

Universidad de las Ciencias Informáticas

Facultad 10



Interfaz Web para la Gestión del Sistema Airesweb

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

Autores: Adalberto Ladrón de Guevara Zamora.

Javier Ernesto Domínguez Viltres.

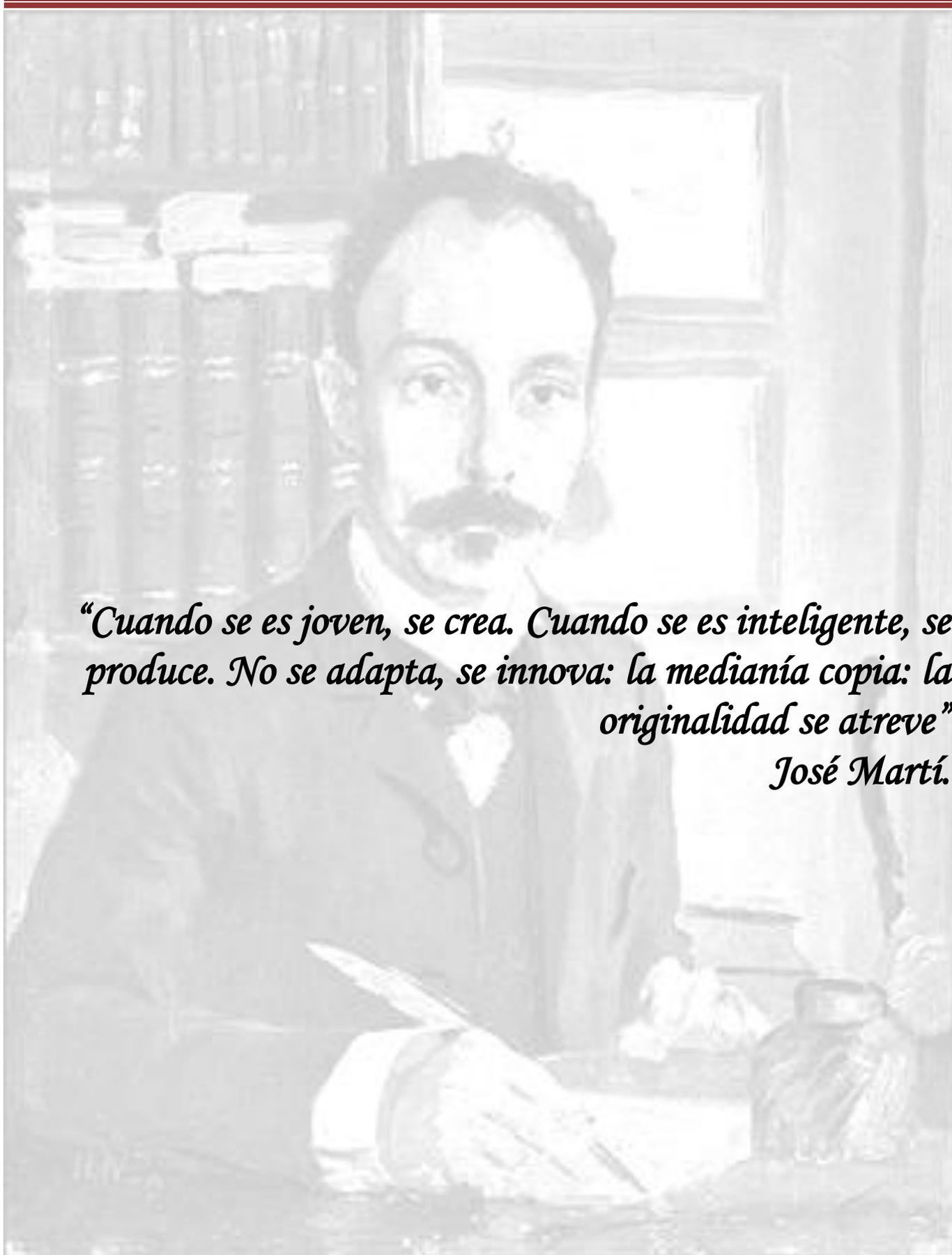
Tutor(es): Ing. Adrian Gallardo Palay.

MsC. Clara Gisela Scot Bigñott.

Ciudad de La Habana, Cuba.

Junio 2010

Año 52 del Triunfo de la Revolución Cubana



“Cuando se es joven, se crea. Cuando se es inteligente, se produce. No se adapta, se innova: la medianía copia: la originalidad se atreve”

José Martí.

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 firmo la presente a los ____ días del mes de _____ del año 2010.

Adalberto Ladrón de Guevara Zamora

Firma del Autor

Javier Ernesto Domínguez Viltres

Firma del Autor

Ing. Adrian Gallardo Palay

Firma del Tutor

Clara Gisela Scott Bigñot

Firma de la Tutora

Resumen

La toma de decisiones más eficiente, mejorar la estrategia de posicionamiento en internet, optimizar el diseño de las páginas web, hacer un adecuado seguimiento de los clientes, averiguar la procedencia y el destino de los visitantes, aumentar la fidelidad de los clientes, entre otras utilidades, son aspectos que el análisis de registros web brinda. Analizador Inteligente de Registros Web (Airesweb), es un software de este tipo. Esta nueva versión del sistema, desarrollada en la Universidad de las Ciencias Informáticas (UCI), genera estadísticas a partir de registros generados por un código script insertado en las páginas web que se quieran monitorizar y cuenta con un núcleo más potente y una base de datos mejor estructurada.

El presente trabajo de diploma se centra en la necesidad de crear una Interfaz Web (IW) que permita gestionar la información almacenada por el sistema Airesweb. Para darle cumplimiento a la siguiente investigación se realizó un estudio de las interfaces web de los principales sistemas de análisis de registros web. Se realizó un estudio de las tecnologías, herramientas y lenguajes utilizados en el desarrollo de interfaces web. Se identificaron y documentaron las funcionalidades que la solución propuesta debía cumplir. El proceso de desarrollo de software estuvo guiado por la metodología ágil SXP.

Se logró una IW funcional y eficiente, que permite gestionar el contenido del análisis de la información procesada por el sistema Airesweb.

Palabras claves: Airesweb, análisis, analítica web, archivo de registros web, estadísticas, Interfaz Web, internet, Symfony.

Índice de Contenido

Resumen	III
Índice de Tablas	VII
Introducción	8
Capítulo 1. Fundamentación Teórica	12
1.1. Trabajos Similares.....	12
1.1.1. Ámbito Internacional.....	12
1.1.2. Ámbito Nacional.....	16
1.2. Interfaz.....	16
1.3. Interfaces Gráficas de Usuario.....	17
1.4. Interfaz Web.....	18
1.4.1. Concepto.....	18
1.4.2. Evolución de las interfaces web de los analizadores de reportes de registros de servidores web.....	18
1.5. Características de la IW a desarrollar.....	19
1.6. Arquitectura.....	19
1.6.1. Modelo Cliente-Servidor.....	20
1.7. Lenguajes, tecnologías y herramientas de desarrollo utilizadas.....	21
1.7.1. Lenguajes de programación.....	21
1.7.2. Lenguaje de Modelado.....	26
1.7.3. Tecnologías.....	26
1.7.4. Frameworks para el lenguaje PHP.....	27
1.7.5. Servidor de Aplicaciones Web.....	32
1.7.6. Herramienta de Programación.....	33
1.7.7. Herramienta de Modelado.....	34
1.8. Metodología de Desarrollo de Software.....	34
1.9. Conclusiones Parciales.....	35
Capítulo 2. Análisis y desarrollo de la solución propuesta	37
2.1. Descripción del Problema.....	37
2.2. Solución Propuesta.....	37
2.3. Lista de Reserva del Producto.....	39

2.4. Historias de Usuario y Tareas Ingenieriles.....	42
2.5. Plan de Liberación.....	47
2.6. Implementación del MVC por Symfony.....	48
2.7. Organización de la Aplicación.....	49
2.8. Patrones de Diseño Utilizados.....	50
2.9. Validación y Tratamientos de Errores.....	51
2.10. Seguridad: Autenticación y Autorización.....	51
2.11. Diagrama de Despliegue.....	52
2.12. Diagrama de Componentes.....	52
2.13. Conclusiones parciales.....	53
Capítulo 3. Validación de la solución propuesta.....	54
3.1 Pruebas.....	54
3.1.1. Pruebas de Aceptación.....	54
3.1.2. Casos de Pruebas.....	54
3.2. Conclusiones parciales.....	55
Conclusiones.....	56
Recomendaciones.....	57
Referencias Bibliográficas.....	58
Bibliografía.....	61
Anexo A. Figuras Relacionadas.....	63
Anexo C. Diagramas de Componentes.....	69
Glosario de Términos.....	72

Índice de Figuras

Ilustración 1.1: Modelo Vista Controlador.....	21
Ilustración 1.2: Esquema de metodología SXP	35
Ilustración 2.1: Funcionamiento del patrón MVC.	49
Ilustración 2.2: Flujo de trabajo del MVC en Symfony.	49
Ilustración 2.3: Organización de la aplicación.	50
Ilustración 2.4: Diagrama de Despliegue de la aplicación.	52
Ilustración 2.5: Diagrama de Componentes general.....	53
A 1: Interfaz Web de WebTrends.....	63
A 2: Interfaz Web de WeblogExpert.	64
A 3: Interfaz Web de AwStats.	65
A 4: Interfaz Web de GoogleAnalytics.	66
A 5: Interfaz Web de Piwik.	67
A 6: Interfaz Web de Airesweb (Versión Anterior).	68
B 1: Diagrama de Componentes de Symfony.....	69
B 2: Diagrama de Componentes Acciones.	70
B 3: Diagrama de Componentes Plantillas.	71

Índice de Tablas

Tabla 2.1: Lista de Reserva del Producto (LRP).....	42
Tabla 2.2: HU Iniciar sesión.	43
Tabla 2.3: Tareas Ingenieriles para la HU Iniciar sesión.	43
Tabla 2.4: HU Insertar sesión.	44
Tabla 2.5: Tareas Ingenieriles para la HU Insertar usuario.	45
Tabla 2.6: HU Generar código javascript.	45
Tabla 2.7: Tareas Ingenieriles para la HU Generar código javascript.....	46
Tabla 2.8: HU Verificar si el sitio envía datos.....	46
Tabla 2.9: Tareas Ingenieriles para la HU Verificar si el sitio envía datos.	47
Tabla 2.10 : Plan de Liberación.	48

Introducción

Internet es una red de computadoras que comparten información unas con otras alrededor de todo el mundo. Una forma moderna y rápida de estos intercambios de datos se realiza mediante los sitios web, que no son más que sitios (localizaciones) que contienen documentos (páginas web) organizados jerárquicamente y que pueden contener una combinación de gráficos, textos, videos, audio y otros materiales estáticos o dinámicos. Cada día que pasa el uso de internet se hace más indispensable en los trabajos personales y de empresas de las personas, o simplemente para interactuar a grandes distancias. Con este aumento del uso de internet ha crecido también el flujo de los datos y la competencia e inversión de las empresas en diseño, programación y promoción.

Es imprescindible poner cuidado en el mantenimiento del sitio web, analizarlo y estudiarlo. Las estadísticas de tráfico de la web son la primera y principal fuente de información cuando se mida el rendimiento. Es importante ver de dónde vienen las visitas, qué páginas son las que usan para llegar al sitio que se está analizando, qué visitan cuando están dentro, en qué lugares de la web se detienen más y qué recursos o herramientas no utilizan nunca. Aplicar mejoras constantes sobre la base de estos análisis propicia que la imagen o información sea bien recibida por el visitante, posible cliente, y que se cumplan los objetivos perseguidos por el sitio web.

Para dar solución al problema del seguimiento y revisión de estadísticas web existen múltiples herramientas que analizan los registros web. Interpretar estos registros de forma bruta es de difícil comprensión, de ahí la necesidad de usar estas herramientas para llevarlos a un formato fácilmente interpretable.

Una Interfaz Web es un caso particular de las interfaces de manipulación directa, también llamadas GUIs por sus siglas en inglés, estas son prácticamente universales. Las interfaces que utilizan ventanas, iconos y menús se han convertido en estándar en los materiales computacionales [1].

Las GUIs son una especialización de las interfaces de usuario. Una interfaz de usuario es el vínculo entre el usuario y el programa de computadora. 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 es posible que el programa haga lo que el usuario quiere hacer. Un programa muy poderoso con una interfaz pobremente elaborada tiene poco valor para un usuario no experto [3].

Las interfaces web 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.

A petición del Centro de Información para la Prensa (CIPRE) se desarrolló en la facultad diez de la Universidad de las Ciencias Informáticas (UCI) un software de análisis de tráfico web que lleva por nombre Airesweb, sistema que en sus inicios realizaba un análisis inteligente de los registros web, estaba diseñado para instalarse en un servidor Debian GNU/Linux, contaba con una base de datos donde se almacenaba la información generada por el análisis de los registros web y con una IW para la gestión de la información procesada por el sistema.

Debido al rápido cambio y desarrollo de las tecnologías se identificaron nuevos requisitos que conllevaron a la realización de una nueva versión del sistema Airesweb. Esta nueva versión del sistema genera estadísticas a partir de registros generados por un código script insertado en las páginas web que se quieran monitorizar; cuenta además con un módulo procesador de datos desarrollado en C/C++ que aumentó la eficiencia y la velocidad, y con una base de datos mejor estructurada.

El uso de nuevas tecnologías y formas de mostrar la información en las interfaces web de los sistemas similares, y el surgimiento de los nuevos requerimientos funcionales, dificultó el uso de la IW del sistema de análisis que se había realizado inicialmente.

A partir de los planteamientos anteriores, se identificó por parte del proyecto Analizadores Inteligentes de Registros (Aires), perteneciente al Departamento de Soluciones Informáticas para Internet (SINI), la necesidad de que los usuarios puedan interactuar con el nuevo sistema Airesweb sin tener conocimientos avanzados, dando lugar al siguiente **problema a resolver**: ¿Cómo gestionar la información procesada por el nuevo sistema Airesweb?. El **objetivo general** puede definirse como: Desarrollar una IW para la gestión del nuevo sistema Airesweb.

En contraste con el problema a resolver y el objetivo general, se define como **objeto de estudio** las interfaces web de gestión y el **campo de acción** será las interfaces web de los sistemas de análisis de registros web. Del mismo se derivan los siguientes **objetivos específicos**:

- Realizar estudio del estado del arte de las aplicaciones similares.
- Definir las características de la IW para la gestión del sistema Airesweb.

- Implementar la IW para la gestión del sistema Airesweb.
- Evaluar la IW para la gestión del sistema Airesweb.

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

- Investigar sistemas similares al que se va a desarrollar e identificar en los mismos la importancia que tienen las GUIs.
- Investigar y definir las tecnologías del lado del cliente.
- Investigar y definir las tecnologías del lado del servidor.
- Seleccionar las 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.
- Investigar sobre la estructura de una IW de este tipo e identificar, describir y modelar las funcionalidades que esta ha de tener.

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

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

- Artefactos resultantes del análisis y diseño.
- IW del sistema Airesweb.
- Facilitar 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 la usabilidad de un sitio web.
- Por otra parte, contribuirá a la utilización de herramientas de análisis de registros web y fomentará el uso de la analítica web y del software libre en Cuba.

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

Métodos teóricos:

- El **Analítico-Sintético** se aplicará para entender las interfaces 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 interfaces web a partir del análisis de su evolución y las etapas principales por las que han transitado.

Métodos empíricos:

- 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 UML se utilizará para reflejar la estructura, relaciones internas y características de la solución a través de diagramas.

El presente trabajo está estructurado en tres capítulos, los cuales en su conjunto abordarán todo el contenido referente al estudio y desarrollo de una solución para gestionar la información procesada por el nuevo sistema Airesweb.

