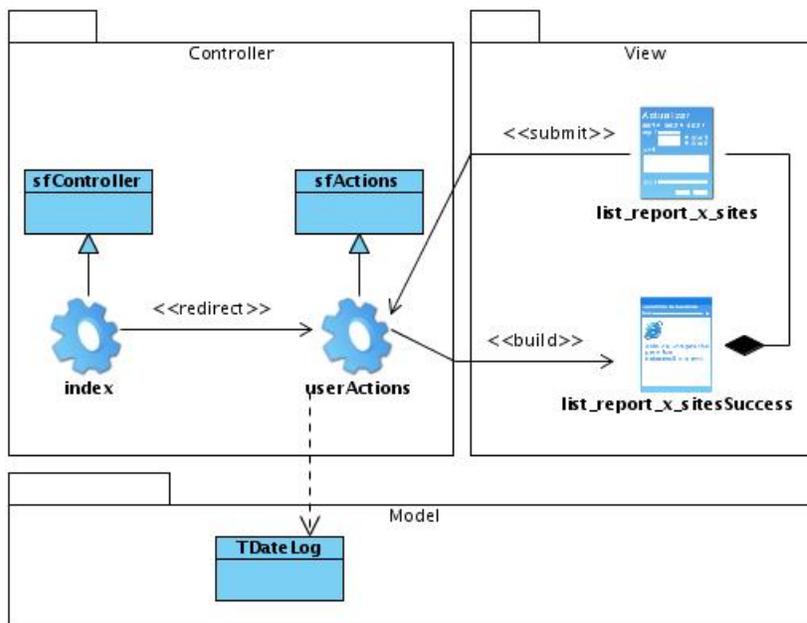


Figura 3.11: DCD Seleccionar fecha a monitorizar

Módulo Reporte

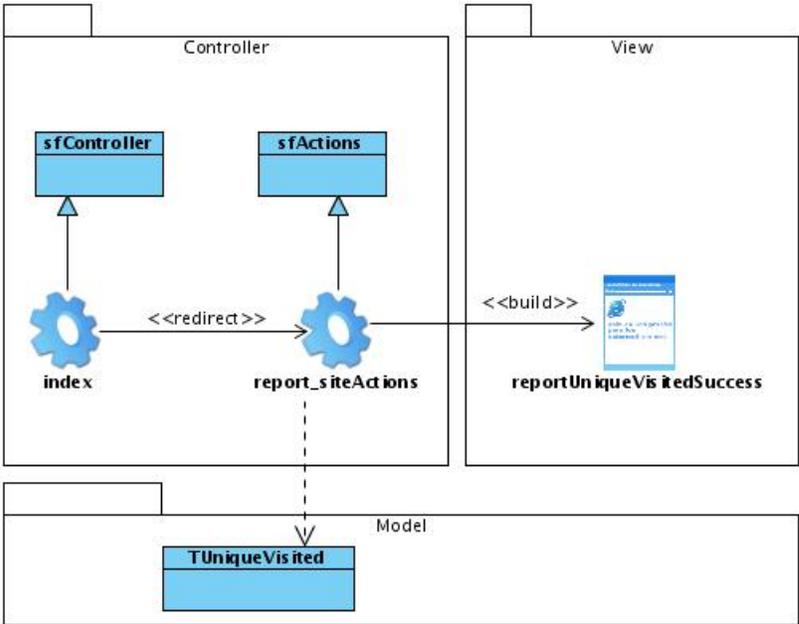


Figura 3.12: DCD Mostrar reportes sección mostrar visitantes únicos

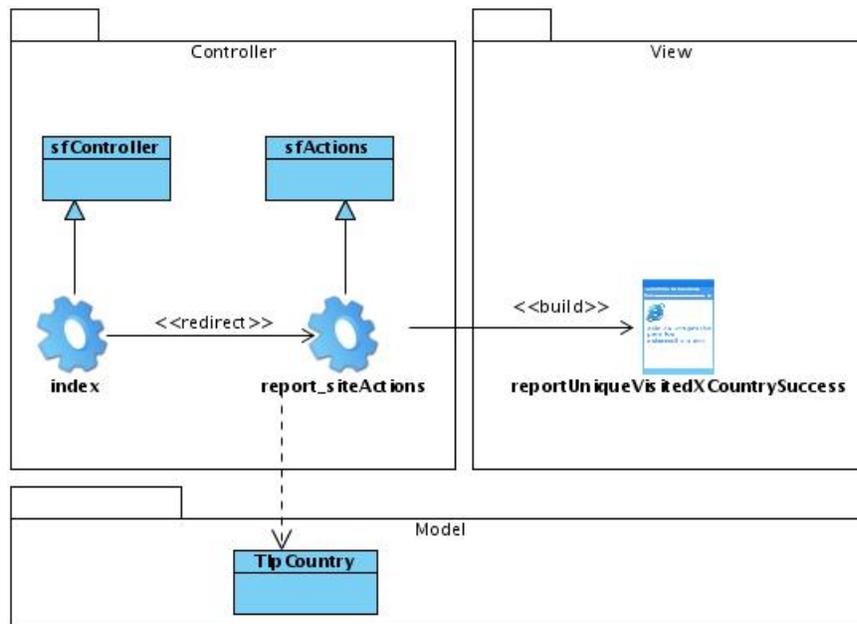


Figura 3.13: DCD Mostrar reportes sección mostrar visitantes por países

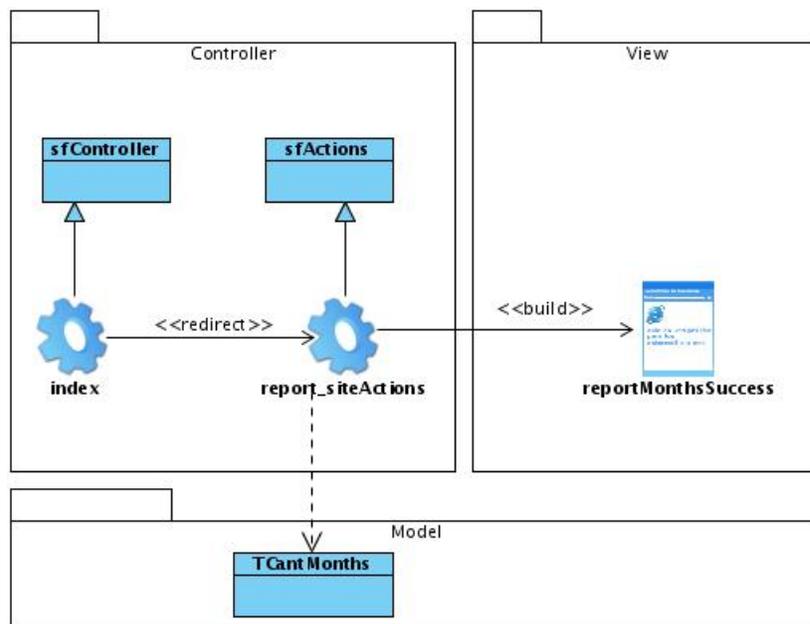


Figura 3.14: DCD Mostrar reportes sección mostrar visitantes por mes

Los restantes diagramas de Clases del Diseño se encuentran en el anexo D.

### 3.2.4. Diagramas de Interacción

#### Módulo Usuario

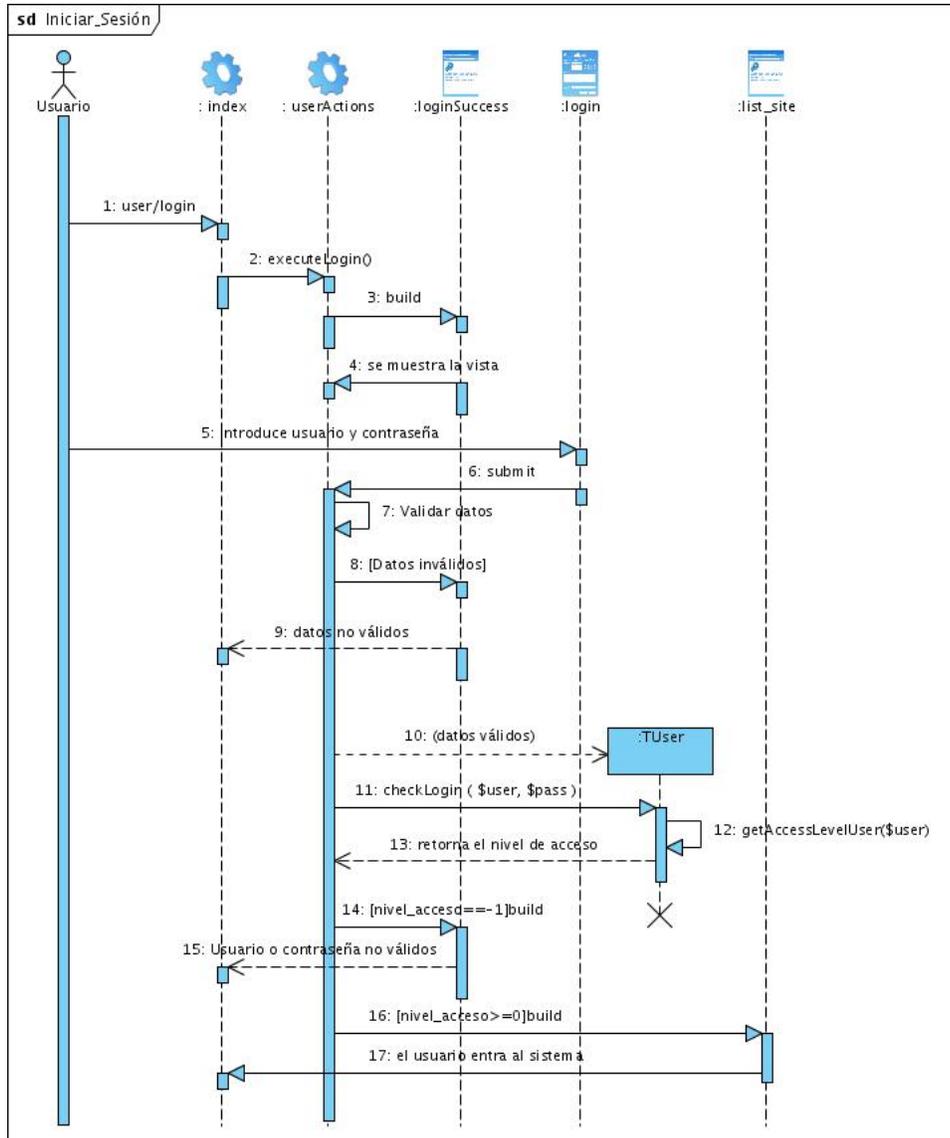


Figura 3.15: Iniciar sesión

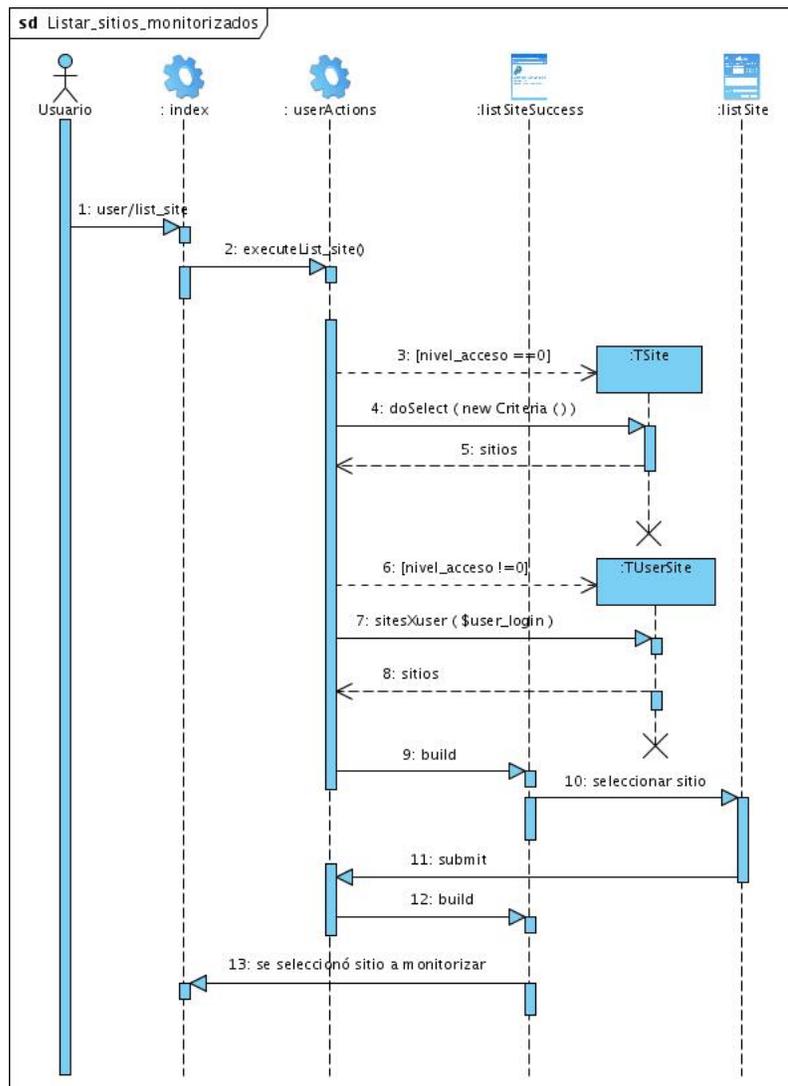


Figura 3.16: Listar sitios monitorizados

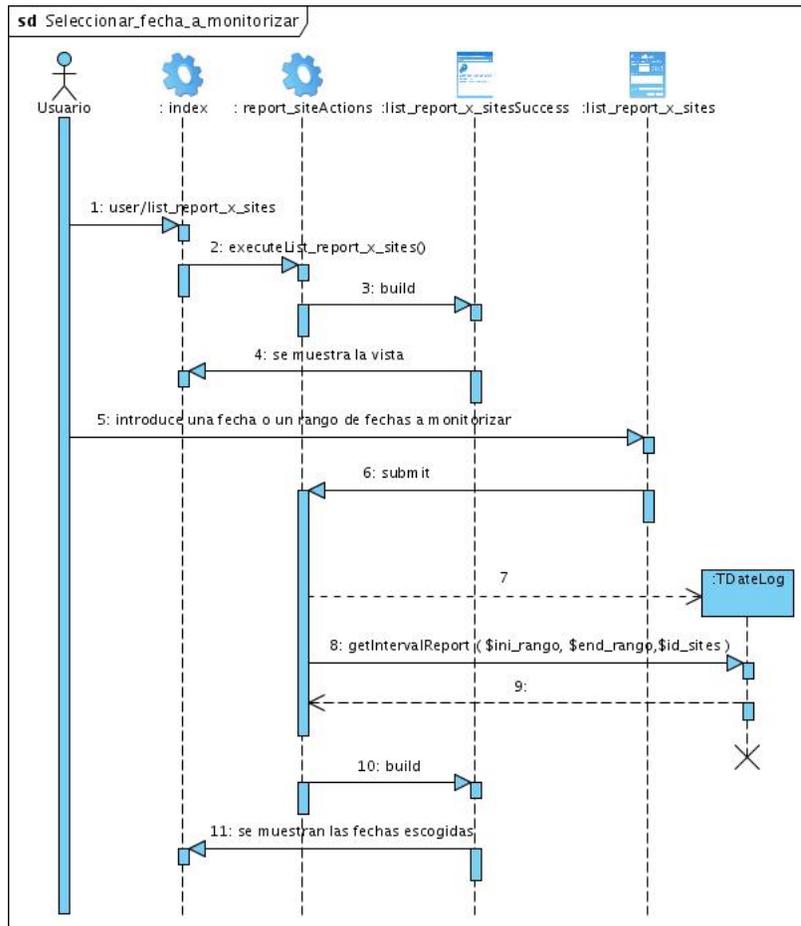


Figura 3.17: Seleccionar fecha a monitorizar

Módulo Reporte

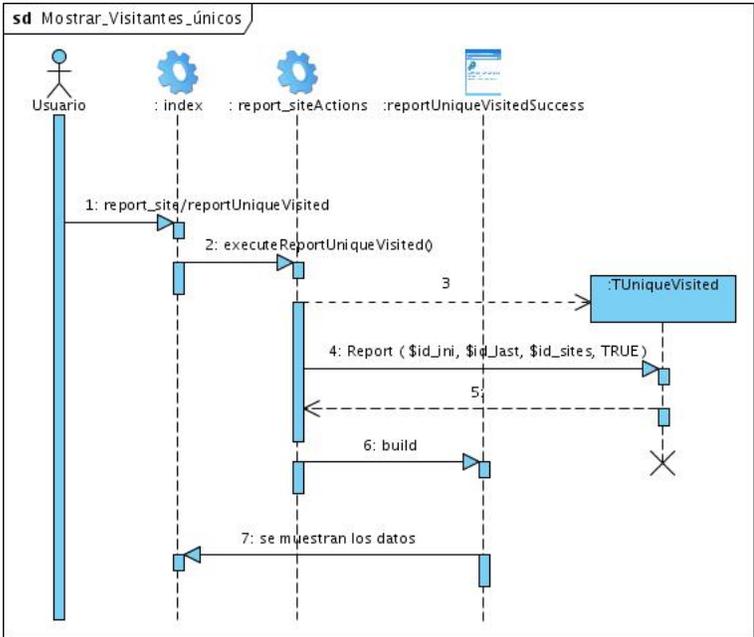


Figura 3.18: Mostrar visitantes únicos

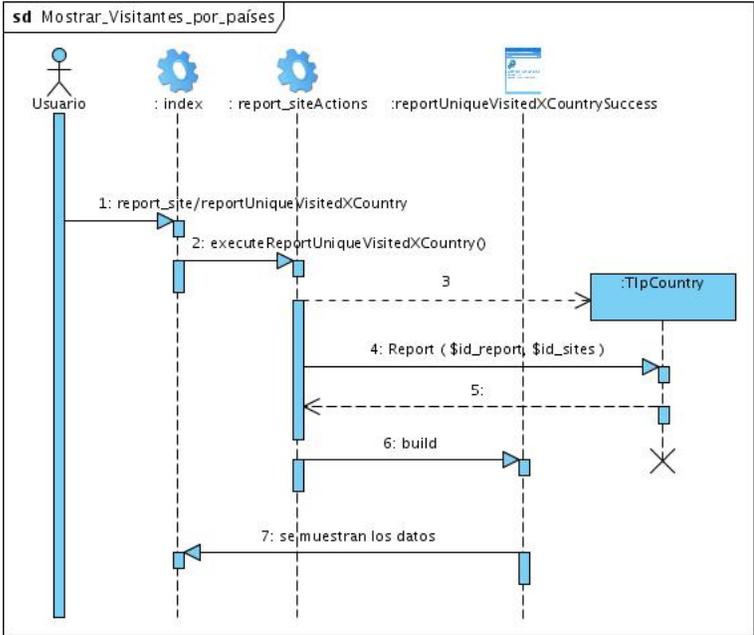


Figura 3.19: Mostrar visitantes por países

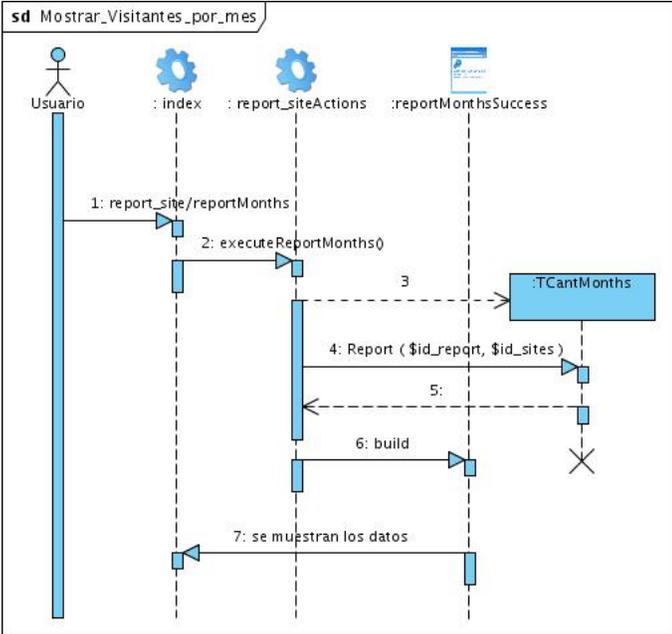


Figura 3.20: Mostrar visitantes por mes

Los restantes diagramas de Interacción se encuentran en el anexo E.

### **3.2.5. Descripción de las clases**

La descripción de las clases se encuentra en el documento adjunto *diagrama\_clases.pdf*.

### **3.2.6. Diseño de la Base de Datos**

Teniendo en cuenta que la base de datos que se utiliza en la aplicación es uno de los elementos principales de Airesweb, puede verse como un componente heredado que no se necesita modificar, por lo que solo se hará una descripción de las tablas que esta posee y sus atributos. A pesar de esto se ha presentado anteriormente el Diagrama Entidad Relación (DER) para una mejor comprensión del sistema Airesweb y su BD. La descripción de las tablas está en el documento adjunto *descripción\_bd.pdf*.

### **3.2.7. Interfaz**

Para el diseño de la presentación se generó el documento de las pautas que han de regir el diseño de la aplicación, estas aparecen en el documento adjunto *pautas\_diseño.pdf*.

### **3.2.8. Concepción de la ayuda**

Un elemento importante de cualquier aplicación es la ayuda que los usuarios necesitan para hacer uso de la misma. En este sentido se presenta un manual de usuario (ver documento adjunto *manual\_usuario\_airesweb.pdf*) que contribuya a facilitar el uso de la aplicación.

