

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

**DESARROLLO DE UN SISTEMA DE CONSTRUCCIÓN DE WEBS 2.0
E INTEGRACIÓN CON UN SISTEMA DE VENTA DE DOMINIOS**

Tesis para optar por el Título de Ingeniero Informático, que presenta el bachiller:

Jonathan Olger Nieto Lajo

ASESOR: Luis Alberto Flores García

Lima, 11 de Junio del 2011

Resumen

Hoy en día, contar con un sitio web 2.0 constituye una ventaja competitiva muy importante para las empresas unipersonales y pymes (pequeñas y medianas empresas). Una parte de dichas empresas no cuentan con una herramienta sencilla e intuitiva que les facilite la creación de su propia página web.

En el proceso de desarrollar o adquirir un sitio web se presentan un conjunto de ventajas y desventajas de acuerdo a un número de factores como, quién desarrolla el sitio web, dónde se aloja el sitio web y quién brinda mantenimiento al sitio web. La elección en cada uno de estos factores, como si el desarrollo lo realiza la misma empresa o se encarga a un proveedor, determina estas ventajas y desventajas, como que el hecho de realizar cambios significativos en el sitio web sea más lento.

Se propone brindar un servicio que permita a una empresa gozar de mayores ventajas en los factores mencionados mediante la utilización de un conjunto de herramientas que le faciliten la creación y mantenimiento de su sitio web. Así también, automatizar la compra de un nombre de dominio y despliegue que permitan el acceso a millones de visitantes del Internet al sitio web.

El Sistema de Construcción de Webs 2.0, es una herramienta muy útil y sencilla cuyo objetivo es facilitar la creación de sitios web para pequeñas empresas. La creación se realiza mediante una interfaz sumamente intuitiva y sencilla. Consta de un conjunto de plantillas pre-configuradas sobre las cuales se puede iniciar con la creación del sitio web. Cuenta con un conjunto muy variado de bloques de contenido entre los que se tiene: Texto, Imagen, Galería, Video, Audio, Facebook, Twitter y RSS. También se cuenta con integración con las principales redes sociales mediante los bloques de contenido de Facebook y Twitter. Con el objetivo de que el uso del sistema sea fácil y sencillo, para el caso de los bloques de contenido, se utilizan las funcionalidades de jalar y situar (drag and drop) y redimensionar (resize). El sistema también permite publicar automáticamente un nuevo sitio web a Internet por medio de un nombre de dominio.

El proyecto se desarrolla en un escenario real en el cual se pueden aplicar las mejores prácticas en cuanto a gestión y desarrollo de proyectos.

TEMA DE TESIS PARA OPTAR EL TÍTULO DE INGENIERO INFORMÁTICO

TÍTULO: DESARROLLO DE UN SISTEMA DE CONSTRUCCIÓN DE WEBS 2.0
E INTEGRACIÓN CON UN SISTEMA DE VENTA DE DOMINIOS

ÁREA: SISTEMAS DE INFORMACIÓN

PROPONENTE: Ing. Luis Alberto Flores García

ASESOR: Ing. Luis Alberto Flores García

ALUMNO: Jonathan Olger Nieto Lajo

CÓDIGO: 20030301

TEMA N°: 412

FECHA: San Miguel, 20 de marzo de 2012



DESCRIPCIÓN

En la actualidad tener un sitio web 2.0 constituye una ventaja competitiva muy importante para las empresas unipersonales, SOHOs (small office/home office, empresas de 1 a 10 trabajadores), y pymes (pequeñas y micro empresas). Estas empresas no cuentan con una herramienta intuitiva y sencilla que les facilite la creación de su propia página web.

En el proceso de desarrollar o adquirir un sitio web como también en su mantenimiento se dan un conjunto de ventajas y desventajas de acuerdo a la solución elegida por la empresa.