Capítulo 1: “Fundamentación Teórica”: Contiene la fundamentación teórica de la investigación, se exponen los lenguajes, herramientas, tecnologías y metodologías en los que se apoya la solución propuesta para el problema. Se realiza un análisis crítico y valorativo de las tendencias actuales y el estado del arte de las interfaces web de los sistemas de análisis inteligente de registros web.

Capítulo 2: “Análisis y desarrollo de la solución propuesta”: En este capítulo se realiza el desarrollo de la solución. Se explica toda la dinámica del proyecto a través de las historias de usuario, prototipos de interfaz de usuario y el plan de liberación para las entregas intermedias.

Capítulo 3: “Validación de la solución propuesta”: En este capítulo se detallan las pruebas realizadas al software así como el impacto que tendrá el sistema una vez implantado.

Capítulo 1. Fundamentación Teórica.

Se denomina IW al conjunto de elementos de la pantalla que permiten al usuario realizar acciones sobre un sitio web que está siendo visitado. Se considera parte de la IW a sus elementos de identificación, de navegación, de contenidos y de acción; deben estar preparados para ofrecer servicios determinados al usuario, con el fin de que éste obtenga lo que vino a buscar cuando visitó el sitio web. Por lo anterior, cada uno de los elementos que sean integrados dentro de la IW debe estar pensado para causar un efecto sobre el usuario y utilizarlos con un propósito determinado. Aunque sigan estrechamente vinculadas a las aplicaciones de escritorio han ido ganando un espacio en la Web, con la estandarización de esta. En el marco de las interfaces web de los sistemas de análisis de registros web, una IW permitirá gestionar los datos generados por el sistema Airesweb, precisándose para su desarrollo la selección de lenguajes tecnologías y herramientas adecuadas, así como la metodología que guiará el proceso de desarrollo.

1.1. Trabajos Similares.

Un analizador de tráfico web es un software que captura información importante de los registros de los servidores web. El sistema Airesweb es un ejemplo de este tipo de software y ante la necesidad de una IW para el mismo se hace necesario un estudio de soluciones similares a nivel mundial.

1.1.1. Ámbito Internacional.

Las interfaces web son utilizadas en los sistemas de análisis de reportes de servidores web por ser una de las formas más económicas y sencillas para llevar dicha información a los usuarios finales. En las figuras A.1, A.2, A.3, A.4 y A.5 se muestran las interfaces web de los sistemas de análisis *Webtrends*, *Advanced Web Statistics (AwStats)*, *WeLog Expert*, *Google Analytics* y *Piwik* respectivamente.

Webtrends

Webtrends es la solución 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 como:

- Lanzamiento de nuevos productos o servicios.
- Optimización de las relaciones con sus clientes.
- Reforzamiento de imagen corporativa.

Webtrends ha dado un nuevo paso adelante en el mundo de la analítica web, abriendo nuevas expectativas en la forma tradicional de hacer marketing online. Su licencia es propietaria y su precio se basa en la cantidad de páginas analizadas [4].

AwStats

Es una poderosa y completa herramienta que genera estadísticas gráficas avanzadas para sitios web. Este analizador de logs trabaja como un CGI (Common Gateway Interface) y desde línea de comandos muestra en pocas páginas toda la información que los registros contienen.

AwStats puede ser utilizado en la mayoría de los sistemas operativos. Es una aplicación muy popular como herramienta de administración de servidores. Utiliza la GNU General Public Licence y está programado en Perl [5].

Puede analizar archivos log desde IIS (formato de log W3C), archivos log de Apache/Linux, Windows y Macintosh. Además analiza archivos log de tipo XQX (formato log Common Log Format 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).
- Sistemas Operativos usados.
- Visitas de robots (307 robots detectados).
- Buscadores, palabras clave, frases clave usadas para encontrar el sitio.
- Errores http.
- Otros reportes personalizados sobre URL, parámetros de URL, referidos.
- Cantidad de veces que el sitio es añadido a Favoritos.
- Ratio de navegadores con soporte Java, Flash, RealG2 Reader, QuickTime, Reader, WMA

Reader, PDF Reader.

WebLog Expert

WebLog Expert es un rápido y potente analizador de registro de servidores. Brinda las siguientes informaciones sobre los visitantes del sitio monitorizado:

- 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 otras estadísticas.

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 IIS. Incluso puede leer registros comprimidos a GZ y ZIP. Su interfaz es intuitiva y un asistente guiará de una manera rápida y fácil a crear su perfil de sitio y qué analizar [6].

Google Analytics

Proporciona información muy valiosa sobre el tráfico del sitio web y la eficacia del plan de marketing. Posee funciones potentes, flexibles y fáciles de usar. Google Analytics ayuda a diseñar anuncios más orientados, a mejorar las iniciativas de marketing y a crear sitios web que generen más conversiones [7].

Funcionalidades que presenta este analizador:

- Retorno de la inversión publicitaria.
- Informes personalizados.
- Seguimiento de varios canales y de elementos multimedia.
- Cómo compartir y comunicar la información.
- Consulta de datos.
- Integración y fiabilidad con la garantía de Google.

Algunas de las principales características que presenta Google Analytics son:

- Analiza a fondo el comportamiento de los usuarios y su interacción con el sitio web para identificar formas de optimizar este último.

- Se instala fácilmente, es exhaustivo, fácil de utilizar y presenta la información con claridad.
- Es una herramienta dinámica.
- La información se encuentra de forma centralizada: resultados de la inversión publicitaria online, análisis web y supervisión del comportamiento del usuario.
- Ofrece información detallada sobre la evolución del tráfico del sitio y también señala el punto en el que los usuarios lo abandonan.

Estas características permiten evaluar el diseño del sitio y mejorar el rendimiento obteniendo análisis reales. Por lo que justifican los cambios que se le incorporen al sitio con datos reales y aplicar ajustes que ayuden a los consumidores.

Piwik

Piwik es un software de código abierto (licencia GPL) para el análisis del tráfico web. Este analizador de servidores permite obtener datos y gráficos de:

- Número de visitas.
- Páginas más vistas.
- Motores de búsqueda más usados.
- Palabras claves que utiliza.
- El lenguaje que hablan.
- Sus páginas populares.

O sea, toda la información necesaria para sacar conclusiones acerca del tráfico en la web que se desee analizar, proporcionando informes detallados en tiempo real de los usuarios que visitan el sitio. Piwik pretende ser una alternativa de código abierto a Google Analytics.

Este analizador consta de tres puntos que lo diferencian claramente de otras soluciones:

- Las funciones se integran en plugins, de tal forma que se añadan sólo los que se necesitan.
- El usuario es propietario de los datos. Se almacenan en la base de datos y sólo él tiene acceso a ellos.
- La interfaz de usuario es totalmente personalizable.

Posee funcionalidades y opciones como:

- Posee una interfaz personalizable por medio de widgets (componente gráfico, o control, con el cual el usuario interactúa).

- Tiene una forma clara y sencilla de mostrar los datos.
- Permite configurar varios sitios independientes.
- También permite crear usuarios y asignarles sus privilegios.
- Está disponible en varios idiomas incluido el español.
- Es extensible por medio de plugins.
- Integra los datos de campañas servidas con OpenX.
- Los datos son mostrados en tiempo casi real.
- Está desarrollado a partir de PHPMyVisites.
- Requiere un servidor con PHP y servidor de bases de datos MySQL [8] [9].

Las interfaces web de los sistemas analizados anteriormente no constituyen una solución que pueda adaptarse a Airesweb ya que Webtrends, WebLog Expert y Google Analytics son privativos, esto impide cualquier reproducción parcial o total, en cambio AwStats y Piwik aunque si son libres su código no es lo totalmente modular para que sea reutilizable.

1.1.2. **Ámbito Nacional.**

En Cuba el desarrollo informático ha ido creciendo vertiginosamente, antes de crearse la UCI no se había explorado lo suficiente el tema de la analítica web, por lo que no se tenían soluciones a problemas relacionados con este aspecto. Con el surgimiento de la UCI el mundo de la informática en Cuba toma otro sentido, se inicia un estudio sobre la analítica web, la cual es una recopilación, medición, evaluación y explicación racional de los datos obtenidos de Internet, con el propósito de entender y optimizar el uso del sitio web de una determinada organización.

Se identificó por parte del grupo de proyecto Aires la ausencia de soluciones de este tipo en nuestro país, por lo que surge Airesweb; siendo el primer software de análisis de registros de servidores web de su tipo en el país. En la Figura A. 6 se muestra la IW del sistema Airesweb . La aplicación es libre pero su IW no se adapta a las nuevas potencialidades en cuanto a la gestión de información ni a la nueva estructura de la base de datos, por lo que tampoco constituye una solución factible.

Se llega a la conclusión entonces de que el desarrollo de una IW sería la solución adecuada para realizar las tareas de gestión de la información generada por Airesweb.

1.2. **Interfaz.**

La Real Academia Española define el término interfaz (del inglés interface, superficie de contacto) como una conexión física y funcional entre dos aparatos o sistemas independientes. Generalizando

esta definición, dados dos sistemas que se deben comunicar entre ellos la interfaz será el mecanismo, entorno o herramienta que hace posible dicha comunicación.

El Webster Dictionary brinda en cambio, una semántica más generosa. Define como interfaz a una superficie considerada el límite común entre dos cuerpos o espacios, un límite o interconexión común entre sistemas, equipos, conceptos o seres humanos.

Cuando uno de los sistemas que interactúa es un ser humano se convierte en el concepto de Interfaz Gráfica de Usuario.

1.3. Interfaces Gráficas de Usuario.

Con la aparición de los ordenadores y su posterior popularización se hizo patente la creación de mecanismos que hicieran posible a un usuario medio comunicarse de forma correcta con los sistemas operativos y las aplicaciones que en ellos se estaban ejecutando, sistemas que no son más que interfaces de usuario, o sea, es un artefacto tecnológico de un sistema interactivo que permite a través de una representación visual una interacción entre el usuario y un sistema informático.

La Interfaz Gráfica de Usuario (Graphical User Interface (GUI)) utiliza imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz, lo que posibilita un mejor entendimiento del usuario con aplicaciones informáticas. 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.

Estas interfaces se crearon en un principio en modo comandos, pero su utilidad práctica era limitada, ya que no eran aptas para usuarios que no poseían conocimientos avanzados de informática. Fue necesario entonces crear interfaces basadas en iconos y menús accesibles por medio del mouse, surgiendo los entornos de ventanas, como los que poseen los sistemas operativos MAC y Windows, que empezaron a utilizar interfaces gráficas de usuario, verdaderas protagonistas del mundo informático actual.

Si las interfaces gráficas desaparecieran la gran mayoría de los usuarios de aplicaciones informáticas quedarían anulados. Tanto es así que sistemas operativos y programas que antes funcionaban casi exclusivamente en modo de comandos han tenido que actualizarse y ofrecer a los usuarios interfaces gráficas equivalentes para no quedarse olvidados por el público general.

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.

1.4. Interfaz Web.

En el mundo se produjo un fenómeno sorprendente, llamado a revolucionar la comunicación entre seres humanos: Internet y la World Wide Web (WWW). Con la aparición de la Web se hizo posible que cualquier persona pudiera ofrecer información particularizada a los demás y encontrar documentos interactivos sobre cualquier tema, relacionados unos con otros mediante enlaces que permitían saltar de página en página alrededor del mundo. Las páginas web propiciaron la aparición de las interfaces web.

Concepto.

Las interfaces web son interfaces gráficas de usuario con elementos comunes de presentación y navegación que deben servir de intermediarias entre usuarios genéricos, no acostumbrados generalmente al uso de aplicaciones informáticas, y sistemas de información y procesos transaccionales, debiendo posibilitar la localización de la información deseada, el entendimiento claro de las funcionalidades ofrecidas, la realización práctica de tareas específicas por parte de los usuarios y la navegación intuitiva por las diferentes páginas que forman el sitio web [45].

Buscando una homogeneidad entre los millones de páginas web que existen actualmente en Internet, el diseño de las mismas ha evolucionado con el tiempo hacia un esquema general, ofreciendo interfaces bien definidas, con un conjunto de componentes gráficos y funcionales similares que hacen posible que sea cual sea el usuario que accede a un sitio web la comunicación entre ellos sea posible y efectiva.

Evolución de las interfaces web de los analizadores de reportes de registros de servidores web.

Las primeras herramientas de analítica web surgieron en Estados Unidos y un tiempo después comenzó a desempeñar un papel céntrico en aquellas organizaciones con cierta presencia en Internet [44] .

Inicialmente, en los primeros años de 1990, los esfuerzos de las compañías proveedoras de este tipo de herramientas, se concentraban en el análisis de los archivos “log” generados por los servidores de páginas web. Estos archivos, con estructura de texto plano, recogen importantes variables acerca de la interacción de un visitante con la página web, como por ejemplo páginas vistas, número de hits y errores 404. Sin embargo, a pesar del valor de esta información, en los primeros años del siglo XXI, los diferentes departamentos integrantes de las organizaciones empezaron a demandar más información,

naciendo el análisis en tiempo real, y el etiquetado de las páginas o “tagging”.

1.5. Características generales de la IW a desarrollar.

La IW a desarrollar ayudará al usuario a trabajar con mayor facilidad cuando interactúe con la aplicación, mostrándole las estadísticas que necesite conocer respecto al sitio que administre. La misma mostrará reportes estadísticos como datos del visitante, origen de tráfico y contenido. Presentará características que lo conviertan en una aplicación de fácil y cómodo trabajo: evolución continua, inmediatez, seguridad, estética, rapidez y usabilidad.

1.6. Arquitectura.

La arquitectura de software de un programa o sistema de computación es la estructura que comprende los componentes del software, las propiedades visibles y las relaciones entre ellos [10].

Los sitios web son una conjunción muy compleja de distintos sistemas integrados entre sí (Bases de datos, servidores, redes, componentes de backup y seguridad). La aceptación y calidad de una IW depende directamente de su arquitectura. En el diseño se requiere un firme conocimiento de las tecnologías aplicadas y lenguajes programación, dado que estas están en continuo desarrollo y avanzan día a día. Un mal diseño arquitectónico puede causar problemas usuales como bajo rendimiento, insuficiente mantenimiento y baja disponibilidad. Cuando se desarrollan IW es preciso tener en cuenta una serie de requerimientos y restricciones que influyen directamente en las decisiones arquitectónicas y es tal la importancia de estas que pueden conducir al éxito o fracaso de la aplicación.

Cuando se trate de entender la arquitectura de un software es importante que no se realice desde un solo punto de vista. Es aconsejable que se modele desde cuatro vistas diferentes y una vista de casos de usos para ilustrar y modelar las otras. Este modelo es conocido como 4 + 1 vistas arquitectónicas, las cuales se describen brevemente a continuación:

- **Vista de diseño:** comprende las clases, interfaces y colaboraciones entre ellas como un reflejo del problema y su solución. Esta vista sustenta los requerimientos funcionales del sistema.
- **Vista de procesos:** comprende los hilos y procesos que forman el sistema en tiempo de ejecución, teniendo en cuenta los mecanismos de control de concurrencia y sincronización.
- **Vista de implementación:** está constituida por los componentes y ficheros que componen el software y que son usados para obtener un sistema ejecutable.
- **Vista de despliegue:** está conformada por los nodos que forman la plataforma de hardware

donde el sistema se ejecuta y se enfoca en la distribución, entrega e instalación del mismo.

Por último, la **vista de casos** de uso persigue describir el comportamiento del sistema según la visión de los analistas, probadores y usuarios finales. Contiene los Casos de Uso claves para guiar el diseño de la arquitectura en las fases iniciales y para validar las restantes vistas en las fases finales del desarrollo [10].