## **3.3. Conclusiones**

La generación de algunos artefactos relacionados con el flujo de análisis y diseño, teniendo en cuenta la arquitectura MVC que Symfony establece, permitió obtener una mayor comprensión de la aplicación y definir los principios que guiarán la implementación y organización de la misma. Además se analizaron las estrategias a seguir para dar respuesta a los requisitos no funcionales relacionados con la seguridad, se obtuvo

un prototipo de interfaz y las pautas de diseño para la misma, así como la concepción de la ayuda de la aplicación.

---

## Capítulo 4

# Implementación y prueba

---

En el presente capítulo se trata el flujo de trabajo de implementación y pruebas, este flujo es en el que se construye el modelo necesario para desarrollar el proceso de implementación del sistema con los diagramas de componentes definidos. También se elabora el diagrama de despliegue donde se representan los nodos necesarios, en los que se distribuye la aplicación. Además se abordarán los temas relacionados con las pruebas aplicadas al software.

### 4.1. Implementación

#### 4.1.1. Diagrama de Despliegue

El diagrama de Despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final. Es un grafo de nodos unidos por conexiones de comunicación donde cada nodo puede contener instancias de componentes. En general un nodo puede ser una unidad de computación de algún tipo, desde un sensor hasta un *mainframe* y las instancias de componentes de software pueden estar unidas por relaciones de dependencia[38].

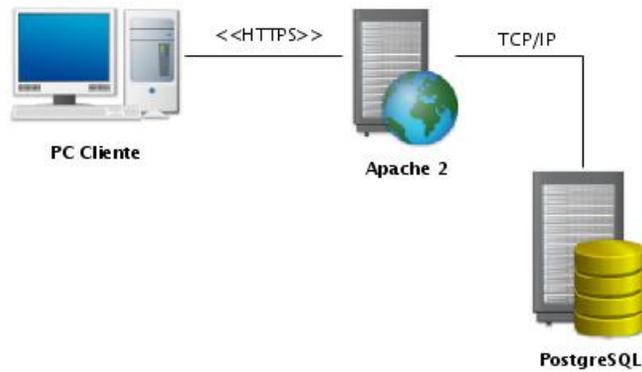


Figura 4.1: Diagrama de Despliegue de la aplicaci n.

#### 4.1.2. Diagramas de Componentes

El diagrama de Componentes muestra las organizaciones y dependencias l gicas entre componentes de software, sean estos ficheros de c digo fuente, binarios o ejecutables. Los elementos de modelado que lo conforman son los componentes y paquetes y muestran la estructura del sistema en t rminos de implementaci n a un alto nivel[39].

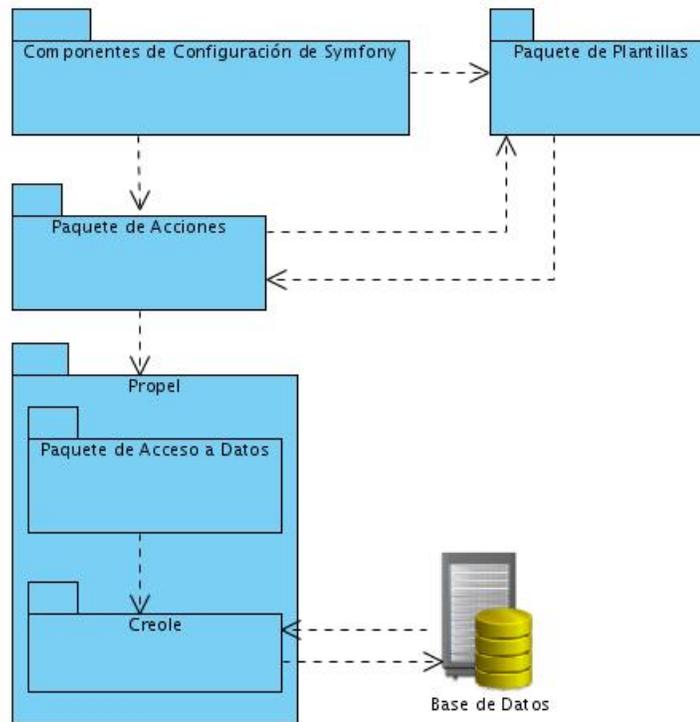


Figura 4.2: Diagrama de componentes general.

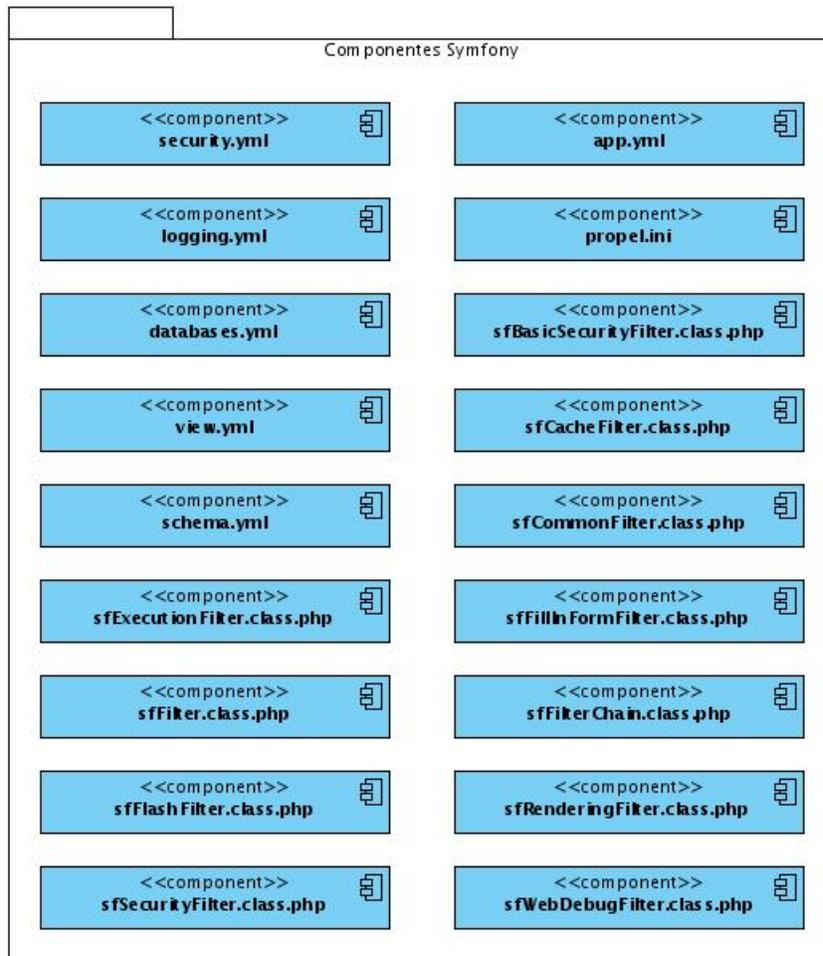


Figura 4.3: Diagrama de componentes de Symfony.

Los restantes diagramas de Componentes se encuentran en el anexo F.

## 4.2. Modelo de prueba

El desarrollo de sistemas de software implica una serie de actividades de producci n en las que las posibilidades de que aparezca el fallo humano son enormes. Los errores pueden empezar a darse desde el primer momento del proceso en el que los objetivos pueden estar especificados de forma err nea o imperfecta as  como en posteriores pasos de dise o y desarrollo. Debido a la imposibilidad humana de trabajar y comu-

nicarse de forma perfecta el desarrollo del software ha de ir acompañado de una actividad que garantice la calidad[40].

Las pruebas del software son un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación[40]. Existen diferentes estrategias, niveles, tipos y métodos de pruebas. Para realizar estas pruebas se diseñan casos de pruebas.

Las estrategias que se pueden seguir son varias, entre las que están incluir técnicas de detección de errores aplicados a los modelos de Análisis y Diseño, cambiar la estrategia para las pruebas de unidad e integración. El diseño de casos de prueba debe tener en cuenta las características propias del software. Los niveles se aplican para diferentes objetivos, entre ellos están las prueba de desarrollador, independiente, de Unidad, de Integración, de sistema de aceptación. Los tipos de pruebas son varios, pueden ser de funcionalidad, usabilidad, fiabilidad, rendimiento, y soportabilidad. Los métodos pueden ser pruebas de caja negra y pruebas de caja blanca.

Cualquier producto de ingeniería puede probarse de una de estas formas: (1) conociendo la función específica para la que fue diseñado el producto, se pueden llevar a cabo pruebas que demuestren que cada función es completamente operativa y al mismo tiempo buscando errores en cada función, (2) conociendo el funcionamiento del producto, se pueden desarrollar pruebas que aseguren que todas las piezas encajan, o sea, que la operación interna se ajusta a las especificaciones y que todos los componentes internos se han probado de forma adecuada. El primer enfoque de pruebas se denomina prueba de caja negra y el segundo, prueba de caja blanca[40]. Para la validación del sistema propuesto se llevará el método de caja negra.

#### **4.2.1. Pruebas de caja negra**

Cuando se considera el software de computadora, la prueba de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea los casos de pruebas pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto[40].

Permite obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Intenta encontrar errores de las siguientes categorías: funciones incorrectas o ausentes; errores de interfaz, en estructuras de datos o en acceso a bases de datos externas; errores de

rendimiento, de inicializaci n y de terminaci n[40].

En el documento adjunto *pruebas.pdf* est  la descripci n de los casos de pruebas.

### **4.3. Conclusiones**

Tras el flujo de implementaci n y prueba el sistema qued  desarrollado. En los diagramas generados pudo ilustrarse la relaci n entre los principales componentes del sistema y como estar  distribuido este. Las pruebas realizadas permitieron aumentar la calidad final de la soluci n. La aplicaci n final aprovech  las ventajas de Symfony y qued  lista para integrarse con Airesweb.

# Conclusiones

---

Después de estudiar varias soluciones existentes a nivel internacional de interfaces web de sistemas de análisis de registros, y ninguna contar con las características necesarias para añadirlas al sistema Airesweb se decidió desarrollar una IW propia. Con el desarrollo de esta se llegó a las siguientes conclusiones:

- El estudio de las interfaces de los principales analizadores de archivos de registros y profundizar en la historia, evolución y tendencias actuales de las aplicaciones Web permitió elegir las herramientas, tecnologías y la metodología para guiar el proceso de desarrollo.
- A partir de la descripción del problema y del levantamiento de requisitos se contruyeron los Casos de Uso, aspecto fundamental y de gran ayuda para las fases posteriores.
- La generación de algunos artefactos relacionados con el flujo de análisis y diseño, teniendo en cuenta la arquitectura MVC que Symfony establece, permitió obtener una mayor comprensión de la aplicación y definió los principios que guiaron la implementación del sistema.
- El uso de Symfony permitió acelerar el desarrollo de la aplicación, ahorrando tiempo y esfuerzos. Además la realización de las pruebas permitió que la aplicación quedara con mayor calidad.

Como conclusión general, puede afirmarse que se le dieron cumplimiento a todos los objetivos planteados al inicio del trabajo y se verificó la validez de la idea a defender materializada en la solución obtenida. Actualmente está implantada en el CIPRE, donde se están obteniendo los primeros resultados.

# Recomendaciones

---

Por la propia naturaleza cambiante de las aplicaciones Web y de los sistemas informáticos en general, y para tratar de aumentar las ventajas competitivas del sistema Airesweb y la IW como parte importante de este, se recomienda:

- Permitir mayor flexibilidad en la gestión de los roles y los sitios, asegurando que se puedan definir con mayor exactitud los permisos de cada usuario.
- Aprovechar las funcionalidades que brindan varios plugin de Symfony relacionados con Ajax, para mejorar la interacción con el usuario.
- Agregar la funcionalidad de generar reportes en formato portable(PDF) y el envío automático de este vía email.
- Estudiar otras metodologías que pudieran agilizar el desarrollo de futuras versiones.
- Permitir la opción a los usuarios de gestionar la forma de los gráficos generados, tanto tipos de gráfico, como formato.
- Utilizar los mecanismos de internacionalización de symfony para aumentar las posibilidades de uso de la IW del sistema Airesweb.

# Bibliografía

---

- [1] Sistema de estadísticas del sitio awstats. [http://wiki.gleducar.org.ar/wiki/Sistema\\_de\\_estad%C3%ADsticas\\_del\\_sitio\\_awstats](http://wiki.gleducar.org.ar/wiki/Sistema_de_estad%C3%ADsticas_del_sitio_awstats), 1 Abril 2009.
- [2] AWStats oficial web site. <http://awstats.sourceforge.net/>, 1 Abril 2009.
- [3] Webtrends. <http://www.dvc.es/es/alianzas/alianzas/Alianza8.html>, 1 Abril 2009.
- [4] Web Site Statistics Tools Reviews and News. <http://www.analyticalnews.org/statistical-analysis-products.html>, 20 Enero 2009.
- [5] WebLog Expert. <http://www.weblogexpert.com/>, 1 Abril 2009.
- [6] Brad A. Myers. *User interface software technology*. ACM, Marzo, 1996.
- [7] Ben Shneiderman *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison Wesley, 1998.
- [8] Cuauhtémoc Rivera Loaiza, Dr. Edgar L. Chávez González. Generador automático de mallas. <http://www.fismat.umich.mx/~crivera/tesis/node6.html>, 7 Mayo 2009.
- [9] Brad Myers, Scott E. Hudson, and Randy Pausch. *Past, Present and Future of User Interface Software Tools*. 1999.
- [10] Adriana Elba Martín. *Personalización de aplicaciones web*. Facultad de Informática – Universidad Nacional de La Plata – ARGENTINA, 2003.

- 
- [11] Valentín Moreira Gibaja. Las aplicaciones web en el entorno empresarial <http://www.camaravalencia.com/colecciondirectivos/leerArticulo.asp?intArticulo=2606>, 27 Febrero 2009.
- [12] Gerti Kappel et al. *Web Engineering*. John Wiley & Sons Ltd., 2006.
- [13] Leon Shklar, Richard Rosen. *Web Application Architecture: Principles, protocols and practices*. John Wiley & Sons Ltd, 2003.
- [14] Cristian Darie, Bogdan Brinzarea, Filip Cherecheș-Toșa, Mihai Bucica. *AJAX and PHP: Building Responsive Web Applications*. Packt Publishing Ltd, 2006.
- [15] Carles Mateu i Piñol. Introducción a las aplicaciones web. [http://www.cibernetia.com/manuales/introduccion\\_aplicaciones\\_web/3\\_historia\\_aplicaciones\\_web.php](http://www.cibernetia.com/manuales/introduccion_aplicaciones_web/3_historia_aplicaciones_web.php), 27 febrero 2009.
- [16] Sergio Luján Mora. *Programación de aplicaciones web: historia, principios básicos y cliente web*. Editorial Club Universitario, Alicante, 2002.
- [17] Roger S. Pressman. *Ingeniería del software un enfoque práctico*. McGraw-Hill, 2003.
- [18] Christian Van Der Henst S. ¿Qué es la Web 2.0?. <http://www.maestrosdelweb.com/editorial/web2/>, 3 abril 2009.
- [19] San Murugesan y Athula Ginige. *Web Engineering*. Springer Berlin / Heidelberg, 2001.
- [20] Rubén Álvarez. Desarrollo Web. Lenguajes de lado servidor. <http://www.desarrolloweb.com/articulos/243.php>, 10 mayo 2009.
- [21] Agustín Dondo. ¿Por qué elegir php? <http://www.programacion.com/php/articulo/porquephp/>, 7 Febrero 2009.
- [22] Andi Gutmans, Stig Sather Bakken, Derick Rethans. *PHP 5 Power Programming*. PRENTICE HALL, Professional Technical Reference, 2005.
- [24] Fabien Potencier, François Zaninotto. *Symfony la guía definitiva*, 2008.

- [25] Laya Franklin, Jonathan Ramos. *Evaluación de resultados de componentes del Módulo de Administración de Servicio Comunitario aplicando técnicas de medición de rendimiento y métricas de software, para apoyar la selección de un framework de desarrollo JEE, para la capa de presentación. Caso de estudio, JavaServer faces 1.2 y Struts 2.* Facultad de Ciencias – Universidad Central de Venezuela – Venezuela, 2009.
- [26] PHP Development Tools Project. <http://www.eclipse.org/pdt/>, 10 Mayo 2009.
- [27] Alexander Makarov. The Big PHP IDE Test: Why Use One And Which To Choose. <http://www.smashingmagazine.com/2009/02/11/the-big-php-ides-test-why-use-oneand-which-to-choose/>, 11 Mayo 2009.
- [28] Miguel Angel Cruz Condor. *Ingeniería Web.* 2007.
- [29] Gregor Engels et al. *Web Engineering: The Discipline of Systematic Development of Web Applications.* John Wiley & Sons Ltd., 2006.
- [30] Efraín Villanueva, Diego Fernando Molina. Aporte a Rup para el Desarrollo de Aplicaciones Web Transaccionales-Requerimientos no Funcionales y Administración de Riesgos. <http://www.javeriana.edu.co/biblos/tesis/ingenieria/Tesis205.pdf>, 11 Mayo 2009.
- [31] Ivar Jacobson et al *El Proceso Unificado de Desarrollo de Software.* Addison Wesley., 2000.
- [32] UML Resource Page. <http://www.uml.org/>, 11 Mayo 2009.
- [33] Visual Paradigm for UML. <http://www.visual-paradigm.com/product/vpum1/>, 11 Mayo 2009.
- [34] Javier J. Gutiérrez. *¿Qué es un framework web?*, 2006.
- [35] Modelo de Dominio. [http://iie.fing.edu.uy/ense/asign/desasoft/practico/hoja8/ejemplos\\_clase2.pdf](http://iie.fing.edu.uy/ense/asign/desasoft/practico/hoja8/ejemplos_clase2.pdf), 20 Abril 2009.
- [36] Craig Larman. *UML y patrones Tomo II.* PEARSON, 2003.
- [37] Ana Fernandez Vilas. Diagrama de Componentes. <http://www-gris.det.uvigo.es/~avilas/UML/node49.html>, 15 Mayo 2009

- [38] Ana Fernandez Vilas. Diagrama de Despliegue. <http://www-gris.det.uvigo.es/~avilas/UML/node50.html>, 15 Mayo 2009

# Referencias bibliográficas

---

- [1] Ben Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison Wesley, Reading, Massachusetts., 1998.
- [2] Brad A. Myers. *User interface software technology*. ACM, March 1996.
- [3] Dr. Edgar L. Chávez González Cuauhtémoc Rivera Loaiza. *Generador automático de mallas* [Internet]. Disponible en: <<http://www.fismat.umich.mx/~crivera/tesis/node6.html>> [Citado 7 mayo 2009].
- [4] Scott E. Hudson Brad Myers and Randy Pausch. *Past, present and future of user interface software tools*. 1999.
- [5] *Sistema de estadísticas del sitio awstats* [Internet]. Disponible en: <[http://wiki.gleducar.org.ar/wiki/Sistema\\_de\\_estad%C3%ADsticas\\_del\\_sitio\\_awstats](http://wiki.gleducar.org.ar/wiki/Sistema_de_estad%C3%ADsticas_del_sitio_awstats)> [Citado 1 Abril 2009].
- [6] *AWStats official web site* [Internet]. Disponible en: <<http://awstats.sourceforge.net/>> [Citado 1 Abril 2009].
- [7] *Webtrends* [Internet]. Disponible en: <<http://www.dvc.es/es/alianzas/alianzas/Alianza8.html>> [Citado 1 Abril 2009].
- [8] *Web Site Statistics Tools Reviews and News* [Internet]. Disponible en: <<http://www.analyticalnews.org/statistical-analysis-products.html>> [Citado 20 Enero 2009].
- [9] *WebLog Expert* [Internet]. Disponible en: <<http://www.weblogexpert.com/>> [Citado 1 Abril 2009].

- 
- [10] Adriana Elba Martín. *Personalización de aplicaciones web*. Magister en ingeniería de software, Facultad de Informática – Universidad Nacional de La Plata – ARGENTINA, 2003.
- [11] Richard Rosen Leon Shklar. *Web Application Architecture: Principles, protocols and practices*, chapter Introduction, page 22. John Wiley & Sons Ltd., 2003.
- [12] Gerti Kappel et al. *Web Engineering: The Discipline of Systematic Development of Web Applications*, chapter An Introduction to Web Engineering, pages 1–22. John Wiley & Sons Ltd., 2006.
- [13] Filip Cherecheș-Toșa Mihai Bucica Cristian Darie, Bogdan Brinzarea. *AJAX and PHP: Building Responsive Web Applications*, chapter AJAX and the Future of Web Applications, pages 1–21. Packt Publishing Ltd., 2006.
- [14] Carles Mateu i Piñol. *Introducción a las aplicaciones web* [Internet]. Disponible en: <[http://www.cibernetia.com/manuales/introduccion\\_aplicaciones\\_web/3\\_historia\\_aplicaciones\\_web.php](http://www.cibernetia.com/manuales/introduccion_aplicaciones_web/3_historia_aplicaciones_web.php)> [Citado 27 febrero 2009].
- [15] Sergio Luján Mora. *Programación de aplicaciones web: historia, principios básicos y cliente web.*, chapter Historia de Internet, pages 5–39. Editorial Club Universitario, Alicante, 2002.
- [16] Christian Van Der Henst S. *¿Qué es la Web 2.0?* [Internet]. Disponible en: <<http://www.maestrosdelweb.com/editorial/web2/>> [Citado 3 abril 2009].
- [17] Roger S. Pressman. *Ingeniería del software. Un enfoque práctico*, chapter Ingeniería Web, pages 521–540. McGraw-Hill, 2002.
- [18] San Murugesan y Athula Ginige. *Web Engineering*, chapter Web Engineering: Introduction and Perspectives, pages 1–30. Springer Berlin / Heidelberg, 2001.
- [19] Gerti Kappel et al. *Web Engineering: The Discipline of Systematic Development of Web Applications*, chapter An Introduction to Web Engineering, pages 65–67. John Wiley & Sons Ltd., 2006.
- [20] Richard Rosen Leon Shklar. *Web Application Architecture: Principles, protocols and practices*, chapter Introduction, pages 201–205. John Wiley & Sons Ltd., 2003.