Se presentan cinco escenarios:

1. La empresa cliente contrata un proveedor para que desarrolle y mantenga su sitio web y también contrata un proveedor para que le brinde el servicio de alojamiento de sitios web.
2. La empresa cliente contrata un proveedor para que desarrolle y mantenga su web pero ella misma lo aloja.
3. La empresa cliente desarrolla y mantiene su propio sitio web y contrata a un proveedor para que le brinde el servicio de alojamiento de sitios web.
4. La empresa cliente desarrolla y mantiene su propio sitio web y ella misma lo aloja.
5. La empresa cliente contrata un proveedor para que desarrolle su sitio web, pero ella misma le brinda mantenimiento. En este caso puede contratar a un proveedor, como también ella misma puede alojar el sitio web.

Cada uno de los escenarios mencionados tiene un conjunto de ventajas y desventajas, las cuales se deben considerar al momento de elegir una de las opciones de acuerdo a las necesidades del usuario.

Así se tiene que cuando se contrata a un proveedor para la realización de una página web se realiza un solo pago, sin embargo si surgen cambios, se necesitan realizar una serie de pagos por cada uno de ellos, pagos que en general no son bajos. En contraste si la empresa realiza su propio sitio web necesita contratar personal que lo haga, lo cual facilita y acelera el proceso de cambio en la misma. Sin embargo se consumen recursos en actividades que no son parte del núcleo del negocio. En un caso intermedio la empresa contrata a un proveedor para que desarrolle el sitio web y ella misma le brinda mantenimiento. De esta manera si existen muchos cambios o actualizaciones pequeñas el gasto es mínimo, pero si se requieren actualizaciones o cambios considerables el gasto es grande.

Según el cuarto escenario, se encuentran un conjunto de servicios de auto desarrollo de sitios web, en donde la empresa construye por medio de una interfaz amigable su propio sitio web y tiene la facultad de cambiarlo cuando desea. Sin embargo, estos servicios a pesar de ofrecer muchas funcionalidades, no son muy fáciles de utilizar, como tampoco permiten dar pleno control del diseño al usuario.

Se necesita brindar un servicio que permita a una empresa realizar su propio sitio web utilizando un conjunto de herramientas que le faciliten esta tarea, englobadas en un sistema. Así también, automatizar la compra de un dominio por el cual acceder al sitio web. De esta forma se permite que la empresa pueda construir su propia web como también de actualizarla según requiera y brindar acceso a través de un nombre de dominio.

El Sistema de Construcción de Webs 2.0, será una herramienta muy útil para facilitar la creación de sitios web al tipo de empresas indicado. El sistema permite la creación de sitios web mediante una interfaz sumamente intuitiva y sencilla. Así como publicar automáticamente estos sitios web mediante un nombre de dominio.

OBJETIVO

Realizar el análisis, diseño e implementación de un sistema que permita brindar un servicio intuitivo y sencillo de construcción de un sitio web 2.0 como también la integración de este sistema con otro de venta de dominios.

OBJETIVOS ESPECÍFICOS

Los objetivos específicos del presente proyecto son:

- Realizar un estudio de las principales funcionalidades y requisitos que requiere un sitio web empresarial.
- Realizar el análisis y evaluación de las funcionalidades factibles de implementación en el sistema.
- Diseñar el sistema considerando las funcionalidades factibles de implementación.
- Implementar el sistema en base al diseño realizado, como también realizar los casos de prueba correspondientes.
- Integrar el sistema de creación de webs 2.0 con el sistema de gestión y venta de dominios.
- Desplegar el sistema de creación de webs 2.0 y realizar las correspondientes pruebas de integración.

ALCANCE

El sistema se basará en la realización de un sitio web a partir de plantillas pre-configuradas. Cada sitio web tendrá un conjunto de páginas. Cada página puede tener bloques de diferentes tipos como Texto, Imagen, Galería, Video, Audio, Facebook, Twitter y RSS.

El proyecto también incluye la configuración en el sitio web de título, colores, formato (disposición de cabecera, cuerpo y pie de página), menú y fuente.