1.6.1. Modelo Cliente-Servidor

Es un modelo de computación en el que el procesamiento requerido para ejecutar una aplicación o conjunto de aplicaciones relacionadas se divide entre dos o más procesos que cooperan entre sí. Esta tecnología Cliente-Servidor es utilizada por las aplicaciones en internet donde los principales componentes del modelo:

- **Clientes:** interactúan con el usuario, usualmente en forma gráfica. Generalmente se comunican con procesos auxiliares que se encargan de establecer conexión con el servidor, enviar el pedido, recibir la respuesta, manejar las fallas y realizar actividades de sincronización y de seguridad.
- **Servidor Web:** es un software específico que soporta varios protocolos entre los cuales están HTTP y HTTPS, proporcionan un servicio al cliente y devuelven los resultados.
- **Servidor de Base de Datos:** Almacena información persistente.

Para que los clientes y los servidores puedan comunicarse se requiere una infraestructura de comunicaciones, la cual proporciona los mecanismos básicos de direccionamiento y transporte [11].

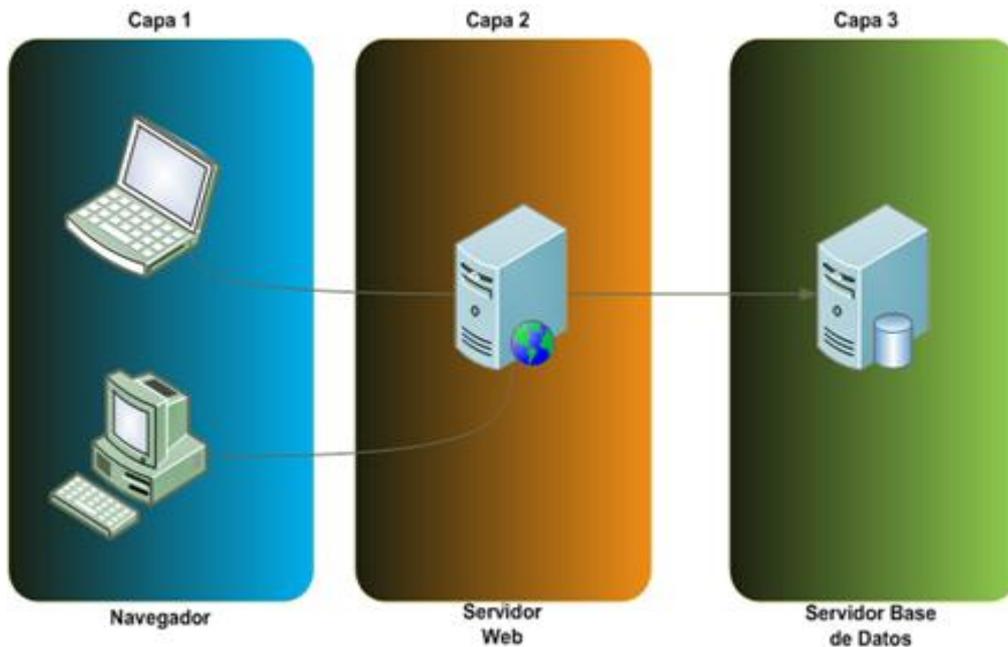


Ilustración 1.1: Modelo Vista Controlador

1.7. Lenguajes, tecnologías y herramientas de desarrollo utilizadas.

1.7.1. Lenguajes de programación.

A la hora de escoger un lenguaje de programación hay que definir bien en qué se utilizará, así como la experiencia de los desarrolladores. Actualmente en Internet los lenguajes comúnmente más utilizados son: PHP, ASP, Java, Ruby on Rail, Python y Perl.

PHP

Cuando se desarrolla una IW bajo una plataforma UNIX o Linux, conviene elegir entre Perl y PHP. PHP es más simple a la hora de escribir scripts y es más productivo en proyectos no tan grandes [12].

PHP es usado en el 33,6% de los sitios web, lo cual no debe extrañar porque es el lenguaje más utilizado [13]. Acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje de código abierto interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor, este lenguaje script es utilizado para la generación de páginas web dinámicas [14] [15].

Un sitio con páginas webs dinámicas es aquel que permite interactuar con el visitante, de modo que cada usuario que visita la página vea la información modificada para propósitos particulares [15].

En PHP el código es ejecutado en el servidor, es extremadamente simple para los principiantes y brinda varias características avanzadas para los programadores profesionales. Al ser un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, por lo que es además independiente del servidor. PHP procesa la información de formularios, genera páginas con contenidos dinámicos, o envía y recibe cookies. Puede ser utilizado en cualquiera de los principales sistemas operativos del mercado actual, incluyendo Linux, muchas variantes UNIX además, Microsoft Windows, Mac OS X, RISC OS. Soporta la mayoría de servidores web, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet.

De modo que con PHP se tiene la libertad de elegir el sistema operativo y el servidor web deseado. También tiene la posibilidad de soporte para una gran cantidad de bases de datos, incluye InterBase, FrontBase, mSQL, Direct MS-SQL, MySQL, Oracle (OCI7 and OCI8), Ovrimos, PostgreSQL. Escribir un interfaz web para una base de datos es una tarea simple con PHP [14].

Como PHP es un lenguaje script hay que tener presente varias cosas. Las cuatro características más importantes son simplicidad, seguridad, estabilidad y velocidad. En cuanto a la velocidad, se integra muy bien a otro software y requiere pocos recursos de sistema. Garantiza la estabilidad pues utiliza su propio sistema de administración de recursos y posee un sofisticado método para manejar variables, lo que lo hace un sistema robusto. Posee protección contra ataques proveyendo diferentes niveles de seguridad. La simplicidad no es más que la posibilidad brindada al desarrollador para generar código rápidamente, desarrolladores con experiencia en C y C++ pueden aprender y adaptarse a PHP fácilmente.

Ventajas:

- Muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- Capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- Capacidad de expandir su potencial utilizando módulos.
- Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Incluye gran cantidad de funciones.

- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

PHP 5

Fue diseñado con intenciones de ser compatible con versiones anteriores tanto como fuese posible, no obstante existen algunos cambios que incluyen:

- Un nuevo modelo de Programación Orientada a Objetos basado en el motor Zend 2.0.
- Una nueva extensión para el soporte mejorado de MySQL.
- Soporte nativo incorporado para SQLite.
- Una nueva constante de reporte de errores para sugerencias sobre el código en tiempo de ejecución.
- Un grupo de nuevas funciones para simplificar la creación de código y reducir la necesidad de crear sus propias funciones para muchos procedimientos comunes.

Teniendo en cuenta lo antes planteado, sus características, ventajas y desventajas se decidió utilizar PHP en su versión 5.0.

HTML

Es un lenguaje de programación que se utiliza para el desarrollo de páginas web. Se trata de la sigla de Hypertext Markup Language, es decir, Lenguaje de Marcas de Hipertexto. Es utilizado para la publicación de hipertexto en la Web y desarrollado con la idea de que cualquier persona o tipo de dispositivo pueda acceder a la información en la Web. Utiliza etiquetas que marcan elementos y estructuran el texto de un documento [16]. HTML es el lenguaje en que se escriben los millones de documentos que hoy existen en el World Wide Web. Al acceder a uno de estos documentos, el cliente (Netscape, IE, Mosaic, Lynx, IBrowse) los interpreta y los despliega [17]. Desde el surgimiento de internet se han publicado sitios web gracias al lenguaje HTML.

Es un lenguaje estático, desarrollado por el World Wide Web Consortium (W3C). Está constituido por elementos que el navegador interpreta y los despliega en la pantalla de acuerdo con su objetivo. Existen elementos para disponer de imágenes, hipervínculos que permiten ir a otra página, listas y tablas para tabular los datos. Los elementos son la estructura básica de este lenguaje, ya que tienen dos propiedades: atributos y contenido. El marcado estructural es el que describe el propósito del texto, aunque no define cómo se verá el elemento.

El marcado presentacional es el que describe la apariencia del texto, sin importar su función. HTML permite describir la estructura y el contenido en forma de texto. El principio esencial del lenguaje es

que se escribe mediante etiquetas, que definen la estructura y el formato del documento; esas etiquetas son leídas por el navegador que es el que permite que se muestren las funciones HTML en nuestra máquina. Por otra parte, el HTML permite incluir scripts, códigos que pueden modificar el comportamiento de los navegadores web y de otros procesadores de HTML.

Ventajas:

- Sencillo que permite describir hipertexto.
- Texto presentado de forma estructurada y agradable.
- No necesita de grandes conocimientos cuando se cuenta con un editor de páginas web.
- Archivos pequeños.
- Despliegue rápido.
- Lenguaje de fácil aprendizaje.
- Lo admiten todos los exploradores [17].

CSS

Originalmente, las páginas HTML sólo incluían información sobre sus contenidos de texto e imágenes. Al crear una página web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos. Una vez creados los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos y posición de cada elemento dentro de la página.

CSS (Cascading Style Sheets), en español Hojas de Estilo en Cascada, es un mecanismo para dar estilo a documentos HTML y XML, que consiste en reglas simples a través de las cuales se establece cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores control total sobre el estilo y formato de sus documentos separando contenido y presentación [18].

CSS es la mejor forma de separar el contenido de las presentaciones y es imprescindible para crear páginas web complejas. Es un mecanismo que describe cómo se va a mostrar un documento en pantalla o como se va a imprimir. Esta forma de describir los estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos, ayudándolos a crear diseño más específico y personalizado para cada elemento de la estructura. Se utiliza para separar el contenido de la

presentación y está basado en una serie de reglas que rigen el estilo de los elementos en los documentos estructurados formando de esta forma la sintaxis de las hojas de estilo.

Usar CSS tiene varias ventajas. El hecho de que el estilo se puede guardar completamente por separado del contenido posibilita por ejemplo almacenar todos los estilos de presentación para una web de 10.000 páginas en un sólo archivo de CSS. Una de las principales características de CSS es su flexibilidad y las diferentes opciones que ofrece para realizar una misma tarea.

Posibilita un mayor control de la presentación agilizando de esta manera su actualización. Los usuarios pueden especificar su propia hoja de estilo, permitiéndoles modificar el formato de un sitio web según sus necesidades, de manera que por ejemplo, personas con deficiencias visuales puedan configurar su propia hoja de estilo para aumentar el tamaño del texto, aumentando de esta forma la accesibilidad. El ahorro en el ancho de banda es otra ventaja, ya que la hoja de estilo se almacena en caché después de la primera solicitud y se puede volver a usar para cada página del sitio, de esta forma, contribuye además con menos costes de almacenamiento y más velocidad a la hora de cargar las páginas web.

JavaScript

JavaScript un lenguaje de programación interpretado, no requiere compilación; dicho de manera más sencilla los programas hechos en Javascript se pueden probar en cualquier navegador sin necesidad de procesos intermedios. Fue creado en la empresa Netscape Communications y es similar a Java aunque no guarda relación directa con este lenguaje de programación, pues no está orientado a objetos.

Desde su aparición siempre fue utilizado en la mayoría de los sitios de Internet, aunque hay que destacar que el surgimiento y desarrollo de aplicaciones Ajax con JavaScript ha dado un protagonismo sin igual dentro de los lenguajes de programación. Esta popularidad se ha extendido a otras aplicaciones y otros entornos no relacionados con la web [19].

Se decide utilizar este lenguaje además porque es la base del análisis de registros web, pues a través de un código javascript embebido en las páginas web a analizar es que el sistema Airesweb recibe información del sitio web en cuestión.

1.7.2. Lenguaje de Modelado.

UML

Lenguaje Unificado de Modelado Unified Modeling Language (UML), por sus siglas en inglés, es uno de los lenguajes de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar 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 [37].

1.7.3. Tecnologías.

AJAX

Una característica esencial de toda aplicación Web 2.0 es la de ofrecer una interfaz especialmente ágil y flexible. El denominador común es la utilización de la tecnología AJAX (Asynchronous Javascript And XML), lo que permite agilizar la interacción entre el navegador y el propio usuario [20]. Ajax no es una tecnología en sí misma. En realidad, se trata de varias tecnologías independientes que se unen de formas nuevas y sorprendentes. Las tecnologías que la forman son: XHTML y CSS para crear la presentación basada en estándares, DOM para la interacción y manipulación dinámica de los elementos de presentación XML para el intercambio y la manipulación de información y JavaScript, para unir todas las demás tecnologías [21].

En las aplicaciones web tradicionales cuando un usuario interactúa con el sistema genera llamadas al servidor, una vez procesada dicha petición el servidor devuelve su respuesta en una nueva página HTML. Esta forma tradicional funciona correctamente pero no causa buena impresión al usuario, si se realizan llamadas continuas, el usuario debe esperar a que se cargue la página nuevamente con los cambios solicitados, entonces el uso de la aplicación se convierte en algo tedioso.

AJAX mejora la interacción del usuario con la aplicación ya que el intercambio de información con el servidor se produce en un segundo plano. AJAX crea una capa intermedia que mejora la respuesta de la aplicación.

Las peticiones HTTP al servidor se sustituyen por peticiones JavaScript. Las peticiones más simples no requieren intervención del servidor, por lo que la respuesta es inmediata. Si la interacción requiere una respuesta del servidor, la petición se realiza de forma asíncrona mediante AJAX. En este caso, la

interacción del usuario tampoco se ve interrumpida por recargas de página o largas esperas por la respuesta del servidor.

Ventajas:

- Utiliza tecnologías ya existentes.
- Soportada por la mayoría de los navegadores modernos.
- Interactividad. El usuario no tiene que esperar hasta que lleguen los datos del servidor.
- Portabilidad (no requiere plugins como Flash y Applet de Java)
- Mayor velocidad, esto debido que no hay que retornar toda la página nuevamente.
- La página se asemeja a una aplicación de escritorio [22].

1.7.4. Frameworks para el lenguaje PHP.

La evolución en el desarrollo de aplicaciones web para múltiples usos, utilizando a PHP como lenguaje de programación y teniendo como mejorar la rapidez, productividad y profesionalidad de los productos que se obtengan, ha propiciado el surgimiento en el mercado para tal fin, de un conjunto de soportes que incluyen a su vez un sin número de bibliotecas para desarrollar y unir los diferentes componentes de un proyecto los denominados Frameworks.

Un Framework, en el argot utilizado por los desarrolladores de software, es una estructura de soporte definido, mediante la cual otro proyecto de software puede ser organizado y desarrollado [23]. Es una estructura conceptual y tecnológica compuesta por librerías, componentes y clases que facilitan el desarrollo ágil, seguro y escalable de una aplicación [24]. Simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, un Framework proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Por último, facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

En términos simples es una estructura lógica de programación que nos ayudará a crear mejores programas. En el mundo existen empresas que demandan soluciones rápidas en aplicaciones web, sin embargo, los programadores por estar contra el tiempo a veces no planifican la construcción de estas estructuras, el resultado, muchas aplicaciones web que pueden funcionar y parecer bonitas por fuera pero por dentro son un verdadero caos.

Los frameworks están diseñados para facilitar el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los

tediosos detalles de bajo nivel. Ofrecen una infraestructura que permite a los desarrolladores tener un código más ordenado, limpio y fácil de actualizar, un código más seguro, robusto y mucho más eficiente.

El desarrollo de aplicaciones web está actualmente muy condicionado por el uso de frameworks; si se concibe un producto medianamente grande que englobe un conjunto de funcionalidades más o menos complejas, es muy recomendable el uso de un framework. Para PHP existen un conjunto de ellos, algunos con características similares y ventajas sobre el resto, por ejemplo: CakePHP, Zend Framework, Symfony, CodeIgniter, Prado, Kumbia, Qcodo, Seagull, Sitellite, SolarPHP, sQeletor, Studs, struts4php, Core Enterprise PHP, entre otros. Los más utilizados por los desarrolladores a nivel mundial coinciden con los siguientes: CodeIgniter, CakePHP, Zend Framework, Kumbia y Symfony.