- 
- [21] Rubén Álvarez. *Desarrollo Web. Lenguajes de lado servidor* [Internet]. Disponible en: <<http://www.desarrolloweb.com/articulos/243.php>> [Citado 10 mayo 2009].
- [22] Derick Rethans Andi Gutmans, Stig Sather Bakken. *PHP 5 Power Programming*, chapter PHP 5 Basic Language, pages 10–13. PRENTICE HALL, Professional Technical Reference, 2005.
- [23] M. A. Álvarez. *¿Qué es PHP?*, page 87. 2004.
- [24] Agustín Dondo. *¿Por qué elegir PHP?* [Internet]. Disponible en: <<http://www.programacion.com/php/articulo/porquephp/>> [Citado 7 febrero 2009].
- [25] Daniel López. *PHP4 y PHP5: ¿Cuál elegir? ¿Migrar o no Migrar? El advenimiento de PHP6* [Internet]. Disponible en: <<http://www.maestrosdelweb.com/editorial>> [Citado 10 mayo 2009].
- [26] *UML Resource Page* [Internet]. Disponible en: <<http://www.uml.org/>> [Citado 11 Mayo 2009].
- [27] Javier J. Gutiérrez. *¿qué es un framework web?* 2006.
- [28] François Zaninotto Fabien Potencier. *Symfony la guía definitiva*. 2008.
- [29] Jonathan Ramos Laya Franklin. *Evaluación de resultados de componentes del Módulo de Administración de Servicio Comunitario aplicando técnicas de medición de rendimiento y métricas de software, para apoyar la selección de un framework de desarrollo JEE, para la capa de presentación. Caso de estudio, JavaServer faces 1.2 y Struts 2*. Licenciados en computación, Facultad de Ciencias – Universidad Central de Venezuela – Venezuela, 2009.
- [30] *PHP Development Tools Project* [Internet]. Disponible en: <<http://www.eclipse.org/pdt/>> [Citado 10 mayo 2009].
- [31] *Visual Paradigm for UML* [Internet]. Disponible en: <<http://www.visual-paradigm.com/product/vpuml/>> [Citado 11 Mayo 2009].
- [32] Miguel Angel Cruz Condor. *Ingeniería web*. 2007.
- [33] Gregor Engels et al. *Web Engineering: The Discipline of Systematic Development of Web Applications*, chapter The Web Application Development Process, pages 197–218. John Wiley & Sons Ltd., 2006.

- [34] Diego Fernando Molina Efraín Villanueva. *Aporte a Rup para el Desarrollo de Aplicaciones Web Transaccionales-Requerimientos no Funcionales y Administración de Riesgos* [Internet]. Disponible en: <<http://www.javeriana.edu.co/biblos/tesis/ingenieria/Tesis205.pdf>> [Citado 11 Mayo 2009].
- [35] Ivar Jacobson et al. *El Proceso Unificado de Desarrollo de Software*, chapter El Proceso Unificado: dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental, pages 1–12. Addison Wesley, 2000.
- [36] *Modelo de Dominio* [Internet]. Disponible en: <[http://iie.fing.edu.uy/ense/assign/desasoft/practico/hoja8/ejemplos\\_clase2.pdf](http://iie.fing.edu.uy/ense/assign/desasoft/practico/hoja8/ejemplos_clase2.pdf)> [Citado 20 Abril 2009].
- [37] Craig Larman. *UML y patrones Tomo II*. PEARSON, 2003.
- [38] Ana Fernandez Vilas (2001). *Diagrama de Despliegue* [Internet]. Disponible en: <<http://www-gris.det.uvigo.es/~avilas/UML/node50.html>> [Citado 15 mayo 2009].
- [39] Ana Fernandez Vilas (2001). *Diagrama de Componentes* [Internet]. Disponible en: <<http://www-gris.det.uvigo.es/~avilas/UML/node49.html>> [Citado 15 mayo 2009].
- [40] Roger S. Pressman. *Ingeniería de software, un enfoque práctico*, chapter Técnicas de prueba del software, pages 281–299. McGraw-Hill, 2001.

## Anexo A

# Figuras relacionadas

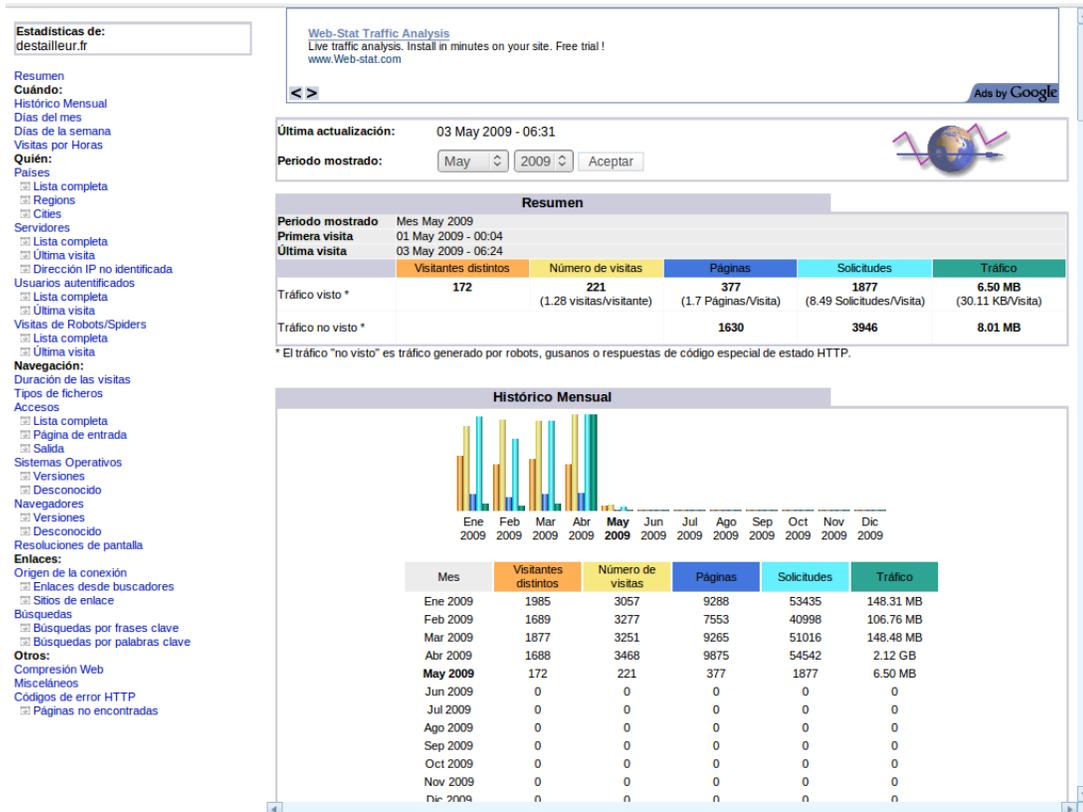


Figura A.1: Interfaz Web de Awstats.

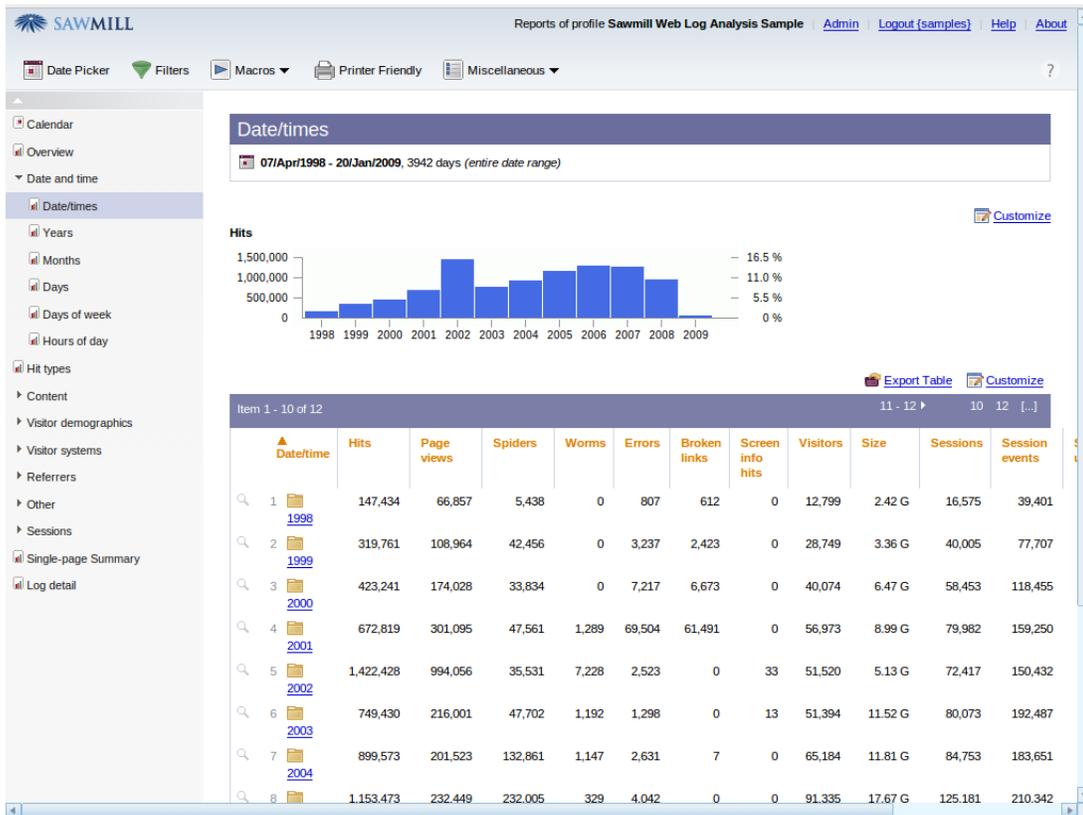


Figura A.2: Interfaz Web de Sawmill.

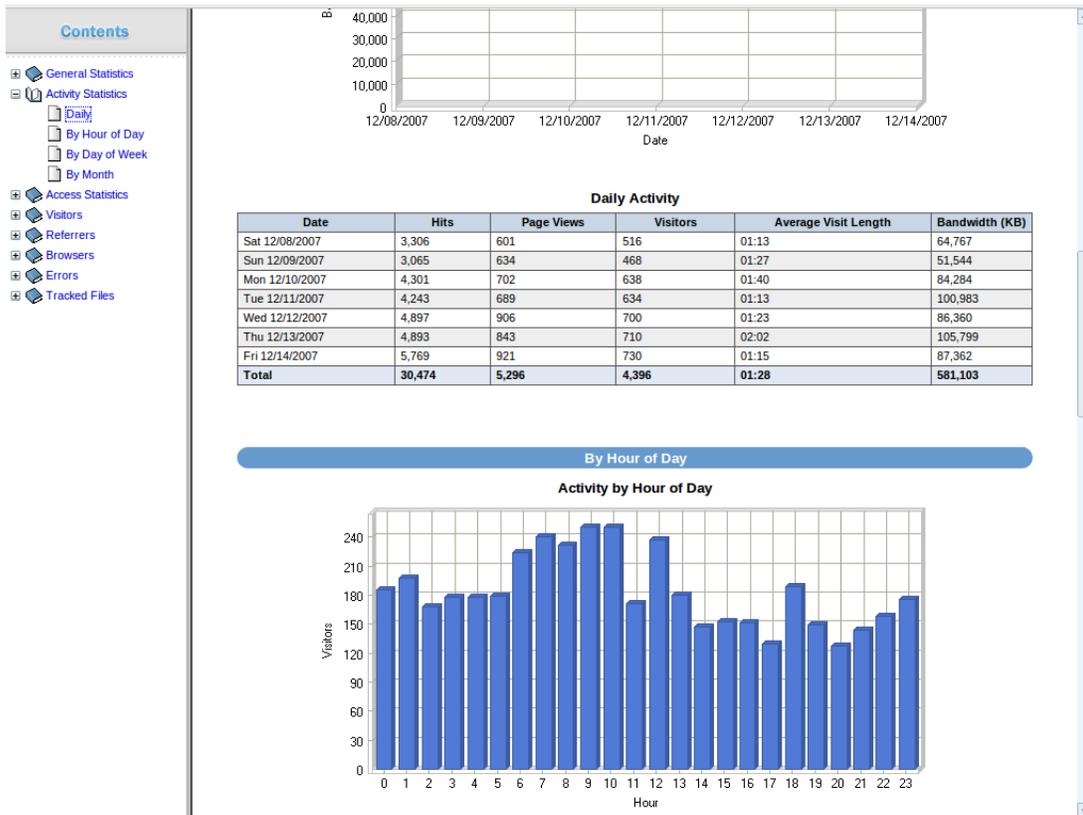
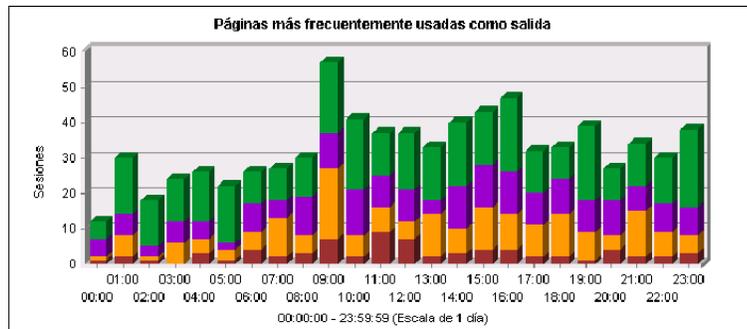


Figura A.3: Interfaz Web de WebLog Expert.

- Contenido**
- Estadísticas generales
  - Recursos Accedidos
    - Páginas principales
    - Páginas más frecuentemente usadas
    - Formularios presentados y solicitados
    - Páginas más frecuentemente usadas como salida
    - Páginas más frecuentemente usadas como entrada
    - ▶ **Págs. más frecuentemente usadas como salida**
    - Páginas de acceso simple
    - Directorios de mayor acceso
    - Rutas más usuales por el sitio
    - Archivos más frecuentemente usados
    - Tipos y tamaños de archivo transferidos
  - Visitantes y Demografía
  - Estadística de Actividad
  - Estadística Técnica
  - Referidores & Palabras Clave
  - Exploradores & Plataformas
  - Estadísticas de depuración
  - Glosario

**Páginas más frecuentemente usadas** Help ?

El Gráfico y la Tabla de páginas más frecuentemente usadas como salida muestran las páginas que los visitantes del sitio vieron en último lugar.



Páginas más frecuentemente usadas como salida			
	Páginas	% del total	Sesiones
1	Cuba, Agencia Cubana de Noticias (ACN), división de la Agencia de Información <a href="http://www.ain.cu/">http://www.ain.cu/</a>	14.85%	340
2	<a href="http://www.ain.cu/imagenes/ b-curio/">http://www.ain.cu/imagenes/ b-curio/</a>	8.43%	193
3	<a href="http://www.ain.cu/2009/abril/ b-curio/">http://www.ain.cu/2009/abril/ b-curio/</a>	7.82%	179
4	<a href="http://www.ain.cu/mayo01-09/ b-curio/">http://www.ain.cu/mayo01-09/ b-curio/</a>	3.1%	71
5	<b>Celebración del Primero de Mayo en las provincias cubanas</b> <a href="http://www.ain.cu/mayo01-09/ provincias.htm">http://www.ain.cu/mayo01-09/ provincias.htm</a>	1.52%	35
6	<b>КУБИНСКОЕ АГ&amp;#</b> <a href="http://www.ain.cu/dioma/ruso/ principalruso.html">http://www.ain.cu/dioma/ruso/ principalruso.html</a>	1.52%	35
7	<a href="http://www.ain.cu/ guerrabiologica/may2402cmpruebas.htm">http://www.ain.cu/ guerrabiologica/may2402cmpruebas.htm</a>	1.26%	29
8	<b>INTERNACIONALES / Agencia Cubana de Noticias</b> <a href="http://www.ain.cu/secciones/ internacionales.htm">http://www.ain.cu/secciones/ internacionales.htm</a>	1.13%	26
9	<a href="http://www.ain.cu/2007/ fidelarticulos.htm">http://www.ain.cu/2007/ fidelarticulos.htm</a>	1%	23
10	<b>Recetas de belleza con miel</b> <a href="http://www.ain.cu/mujer/ recetasmiel.htm">http://www.ain.cu/mujer/ recetasmiel.htm</a>	0.96%	22
11	<a href="http://www.ain.cu/2009/mayo/ b-curio/">http://www.ain.cu/2009/mayo/ b-curio/</a>	0.91%	21

Figura A.4: Interfaz Web de Webtrends.

---

## Anexo B

# Diagramas de clases del Análisis

---

### Módulo Usuario

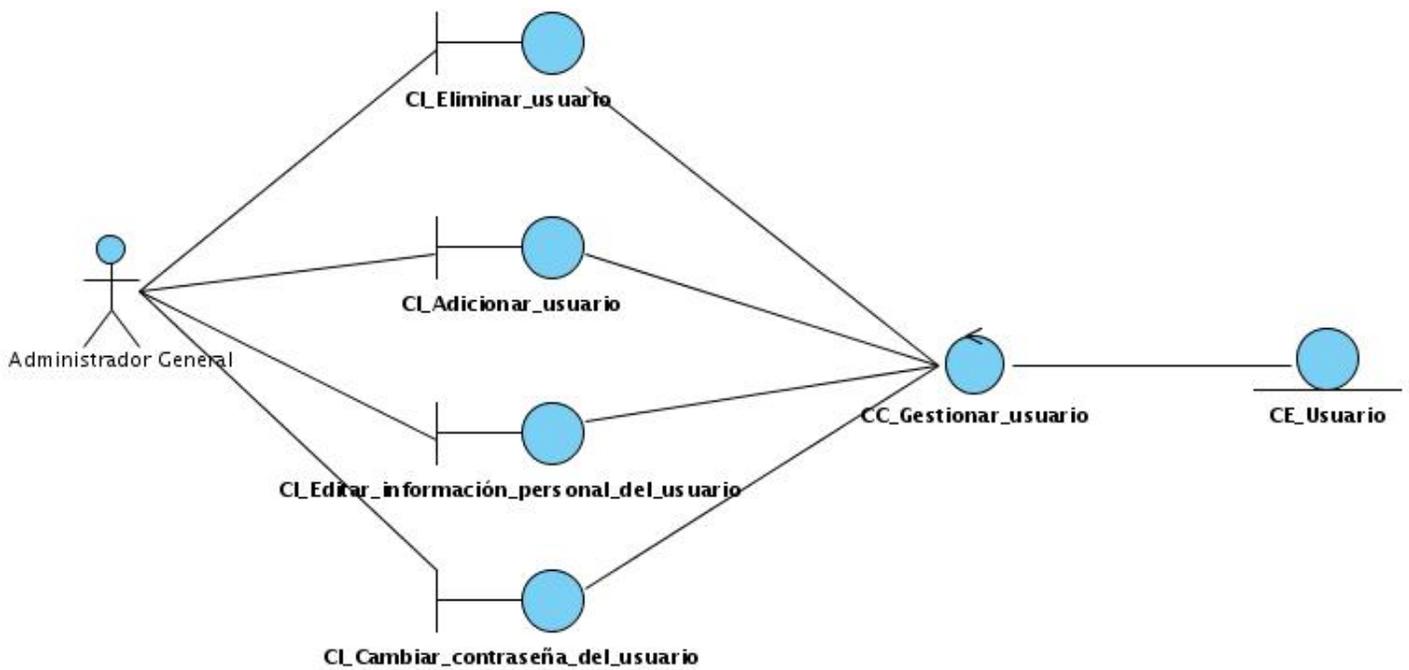


Figura B.1: DCA Administrador general Gestionar usuario y Cambiar contraseña del usuario