La realización de un sitio web debe ser un proceso fácil e intuitivo por ello por ejemplo para el caso de los bloques y su posicionamiento se debe poder jalar y situar (drag and drop), así como redimensionar (resize).

También se considera como parte del proyecto la integración del mismo con el sistema de venta de dominios, el cual incluye el proceso de compra.

No se considera como parte del proyecto el análisis, diseño y construcción del sistema de venta de dominios.

Máximo: 100 páginas

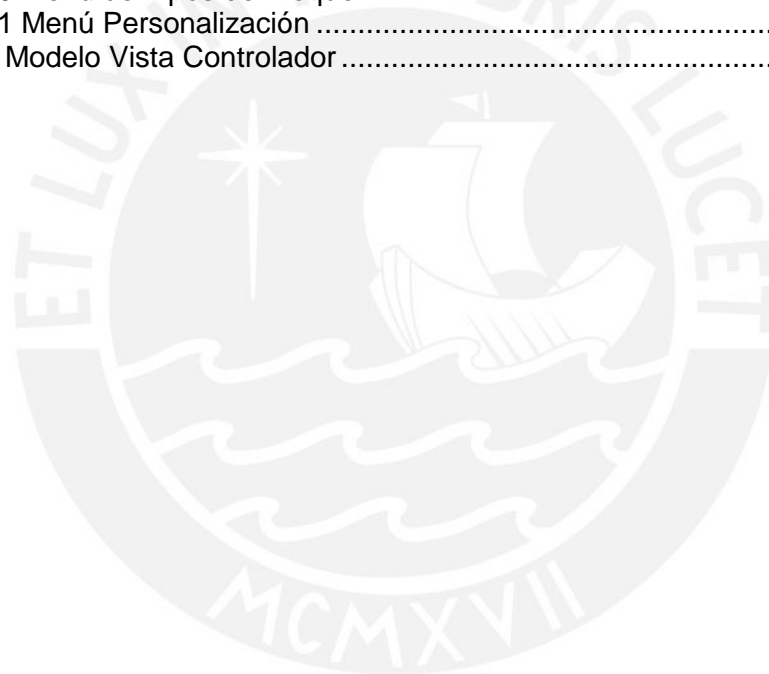
Tabla de Contenido

Resumen.....	I
Tabla de Contenido.....	V
Índice de Figuras.....	VII
Índice de Tablas.....	VIII
1. Generalidades	1
1.1. Definición del Problema	1
1.2. Marco Conceptual.....	4
1.2.1. Proceso de petición de una página web.....	4
1.2.2. HTML, CSS y Javascript.....	5
1.2.3. Nombres de Dominio	7
1.2.4. Web 2.0	7
1.2.5. WYSIWYG	9
1.2.6. Wiki.....	10
1.2.7. RSS	10
1.2.8. Facebook.....	11
1.2.9. Twitter	11
1.2.10. Sistema de Venta de Dominios	12
1.3. Estado del Arte	12
1.3.1. Servicios Existentes	12
1.3.1.1. Google Sites (https://sites.google.com/)	12
1.3.1.2. Negocio en Línea (http://www.negocioenlinea.com/)	12
1.3.1.3. Yahoo Small Business (http://smallbusiness.yahoo.com/webhosting)	12
1.3.1.4. About Me (https://about.me/)	13
1.3.1.5. CMS Drupal.....	13
1.3.1.6. CMS Joomla.....	13
1.3.2. Cuadro comparativo.....	14
1.4. Plan de Proyecto.....	15
1.4.1. Estructura de Descomposición del Trabajo (EDT).....	18
1.4.2. Diagrama de Gantt.....	19
1.5. Descripción y sustentación de la solución	22
2. Análisis	24
2.1. Metodología aplicada para el desarrollo de la solución	24
2.2. Identificación de Requerimientos	27
2.2.1. Lista de requerimientos.....	27
2.2.2. Casos de Uso	30
2.3. Análisis de la solución.....	33
2.3.1. Análisis de factores técnicos	33
2.3.2. Análisis de factores económicos	33
2.3.3. Restricciones de costo y tiempo.....	34
2.3.4. Análisis de viabilidad.....	35
2.3.5. Clases de Análisis.....	36
3. Diseño	38
3.1. Arquitectura de la solución	38
3.1.1. Representación de la arquitectura.....	38
3.1.2. Metas y Restricciones de la Arquitectura	39
3.1.3. Vista Lógica	40
3.1.4. Vista de Despliegue	41
3.1.5. Vista de Implementación.....	42
3.2. Diseño de Interfaz Gráfica.....	46
3.2.1. Pantalla de selección de plantilla de un sitio web.....	46
3.2.2. Pantalla principal de edición de un sitio web	48
3.2.2.1. Menú Principal.....	48