CodeIgniter

CodeIgniter es un sencillo framework escrito para que funcione sobre PHP 4 y PHP 5. A continuación se describen un conjunto de características:

- Implementa el patrón MVC (Modelo-Vista-Controlador) y cuenta con una gran cantidad de helpers que le facilitan el trabajo a los desarrolladores.
- Implementa el patrón Active Record como capa de abstracción de base de datos, aunque no posee capa de ORM.
- Es compatible con algunos de los motores de base de datos más usados (MySQL, PostgreSQL, SQL Server, SQLite y Oracle).
- El mecanismo de control de sesiones es a través de cookies y puede usar base de datos para el control de las mismas.
- Posee una gran comunidad de desarrolladores y una amplia documentación, tanto en inglés como español [25].

Kumbia

Diseñado para acelerar el desarrollo de aplicaciones web sin pérdida de tiempo y con eficacia, con una buena documentación en español sobre su uso en Internet. Reúne el conjunto de herramientas para la automatización de tareas que quitan tiempo al desarrollador, permitiéndole invertir sus esfuerzos en los aspectos esenciales del desarrollo. Está escrito en PHP 5 y liberado bajo la licencia GPL de software libre. El Modelo de Objetos de Kumbia es representado en tres capas:

- Abstracción del sistema de gestión de bases de datos.
- Mapeo Objeto Relacional.

- Modelo Vista Controlador.

“Kumbia es innovador y su principal enfoque es desarrollar herramientas y escribir cada componente del framework pensando en que sea fácil de usar para cualquiera que lea su documentación” [26].

Cumple con los requisitos esenciales para construir un proyecto web:

- Soporta las mejores prácticas de diseño y patrones más eficientes.
- Multiplataforma, sencillo, fácil de aprender y configurar.
- Permite la creación de aplicaciones de código legible y fácil de mantener.
- Se integra con los sistemas gestores de bases de datos más populares en la actualidad, como por ejemplo: MySQL, PostgreSQL y Oracle.
- Administración de Caché.
- Scaffolding Avanzado.
- Implementa el patrón arquitectónico MVC.
- Brinda soporte para el uso de AJAX para asegurar una mejor interactividad y experiencia del usuario.
- Posee herramientas para la generación de formularios.
- Implementa mecanismo de seguridad para evitar accesos no autorizados a la aplicación.
- Presenta herramientas que complementan su funcionalidad; tal es el caso de FPDF para los reportes en formato PDF, Prototipo como framework de Javascript unido a Scriptaculous para la manipulación de la presentación y lograr los efectos visuales compatibles con varios navegadores, PHPMailer para el envío de correo electrónico y Smarty como motor de plantillas potente y fácil de usar.

CakePHP

Es un marco de trabajo para el desarrollo rápido y robusto de aplicaciones web, con excelentes características para lograr un producto con los requisitos que se exigen de un sistema web como su portabilidad, seguridad, reutilización de componentes, refactorización y modularización del código fuente. Dentro de las características más significativas se relacionan:

- Automatiza un conjunto de tareas que pueden resultar molestas en algunos casos por su naturaleza trivial.
- Emplea el patrón arquitectónico MVC y accede a las bases de datos con una capa ORM.
- Urls amigables.
- Uso de plantillas personalizables.

- Validación de datos de entrada integrada y componentes de seguridad y sesión.
- Compatibilidad con las versiones 4 y 5 de PHP.
- Helpers integrados (fragmentos de código encapsulado en funciones PHP para formularios, HTML, JavaScript, AJAX y otros [27]).

Zend Framework

Es el framework desarrollado por la empresa que respalda comercialmente a PHP. Basado en la arquitectura MVC y escrito en PHP 5. Otros aspectos que lo caracterizan son:

- Los componentes que representan la biblioteca estándar del framework son completamente extensibles y pueden usarse individualmente por el bajo acoplamiento que mantienen.
- Zend Framework posibilita la integración con PHPUnit para el desarrollo de pruebas unitarias a la aplicación que se desarrolle con el mismo.
- Soporta diferentes motores de bases de datos, haciendo uso del patrón Active Record como capa de abstracción de base de datos; aunque no posee una capa de ORM propio, puede integrarse con el framework Doctrine.
- Contiene un conjunto de módulos para manejo de canales RSS, servicios web (Amazon, Flickr, Yahoo) y módulos de autenticación.
- Cuenta con una amplia documentación, aunque la mayoría en inglés [28].

Symfony

Es un Framework de aplicación de web para proyectos PHP. Está dirigido a la aceleración de la creación y el mantenimiento de aplicaciones web, de igual forma a reemplazar los códigos de tareas repetidos, con poder, control y placer [29]. Simplifica el desarrollo de aplicaciones mediante la automatización de muchas tareas comunes y rutinarias [30].

Está desarrollado en PHP5, se puede utilizar en plataformas Unix, Linux y Windows. Requiere de una instalación, configuración y líneas de comando, incorpora el patrón MVC, soporta AJAX, plantillas y un gran número de bases de datos. Se encarga de todos los aspectos comunes de las aplicaciones web, dejando que el programador se dedique a aportar valor desarrollando las características únicas de cada proyecto. Aumenta la productividad y ayuda a mejorar la calidad de las aplicaciones web aplicando las buenas prácticas y patrones de diseño que se han definido para la web.

Es fácil de instalar y configurar en sistemas Windows, Mac y Linux. Funciona con todas las bases de datos comunes (MySQL, PostgreSQL, SQLite, Oracle, MS SQL Server). Preparado para aplicaciones

empresariales, ya que se puede adaptar con facilidad a las políticas y arquitecturas propias de cada empresa u organización. Flexible hasta cualquier límite y extensible mediante un completo mecanismo de plugins. Publicado bajo licencia MIT (significa que cualquier persona que obtenga una copia de Symfony podrá utilizarlo sin restricciones) de software libre y apoyado por una empresa comprometida con su desarrollo. Traducido a más de 40 idiomas y fácilmente traducible a cualquier otro idioma.

Desde su primera versión Symfony ha sido creado sólo para PHP5, para obtener el máximo rendimiento de PHP y aprovechar todas sus características. Ha sido probado con éxito durante años en varias aplicaciones gigantescas y en otros pequeños y medianos. Es infinitamente escalable si se disponen de los recursos necesarios. En cuanto a soporte, Symfony sigue una política de tipo LTS (long term support), por la que las versiones estables se mantienen durante 3 años sin cambios pero con una continua corrección de errores. A los creadores de Symfony les interesan aspectos como el rendimiento, la buena documentación y el soporte muy largo. Su código fuente incluye más de 9.000 pruebas unitarias y funcionales.

Symfony cuenta con miles de páginas de documentación distribuidas en varios libros gratuitos y decenas de tutoriales. En la actualidad es uno de los Frameworks conocidos y utilizados por los desarrolladores que desean crear sus aplicaciones de manera ágil utilizando patrones de diseño como el Modelo-Vista-Controlador (MVC). Es adecuado tanto para programadores expertos como para principiantes del desarrollo web. Si se desarrolla aplicación web mediana o compleja, en las que es importante la productividad de su desarrollo, la facilidad de su mantenimiento, su seguridad y la garantía de utilizar librerías probadas en algunos de los sitios web más grandes de Internet, entonces se debe utilizar un buen Framework como Symfony.

Se elige Symfony como Framework de desarrollo porque reúne las mejores prácticas de desarrollo web e integra muchas otras librerías y herramientas que aportan fortaleza al producto final. Usa además Doctrine o Propel como ORM (Object Relational Mapping) a opción del desarrollador, para el mapeo de objetos relacional, que constituyen los dos mejores en esta área. Unido a lo plasmado anteriormente las herramientas de generación de código, la interfaz de línea de comando para la instalación del sistema desarrollado y otras tareas comunes automatizadas y la amplia documentación que posee en Internet; convierten a Symfony en uno de los más potentes Framework para PHP actuales. Cuenta con una amplia comunidad de desarrolladores que constituye la principal fuente de retroalimentación para el desarrollo de versiones posteriores, en las que se corrigen errores en las versiones anteriores y se agregan nuevas funcionalidades. Además de seguir una política LTS (Long

Term Support), por lo que las versiones estables se mantienen durante un período de 3 años con una continua corrección de errores.

1.7.5. Servidor de Aplicaciones Web.

Apache

Apache surge a partir de un servidor web de la NCSA (National Center for Supercomputing Applications de la Universidad de Illinois) como corrección a errores y mejoras importantes al producto inicial. En 1999 los desarrolladores fundan la Apache Software Foundation, organización que da soporte legal al desarrollo de Apache y que lo mantiene actualmente bajo la licencia Apache Licence 2.0, que lo convierte en software libre [31].

Apache, en su versión 2, hace del servidor web una solución más flexible y escalable, ofreciendo un conjunto de mejoras. Dentro de sus características están:

- Soporte para el protocolo HTTP 1.1.
- Sencillo.
- Soporte para CGI (Common Gateway Interface).
- Soporte para FastCGI.
- Soporte de Host Virtuales.
- Soporte de autenticación HTTP.
- Perl integrado.
- Soporte de scripts PHP.
- Soporte de servlets de Java.
- Servidor proxy integrado.
- Estado del servidor y adaptación de registros.
- Soporte de Server Side Include (SSI).
- Soporte de Secured Socket Layer (SSL).
- Módulos multiproceso.
- Portable en tiempo de ejecución [32].

Se escoge como servidor web Apache en su versión dos por las características en cuanto a rendimiento, seguridad y escalabilidad. Como estrategia de la Universidad de las Ciencias Informáticas de uso de software libre en sus sistemas, Apache constituye la alternativa libre en el hospedaje de aplicaciones para la web, con soporte nativo para PHP, lenguaje también seleccionado para el

desarrollo de la solución propuesta. Es el servidor web por excelencia para sistemas operativos GNU/Linux.

1.7.6. Herramienta de Programación.

En la actualidad un IDE de programación es una herramienta indispensable para el desarrollo de un proyecto, son varios los que están disponibles para la implementación utilizando el lenguaje PHP.

Ejemplos de estos son:

- ZendStudio5.
- PhpEdit7.
- PhpDesigner8.
- Netbeans9.
- Eclipse PDT.
- Open Komodo Project66.

La famosa revista online Smashing Magazine muestra un interesante artículo, en el cual a través de una tabla comparativa de IDEs para PHP, abordan las ventajas de cada uno de estos. La tabla se encuentra en el documento adjunto *IDE_PHP.pdf*.

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. 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. Las características del programa se pueden mejorar mediante el uso de plugins [33]. Uno de los plugins básicos para el trabajo con PHP es el PDT.

Eclipse PDT es un conjunto de herramientas y Frameworks que mejoran la productividad de los desarrolladores que usan PHP por las siguientes características. Editor sensible al contexto, el cual resalta el código, asistente de código y autocompletado de código. Soporte para el debug incremental del código de PHP. Extensos Frameworks e Interfaz de aplicación (APIs) que permiten a los desarrolladores extender PDT para crear nuevas herramientas orientadas al desarrollo de PHP [33].

Eclipse PDT es:

- 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.

Por las características antes mencionadas se escogió Eclipse PDT como IDE de programación.

1.7.7. Herramienta de Modelado.

Visual Paradigm

Es una herramienta para modelado UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta Visual Paradigm también proporciona abundantes tutoriales, demostraciones interactivas y proyectos UML. Presenta un entorno para la especificación de los detalles de los casos de uso, incluye la especificación del modelo general y de las descripciones de los casos de casos de usos [38].

1.8. Metodología de Desarrollo de Software.

La selección de una metodología de desarrollo de software así como las herramientas que soportarán al mismo es una tarea necesaria e importante; la calidad del proceso y del producto final estará condicionada en gran medida por estos dos aspectos.

SXP

SXP (SCRUM + XtremeProgramming) es una metodología ágil desarrollada como un híbrido de SCRUM y XP; para coleccionar documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software. Garantizando la eficacia y la eficiencia en el proceso de generación del mismo [34].

Metodologías Ágiles: Surgen como una alternativa a las metodologías tradicionales. Su filosofía se centra en el individuo, la colaboración con el cliente e Se ajustan a proyectos de corta duración y cuya planificación no debe ser estricta [35]. Ofrece una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva fomentando el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo, y cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los

requisitos para llegar al éxito del proyecto. Con el uso de SCRUM para la gestión del trabajo, SXP logra resultados satisfactorios y toma de XP las mejores prácticas que guían el desarrollo del software, como la refactorización y pruebas continuas.

SXP consta de 3 fases, las que se muestran en el gráfico de la figura:

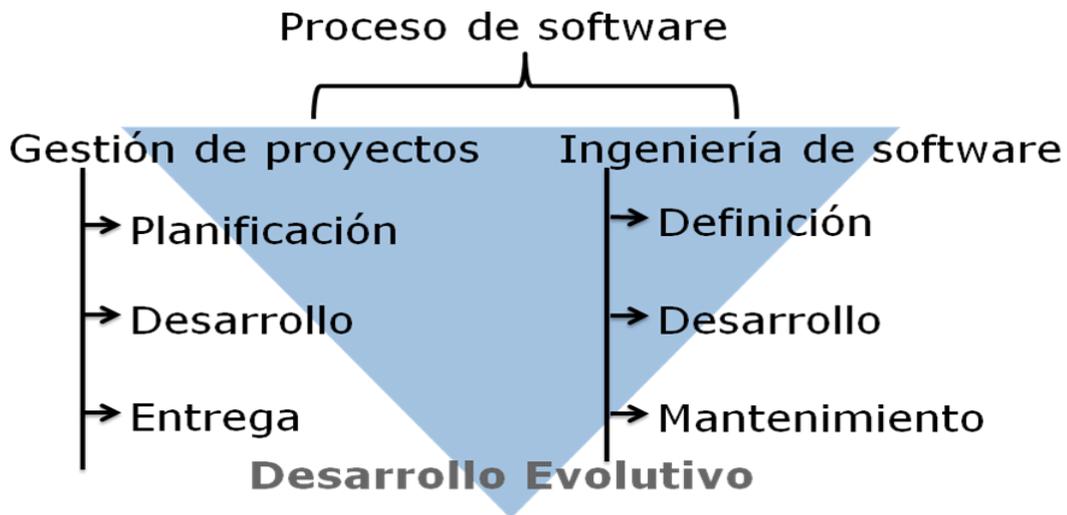


Ilustración 1.2: Esquema de la metodología SXP.

- Fase de Planificación-Definición: su propósito es establecer la visión, fijar expectativas y asegurar financiamiento.
- Fase de Desarrollo: el propósito es implementar un sistema listo para entrega en una serie de iteraciones de 60 días. (El tiempo puede decrementarse a la medida que se está refinando el producto).
- Fase de Entrega-Mantenimiento: se entrega el producto y se le da el soporte necesario.

SXP es ideal para proyectos de corta duración con requisitos cambiantes o no bien definidos, donde prevalezca la retroalimentación entre el cliente y el equipo de trabajo. El desarrollo con SXP se realiza en iteraciones cortas (sprints) a lo largo de 3 fases, dándole cumplimiento a un grupo de actividades, de las que se generan una serie de artefactos que documentan el proceso de desarrollo, obteniendo un release del producto con nuevas funcionalidades [36].

1.9. Conclusiones Parciales.

En el presente capítulo se trataron conceptos generales acerca de las interfaces web con el objetivo de

ubicar al lector en el dominio del problema. Fueron analizados algunos sistemas de análisis de registros web determinándose que es necesaria la implementación de una interfaz web acorde con las necesidades de gestión del sistema Airesweb. Para ello se debe utilizar PHP5 como lenguaje principal en el desarrollo de la solución, como herramienta de desarrollo o IDE a Eclipse PDT y como Framework de desarrollo Symfony en su versión 1.2.9. Resulta válida también la elección de SXP como metodología a emplear en el proceso de desarrollo de software; complementando el proceso de ingeniería con el uso del lenguaje de modelado UML y la herramienta case de modelado Visual Paradigm.