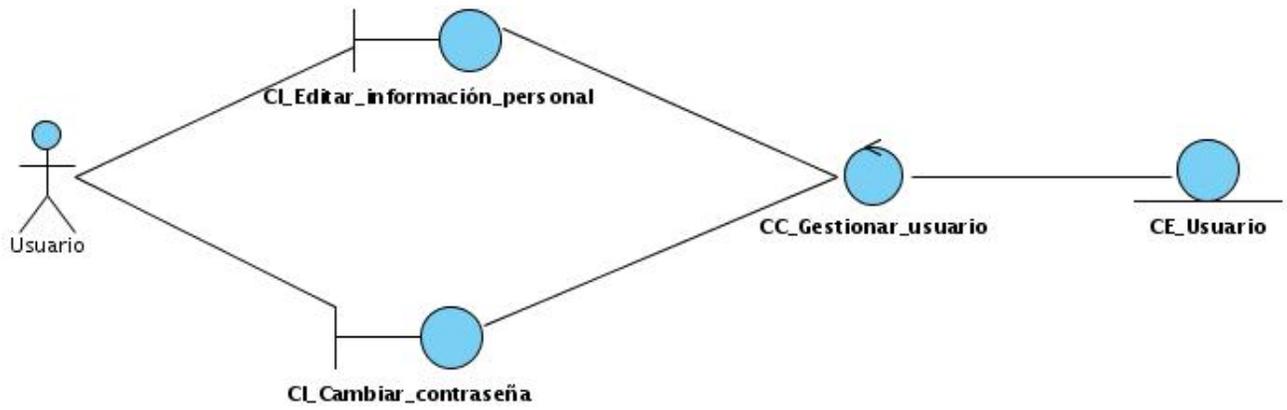


Figura B.2: DCA Editar información personal y Cambiar contraseña personal

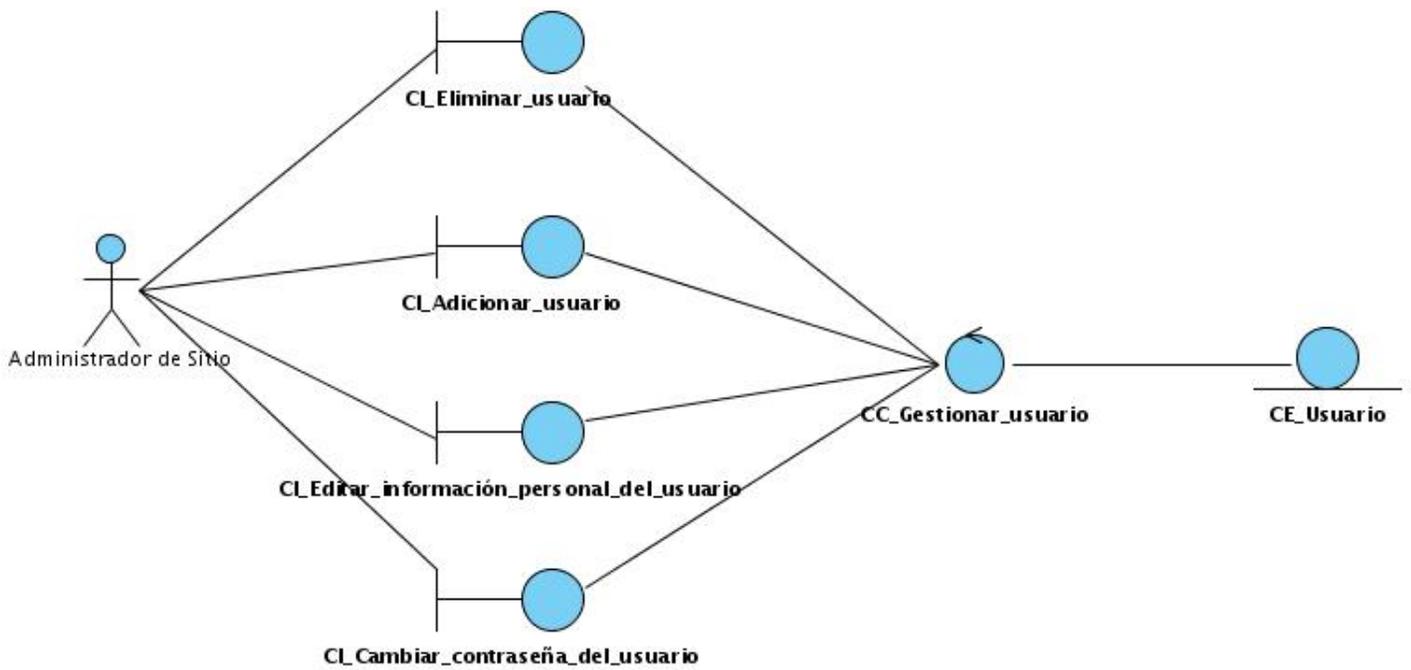


Figura B.3: DCA Gestionar usuario y Cambiar contraseña del usuario

---

## Anexo C

# Casos de Uso expandidos

---

### Módulo Usuario

Caso de Uso:	Gestionar Usuario.
Actores:	Administrador de sitio, Administrador General.
Resumen:	El Caso de Uso se inicia cuando el actor decide eliminar, adicionar o modificar la información personal de algún usuario que administra.
Referencia:	R5
CU asociados:	
Precondiciones:	Actor autenticado en el sistema.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Principal”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor necesita eliminar, adicionar o editar la información personal de algún usuario que administra.</p>	<p>1. El sistema ejecuta alguna de las siguientes acciones:</p> <p>1.1 Si decide adicionar a un usuario ir a la sección "Adicionar Usuario".</p> <p>1.2 Si decide eliminar a un usuario, ir a la sección "Eliminar Usuario".</p> <p>1.3 Si decide editar la información personal de un usuario, ir a la sección "Editar información personal de usuario".</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<b>Flujo Normal de Eventos</b>	
<b>Sección "Adicionar Usuario"</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor entra los datos del nuevo usuario al sistema.</p>	<p>1.1 El sistema verifica que todos los campos estén llenos.</p> <p>1.2 El sistema verifica que el usuario no exista en la Base de Datos.</p> <p>1.3 El sistema almacena los datos del nuevo usuario y emite un mensaje informándole al actor que se realizó la acción satisfactoriamente y así termina el caso de uso.</p>
<b>Flujo alternativo</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>1.1.a Se emite un mensaje para que llene los campos obligatorios y regresa a la acción 1 de esta sección.</p> <p>1.2.a Si el usuario existe se muestra un mensaje informativo y regresa a la acción 1 de esta sección.</p>
<b>Flujo Normal de Eventos</b>	
<b>Sección "Eliminar Usuario"</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor introduce los parámetros para localizar el usuario a eliminar.</p> <p>2. El actor selecciona eliminar al usuario.</p> <p>3. El actor confirma eliminar el usuario seleccionado.</p>	<p>1.1 El sistema muestra el usuario que coincide con los parámetros establecidos.</p> <p>2.1 Es sistema pide confirmación.</p> <p>3.1 El sistema elimina el usuario seleccionado y termina el caso de uso.</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>3.a El actor cancela la acción de eliminar el usuario.</p>	<p>3.1.a El sistema regresa a la acción 2 de esta sección .</p>
<b>Flujo Normal de Eventos</b>	
<b>Sección “Editar información personal de usuario”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor selecciona el usuario a modificar.</p> <p>2. El actor realiza los cambios deseados y selecciona la opción de actualizar.</p>	<p>1.1 El sistema brinda la posibilidad de modificar los datos.</p> <p>2.1 Se verifica que los campos obligatorios estén llenos.</p> <p>2.2 Se actualiza la Información y finaliza el caso de uso.</p>
---	--

**Flujo alternativo**

<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>2.1.a Se emite un mensaje para que llene los campos obligatorios y regresa a la acción 2 de esta sección.</p>

Poscondiciones:	
Prioridad:	Crítico
Especificaciones:	
Complementaria:	

Caso de Uso:	Cambiar contraseña del usuario.
Actores:	Administrador de Sitio, Administrador General.
Resumen:	El Caso de Uso se inicia cuando el actor decide cambiar la contraseña del usuario.
Referencia:	R6

Continúa en la próxima página

CU asociados:	
Precondiciones:	Actor autenticado como Administrador General o Administrador de Sitio.
<b>Flujo Normal de Eventos</b>	
<b>Sección “nombre del escenario”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<ol style="list-style-type: none"> <li>1. El actor solicita cambiar la contraseña de un usuario.</li> <li>2. El actor inserta y confirma la nueva contraseña del usuario.</li> </ol>	<ol style="list-style-type: none"> <li>1 El sistema le muestra un formulario para que inserte y confirme la nueva contraseña.</li> <li>2.1 El sistema valida la información.</li> <li>2.2 El sistema actualiza la contraseña del actor en la Base de Datos.</li> </ol>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Poscondiciones:	El actor finalmente cambia la contraseña del usuario.
Prioridad:	Secundario
Especificaciones:	
Complementaria:	

Caso de Uso:	Gestionar administrador.
Actores:	Administrador General.
Resumen:	El Caso de Uso se inicia cuando el actor decide eliminar, adicionar o modificar la información personal de algún Administrador del sistema.
Continúa en la próxima página	

Referencia:	R7
CU asociados:	
Precondiciones:	Autenticado en el sistema como Administrador General.
<b>Flujo Normal de Eventos</b>	
<b>Sección “Principal”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. El actor necesita eliminar, adicionar o editar la información personal de algún administrador del sistema.</p>	<p>1. El sistema ejecuta alguna de las siguientes acciones:</p> <p>1.1 Si decide adicionar a un administrador ir a la sección “Adicionar Administrador”.</p> <p>1.2 Si decide eliminar a un administrador, ir a la sección “Eliminar Administrador”.</p> <p>1.3 Si decide editar la información personal de un administrador, ir a la sección “Editar información personal de administrador”.</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<b>Flujo Normal de Eventos</b>	
<b>Sección “Adicionar Administrador”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor entra los datos del nuevo administrador al sistema.</p>	<p>1.1 El sistema verifica que todos los campos estén llenos.</p> <p>1.2 El sistema verifica que el administrador no exista en la Base de Datos.</p> <p>1.3 El sistema almacena los datos del nuevo administrador y emite un mensaje informándole al actor que se realizó la acción satisfactoriamente y así termina el caso de uso.</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>1.1.a Se emite un mensaje para que llene los campos obligatorios y regresa a la acción 1 de esta sección.</p> <p>1.3.a Si el usuario existe se muestra un mensaje informativo y regresa a la acción 1 de esta sección.</p>
<b>Flujo Normal de Eventos</b>	
<b>Sección “Eliminar Administrador”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor introduce los parámetros para localizar el administrador a eliminar.</p> <p>2. El actor selecciona eliminar al administrador.</p> <p>3. El actor confirma eliminar el administrador seleccionado.</p>	<p>1.1 El sistema muestra el administrador que coincide con los parámetros establecidos.</p> <p>2.1 El sistema pide confirmación.</p> <p>3.1 El sistema elimina el administrador seleccionado y termina el caso de uso.</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>3.a El actor cancela la acción de eliminar el administrador.</p>	<p>3.1.a El sistema regresa a la acción 2 de esta sección .</p>
<b>Flujo Normal de Eventos</b>	
<b>Sección “Editar información personal de administrador”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
Continúa en la próxima página	

<p>1. El actor selecciona el administrador a modificar.</p> <p>2. El actor realiza los cambios deseados y selecciona la opción de actualizar.</p>	<p>1.1 El sistema brinda la posibilidad de modificar los datos.</p> <p>2.1 Se verifica que los campos obligatorios estén llenos.</p> <p>2.2 Se actualiza la Información y finaliza el caso de uso.</p>
<b>Flujo alternativo</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>2.1.a Se emite un mensaje para que llene los campos obligatorios y regresa a la acción 2 de esta sección.</p>
Poscondiciones:	
Prioridad:	Crítico
Especificaciones:	
Complementaria:	

Caso de Uso:	Cambiar contraseña del administrador.
Actores:	Administrador General.
Resumen:	El Caso de Uso se inicia cuando el actor decide cambiar la contraseña de algún administrador.
Continúa en la próxima página	

Referencia:	R8
CU asociados:	
Precondiciones:	Autenticado en el sistema como Administrador General.
<b>Flujo Normal de Eventos</b>	
<b>Sección “nombre del escenario”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<p>1. El actor solicita cambiar la contraseña del administrador.</p> <p>2. El actor inserta y confirma su nueva contraseña.</p>	<p>1.1 El sistema le muestra un formulario para que inserte y confirme la nueva contraseña.</p> <p>2.1 El sistema valida la información.</p> <p>2.2 El sistema actualiza la contraseña del administrador en la Base de Datos.</p>
<b>Flujo alterno</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<p>2.1 El sistema envía un mensaje en caso de tener problema la contraseña entrada por el actor y retorna al paso 2.</p>
Poscondiciones:	La contraseña del administrador finalmente es modificada y guardada en la Base de Datos del sistema.
Prioridad:	Crítico
Especificaciones:	
Continúa en la próxima página	

Complementaria:	
-----------------	--

Caso de Uso:	Listar sitios monitorizados.
Actores:	Administrador de Sitio, Administrador General, Usuario.
Resumen:	El Caso de Uso se inicia cuando el actor decide mostrar los sitios monitorizados.
Referencia:	R9
CU asociados:	
Precondiciones:	Actor autenticado en el sistema.

**Flujo Normal de Eventos**

**Sección "nombre del escenario"**

<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El actor solicita la opción de listar los sitios monitorizados.	1.1 El sistema realiza la búsqueda en la Base de Datos.  1.2 El sistema muestra los resultados obtenidos.

**Flujo alterno**

<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>

Poscondiciones:	El actor finalmente obtiene una lista con los sitios monitorizados.
Prioridad:	Secundario
Especificaciones:	
Complementaria:	

Continúa en la próxima página

Caso de Uso:	Seleccionar sitio a monitorizar.	
Actores:	Administrador de Sitio, Administrador General, Usuario.	
Resumen:	El Caso de Uso se inicia cuando el actor decide seleccionar un sitio para ver sus reportes.	
Referencia:	R10	
CU asociados:		
Precondiciones:	Actor autenticado en el sistema. Que existan sitios monitorizados.	
<b>Flujo Normal de Eventos</b>		
<b>Sección "nombre del escenario"</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1. El actor selecciona un sitio de los monitorizados para ver sus reportes.	1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos.	
<b>Flujo alterno</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
Poscondiciones:	El actor finalmente selecciona un sitio para ver sus reportes.	
Prioridad:	Secundario	
Especificaciones:		
Complementaria:		

Caso de Uso:	Seleccionar fecha a monitorizar.	
Actores:	Administrador de Sitio, Administrador General, Usuario.	
Continúa en la próxima página		

Resumen:	El Caso de Uso se inicia cuando el actor decide seleccionar una fecha o un rango de fechas para ver los reportes correspondientes a un sitio monitorizado.	
Referencia:	R11	
CU asociados:		
Precondiciones:	Actor autenticado en el sistema. Que exista un sitio seleccionado para ver sus reportes.	
<b>Flujo Normal de Eventos</b>		
<b>Sección "nombre del escenario"</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1. El actor selecciona una fecha o un rango de fechas.	1.1 El sistema realiza la búsqueda en la Base de Datos. 1.2 El sistema muestra los resultados obtenidos.	
<b>Flujo alterno</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
Poscondiciones:	El actor finalmente selecciona una fecha o un rango de fechas.	
Prioridad:	Secundario	
Especificaciones:		
Complementaria:		

---

## Anexo D

# Diagramas de clases del Diseño

---

### Módulo Usuario

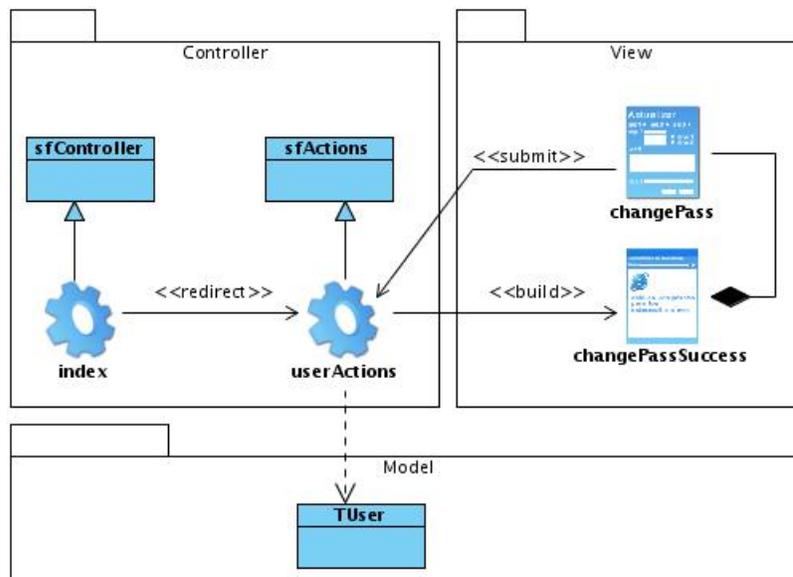


Figura D.1: DCD Cambiar Contraseña

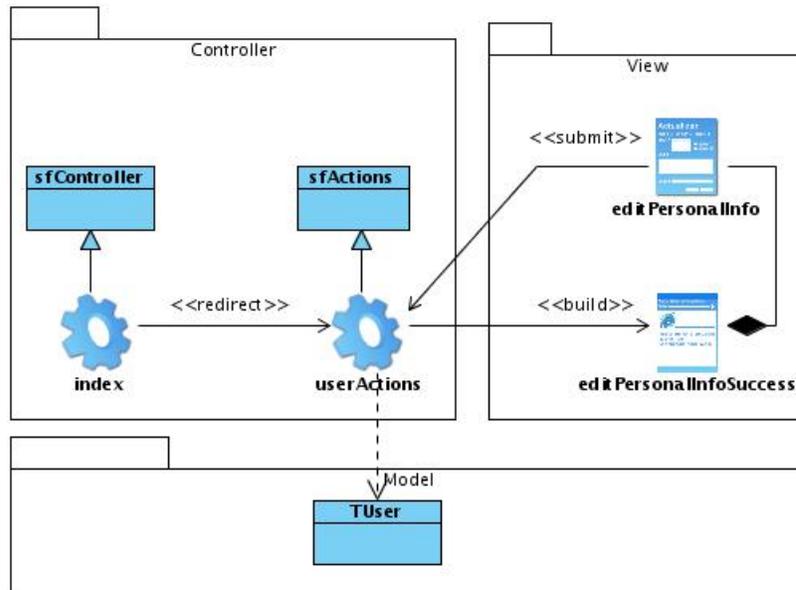


Figura D.2: DCD Editar Información Personal

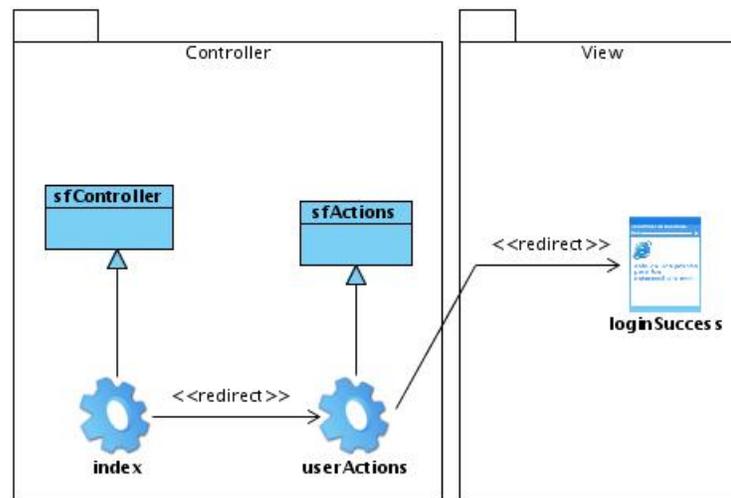


Figura D.3: DCD Finalizar Sesión

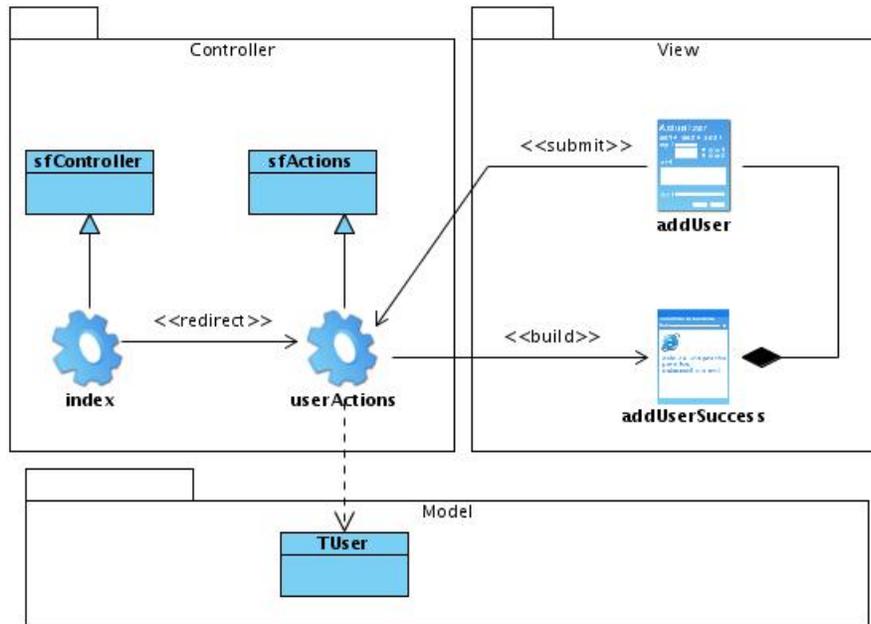


Figura D.4: DCD Gestionar Administrador Escenario Adicionar Administrador

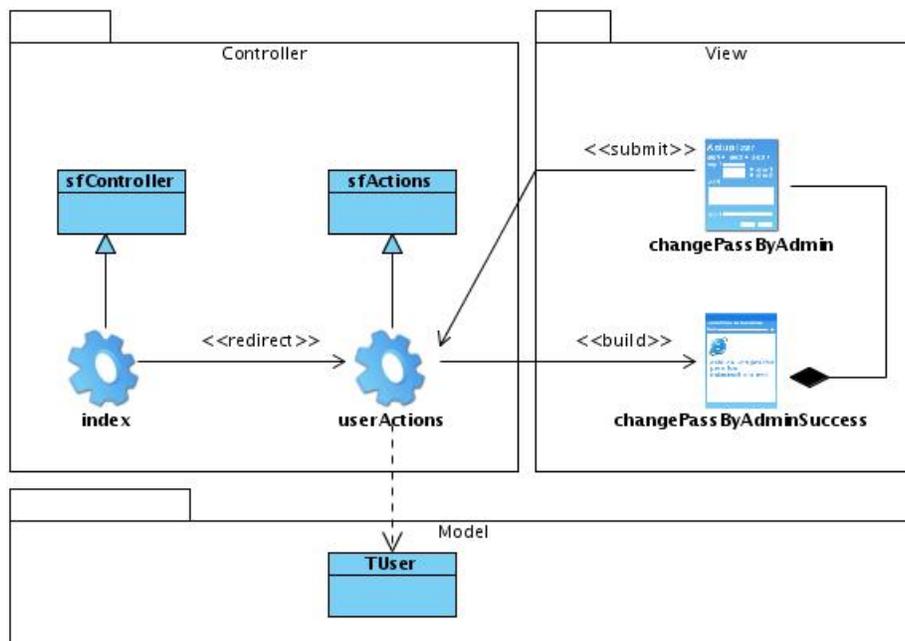


Figura D.5: DCD Gestionar Administrador Escenario Cambiar Contraseña del Administrador