3.2.2.2.	Menú de Tipos de Bloque	49
3.2.2.3.	Menú de Personalización	50
4.	Construcción.....	51
4.1.	Construcción	51
4.1.1.	Tecnologías	51
4.1.1.1.	Lenguajes de Programación	52
4.1.1.2.	Servidor Web.....	55
4.1.1.3.	Tecnologías de Presentación	56
4.1.1.4.	Gestor de Base de Datos	60
4.1.1.5.	Entorno de Desarrollo.....	62
4.1.2.	Patrones de Diseño.....	64
4.1.2.1.	Modelo Vista Controlador	64
4.1.2.2.	Modelo Vista Presentador	66
4.1.3.	Frameworks	67
4.1.3.1.	Framework de PHP	67
4.1.3.2.	Framework de Javascript.....	69
4.1.3.3.	Framework ORM (Object Relational Mapping)	72
4.2.	Pruebas	74
4.2.1.	Estrategia de Pruebas.....	74
4.2.2.	Tipos de Pruebas.....	75
4.2.2.1.	Pruebas Unitarias	75
4.2.2.2.	Pruebas de Integración.....	75
4.2.2.3.	Pruebas de Casos de Uso.....	75
4.2.2.4.	Pruebas de Aceptación.....	75
4.2.3.	Catálogo de Pruebas	76
4.2.3.1.	Módulo de Configuración Sitio Web.....	76
4.2.3.2.	Módulo de Bloques.....	77
4.2.4.	Resultado Pruebas.....	82
5.	Observaciones, conclusiones y recomendaciones.....	85
5.1.	Observaciones	85
5.2.	Conclusiones	86
5.3.	Recomendaciones y Trabajos Futuros	87
	Bibliografía	89

Índice de Figuras

Figura 1.1 Petición de una Página Web	6
Figura 1.2 Esquema de la Web 2.0	8
Figura 1.3 EDT del Proyecto	18
Figura 1.4 Cronograma de actividades del diagrama de Gantt.....	20
Figura 1.5 Periodos de actividades del diagrama de Gantt.....	21
Figura 2.1 Diagrama de clases de análisis	37
Figura 3.1 Vista Lógica de la solución	40
Figura 3.2 Vista de Despliegue de la solución	41
Figura 3.3 Vista de Implementación de la solución.....	43
Figura 3.4 Diagrama de Clases en Cliente	45
Figura 3.5 Pantalla de Selección de Plantillas	47
Figura 3.6 Detalle de Selección de Plantilla	47
Figura 3.7 Pantalla Principal de Edición	48
Figura 3.8 Menú Principal Izquierda.....	49
Figura 3.9 Menú Principal Derecha	49
Figura 3.10 Menú de Tipos de Bloque.....	49
Figura 3.11 Menú Personalización	50
Figura 4.1 Modelo Vista Controlador	65