Capítulo 2. Análisis y desarrollo de la solución propuesta.

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, además de asegurar que todas las partes involucradas tengan un entendimiento común de los objetivos.

En el presente capítulo se recogen las funcionalidades de la solución propuesta reflejadas en las Historias de Usuario así como la planificación de las iteraciones del ciclo de desarrollo. Aborda el diseño a alto nivel de la solución propuesta, describiendo el patrón Modelo-Vista-Controlador, en la variante implementada por el framework Symfony; así como los patrones de diseño empleados y los mecanismos de seguridad que provee el framework. También se describe la estructura modular del sistema propuesto a través de diagramas de Componentes.

2.1. Descripción del Problema.

Anteriormente se expuso la necesidad de poseer una IW para gestionar la información procesada por el sistema Airesweb de una forma eficiente y rápida. Para conocer cuál es la información procesada se aborda a continuación el funcionamiento del sistema. Cuando se desea monitorizar un sitio web es necesario registrarlo en el sistema, el cual genera un código javascrit que debe ser insertado en las páginas a analizar. Al interactuar los usuarios con dicho sitio se genera cierta información que es recibida por un receptor de datos y se guarda en archivos de registros. El contenido de estos archivos es analizado y procesado por el módulo de procesamiento de datos, generando reportes estadísticas que se almacenan en la BD.

Las tareas de gestión de la información por parte del sistema Airesweb dependen de una BD, los reportes estadísticos generados podrían ser consultados desde una herramienta como PgAdminIII, pero habría que tener conocimientos avanzados sobre Estructured Query Language (SQL) y BD por el nivel de complejidad de algunas de las consultas SQL, y también se necesita conocimiento sobre la herramienta en cuestión, con la Interfaz Web estos conocimientos pasan a un segundo plano y cualquier tipo de usuario puede consultar los reportes realizados.

2.2. Solución Propuesta.

La propuesta para dar solución a la problemática planteada consiste en estructurar la IW de modo que cuente con cinco módulos, uno de ellos relacionado con la ayuda del sistema, otro encargado de las

gráficas y nuevas formas de mostrar las estadísticas, otro módulo agrupará las funcionalidades que permiten obtener los reportes a partir de la información almacenada en la BD, otro módulo para el registro de los usuarios y el otro módulo relacionado con las tareas de gestión de sitios.

La IW de Airesweb mostrará los siguientes reportes estadísticos.

Datos del visitante:

- Visitas nuevas y recurrentes.
- Visitas por demografía.
- Visitas por fecha.
- Usuarios únicos por fecha.
- Páginas vistas por fecha.
- Rebotes por fecha.

Origen de Tráfico:

- Navegadores.
- Sistema Operativo.
- Sistema Operativo + Navegador.
- Opciones del Navegador.
- Visión General.
- Fuentes de Tráfico.
- Fuentes de Tráfico Individual.
- Palabras Clave.
- Buscadores.

Contenido:

- Páginas vistas.
- Páginas de entrada.
- Páginas de salida.

La IW de Airesweb presentará características que lo conviertan en una aplicación de fácil y cómodo trabajo:

- Evolución continua: Actualizaciones frecuentes con el propósito de facilitar la navegación del usuario en la aplicación.
- Inmediatez: Tiempo de distribución corto, está definido por la velocidad con que llega la

información al usuario.

- Seguridad: Infraestructura de seguridad.
- Estética: Apariencia confortable y atractiva del sitio.
- Rapidez: Cuenta con un corto tiempo de respuesta en cargar la página web.
- Usabilidad: La interfaz presentará funcionalidades para la mejor interacción de los usuarios con la aplicación.

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

La Lista de Reserva del Producto que se presenta a continuación brindará más detalles de la solución propuesta.

2.3. Lista de Reserva del Producto.

Otra de las actividades más importantes definidas en la metodología SXP es la Lista de Reserva del Producto (LRP), en la cual se recoge en una lista priorizada todo el trabajo a desarrollar en el proyecto.

Cuando un proyecto comienza es muy difícil tener claro todos los requerimientos sobre el producto. Sin embargo, suelen surgir los más importantes que casi siempre son más que suficientes para una iteración.

Esta lista puede crecer y modificarse a medida que se obtienen más conocimientos acerca del producto y del cliente. Con la restricción de que sólo puede cambiarse entre iteraciones. El objetivo es asegurar que el producto definido al terminar la lista es el más correcto, útil y competitivo posible y para esto la lista debe acompañar los cambios en el entorno y el producto.

Esta lista puede estar conformada por requerimientos técnicos y del negocio, funciones, errores a reparar, defectos, mejoras y actualizaciones tecnológicas requeridas.

<i>Prioridad</i>	<i>Ítem*</i>	<i>Descripción</i>	<i>Estimación(Días)</i>	<i>Estimado por</i>

Alta				
	1	Iniciar sesión.	1	GER
	2	Finalizar sesión de usuario.	1	GER
	3	Insertar usuario.	1	GER
	4	Eliminar usuario.	1	GER
	5	Cambiar contraseña.	1	GER
	6	Añadir sitio web a monitorizar.	1	GER
	7	Mostrar los sitios monitorizados para cada usuario.	1	GER
	8	Generar javascript.	2	GER
	9	Escoger sitios.	1	GER
	10	Verificar si el sitio envía datos.	1	GER
	11	Mostrar panel principal con las estadísticas del día anterior.	2	GER
	12	Mostrar en cada página el rango de fecha a seleccionar.	1	GER
	13	Mostrar visitas realizadas por fechas.	1	GER
	14	Mostrar visitas nuevas y recurrentes.	1	GER
	15	Mostrar fidelización del usuario.	1	GER
	16	Mostrar visitas por demografía.	1	GER
	17	Mostrar usuarios únicos por fecha.	1	GER
	18	Mostrar páginas vistas por fecha.	1	GER
	19	Mostrar rebotes por fechas.	1	GER
	20	Mostrar estadísticas de Navegadores	1	GER
	21	Mostrar estadísticas de SO.	1	GER

		rapidez de respuesta depende del hardware y el tráfico de la red donde se instale el sistema, pero también del nivel de optimización del código.		
	5	Debe ir acompañada de un Manual de Usuario para facilitar su trabajo de uso.		

Tabla 2.1: Lista de Reserva del Producto (LRP).

2.4. Historias de Usuario y Tareas Ingenieriles.

A continuación se relaciona una muestra de las Historias de Usuario, los prototipos de interfaz de usuarios y Tareas Ingenieriles asociadas. Esta es solo una planificación inicial, el proceso es cambiante para ir adecuándolo a las necesidades y nuevas propuestas. Todas las decisiones se toman de conjunto con el cliente que es parte del equipo de desarrollo.

Historia de Usuario	
Número: M-1	Nombre Historia de Usuario: Iniciar sesión
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Adrian Gallardo Palay	Iteración Asignada: Sprint 1
Prioridad en Negocio: Alta	Puntos Estimados: 1 semana
Riesgo en Desarrollo: Bajo	Puntos Reales: 2 semanas
Descripción: Para iniciar la sesión el usuario deberá introducir su nombre de usuario y su contraseña, luego dar clic sobre el botón que indica el inicio de dicha sesión.	
Observaciones:	

Prototipo de Interfaz

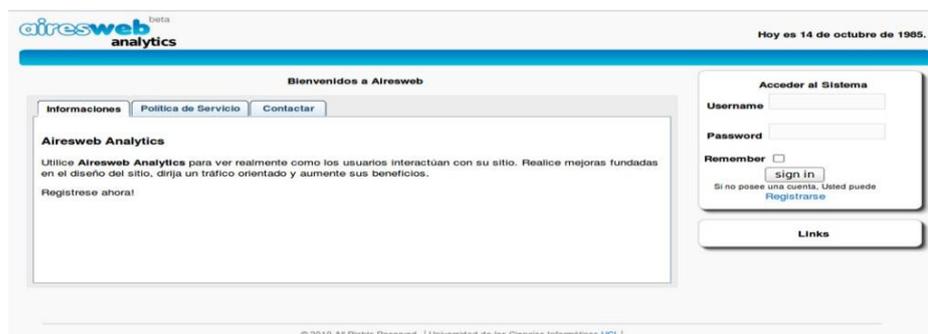


Tabla 2.2: HU Iniciar sesión.

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: M-1
Nombre Tarea: Desarrollar el módulo de usuario	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrian Gallardo Palay	
Descripción: Crear un módulo para controlar la información referente a los usuarios.	
Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: M-1
Nombre Tarea: Iniciar sesión de usuario	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrian Gallardo Palay	
Descripción: Agregar al módulo de usuario la posibilidad de inicio de sesión. Personalizar formulario de inicio de sesión, añadiendo mensajes de validación.	

Tabla 2.3: Tareas Ingenieriles para la HU Iniciar sesión.

Historia de Usuario	
Número: M-3	Nombre Historia de Usuario: Insertar Usuario
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Javier E. Domínguez Viltres	Iteración Asignada: Sprint 1

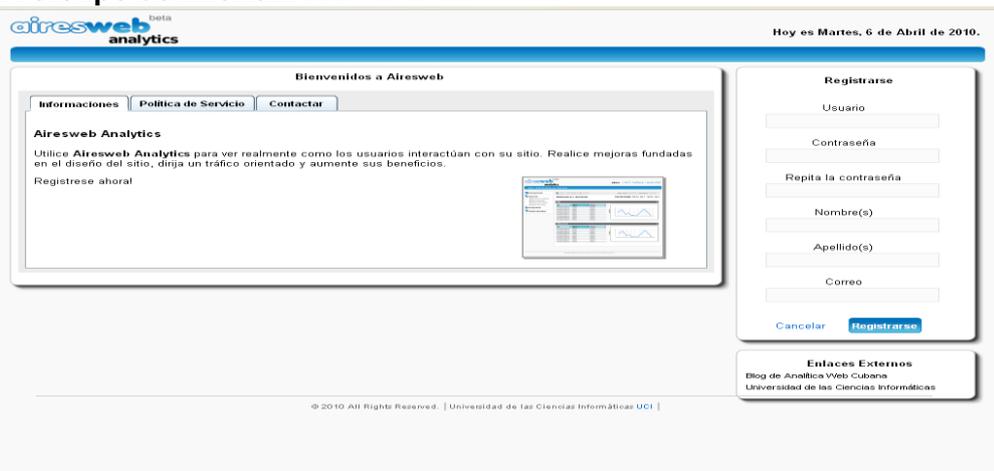
Prioridad en Negocio: Alta	Puntos Estimados: 1 semana
Riesgo en Desarrollo: Bajo	Puntos Reales: 2 semanas
Descripción: Para insertar un usuario debe introducir su nombre y apellidos, contraseña y correo electrónico.	
Observaciones: El sistema debe verificar los datos.	
Prototipo de interfaz:	
	

Tabla 2.4: HU Insertar sesión.

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: M-3
Nombre Tarea: Desarrollar el módulo registro	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier E. Domínguez Viltres	
Descripción: Crear un módulo para controlar el registro de los usuarios.	
Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: M-3
Nombre Tarea: Insertar usuario	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:

Programador Responsable: Javier E. Domínguez Viltres
Descripción: Agregar al módulo de registro la posibilidad de registrar un usuario nuevo. Personalizar formulario de insertar usuario, añadiendo mensajes de validación.

Tabla 2.5: Tareas Ingenieriles para la HU Insertar usuario.

Historia de Usuario	
Número: M-8	Nombre Historia de Usuario: Generar código javascript
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Javier E. Domínguez Viltres	Iteración Asignada: Sprint 1
Prioridad en Negocio: Alta	Puntos Estimados: 1 semana
Riesgo en Desarrollo: Bajo	Puntos Reales: 2 semanas
Descripción: Luego de haber insertado los datos del sitio, el sistema genera un “código javascript”, el mismo debe ser insertado en cada una de las páginas que se desee monitorizar.	
Observaciones:	
Prototipo de interfaz:	

Tabla 2.6: HU Generar código javascript.

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: M-8
Nombre Tarea: Estudiar sobre la generación de “código javascript”	
Tipo de Tarea: Estudio	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrian Gallardo Palay.	
Descripción: Estudiar sobre la generación de “código javascript” en sistemas similares y su funcionamiento.	
Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: M-8
Nombre Tarea: Mostrar “código javascript”	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1

Fecha Inicio:	Fecha Fin:
Programador Responsable: Adrian Gallardo Palay.	
Descripción: Agregar al módulo correspondiente la posibilidad de mostrar el “código javascript” generado luego de haber añadido la información del sitio.	

Tabla 2.7: Tareas Ingenieriles para la HU Generar código javascript.

Historia de Usuario																													
Número: M-10	Nombre Historia de Usuario: Verificar si el sitio envía datos.																												
Modificación de Historia de Usuario Número: Ninguna																													
Usuario: Javier E. Domínguez Viltres	Iteración Asignada: Sprint 1																												
Prioridad en Negocio: Alta	Puntos Estimados: 1 semana																												
Riesgo en Desarrollo: Bajo	Puntos Reales: 2 semanas																												
Descripción: Cada sitio tiene tres estados: Activo (Se han generado estadísticas de la interacción de los usuarios con el sitio web), Poco activo (El sitio no ha generado nuevas estadísticas en un tiempo mayor a los 7 días) e Inactivo (El sitio no ha generado estadísticas o ha dejado de generar estadísticas por un tiempo mayor a los 15 días).																													
Observaciones:																													
Prototipo de interfaz:																													
<p style="text-align: center;">Sitios:</p> <table border="1"> <thead> <tr> <th>Nombre</th> <th>URL</th> <th>Reportes</th> <th>Estado</th> <th>Visitas</th> <th>Tiempo promedio en el sitio</th> <th>Acciones</th> </tr> </thead> <tbody> <tr> <td>AiresWeb Official Site</td> <td>http://10.33.13.13:8080/</td> <td>Ver reportes</td> <td>✓</td> <td>16</td> <td>55.06 seg</td> <td>Editar Eliminar</td> </tr> <tr> <td>Blog de analítica web</td> <td>http://analiticaweb.blogcip.cu/</td> <td>Ver reportes</td> <td>✗</td> <td></td> <td></td> <td>Editar Eliminar</td> </tr> <tr> <td>hayhsdjasd</td> <td>http://www.airesweb1.com</td> <td>Ver reportes</td> <td>✓</td> <td>2</td> <td>746.00 seg</td> <td>Editar Eliminar</td> </tr> </tbody> </table> <p style="text-align: center;">Agregar Sitio</p>		Nombre	URL	Reportes	Estado	Visitas	Tiempo promedio en el sitio	Acciones	AiresWeb Official Site	http://10.33.13.13:8080/	Ver reportes	✓	16	55.06 seg	Editar Eliminar	Blog de analítica web	http://analiticaweb.blogcip.cu/	Ver reportes	✗			Editar Eliminar	hayhsdjasd	http://www.airesweb1.com	Ver reportes	✓	2	746.00 seg	Editar Eliminar
Nombre	URL	Reportes	Estado	Visitas	Tiempo promedio en el sitio	Acciones																							
AiresWeb Official Site	http://10.33.13.13:8080/	Ver reportes	✓	16	55.06 seg	Editar Eliminar																							
Blog de analítica web	http://analiticaweb.blogcip.cu/	Ver reportes	✗			Editar Eliminar																							
hayhsdjasd	http://www.airesweb1.com	Ver reportes	✓	2	746.00 seg	Editar Eliminar																							

Tabla 2.8: HU Verificar si el sitio envía datos.

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: M-10
Nombre Tarea: Estudiar sobre verificación de envío de datos por parte del sitio.	
Tipo de Tarea: Estudio	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:

Programador Responsable: Adalberto L. de Guevara Zamora	
Descripción: Estudiar sobre cómo se puede verificar si los sitios analizados envían datos al sistema en los sistemas similares, así como las características de cada categoría del nivel de actividad.	
Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: M-10
Nombre Tarea: Verificar si el sitio envía datos.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Adalberto L. de Guevara Zamora	
Descripción: Agregar al módulo correspondiente la funcionalidad de verificar si el sitio envía datos o no y mostrar dicha información al usuario. Incluir mensajes por categoría.	