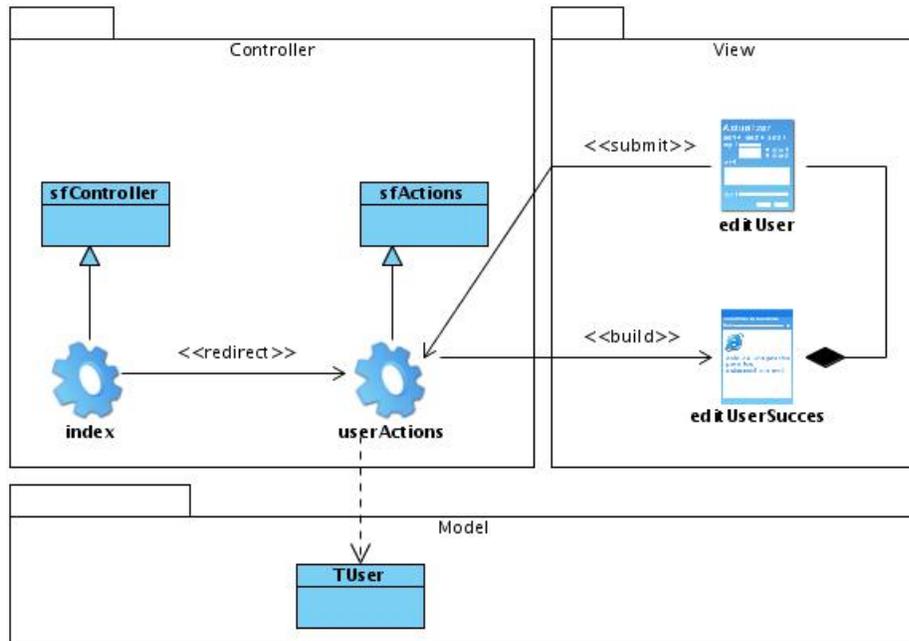


Figura D.6: DCD Gestionar Administrador Escenario Editar Información Personal de Administrador

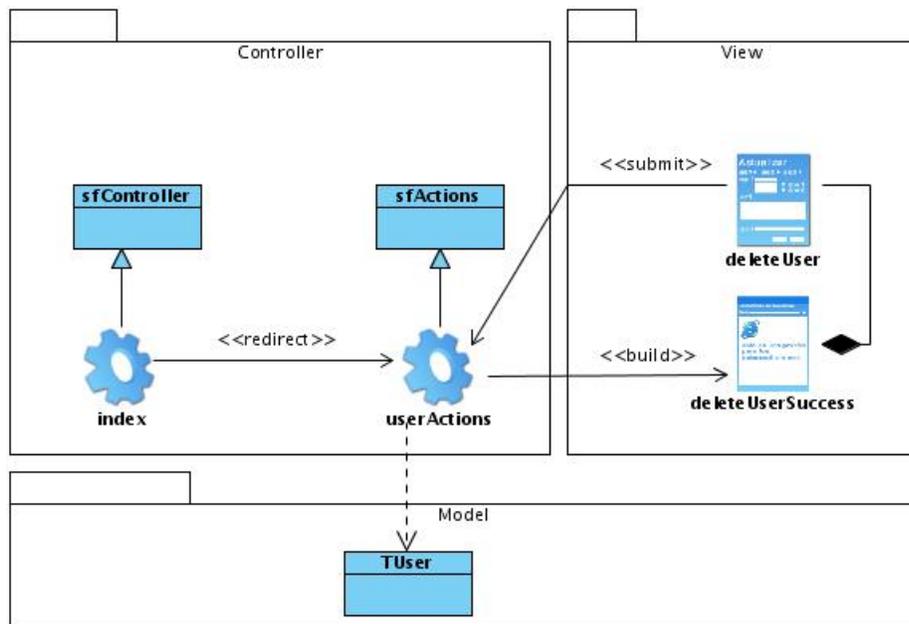


Figura D.7: DCD Gestionar Administrador Escenario Eliminar Administrador

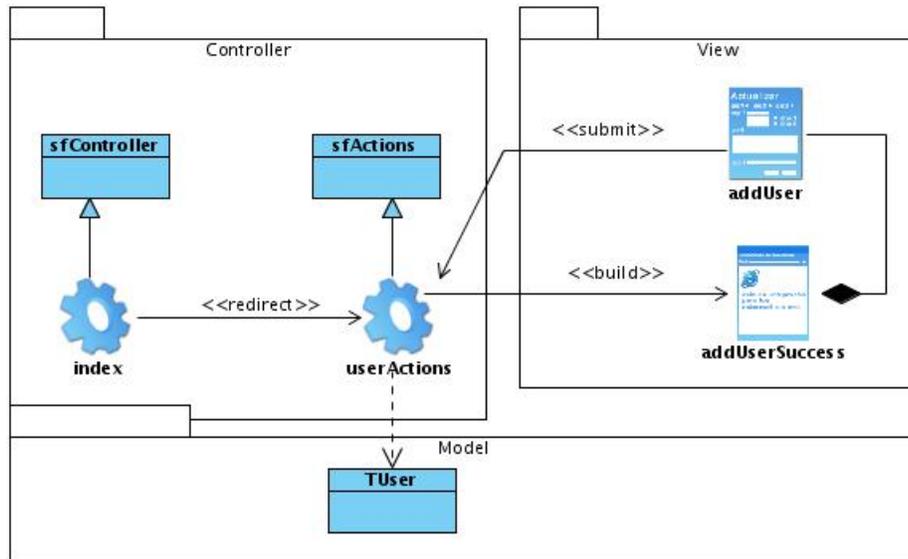


Figura D.8: DCD Gestionar Usuario Escenario Adicionar Usuario

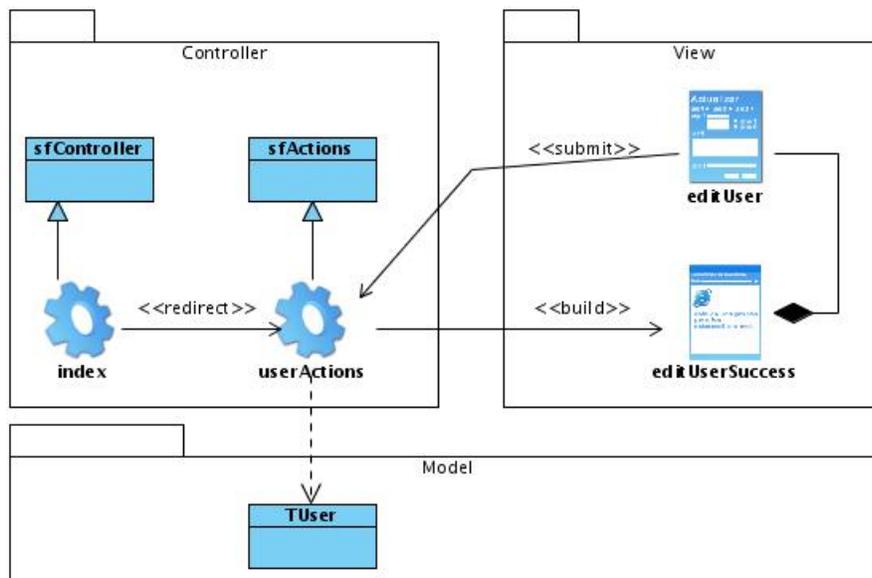


Figura D.9: DCD Gestionar Usuario Escenario Editar Información Personal de Usuario

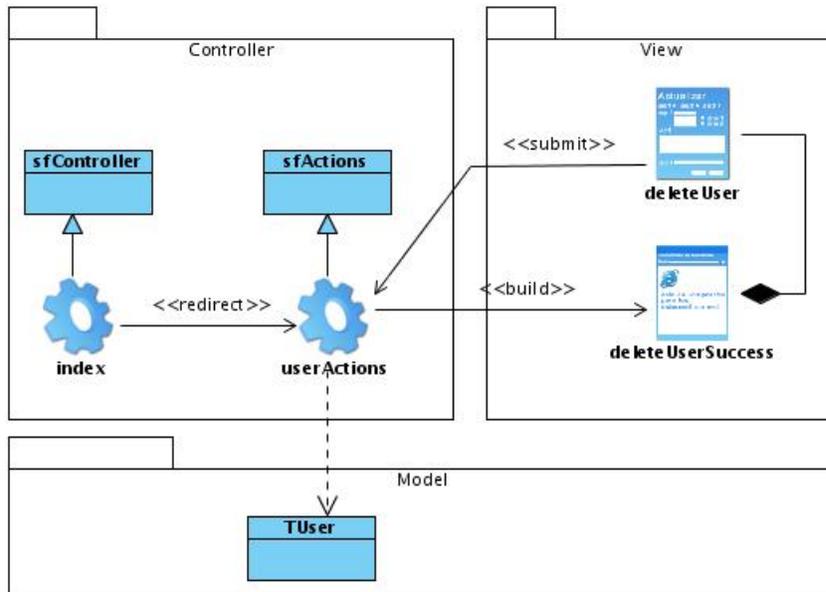


Figura D.10: DCD Gestionar Usuario Escenario Eliminar Usuario

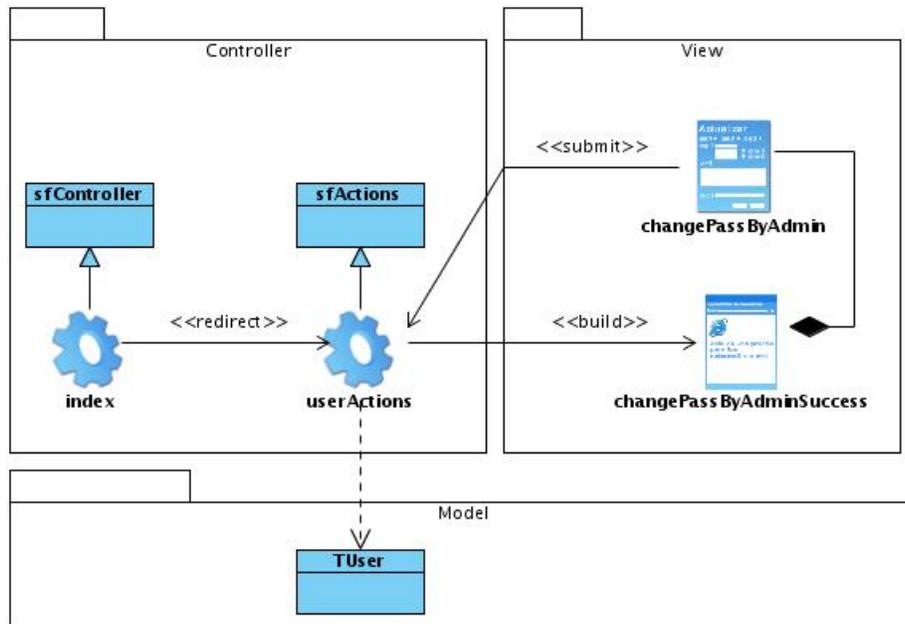


Figura D.11: DCD Gestionar Usuario Escenario Cambiar Contraseña de Usuario

## Módulo Reporte

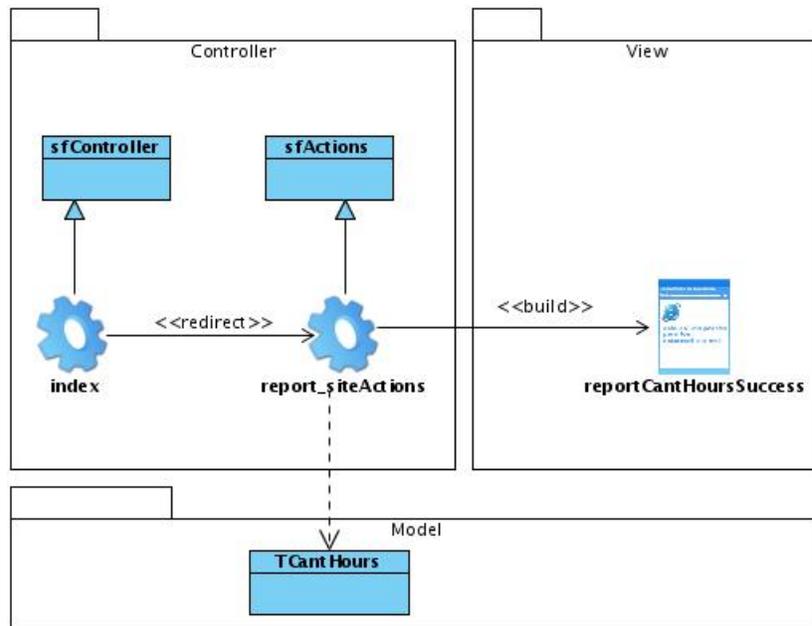


Figura D.12: DCD Mostrar Reportes Sección Mostrar Visitantes por horas

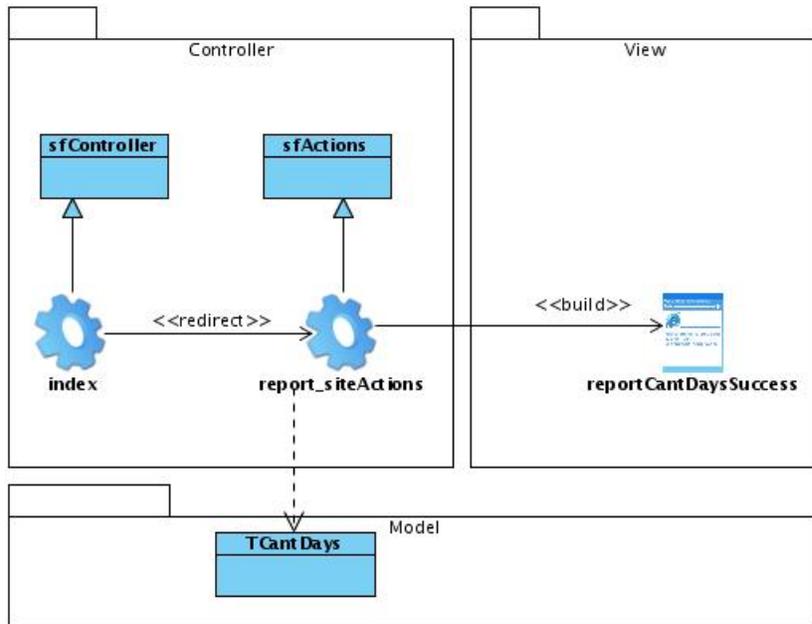


Figura D.13: DCD Mostrar Reportes Sección Mostrar Visitantes por días

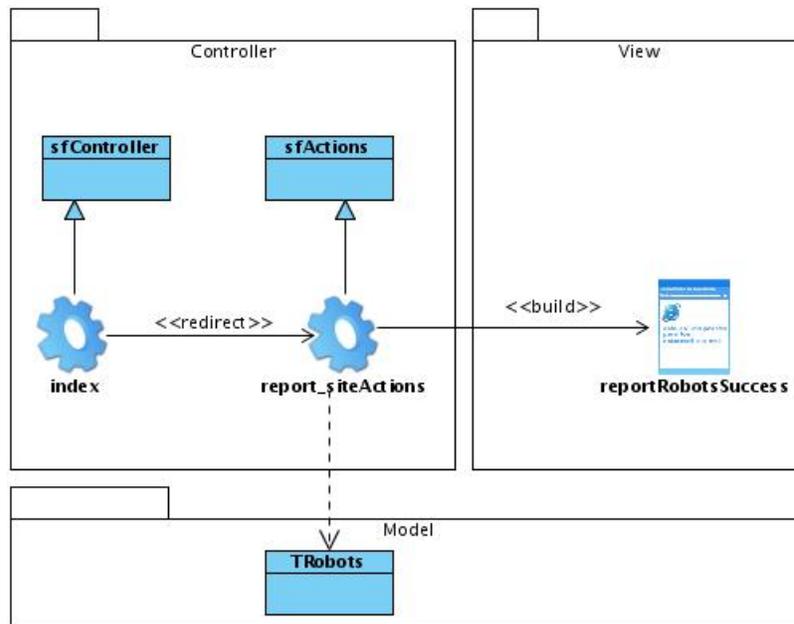


Figura D.14: DCD Mostrar Reportes Sección Mostrar robots

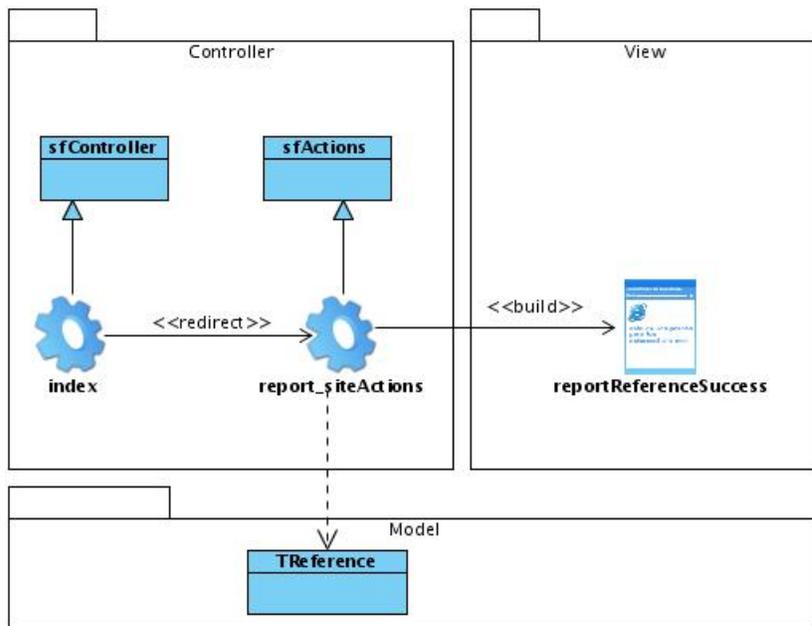


Figura D.15: DCD Mostrar Reportes Sección Mostrar Referencias

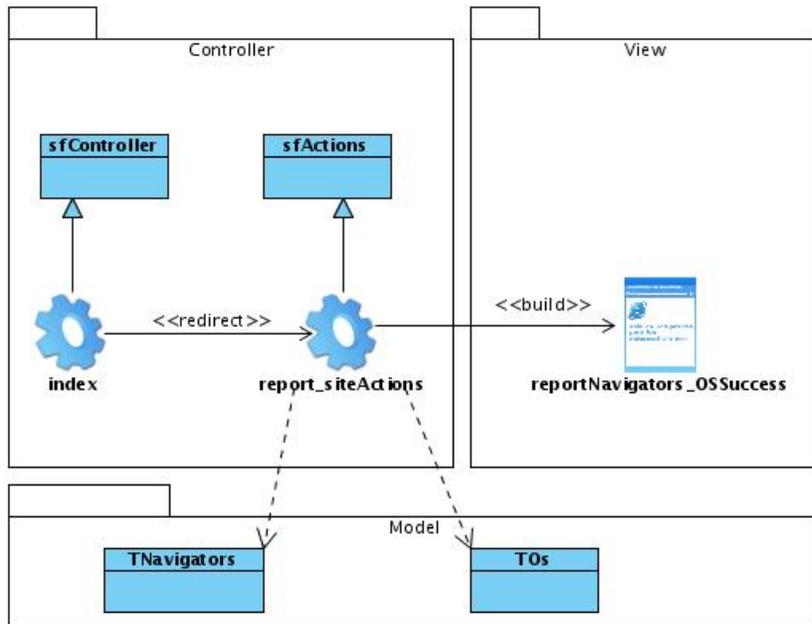


Figura D.16: DCD Mostrar Reportes Sección Mostrar Principales navegadores y sistemas operativos utilizados

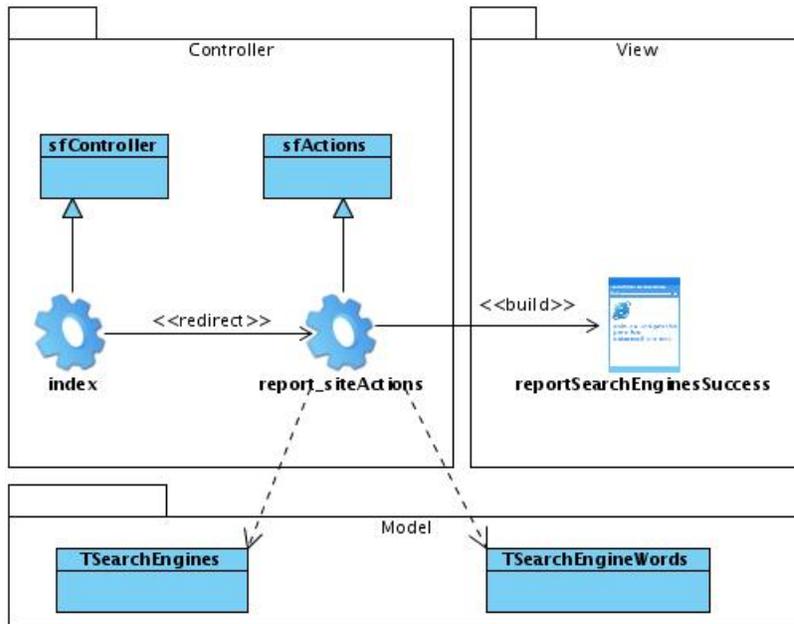


Figura D.17: DCD Mostrar Reportes Sección Mostrar principales buscadores por donde acceden los visitantes