Índice de Tablas

Tabla 1.1 Ventajas y desventajas de escenarios de servicio	3
Tabla 1.2 Cuadro comparativo de características.....	14
Tabla 1.3 Actividades de Gestión del Proyecto	15
Tabla 1.4 Detalle de actividades para cada área de conocimiento	17
Tabla 2.1 Lista de requerimientos funcionales.....	29
Tabla 2.2 Lista de requerimientos no funcionales.....	29
Tabla 2.3 Matriz de trazabilidad de requerimientos con casos de uso	32
Tabla 2.4 Estimación del presupuesto para el proyecto	34
Tabla 4.1 Comparación de lenguajes de programación.....	55
Tabla 4.2 Comparación de tecnologías de presentación	59
Tabla 4.3 Comparación de gestores de bases de datos	61
Tabla 4.4 Comparación de entornos de desarrollo	64
Tabla 4.5 Comparación de patrones de diseño	66
Tabla 4.6 Comparación de frameworks de PHP.....	69
Tabla 4.7 Comparación de frameworks Javascript	72
Tabla 4.8 Comparación de ORMs de PHP	74
Tabla 4.9 Resultado pruebas de Catálogo de Pruebas	83
Tabla 4.10 Resultado pruebas de Checklist de Validación	83
Tabla 4.11 Resultado ponderaciones Características Usabilidad	84



1. Generalidades

En el presente capítulo se explica de forma general el problema a tratar. Se explican los conceptos básicos relacionados con el problema, de forma que se pueda comprender mejor los capítulos restantes del documento. De igual forma se analizan las soluciones existentes, comparando sus características, ventajas y desventajas, se define el plan del proyecto y se realiza la descripción de la solución planteada.

1.1. Definición del Problema

En la actualidad, para que una empresa pueda gozar los beneficios de tener presencia en Internet necesita de un sitio web acorde a sus necesidades, un nombre de dominio al cual enlazar el sitio web y una infraestructura que brinde soporte a este sitio web.

En el proceso de desarrollar o adquirir estos elementos como también en su mantenimiento se obtiene un conjunto de ventajas y desventajas de acuerdo a la solución elegida por la empresa.

Se presentan cinco escenarios:

1. La empresa cliente contrata un proveedor para que desarrolle y mantenga su sitio web y también contrata un proveedor para que le brinde el servicio de alojamiento de sitios web (Servicio de hosting).
2. La empresa cliente contrata un proveedor para que desarrolle y mantenga su web pero ella misma lo aloja.
3. La empresa cliente desarrolla y mantiene su propio sitio web y contrata a un proveedor para que le brinde el servicio de alojamiento de sitios web.
4. La empresa cliente desarrolla y mantiene su propio sitio web y ella misma lo aloja.
5. La empresa cliente contrata un proveedor para que desarrolle su sitio web, pero ella misma le brinda mantenimiento. En este caso puede contratar a un proveedor, como también ella misma puede alojar el sitio web.

A continuación en la tabla 1.1 se detallan las ventajas y desventajas para cada una de los componentes de los escenarios mencionados anteriormente.

	Ventajas	Desventajas
Se contrata proveedor para que desarrolle y mantenga el sitio web	<ul style="list-style-type: none"> - No es necesario contratar personal dedicado al desarrollo y mantenimiento. - Se garantiza una mejor calidad si se trata de un proveedor especializado. - Se redirige esfuerzo a los núcleos de negocio. 	<ul style="list-style-type: none"> - A cada cambio o conjunto de cambios que ocurran (dependiendo del proveedor) se debe realizar un pago, que por lo general no es barato. - Tiempo lento de respuesta para actualizaciones. - Si se realizan muchos cambios y actualizaciones, el proceso se vuelve más lento, tedioso y costoso.
Se desarrolla y mantiene el sitio web en la empresa	<ul style="list-style-type: none"> - Ya que hay un equipo a cargo se dan los cambios sin necesidad de realizar pago adicional. - Tiempo de respuesta rápido para actualizaciones. - Si el número de cambios y actualizaciones es grande el proceso continúa siendo rápido y barato. 	<ul style="list-style-type: none"> - Es necesario contratar personal dedicado al desarrollo y mantenimiento. - Es un gasto más a nivel empresarial. - Se ocupa esfuerzo en actividades no pertenecientes al núcleo del negocio.
Se contrata proveedor para que desarrolle sitio web, pero la empresa misma lo mantiene	<ul style="list-style-type: none"> - Si los cambios son pequeños se dan sin necesidad de realizar pago alguno ya que la misma empresa lo hace. - Tiempo de respuesta rápido para actualizaciones pequeñas. - Para cambios y actualizaciones pequeñas el proceso es sencillo. 	<ul style="list-style-type: none"> - Si los cambios son considerables es necesario que el proveedor los realice y por ende se da un gasto. - Tiempo de respuesta lento para actualizaciones considerables. - Para cambios y actualizaciones considerables el proceso es tedioso.