Tabla 2.9: Tareas Ingenieriles para la HU Verificar si el sitio envía datos.

La descripción de las restantes Historias de Usuario y Tareas Ingenieriles se encuentra en el documento adjunto **Historias_Usuario_TI.pdf**.

2.5. Plan de Liberación.

En este paso se define el plan de liberación e iteraciones para realizar las entregas intermedias y la entrega final. Tiene como entrada la relación de Historias de Usuario definidas previamente. Para colocar una historia en cada iteración se tiene en cuenta la prioridad que definió el cliente para dicha Historia. Como resultado de la priorización de Historias se llegó a la siguiente planificación:

Reléase	Orden de la HU a implementar	Duración total
1	M1, M2, M5, M3	3
2	M4, M6, M7, M8, M9, M10, M11, M12, M13, M14, M15, M16, M17	6
3	M18 , M19, M20, M21, M22, M23, M24, M25, M26, M27, M28, M29, M30	5

Tabla 2.10 : Plan de Liberación.

2.6. Implementación del MVC por Symfony.

Symfony es un framework diseñado para optimizar, pues proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Está basado en el patrón arquitectónico MVC, que está formado por tres capas:

- La *capa* del modelo define la lógica de negocio (la BD pertenece a esta capa).Symfony guarda todas las clases y archivos relacionados con el modelo en el directorio lib/model.
- La *vista* es lo que utilizan los usuarios para interactuar con la aplicación (los gestores de plantillas pertenecen a esta capa). En Symfony la capa de la vista está formada principalmente por plantillas en PHP. Estas plantillas se guardan en varios directorios llamados templates repartidos por todo el proyecto. Se descompone en: Layout: es global en toda la aplicación, Lógica de la vista: se puede transformar en un archivo de configuración sencillo, sin necesidad de programarla y las ya mencionadas plantillas: encargadas de visualizar las variables definidas en el controlador.
- El *controlador* es un bloque de código que realiza llamadas al modelo para obtener los datos y se los pasa a la vista para que los muestre al usuario. Es único por cada aplicación y es el único punto de entrada a la misma. Todas las peticiones se canalizan a través de los controladores frontales (index.php y frontend_dev.php). Estos controladores frontales realmente delegan todo el trabajo en las acciones: incluyen el código específico del controlador en cada página [39].

En figuras 2.1 y 2.2 se muestra el funcionamiento del patrón MVC y el flujo de trabajo del MVC en Symfony respectivamente.

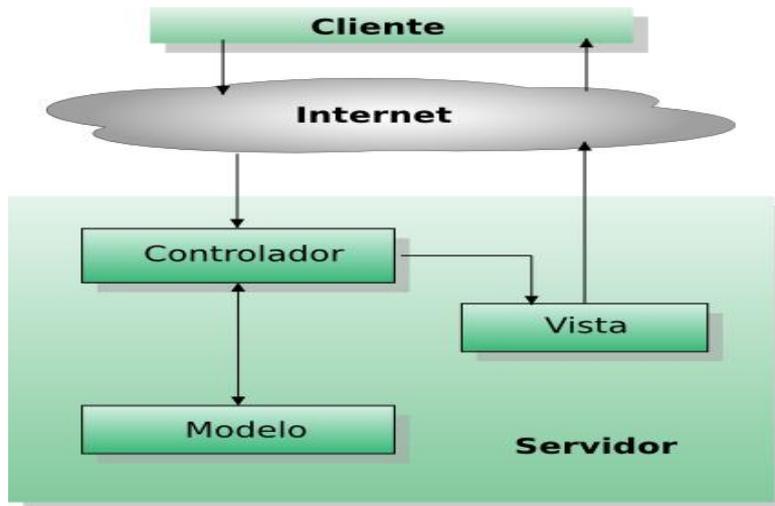


Ilustración 2.1: Funcionamiento del patrón MVC.

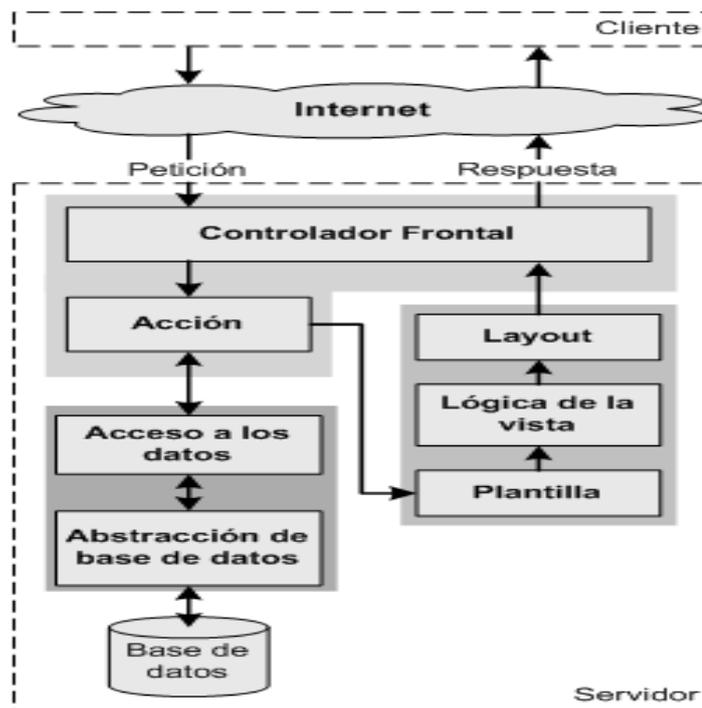


Ilustración 2.2: Flujo de trabajo del MVC en Symfony.

2.7. Organización de la Aplicación.

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

proyecto en una estructura de tipo árbol. Un proyecto en Symfony es un conjunto de servicios y operaciones disponibles bajo un determinado nombre de dominio y que comparten el mismo modelo de objetos [39].

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 generalmente 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 figura 2.3:

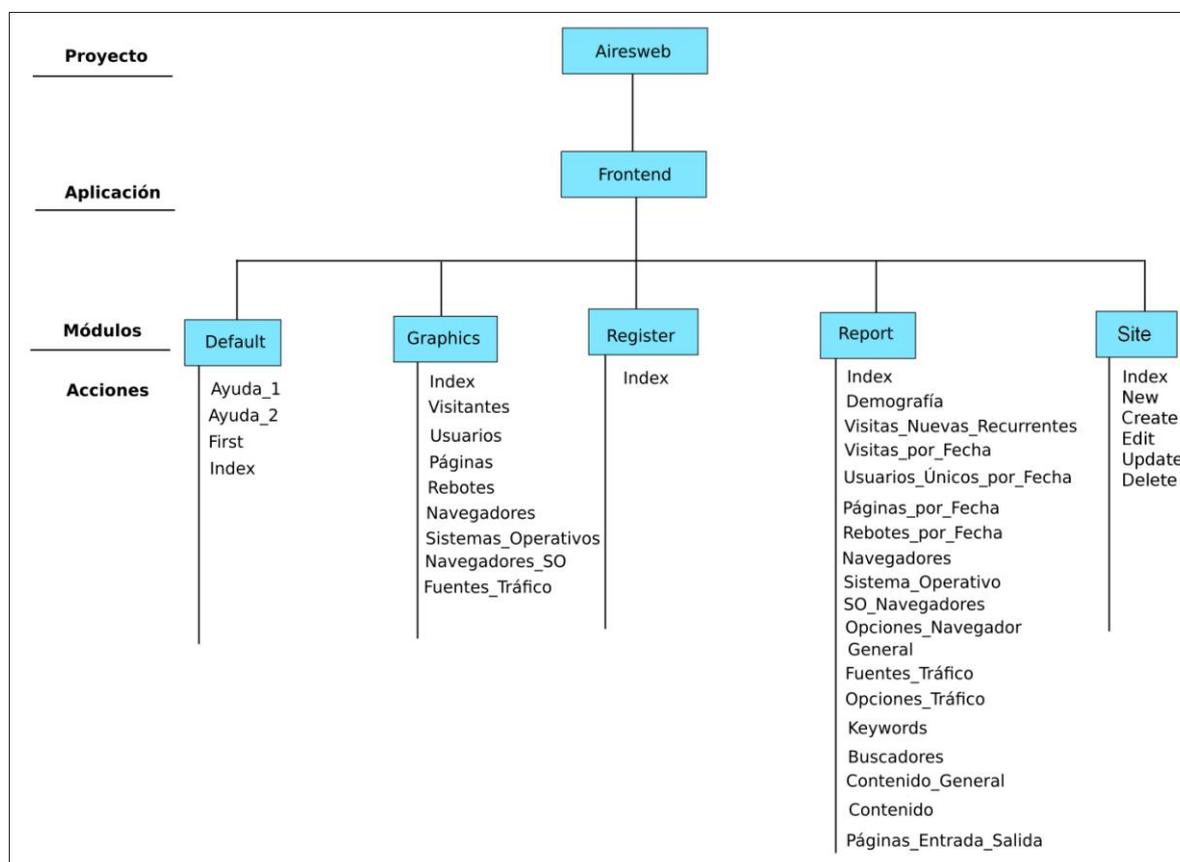


Ilustración 2.3: Organización de la aplicación.

2.8. Patrones de Diseño utilizados.

Symfony sigue las mejores prácticas y patrones de diseño para el desarrollo de aplicaciones Web. Seguidamente se mencionan cuales son los patrones de diseño 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 layout 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 quien 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 [39].

2.9. Validación y Tratamientos de Errores.

La validación y el tratamiento de errores se han diseñado de acuerdo con 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, sin la consecuente promoción de errores que puede traer consigo la programación relacionada con este aspecto [40].

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

2.10. 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 mas 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 [41]. 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.

2.11. Diagrama de Despliegue.

Un diagrama de despliegue modela los aspectos físicos de un sistema, la vista de despliegue estática de un sistema, una configuración de nodos y los componentes que residen en ellos. Describe la arquitectura en tiempo de ejecución de procesadores, dispositivos y los componentes de software que ejecutan esta arquitectura. Además, define la topología del sistema, estructura del hardware y el software que se ejecuta en cada unidad [42].

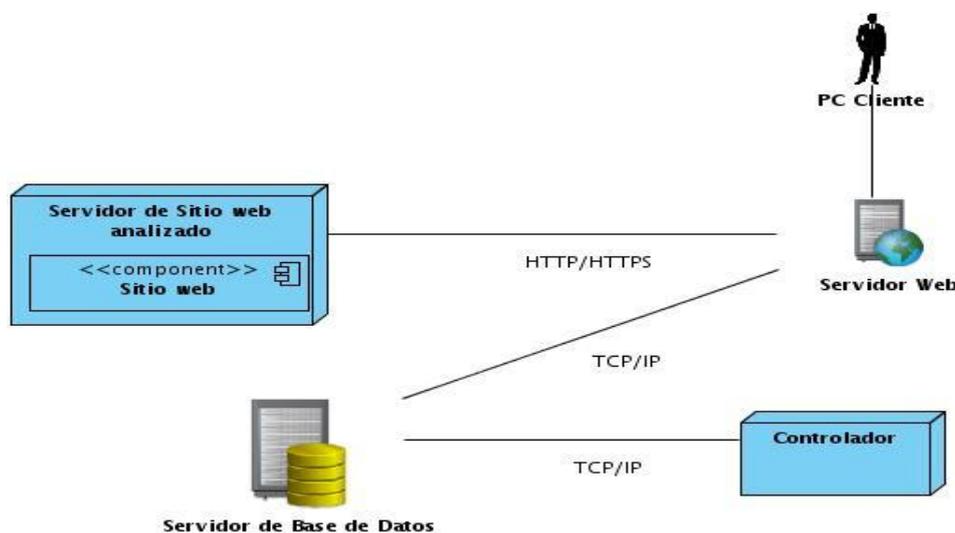


Ilustración 2.4: Diagrama de Despliegue de la aplicación.

2.12. Diagrama de Componentes.

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas. Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente.

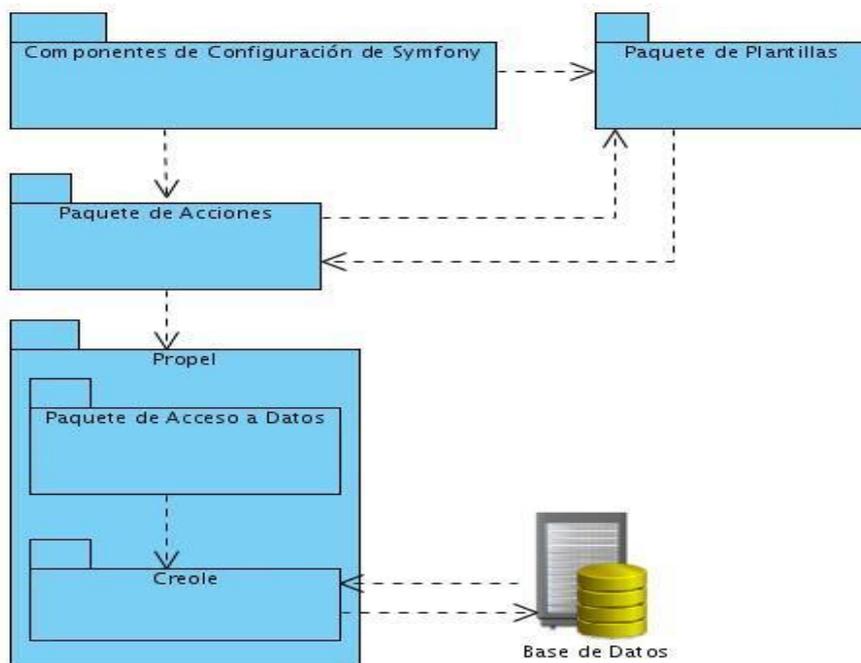


Ilustración 2.5: Diagrama de Componentes general.

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

2.13. Conclusiones parciales.

A partir de la descripción del problema se hizo posible elaborar una propuesta de solución e identificar los requisitos funcionales y no funcionales, desglosados en Historias de Usuario que cada desarrollador debe cumplir a través de una serie de una serie de Tareas Ingenieriles; y se desarrolló un plan de liberación para precisar el tiempo de desarrollo por iteraciones. Se describieron los elementos esenciales de la arquitectura de la solución propuesta, los componentes desarrollados y su integración. Se concibe una solución basada en los patrones de diseño y el patrón arquitectónico que implementa el framework Symfony, unido a los mecanismos de seguridad que provee. A partir de este proceso se alcanzó como resultado una versión funcional de la solución propuesta, que debe ser validada por un proceso de pruebas.

Capítulo 3. Validación de la solución propuesta.

Una parte muy importante en todo desarrollo de software es la etapa de pruebas, en la metodología SXP se requiere de una etapa de pruebas en cada iteración de desarrollo del producto, en estas etapas se le realizan pruebas de al software con el objetivo de verificar si se cumplieron con los objetivos trazados en la iteración.

3.1 Pruebas.

Las pruebas que se le realizan a todo producto de software son cruciales para ganar en la seguridad y la calidad del mismo, de modo que una vez sea entregado al cliente incurra en el menor número de inconvenientes posibles. Es recomendado realizar pruebas al software desde el mismo momento en que se implementa una nueva funcionalidad, logrando así un entorno de desarrollo satisfactorio. La metodología SXP divide las pruebas en dos grupos:

- Pruebas unitarias, desarrolladas por los programadores y encargadas de verificar el código de forma automática.
- Pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente.