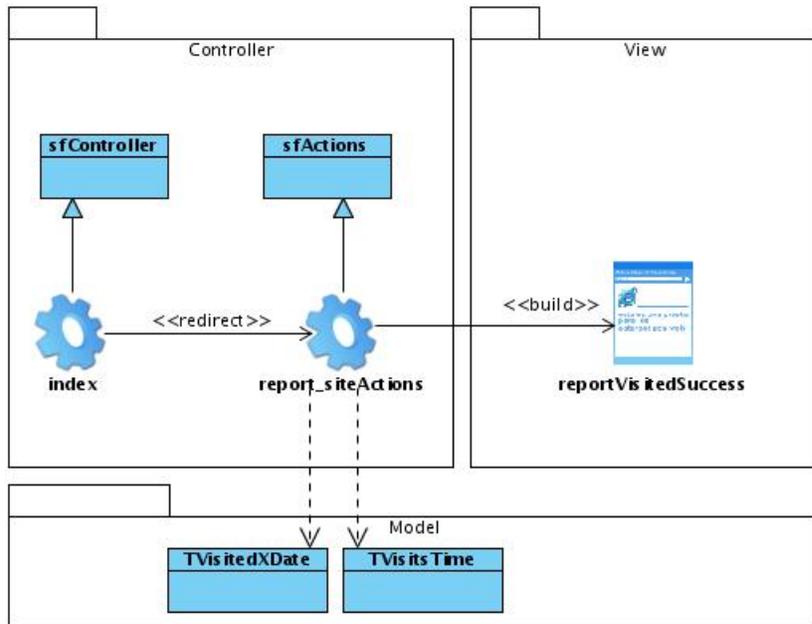


Figura D.18: DCD Mostrar Reportes Sección Mostrar Páginas Vistas por visitas(cambiar a visitas)

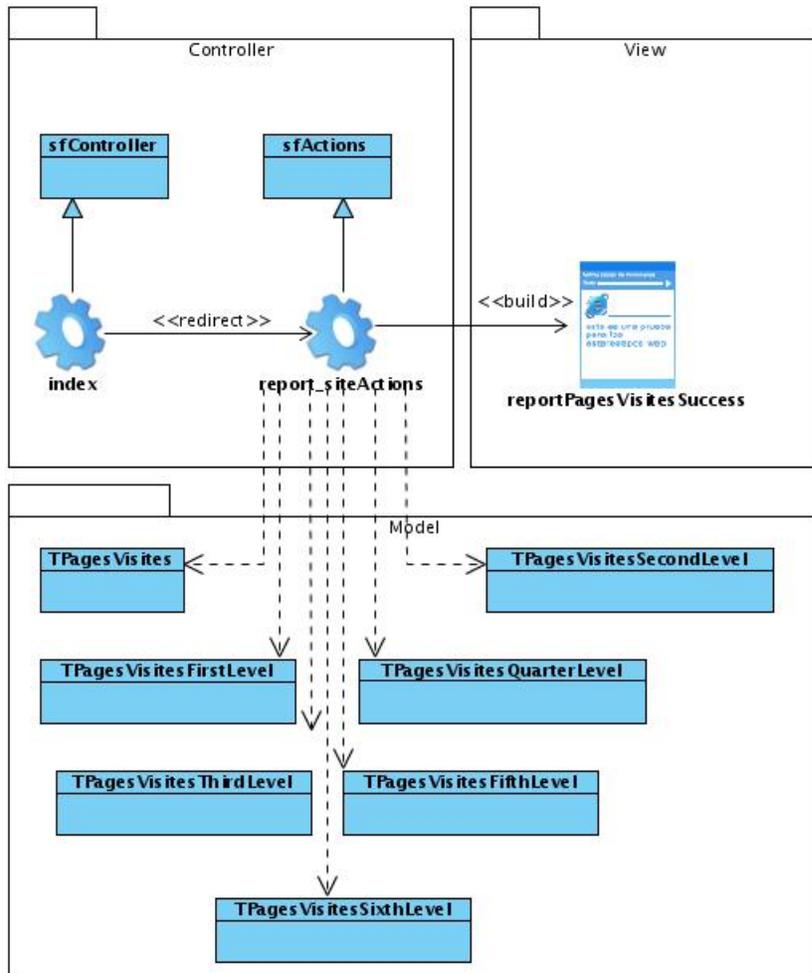


Figura D.19: DCD Mostrar Reportes Sección Mostrar Páginas Vistas

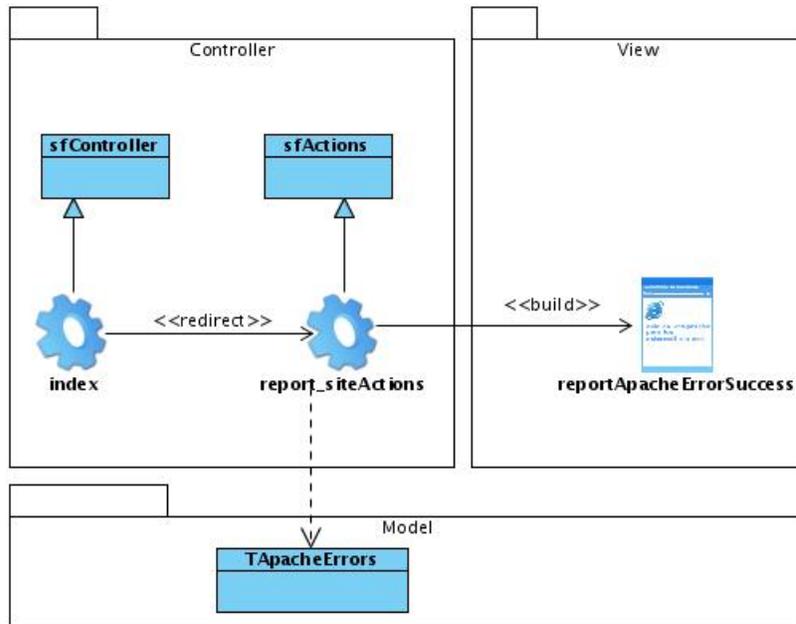


Figura D.20: DCD Mostrar Reportes Sección Mostrar errores de apache

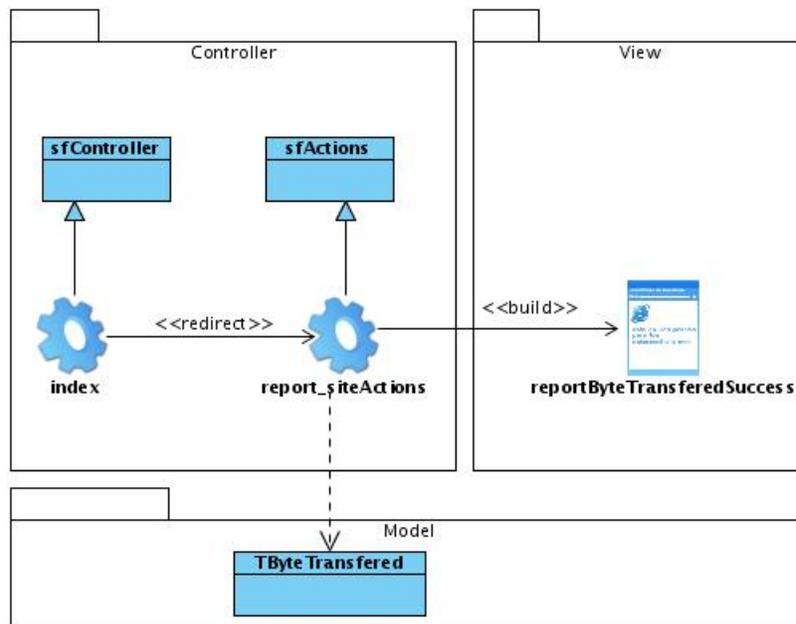


Figura D.21: DCD Mostrar Reportes Sección Mostrar bytes transferidos



## Anexo E

# Diagramas de Interacción

### Módulo Usuario

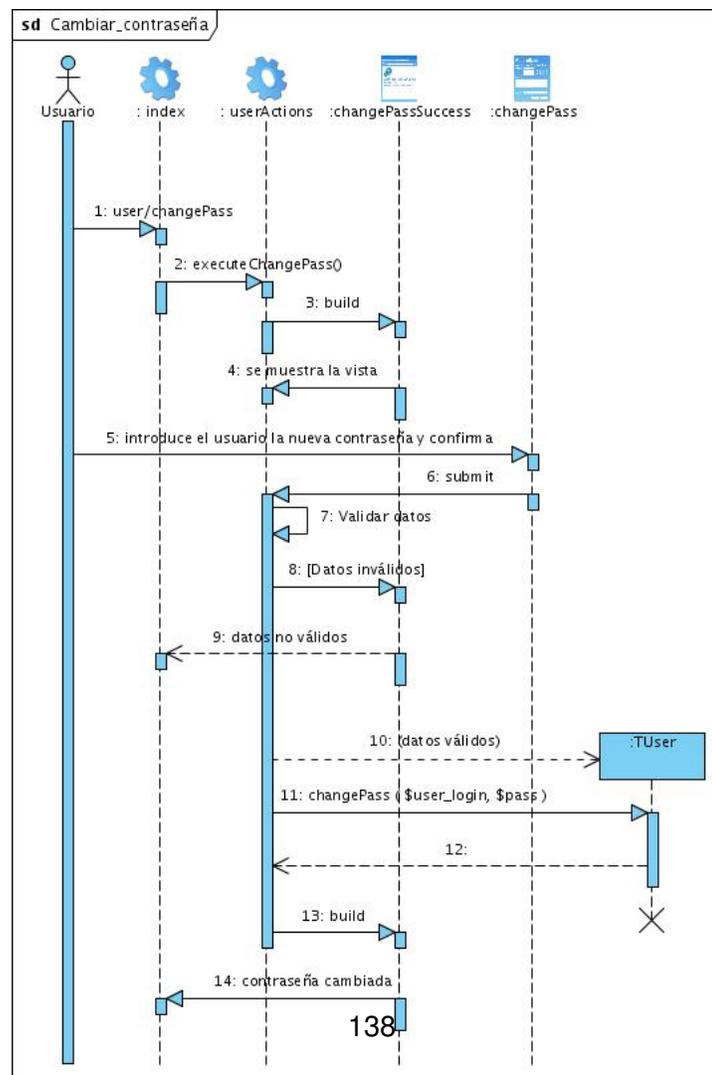


Figura E.1: Cambiar contraseña

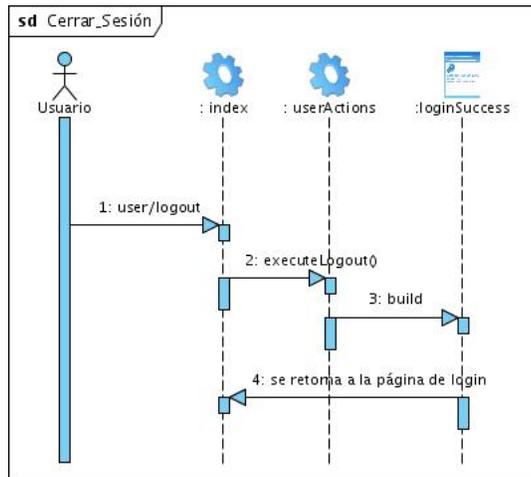


Figura E.2: Cerrar Sesión

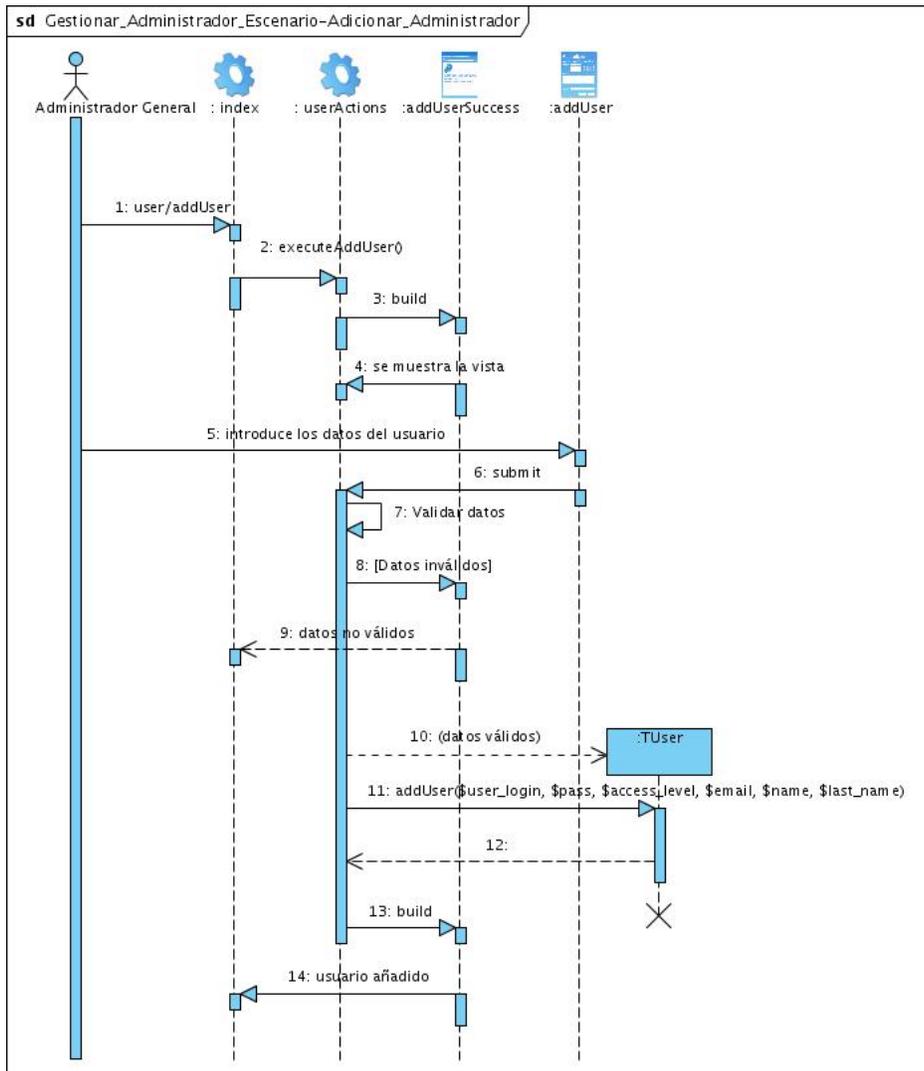


Figura E.3: Gestionar Administrador Escenario Adicionar Administrador

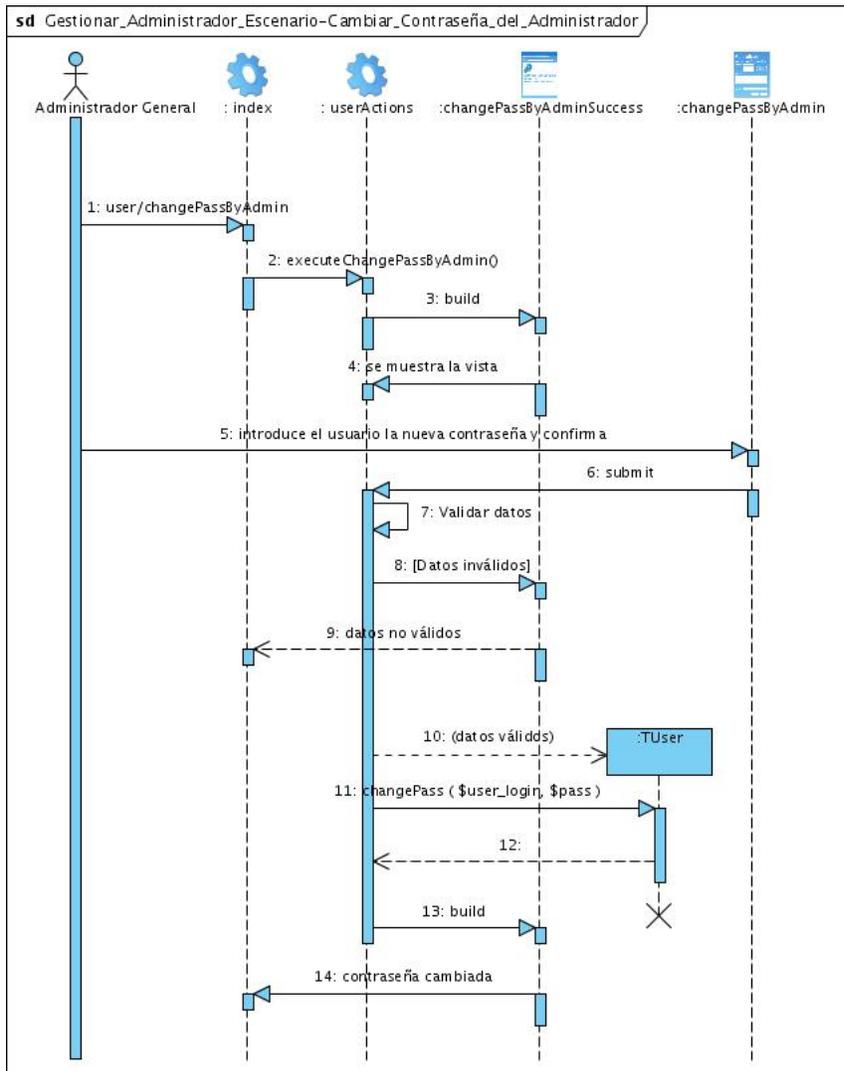


Figura E.4: Gestionar Administrador Escenario Cambiar Contraseña del Administrador

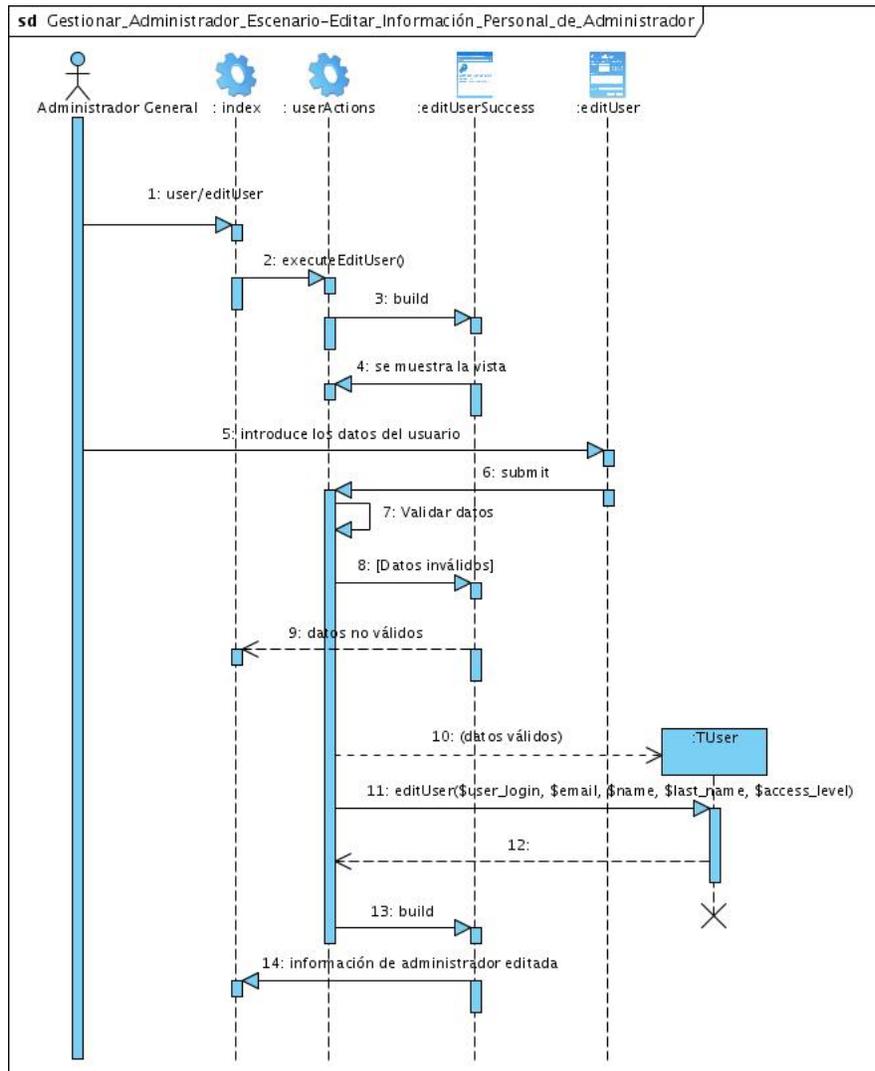


Figura E.5: Gestionar Administrador Escenario Editar Información Personal de Administrador