<p>Se contrata proveedor para alojar sitio web</p>	<ul style="list-style-type: none"> - No es necesario adquirir infraestructura para el sitio web. - No es necesario contratar personal que dé soporte a la infraestructura. - Se redirige esfuerzo a los núcleos de negocio. 	<ul style="list-style-type: none"> - No se tiene control sobre la información del sitio web a nivel físico. - Si se escala muy rápidamente puede llegar a ser muy costoso. - Existe una mayor posibilidad de peligro en la seguridad de la información de acuerdo al proveedor del servicio.
<p>Se aloja el sitio web en la infraestructura de la empresa</p>	<ul style="list-style-type: none"> - Se tiene control sobre la información del sitio web a nivel físico. - Si se escala muy rápidamente el costo es barato. - Existe una menor posibilidad de peligro en la seguridad de la información ya que depende de la empresa misma. 	<ul style="list-style-type: none"> - Es necesario adquirir infraestructura para el sitio web. - Es necesario contratar personal que dé soporte a la infraestructura. - Se ocupa esfuerzo en actividades no pertenecientes al núcleo del negocio.

Tabla 1.1 Ventajas y desventajas de escenarios de servicio

A través de la tabla 1.1 se puede inferir que cuando se contrata a un proveedor para la realización de una página web se realiza un solo pago, sin embargo si surgen cambios, se necesitan realizar una serie de pagos por cada uno de ellos, pagos que en general no son bajos. En contraste si la empresa realiza su propio sitio web necesita contratar personal que lo haga, lo cual facilita y acelera el proceso de cambio en la misma. Sin embargo se consumen recursos en actividades que no son parte del núcleo del negocio. En un caso intermedio la empresa contrata a un proveedor para que desarrolle el sitio web y ella misma le brinda mantenimiento. De esta manera si existen muchos cambios o actualizaciones pequeñas el gasto es mínimo, pero si se requieren actualizaciones o cambios considerables el gasto es grande.

(Potts 2008: 2 - 3)

Según el cuarto escenario, se encuentran un conjunto de servicios de auto desarrollo de sitios web como los ofrecidos por Negocios en Línea (<http://www.negocioenlinea.com/>) de la empresa Movistar, Google Sites (<https://sites.google.com>) y Wix (<https://premium.wix.com>), en donde la empresa construye por medio de una interfaz amigable su propio sitio web y tiene la facultad de cambiarlo cuando desee. Sin embargo, estos servicios a pesar de ofrecer muchas funcionalidades, no son muy fáciles de utilizar, como tampoco permiten dar pleno control del diseño al usuario. A su vez, también se da el caso que, utilizan tecnologías que no permiten el posicionamiento de los sitios en buscadores como es el caso de

Negocios en Línea y Wix, que utilizan Flash, lo cual va en contra del objetivo de un sitio web el cual es obtener la mayor cantidad de visitantes o potenciales clientes.