3.1.1. Pruebas de Aceptación.

Las pruebas de aceptación son pruebas que se crean a partir de las historias de usuario. En ellas se especifican, desde la perspectiva del cliente, los escenarios para probar que una HU ha sido implementada correctamente. Una HU puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo final de estas pruebas es garantizar que los requerimientos hayan sido cumplidos y que la aplicación es realmente lo que el cliente quería. Una HU no se considera terminada hasta que no ha pasado sus pruebas de aceptación.

Se decidió realizar las pruebas de aceptación a la IW debido a que su objetivo es verificar que el sistema cumpla con los requisitos establecidos por el usuario, por lo que significa la satisfacción del cliente con el producto desarrollado. A continuación se muestran los casos de pruebas que fueron realizados al sistema en cada una de las iteraciones.

3.1.2. Casos de Pruebas.

Los casos de prueba son un conjunto de entradas con datos de prueba, unas condiciones de

ejecución, y unos resultados esperados con el propósito de identificar y comunicar las condiciones que se llevarán a cabo en la prueba. Los casos de pruebas son necesarios para verificar la aplicación exitosa y aceptable de los requisitos del producto [43].

En el documento adjunto ***pruebas.pdf*** se describen un conjunto de pruebas diseñadas y ejecutadas para validar la solución propuesta.

3.2. Conclusiones parciales.

En este capítulo se definieron, documentaron y aplicaron las pruebas de aceptación. Se validaron las Historias de Usuario y se comprobó que el sistema deniega las peticiones de los usuarios y mediante la IW muestra los mensajes de error correspondientes ante la entrada de datos incorrectos. Se garantiza entonces la calidad de la solución propuesta que hace lo que debe de hacer.

Conclusiones

- ❖ El estudio de las interfaces web de los principales sistemas de análisis de registros web en el ámbito nacional y extranjero, y las tendencias actuales de las mismas permitió elegir las herramientas, tecnologías, los lenguajes y metodología para guiar el proceso de desarrollo.
- ❖ A partir de la definición de la Lista de Reserva del Producto se construyeron las Historias de Usuario, aspecto fundamental y de gran ayuda para el desarrollo de la solución obtenida.
- ❖ La generación del diagrama de componentes general y otros relacionados, teniendo presente la arquitectura MVC de Symfony, permitió obtener una mayor comprensión del sistema y definió los principios que guiaron la implementación de la solución obtenida.
- ❖ El uso de Symfony permitió acelerar el desarrollo de la aplicación, ahorrando de esta manera tiempo y esfuerzos. Además, se definieron y aplicaron las Pruebas de Aceptación, aumentando la calidad y seguridad de la solución obtenida.
- ❖ Se realizó un estudio del estado del arte de aplicaciones similares, se definieron las características de la IW a desarrollar y se implementó y evaluó la IW para la gestión del sistema Airesweb. Cumpliéndose de esta manera con todos los objetivos planteados al inicio del presente trabajo y comprobándose la validez de la idea a defender materializada en la solución obtenida.

Recomendaciones

Al concluir el siguiente trabajo se recomienda:

- ❖ Agregar la funcionalidad de generar reportes en formato portable (PDF) y el envío automático de este vía email.
- ❖ Agregar la funcionalidad de mostrar a los usuarios los reportes estadísticos de manera personalizada.
- ❖ Desplegar el sistema Airesweb en el país para contribuir a la utilización de herramientas de análisis de registros web, y fomentar el uso de la analítica web y el software libre en Cuba.
- ❖ Lograr la internacionalización de la IW obtenida como solución.

Referencias Bibliográficas.

- [1]. **Shneiderman, Ben.** *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* Massachusetts : Addison Wesley, 1998.
- [2]. **Myers, Brad A.** *User interface software technology.* March 1996.
- [3]. **Loaiza., Dr. Edgar L. Chavez Gonzalez y Cuauhtemoc Rivera.** Interfaz de Usuario. [En línea] 23 de mayo de 2000. [Citado el: 09 de diciembre de 2009.] . [En línea] 23 de mayo de 2000. [Citado el: 09 de diciembre de 2009.] <http://www.fismat.umich.mx/~crivera/tesis/node6.html/>.
- [4]. WebTrends. [En línea] 2009. [Citado el: 10 de enero de 2010.] <http://www.dvc.es/es/alianzas/alianzas/Alianza8.html/>.
- [5]. AWStats Official Website. [En línea] [Citado el: 10 de enero de 2010.] <http://AwStats.sourceforge.net/>.
- [6]. WebLog Expert. [En línea] 2002. [Citado el: 11 de enero de 2010.] <http://www.weblogexpert.com/>.
- [7]. Google Analytics, La solución de análisis web para empresas gratuitas, y más inteligente y fácil de usar. [En línea] [Citado el: 11 de enero de 2010.] <http://www.google.com/analytics/>.
- [8]. Piwik - Web analytics #Open source web analytics. [En línea] [Citado el: 14 de enero de 2010.] <http://piwik.org/>.
- [9]. Piwik – Alternativa opensource a Google Analytics. [En línea] [Citado el: 14 de enero de 2010.] <http://www.incubaweb.com/piwik-alternativa-opensource-a-google-analytics/>.
- [10]. Arquitectura Web. [En línea] 200. [Citado el: 2010 de enero de 15.] <http://www.inteligenciaweb.com/disenio-paginas-web/arquitectura-web.htm/>.
- [11]. EL MODELO CLIENTE/SERVIDOR. [En línea] [Citado el: 17 de enero de 2010.] <http://agamenon.uniandes.edu.co/~revista/articulos/cliser.html/>.
- [12]. 19. ¿Por qué elegir PHP?. Programación en Castellano. [En línea] [Citado el: 23 de enero de 2010.] http://www.programacion.com/articulo/por_que_elegir_php_143/.
- [13]. Distribución mundial de Php. [En línea] [Citado el: 23 de enero de 2010.] <http://www.tufuncion.com/php-mundial/>.
- [14]. PHP: ¿Qué es PHP? - Manual. [En línea] [Citado el: 23 de enero de 2010.] <http://docs.php.net/manual/es/intro-what-is.php/>.
- [15]. LinuxCentro.net - Características de PHP. [En línea] [Citado el: 23 de enero de 2010.]

<http://www.linuxcentro.net/linux/staticpages/index.php?page=CaracteristicasPHP/>.

[16]. W3c Oficina Española. [En línea] [Citado el: 25 de enero de 2010.]
<http://www.w3c.es/divulgacion/a-z/#html/>.

[17]. **Arocena, Francisco.** *Gran Libro de HTML*. 1997.

[18]. W3c Oficina España. [En línea] [Citado el: 26 de enero de 2010.] <http://www.w3c.es/divulgacion/a-z/#css/>.

[19]. **Pérez, Javier E.guíluz.** *Introducción a JavaScript*. 2009.

[20]. Digital. CSIC. [En línea] [Citado el: 27 de enero de 2010.]
http://digital.csic.es/bitstream/10261/5108/1/Comunicacion_TCO-294-2007JS.pdf/.

[21]. **Garrett, Jesse James.** Ajax: A New Approach to Web Applications. [En línea] 18 de febrero de 2005. [Citado el: 27 de enero de 2010.]
<http://www.adaptivepath.com/ideas/essays/archives/000385.php/>.

[22]. Conceptos de Ajax . [En línea] [Citado el: 27 de enero de 2010.]
<http://www.ajaxya.com.ar/temarios/descripcion.php?cod=34&punto=2/>.

[23]. **Age, Clever.** *Libro blanco sobre Frameworks PHP para empresas*. 4 julio de 2008 .

[24]. **Verdoy, Alberto.** Ventajas de usar frameworks para PHP. [En línea] 15 de julio de 2009. [Citado el: 30 de enero de 2010.] <http://www.tucamon.es/contenido/ventajas-de-usar-frameworks-en-php/>.

[25]. **Inc, Ellislab.** *Welcome to CodeIgniter: CodeIgniter User Guide*. 2009.

[26]. **D.Teejda, A.Gutierrez y.** *Kumbia PHP Framework. ¿Por qué programar debería ser más fácil?*

[27]. **Foundation, Inc.Cake Software.** *El manual: 1.2 Colección: El Cookbook*. 2009.

[28]. **Ltd, Zend Technologies.** *Zend Framework*. 2009.

[29]. **Steinmetz, Alejandro.** El framework Symfony. [En línea] 2009. [Citado el: 01 de febrero de 2010.]
<http://knol.google.com/k/alejandro-steinmetz/el-framework-symfony/gbb8pke8a5fx/2#/>.

[30]. **Mateu, Carles.** *Desarrollo de aplicaciones web*. SL : Eureka Media, 2004. Primera Edición.

[31]. **Kabir, Mohammed J.** La biblia del servidor Apache2. [En línea] [Citado el: 02 de febrero de 2010.]
<http://bibliodoc.uci.cu/pdf/reg01737.pdf/>.

[32]. **Franklin, Jonathan Ramos Laya.** *Evaluación de resultados de componentes del Modulo 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.*

Venezuela : Licenciados en computación, Facultad de Ciencias – Universidad Central de Venezuela, 2009.

[33]. PHP Development Tools Project. [En línea] [Citado el: 02 de febrero de 2010.] <http://www.eclipse.org/pdt/>.

[34]. **(UCI), Universidad de las Ciencias Informáticas.** Entorno Virtual de Aprendizaje - Introducción a la Ingeniería de Software. [En línea] [Citado el: 03 de febrero de 2010.] <http://eva.uci.cu/mod/resource/view.php?id=11402/>.

[35]. Principles behind the Agile Manifesto. [En línea] 2001. [Citado el: 03 de febrero de 2010.] <http://agilemanifesto.org/principles.html/>.

[36]. **Romero., Abel Meneses Abad y Gladys Marsi Peñalver.** *SXP, metodología ágil para proyectos de software libre.* Marzo 2009.

[37]. Object Management Group - UML. [En línea] [Citado el: 04 de febrero de 2010.] <http://www.uml.org/>.

[38]. Visual Paradigm for UML. [En línea] [Citado el: 04 de febrero de 2010.] <http://www.visualparadigm.com/product/vpumf/>.

[39]. **Potencier, Francois Zaninotto Fabien.** *Symfony la guía definitiva.* 2008.

[40]. **Larman, Craig.** *UML y patrones Tomo II.* 2003.

[41]. **Daniele, Marcela.** *Teoría 11: El arte de modelar.* Universidad Nacional de Rio Cuarto (UNRC)- Argentina : s.n., 2007.

[42]. Modelamiento Visual y UML. [En línea] [Citado el: 30 de abril de 2010.] <http://www.slideshare.net/dersteppenwolf/modelamiento-visual-y-uml/>.

[43]. **(UCI), Universidad de las ciencias Informáticas.** Entorno Virtual de Aprendizaje. [En línea] [Citado el: 02 de mayo de 2010.] http://eva.uci.cu/file.php/259/Curso_2009-2010/Conferencia_7/Materiales_complementarios/Material_EI_Como_diseñar_casos_de_pruebas_a_partir_de_los_casos_de_uso.doc/.

[44]. Blog de Analítica Web en Cuba. [En línea] [Citado el: 23 de Febrero de 2010.] <http://analiticaweb.blogcip.cu/2009/11/18/celebrado-i-evento-de-analitica-web/>.

[45]. Desarrollo Web. *Componentes de una interfaz web. Interfaces.* [En línea] 22 de Septiembre de 2007. [Citado el: 22 de Febrero de 2010.] <http://www.desarrolloweb.com/articulos/2171.php>.

Bibliografía

1. Blog de Analítica Web. [En línea] <http://analiticaweb.blogcip.cu/>.
2. Manuales de Desarrollo web, diseño, programación, sistemas. [En línea] <http://www.desarrolloweb.com/manuales/>.
3. Kniberg, Henrik. SCRUM Y XP DESDE LAS TRINCHERAS. ¿Cómo hacemos SRUM? s.l. : <http://infoq.com/minibooks/scrum-xp-fromthetrenches>, 2007.
4. Manual rápido para utilizar HTML. Guadalajara : Coordinación General del Sistema para la Innovación del Aprendizaje.Universidad de Guadalajara.
5. Politécnica, Universidad. Metodologías ágiles para el desarrollo de software: EXtreme Programming (XP). Valencia : Laboratorio de Sistemas de Información. Departamento de Sistemas Informáticos y Computación.Facultad de Informática.Universidad Politécnica.
6. Indianapolis, Derick Rethans-Andi Gutmans-Stig Sæther Bakken. PHP 5 Power Programming. Indianapolis : s.n., Septiembre 15, 2004.
7. PHP: Información General-Manual. [En línea] <http://docs.php.net/manual/es/>.
8. Pérez, Javier E.guiluz Symfony.es. [En línea] <http://www.symfony.es/>.
9. Extensible Markup Language (XML). [En línea] 2003. <http://www.w3.org/XML/>.
10. Daniele, Marcela. Teoría 11: EL ARTE DE MODELAR UML (Unified Modeling Language). 2007.
11. Retschitzegger, Werner-Gerti Kappel-Birgit Proll-Siegfried Reich. Web Engineering. The Discipline of Systematic Development. Heidelberg, Germany : dpunkt.verlag, 2003. ISBN: 3-89864-234-8.
12. Day, Robert A. Cómo escribir y publicar trabajos científicos. Tercera edición en español. 525 Twenty-third Street, NW, Washington, DC 20037, EUA : s.n., 2005. Publicación Científica y Técnica No. 598.
13. González, Sayda Coello-Rolando Alfredo Hernández León. EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTIFICA. UCI, Ciudad de la Habana, Cuba : EDUCNIV-Editorial Universitaria, Noviembre del 2002. ISBN: 959-16-0343-6.
14. Machado, Oscar Andrés Casas -Adrian Gallardo Palay. Interfaz Web para el sistema de análisis Airesweb. Trabajo de Diploma para optar por el Título de Ingeniería en Ciencias Informáticas. UCI, Ciudad de la Habana, Cuba : s.n., 19 de mayo de 2009.
15. García, Susel Pino-Abel Meneses Abad-Gladys Marsi Peñalver Romero-Malay Rodríguez Villar-

Raycel Fernández Céspedes. SXP. Metodología ágil para proyectos de software libre. UCI, Ciudad de la Habana, Cuba : s.n., Marzo de 2009.

16. Zaninotto, François-Fabien Potencier. Symfony. La guía definitiva. s.l. : http://www.librosweb.es/symfony_1_0/, 24 de diciembre de 2008.