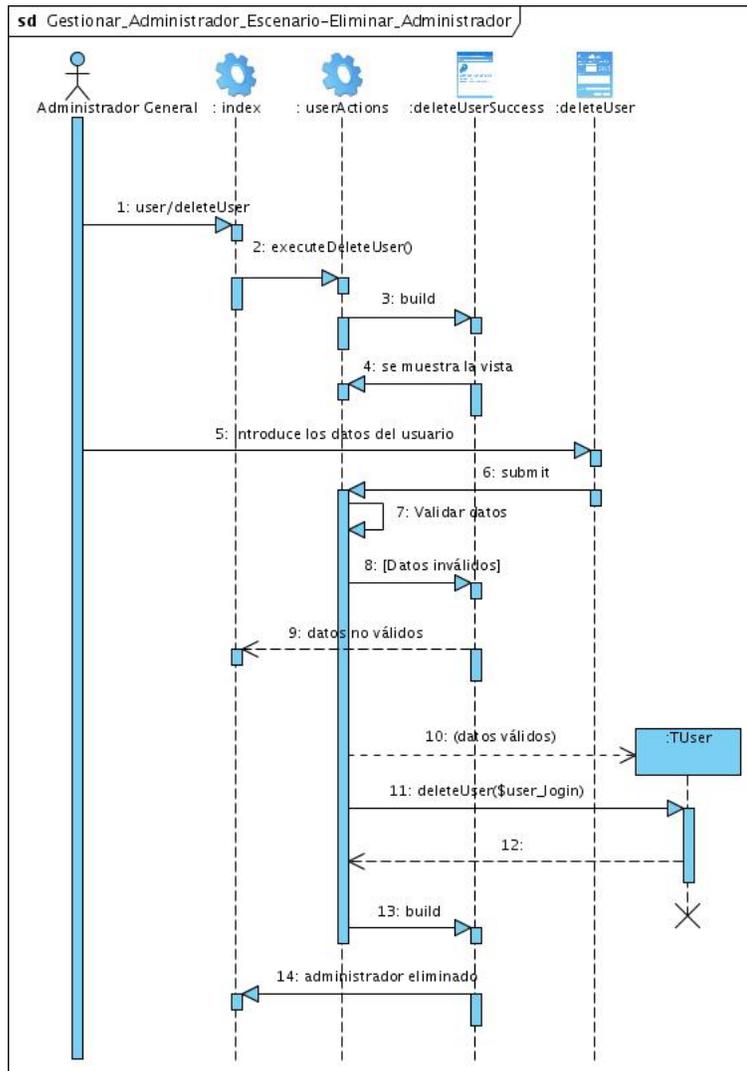


Figura E.6: Gestionar Administrador Escenario Eliminar Administrador

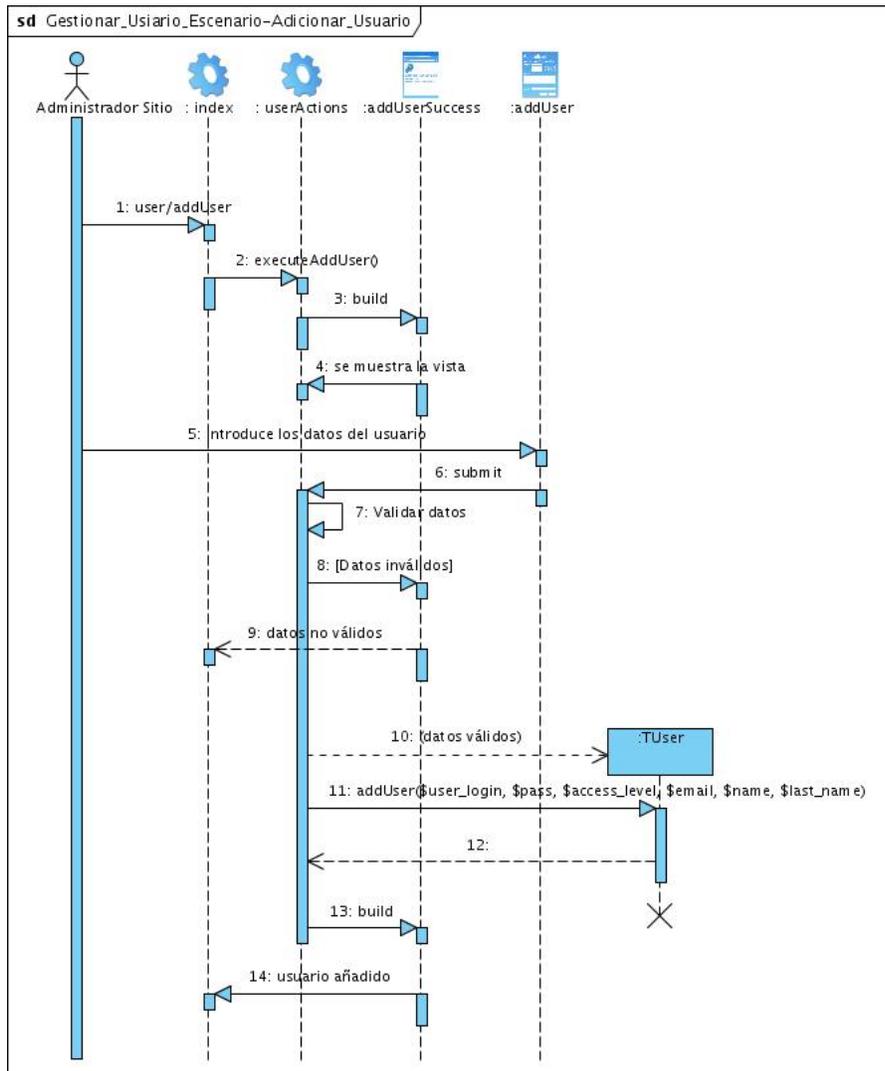


Figura E.7: Gestionar Usuario Escenario Adicionar Usuario

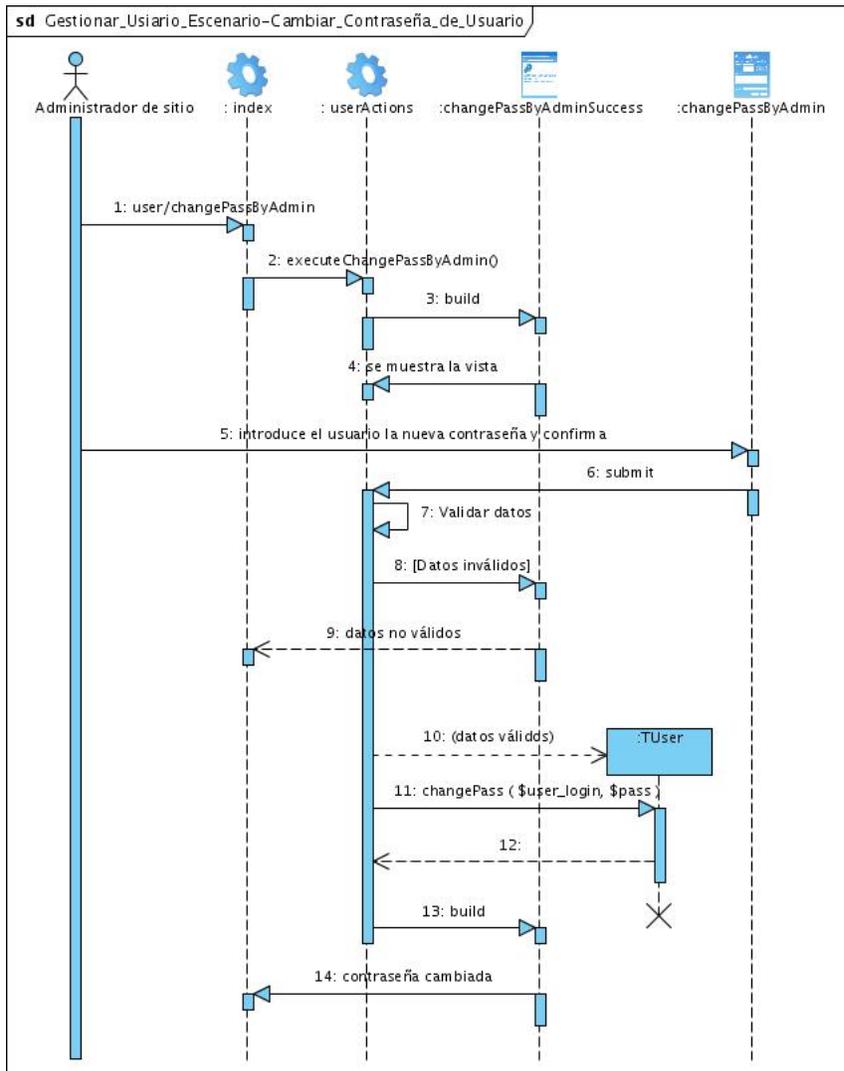


Figura E.8: Gestionar Usuario Escenario Cambiar Contraseña de Usuario

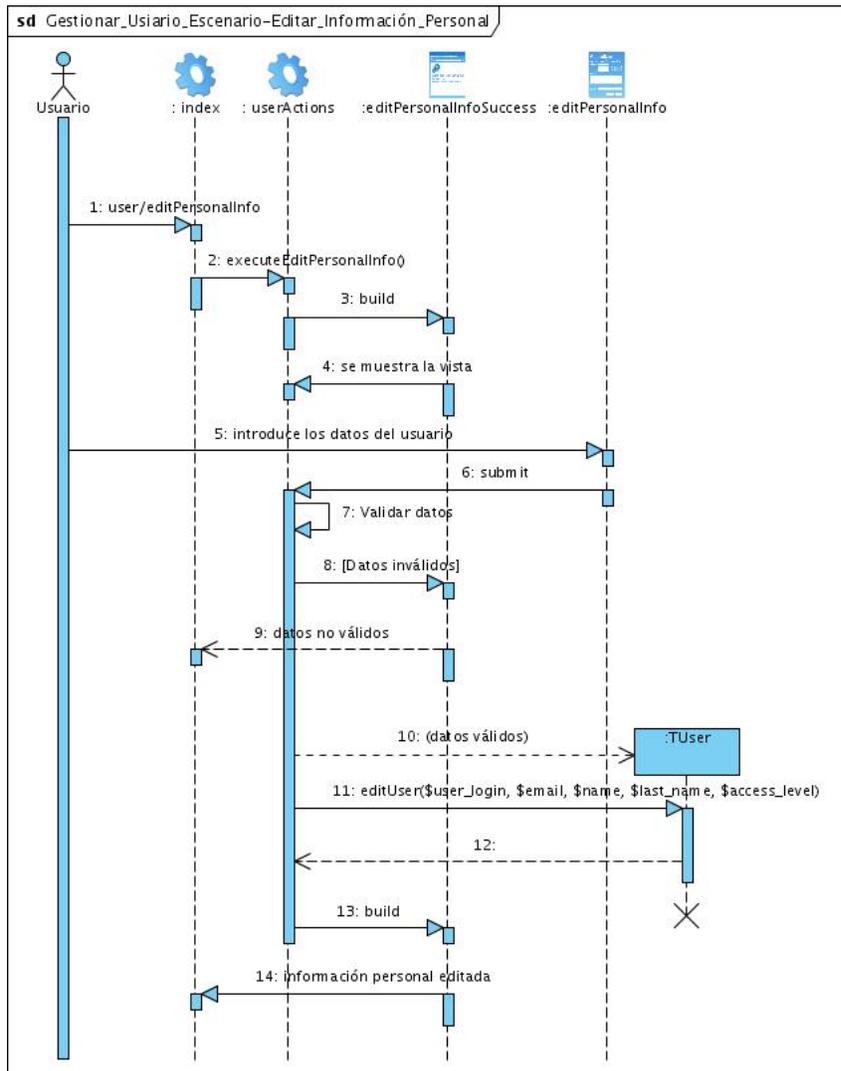


Figura E.9: Gestionar Usuario Escenario Editar Información Personal

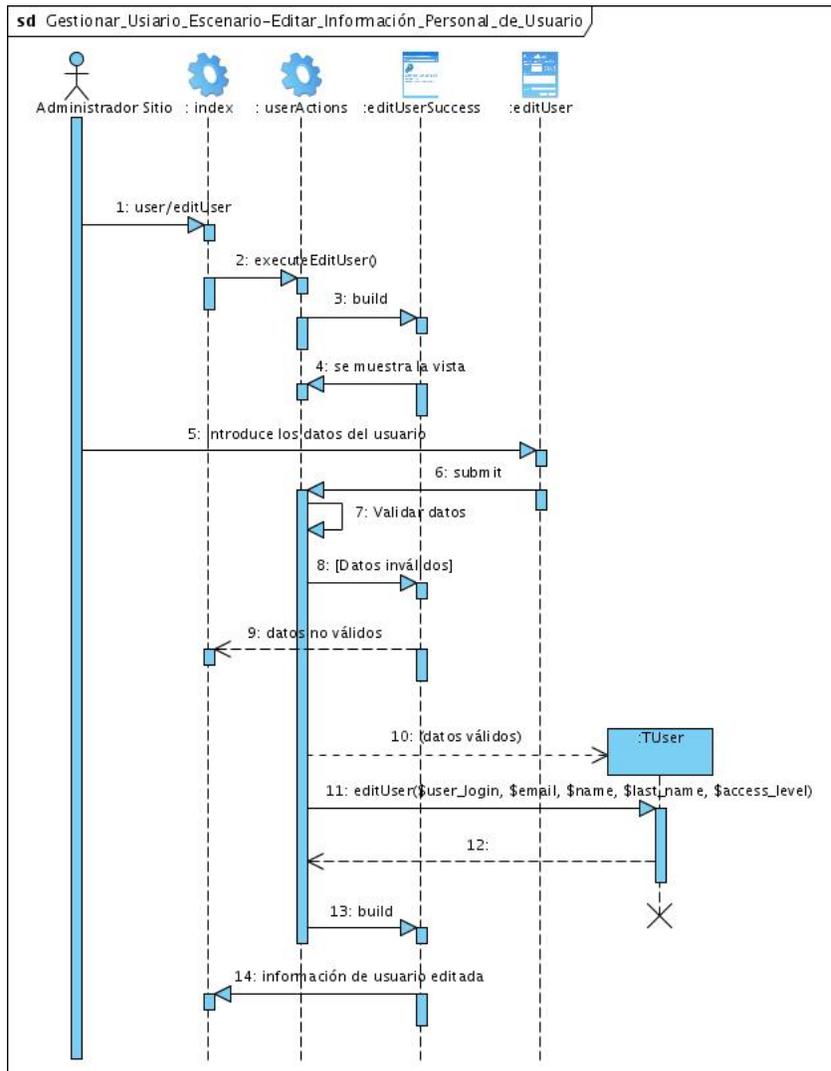


Figura E.10: Gestionar Usuario Escenario Editar Información Personal de Usuario

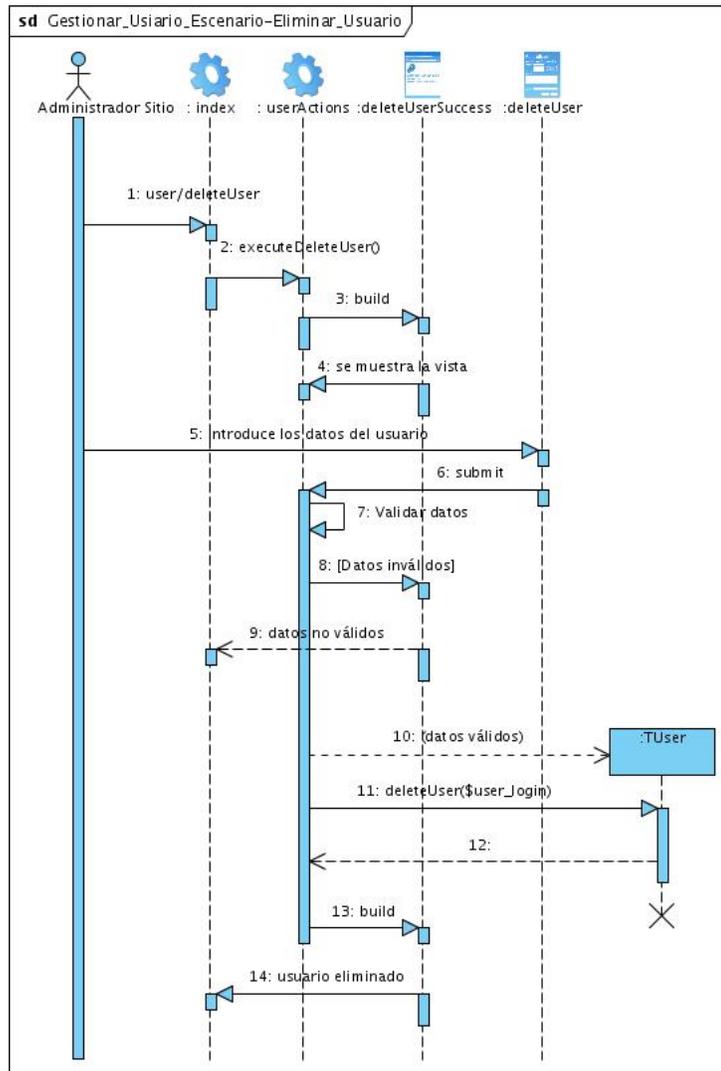


Figura E.11: Gestionar Usuario Escenario Eliminar Usuario

## Módulo Reporte

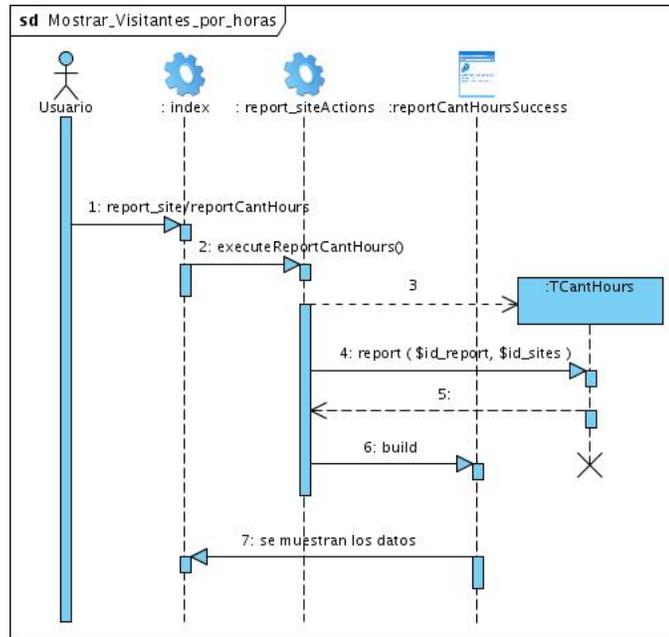


Figura E.12: Mostrar Visitantes por Horas

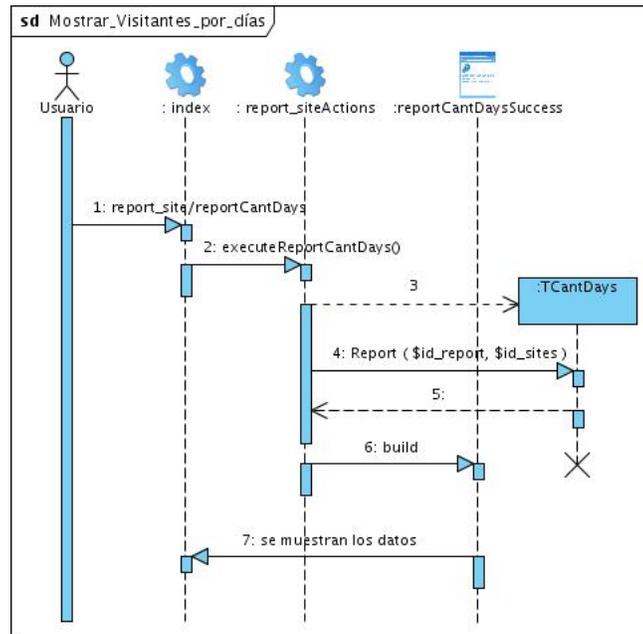


Figura E.13: Mostrar Visitantes por Días

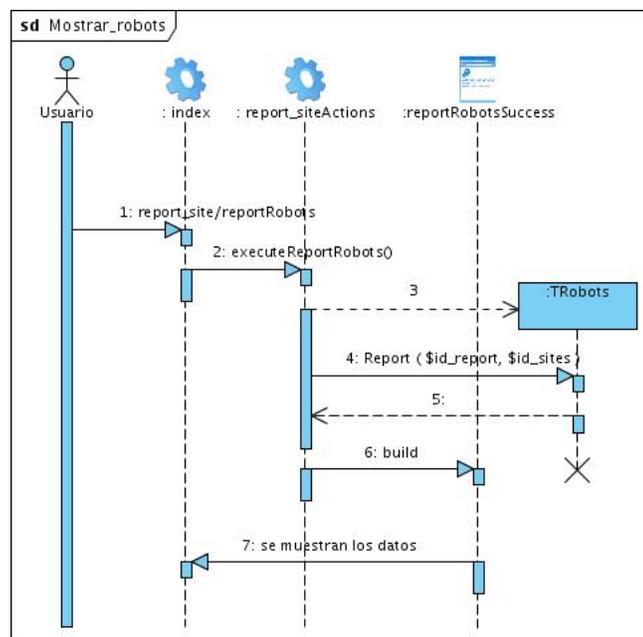


Figura E.14: Mostrar Robots

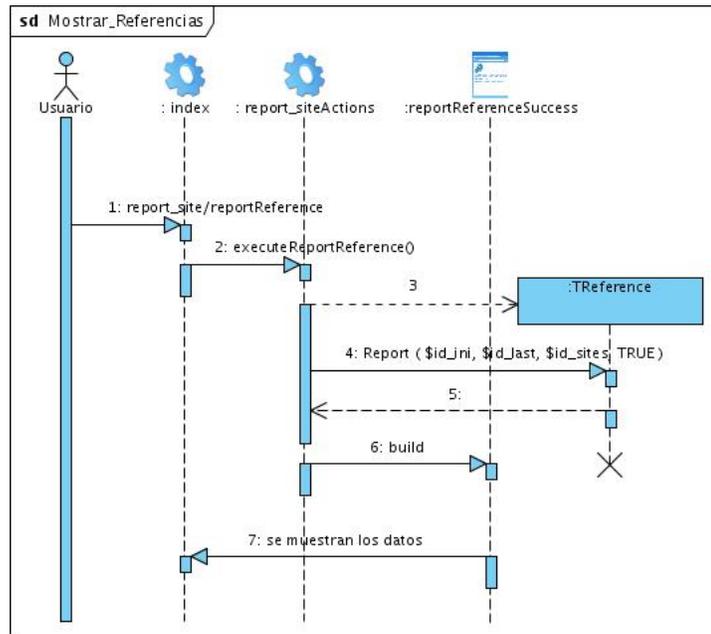


Figura E.15: Mostrar Referencias

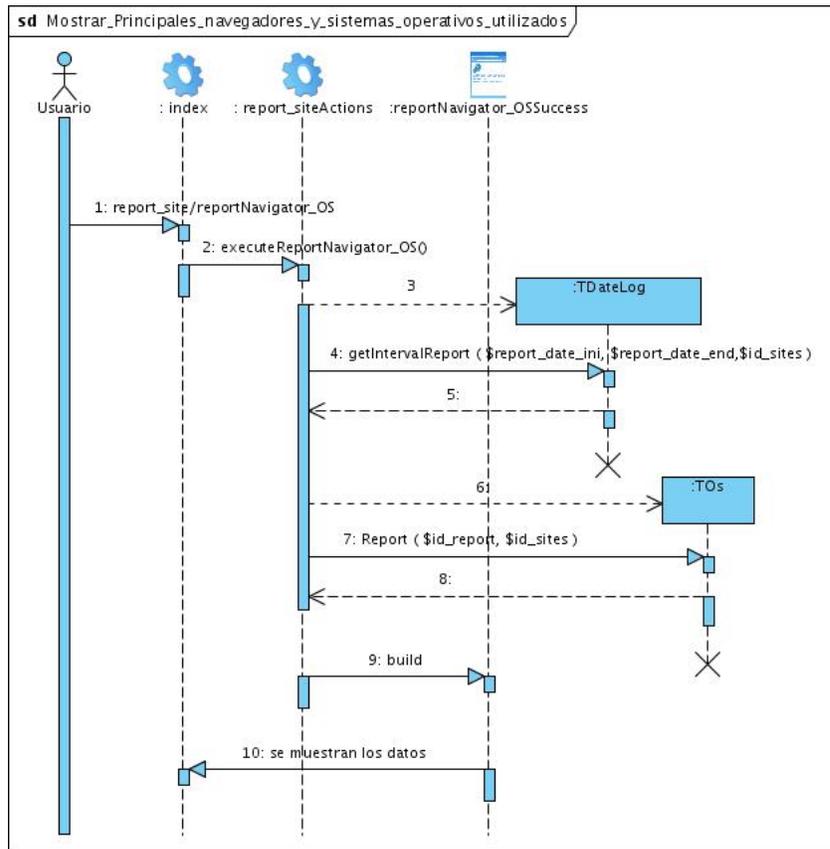


Figura E.16: Mostrar Principales Navegadores y Sistemas Operativos utilizados

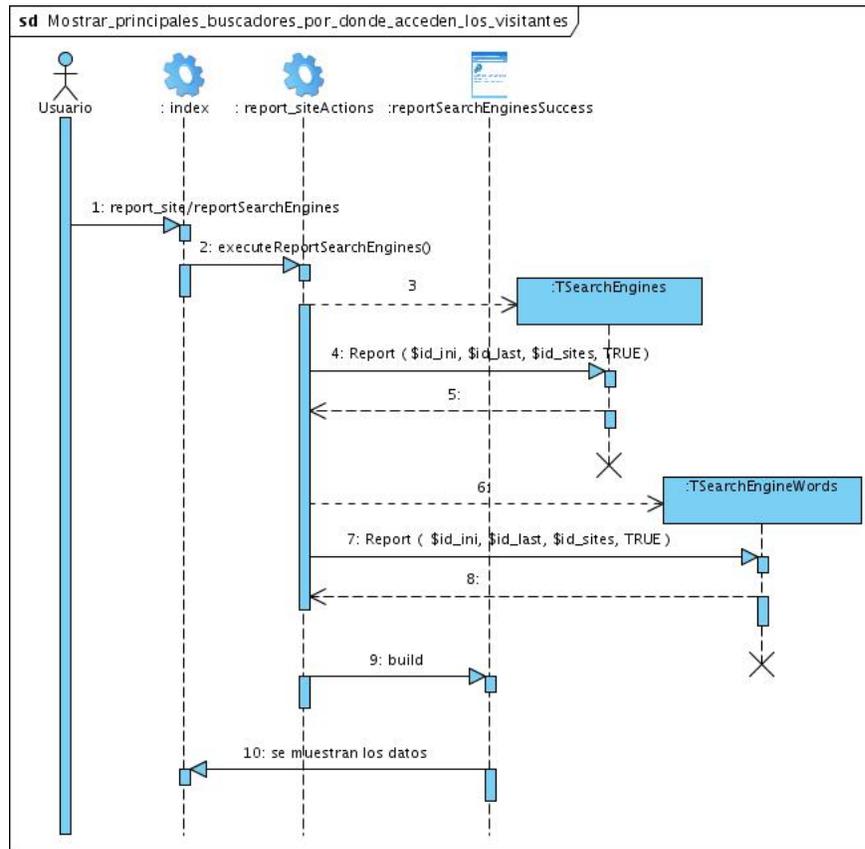


Figura E.17: Mostrar Principales Buscadores por donde acceden los visitantes

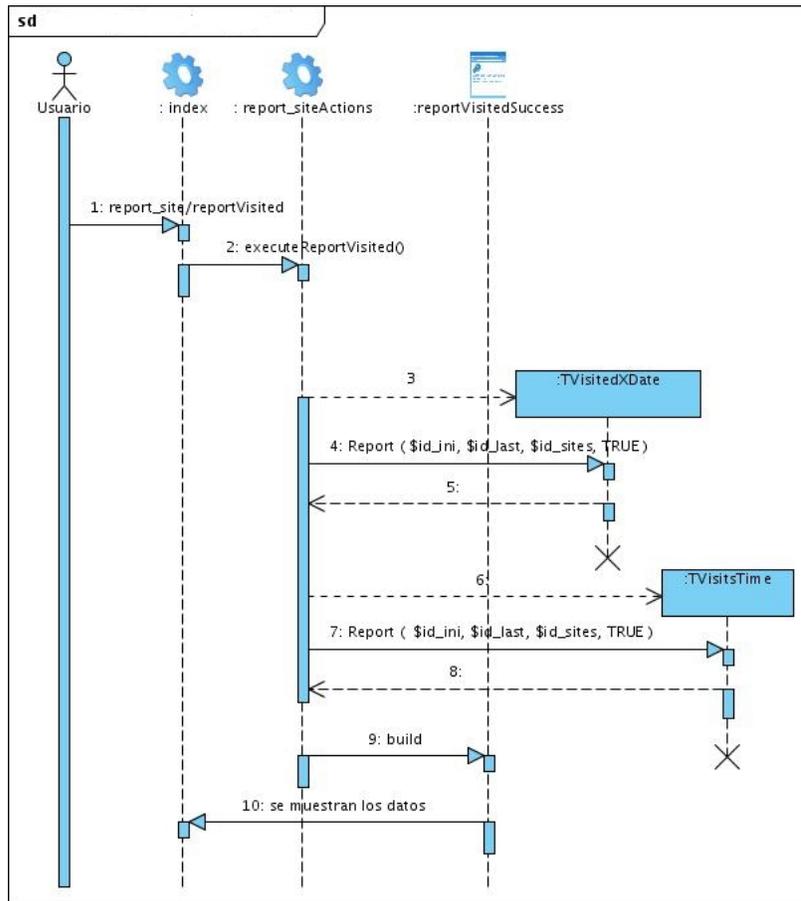


Figura E.18: Mostrar las visitas

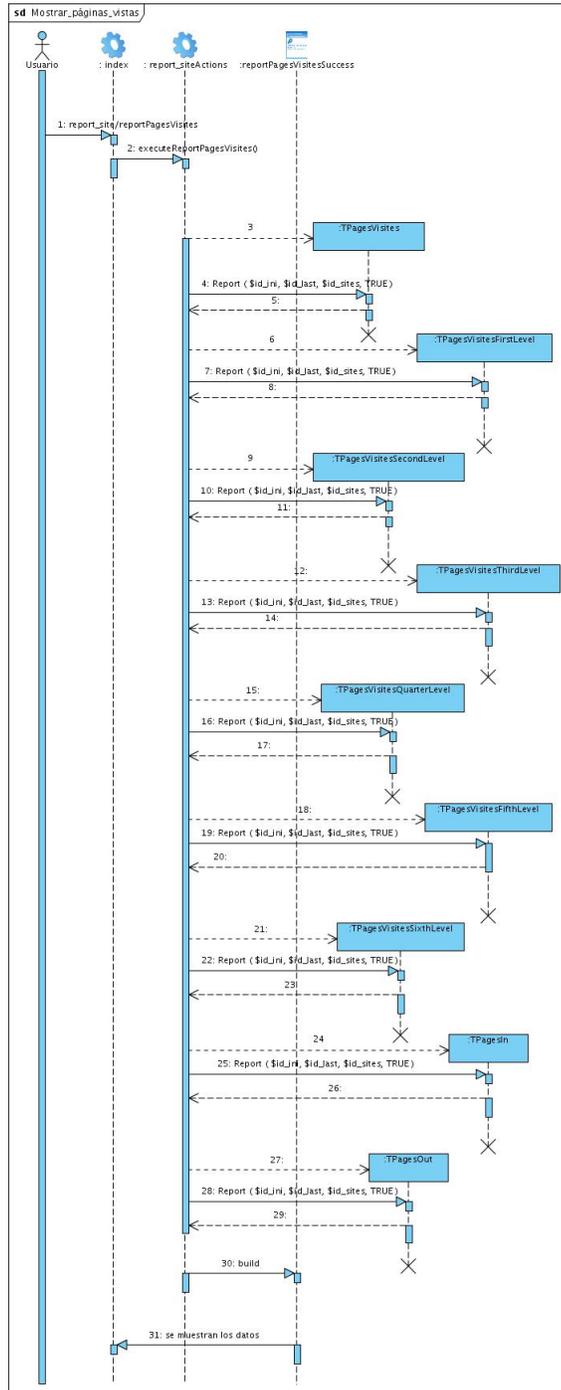


Figura E.19: Mostrar Páginas Vistas

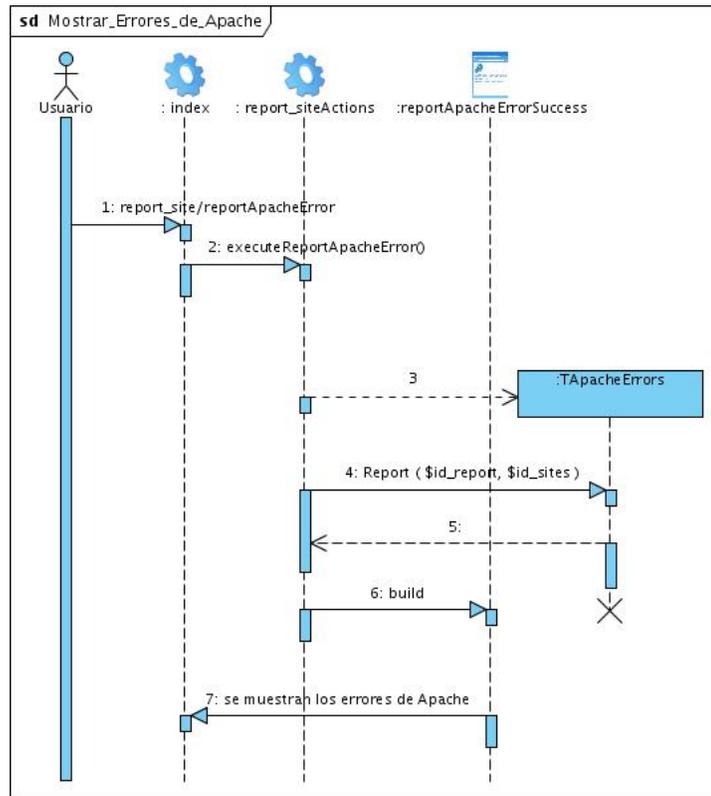


Figura E.20: Mostrar Errores de Apache

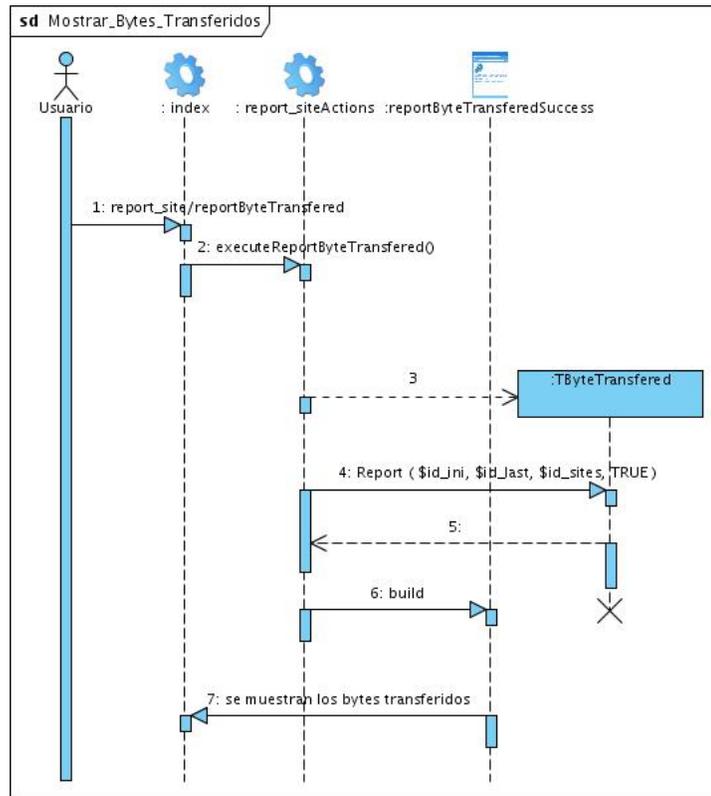
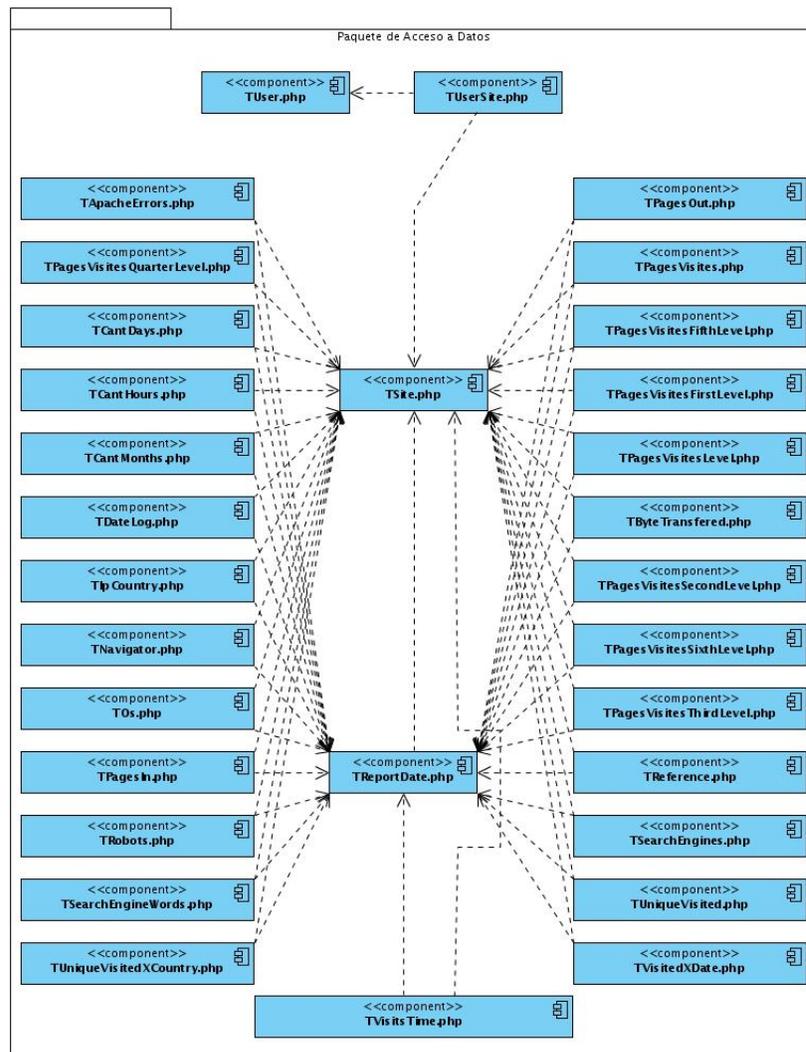


Figura E.21: Mostrar Bytes Transferidos



## Anexo F

# Diagramas de Componentes



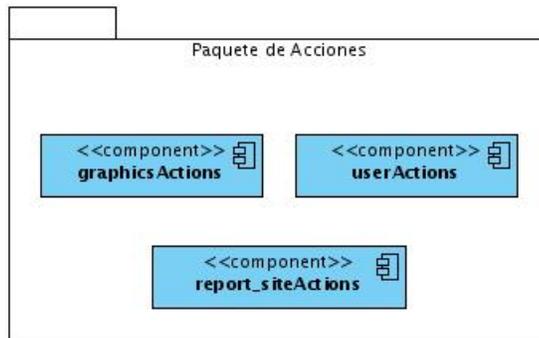


Figura F.2: Diagrama de Componente Acciones.



Figura F.3: Diagrama de Componente Plantillas.

# Glosario de términos

---

<b>UCI</b>	Son las siglas de Universidad de las Ciencias Informáticas la cual fue creada, por el comandante en jefe de la Revolución cubana, al calor de la batalla de ideas; en el año 2002.
<b>AIRES</b>	Analizadores Inteligentes de Registros de Servidores
<b>Airesweb</b>	Analizador Inteligente de Registros de Servidores Web
<b>APIs</b>	Interfaz de aplicación
<b>CIPRE</b>	Centro de Información para la Prensa
<b>IW</b>	Interfaz Web
<b>GUI</b>	Interfaz Gráfica de Usuario
<b>GUIs</b>	Interfaces Gráficas de Usuario
<b>UI</b>	Interfaz de Usuario
<b>IDE</b>	Entorno de Desarrollo Integrado
<b>RUP</b>	Rational Unified Process
<b>UML</b>	Unified Modeling Language
<b>OMG</b>	Object Management Group
<b>SGBD</b>	Sistema Gestor de Base de Datos

<b>BD</b>	Base de Datos
<b>SQL</b>	Structured Query Language
<b>IIS</b>	Internet Information Server
<b>CGI</b>	<i>Common Gateway Interface</i> . Estándar que permite el intercambio de información entre el servidor y un programa externo al servidor. Un programa CGI es un programa preparado para recibir y enviar datos desde y hacia un servidor Web según este estándar. Normalmente se programan en C o en Perl, aunque se puede usar cualquier lenguaje de propósito general.
<b>JSP</b>	<i>Java Server Page</i> . Tecnología de <i>Sun Microsystems</i> que permite crear páginas web dinámicas en el servidor. Equivale a la tecnología ASP de <i>Microsoft</i> . Se programan en <i>Java</i> .