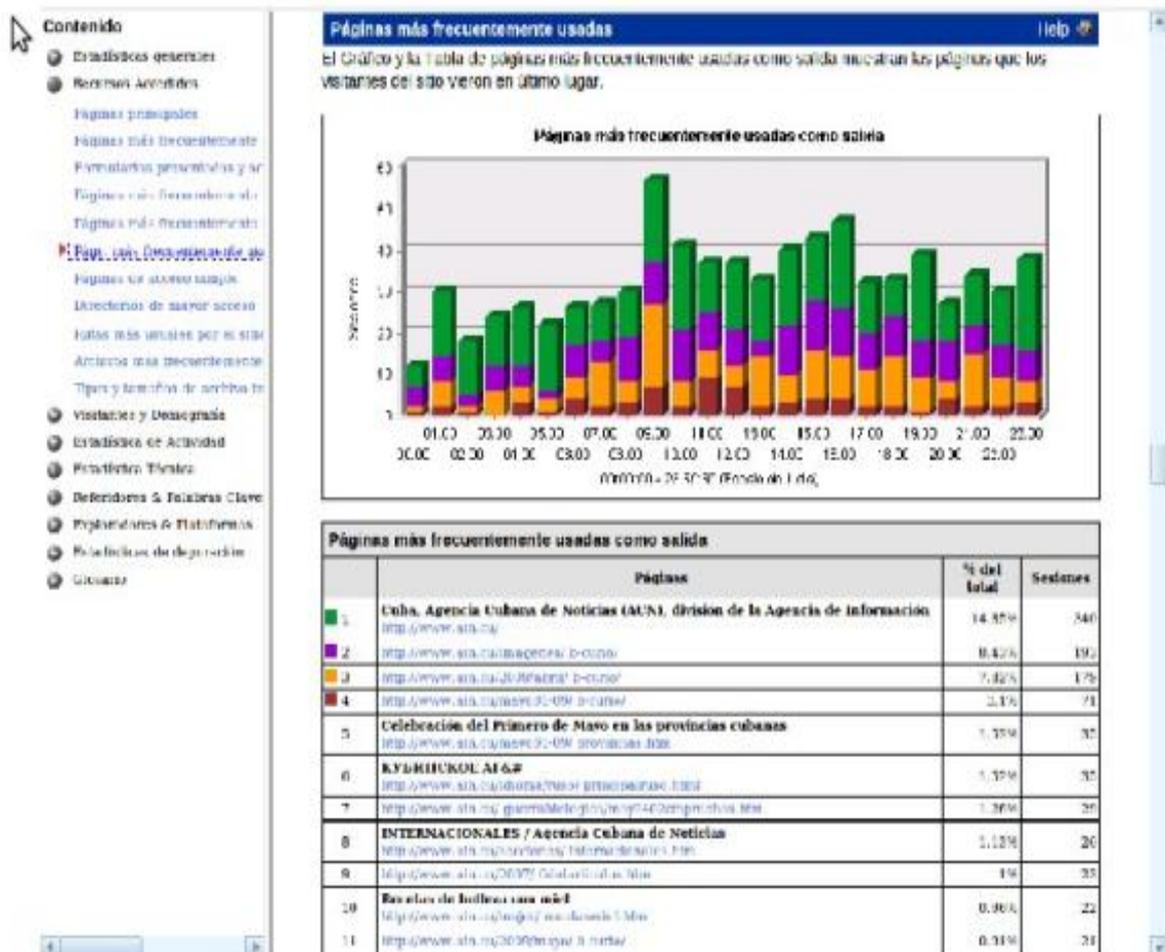
17. Pérez, Javier E.guíluz. Introducción a JavaScript. s.l. : <http://www.librosweb.es/javascript/>, 25 de marzo de 2009.

18. Pérez, Javier E.guíluz. Introducción a XHTML. s.l. : <http://www.librosweb.es/xhtml/>, 17 de diciembre de 2008.

19. Pérez, Javier E.guíluz. Introducción a Ajax. s.l. : <http://www.librosweb.es/ajax/>, 7 de junio de 2008.

20. Pérez, Javier E.guíluz. Introducción a CSS. s.l. : <http://www.librosweb.es/css/>, 8 de mayo de 2009.

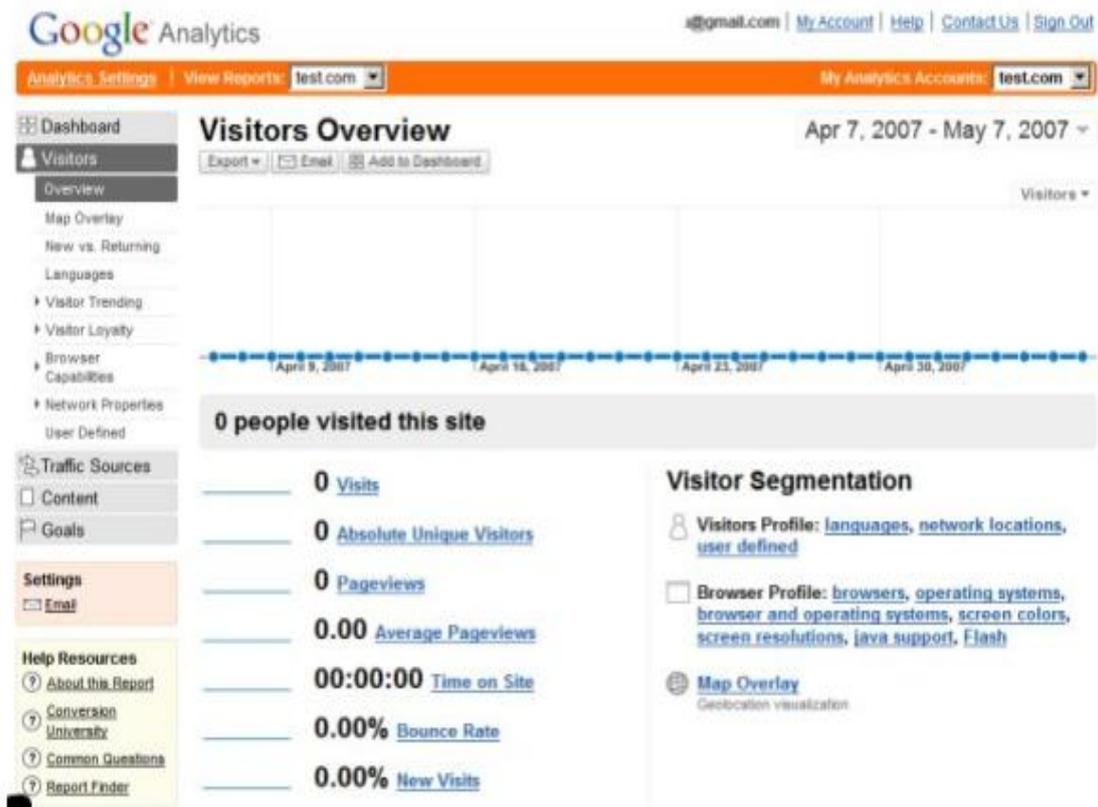
Anexo A. Figuras Relacionadas



A 1: Interfaz Web de WebTrends.



A 2: Interfaz Web de WeblogExpert.



A 4: Interfaz Web de GoogleAnalytics.

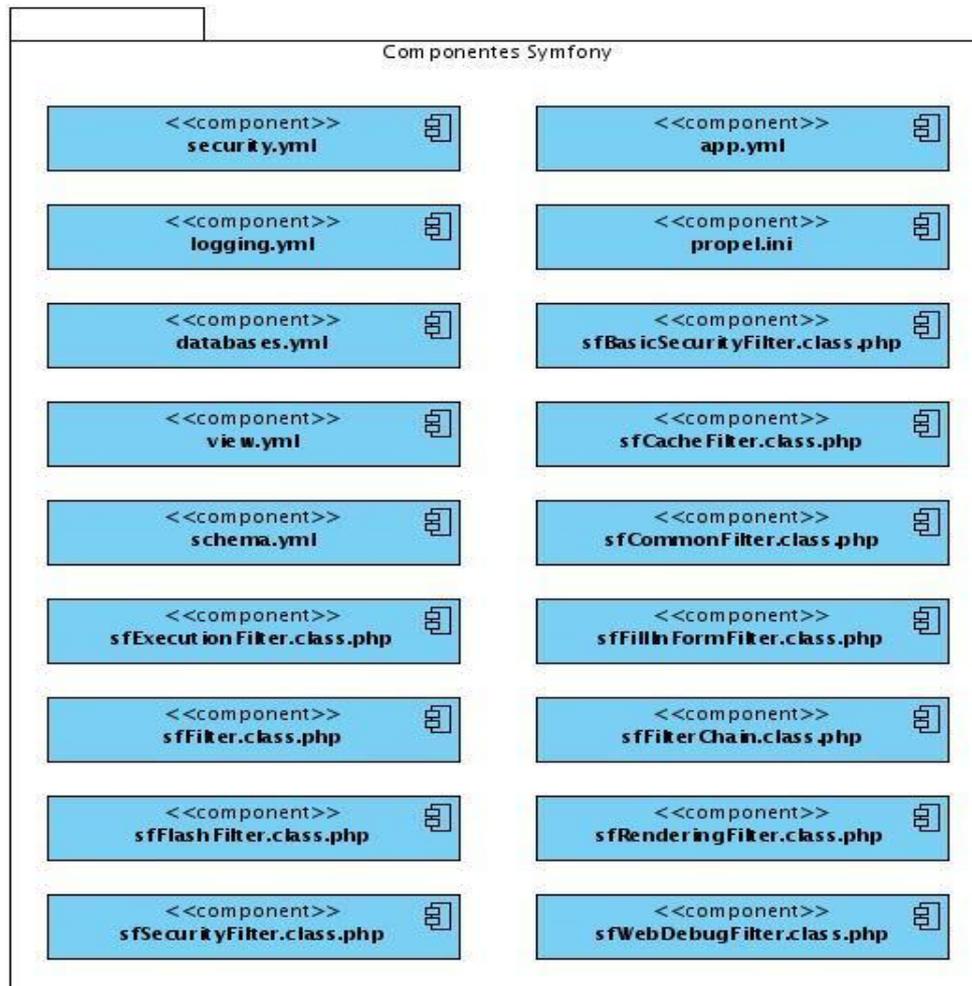


A 5: Interfaz Web de Piwik.

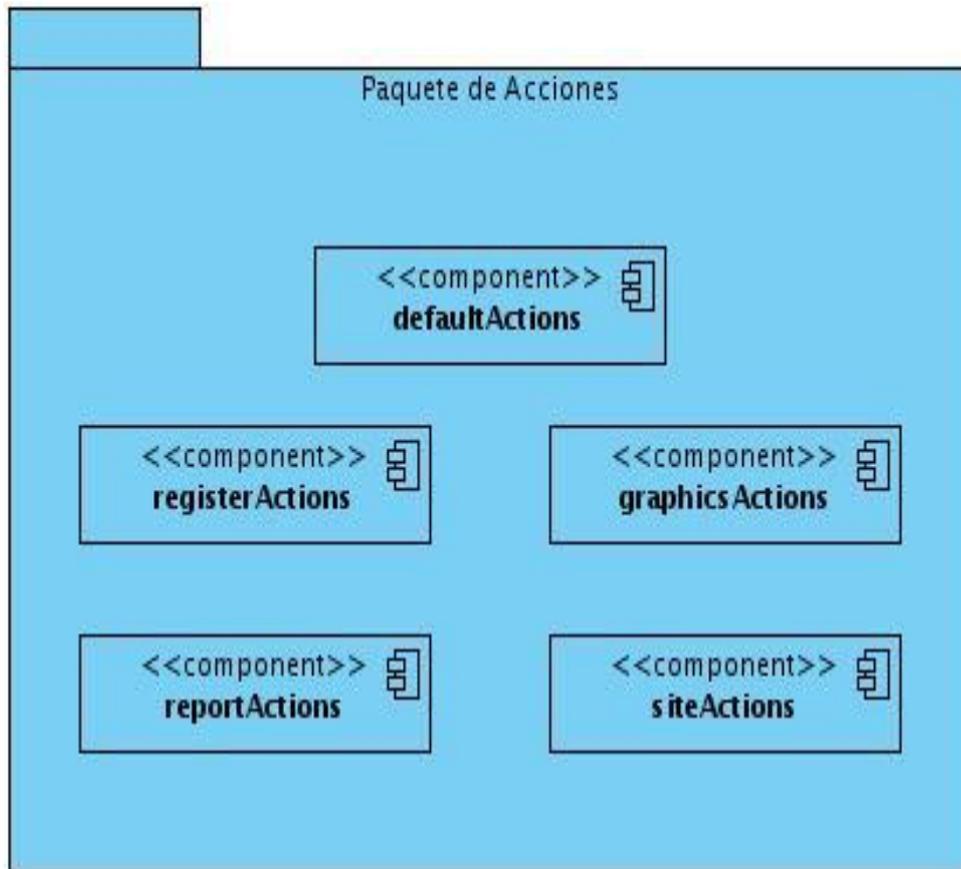


A 6: Interfaz Web de Airesweb (Versión Anterior).

Anexo B. Diagramas de Componentes



B 1: Diagrama de Componentes de Symfony.



B 2: Diagrama de Componentes Acciones.



B 3: Diagrama de Componentes Plantillas.

Glosario de Términos

SXP: *SCRUM + XtremeProgramming (XP)*. Metodología ágil desarrollada como un híbrido de SCRUM y XP. La finalidad es garantizar la eficacia y la eficiencia en el proceso de generación de software.

GUI: Interfaz Gráficas de Usuario.

GUIs: Interfaces Gráficas de Usuario.

Software: Conjunto de programas elaborados por el hombre, que controlan la actuación del computador, haciendo que éste siga en sus acciones una serie de esquemas lógicos predeterminados y pueda desempeñar tareas inteligentes. Suele sustituirse por expresiones tales como programas (informáticos) o aplicaciones (informáticas).

CIPRE: Centro de Información para la Prensa.

SINI: Departamento de Soluciones Informáticas para Internet.

Airesweb: Analizadores Inteligentes de Registros Web.

Aires: Analizadores Inteligentes de Registros de Servidores.

UCI: Universidad de las Ciencias Informáticas. Creada por el Comandante en Jefe Fidel Castro Ruz en el año 2002.

IW: Interfaz Web.

BD: Base Datos.

PHP: Hypertext Preprocessor. Es un lenguaje de código abierto interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor.

UML: *Modelado Unified Modeling* (Lenguaje Unificado de Modelado). Es uno de los lenguajes de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar.

SO: Sistema Operativo.

IDE: Entorno de Desarrollo Integrado.

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).

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*.

HTML: *Hypertext Markup Language*. Es un lenguaje de programación que se utiliza para el desarrollo de páginas web. Es un lenguaje estático, desarrollado por el World Wide Web Consortium (W3C). Está constituido por elementos que el navegador interpreta y los despliega en la pantalla de acuerdo a su objetivo. Es utilizado para la publicación de hipertexto en la Web y desarrollado con la idea de que cualquier persona o tipo de dispositivo pueda acceder a la información en la Web.

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.

CSS: *Cascading Style Sheets*. 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.

AJAX: *Asynchronous Javascript And XML* (JavaScript asíncrono y XML). No es una tecnología en sí misma. En realidad, se trata de varias tecnologías independientes que se unen de formas nuevas y sorprendentes. Es una técnica de desarrollo web que combina Javascript, XML, XSLT, [XHTML](#), CSS y el DOM, para crear aplicaciones interactivas que se ejecutan en el cliente realizando llamadas asíncronas al servidor. Permite realizar cambios en las páginas web sin necesidad de volver a cargarlas.

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.

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.

MVC: *Model-View-Controller* (Modelo-Vista-Controlador). Es una arquitectura donde el Modelo es todo acceso a base de datos, y las funciones que llevan lo que llaman "lógica de negocio". La Vista, en una aplicación web, es el HTML (o XML, WAP, PDF) y lo necesario para convertir datos en HTML y el Controlador es lo que une la vista y el modelo.

ORM: *Object-Relational Mapping* (Mapeo Objeto-Relacional). Es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional.

RF: Requisito Funcional.

RNF: Requisito No Funcional.

SSL: *Secure Socket Layer* (Protocolo de Capa de Conexión Segura). Es un protocolo que se ha convertido en el estándar universal para la autenticación de sitios y para codificar la comunicación entre usuarios y servidores de Web.

SQL: Structured Query Language.

APIs: Interfaz de Aplicación.

LRP: Lista de Reserva del Producto.

HU: Historia de Usuario.

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 sí.

CGI: Common Gateway Interface (en español Interfaz de entrada común) es una importante tecnología de la *World Wide Web* que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web. Especifica un estándar para transferir datos entre el cliente y el programa. Es un mecanismo de comunicación entre el servidor web y una aplicación externa.

CLF: *Common Log Format* (en español Formato Común de Registro) es un formato de archivo de texto estándar utilizado por los servidores web para la generación de archivos de registro. Dado que el formato está estandarizado, los archivos pueden ser analizados por una variedad de programas de análisis de registros.

Smashing Magazine: Revista online (<http://www.smashingmagazine.com>) con información de utilidad para diseñadores gráficos y desarrolladores web. Posee información actualizada en cuanto a las tendencias del diseño y desarrollo de sitios web.