<b>ASP</b>	<i>Active Server Pages</i> . Tecnología propietaria de MICROSOFT que permite crear páginas Web dinámicas en el Servidor. Desarrollada con el objetivo de sustituir la tecnología CGI, ofrece una serie de características que facilitan la programación de aplicaciones Web.
<b>IDC</b>	<i>Internet Database Connector</i> . Conector de bases de datos de Internet. Tecnología propietaria de MICROSOFT que permite generar páginas Web dinámicas a partir de la información almacenada en la base de datos. Es el precursor de ASP.
<b>CSS</b>	<i>Cascading Style Sheets</i> . Tecnología empleada en la creación de páginas Web, que permite un mayor control sobre el lenguaje HTML. Permite hojas de estilo que definen como cada elemento, se tiene que mostrar. CSS ha sido desarrollado por W3C.
<b>HTML</b>	<i>HyperText Markup Language</i> . Lenguaje compuesto de una serie de etiquetas o marcas que permiten definir el contenido y la apariencia de las páginas Web. Aunque se basa en el estándar SGML, no se puede considerar que sea un subconjunto de él. Existen cientos de etiquetas con diferentes atributos. <i>World Wide Web Consortium</i> . Consorcio internacional de compañías y organizaciones involucradas en el desarrollo de Internet y en especial de la WWW. Su propósito es desarrollar estándares y "poner orden" en Internet se encarga de su estandarización. El futuro sustituto de HTML es XHTML.

- DHTML** *Dynamic HTML*. Conjunto de extensiones a HTML que permiten modificar el contenido de una página Web en el cliente sin necesidad de establecer una conexión con el servidor. Se basa en el uso de DOM para acceder al contenido de la página.
- DOM** *Document Object Model*. Especificación que define como se puede acceder a los objetos de un documento HTML (ventanas, imágenes y formularios) a través de un lenguaje script. Básicamente define una jerarquía de objetos.
- SGML** *Standard Generalized Markup Language*. Lenguaje que permite organizar y etiquetar los distintos elementos que componen un documento. Se emplea para manejar grandes documentos que sufren constantes revisiones y se imprimen en distintos formatos de idioma.
- XHTML** *Extensible HyperText Markup Language*. HTML escrito según las normas que marca XML. Por tanto, se trata de una aplicación concreta de XML y no tienen que confundirse entre si.
- XML** *Extensible Markup Language*. Metalenguaje de etiquetado en SGML. Diseñado específicamente para la WWW por W3C. Permite que un usuario diseñe sus propias etiquetas, con sus atributos y reglas de construcción de documentos.
- WWW** *World Wide Web*. También conocida como “la Web“ ó ”la Red“. Sistema mundial de servidores Web conectados a Internet (No todos los ordenadores conectados a Internet forman parte de la WWW).
- W3C** *World Wide Web Consortium*. Consorcio internacional de compañías y organizaciones involucradas en el desarrollo de Internet y en especial de la WWW. Su propósito es desarrollar estándares y ”poner orden“ en Internet.
- HTTP** *HyperText Transfer Protocol*. Es el protocolo que emplea la WWW. Define como se tienen que crear y enviar los mensajes y que opciones debe tener el servidor y el navegador en respuesta a un comando.
- HTTPS** *HyperText Transfer Protocol Security*.