(Smith and Williams 2009: 323 - 328) (Spillane 2011: 179)

En especial las empresas unipersonales, SOHOs (small office/home office, empresas de 1 a 10 trabajadores), y pymes (pequeñas y micro empresas) que desean administrar su propia página web, se encuentran en este problema al no tener una herramienta que les facilite la creación de su propia página web sin destinar abundantes recursos económicos en ello. Sin embargo hay un conjunto de empresas para los cuales no es ideal este servicio, las cuales son las empresas de desarrollo web y de software.

Se propone brindar un servicio que permita a una empresa realizar su propio sitio web utilizando un conjunto de herramientas que le faciliten esta tarea, englobadas en un sistema. Adicionalmente se propone integrar el proceso de compra de un dominio por el cual acceder al sitio web. De esta forma se permite que la empresa pueda construir su propia web como también de actualizarla según requiera y brindar acceso a través de un nombre de dominio. Todo lo anterior se espera pueda realizarse a través de un servicio con un cómodo costo mensual o anual. Muy a diferencia de un servicio de alojamiento de sitios web o servicio de hosting, que no incluye la compra de un dominio ni la construcción de un sitio web; tan solo otorga el alojamiento del sitio web en Internet.

1.2. Marco Conceptual

El sistema a desarrollarse involucra un conjunto de conceptos relacionados con el desarrollo, implantación, acceso y posterior mantenimiento de sitios web.

1.2.1. Proceso de petición de una página web

Para tener presencia en Internet y lograr que las personas accedan a la información que se desea brindar, en primer lugar, se necesita de un sitio web, el cual es un conjunto de archivos estáticos como HTML, CSS y Javascript, y dinámicos como PHP, ASP y JSF, los que por medio de un navegador como Internet Explorer o Firefox,

permiten a las personas visualizar la información. (**Britannica Online Encyclopedia 2011: Website**)

En segundo lugar se necesita de una dirección web o nombre de dominio, el cual no es más que un conjunto de caracteres que identifican a un sitio web. Por ejemplo www.google.com, identifica al sitio web de Google. En tercer lugar se necesita de un servicio de hosting que permita alojar el sitio web para que pueda ser accedido por cualquier persona en el mundo.

A continuación se describe el proceso de petición de una página web representado en la figura 1.1. Una persona cuando ingresa en un navegador una dirección web o nombre de dominio, el navegador envía primero una petición de reconocimiento del nombre de dominio a un servidor DNS el cual lleva registro de los nombres de dominio y sus servidores de hosting asociados. El navegador recibe la respuesta del servidor DNS con el servidor de hosting donde se encuentra alojada la página web, entonces el navegador realiza la petición a ese servidor, el cual contesta con los archivos HTML, CSS y Javascript correspondientes. El navegador al recibir estos archivos genera el sitio web como tal y se lo presenta al visitante. (**Gilmore 2008: 5 - 10**)

1.2.2. HTML, CSS y Javascript

Son el conjunto de tecnologías que componen los sitios web en la actualidad. A continuación se describe cada una de ellas:

HTML (Hypertext Markup Language), es la tecnología de presentación sobre la que se basa el Internet desde sus inicios. Constituye la capa de contenido de una aplicación web ya que en ella se sitúa la información que se desea compartir a través de la web. Esta información se estructura mediante etiquetas, rodeadas por corchetes de apertura y de cierre de la etiqueta ('<' y '>').

(**Lawson and Sharp 2011: 1 - 3**)

Javascript es un lenguaje de programación interpretado que se encarga de la interacción en tiempo real del usuario con una aplicación web. Constituye la capa de comportamiento de una aplicación web ya que ella dirige las interacciones posibles del usuario en el navegador. Esta capa es la que permite que las aplicaciones web tengan una interacción y apariencia parecidas a las aplicaciones de escritorio.

(Olsson and O'Brien 2008: 1 - 2)

CSS (Cascading Style Sheets), es la tecnología de presentación estética de una aplicación web. Constituye la capa de presentación ya que controla los estilos, formas, diseños y colores en los cuales se presentará un sitio web al usuario.

(Olsson and O'Brien 2008: 1 - 3)

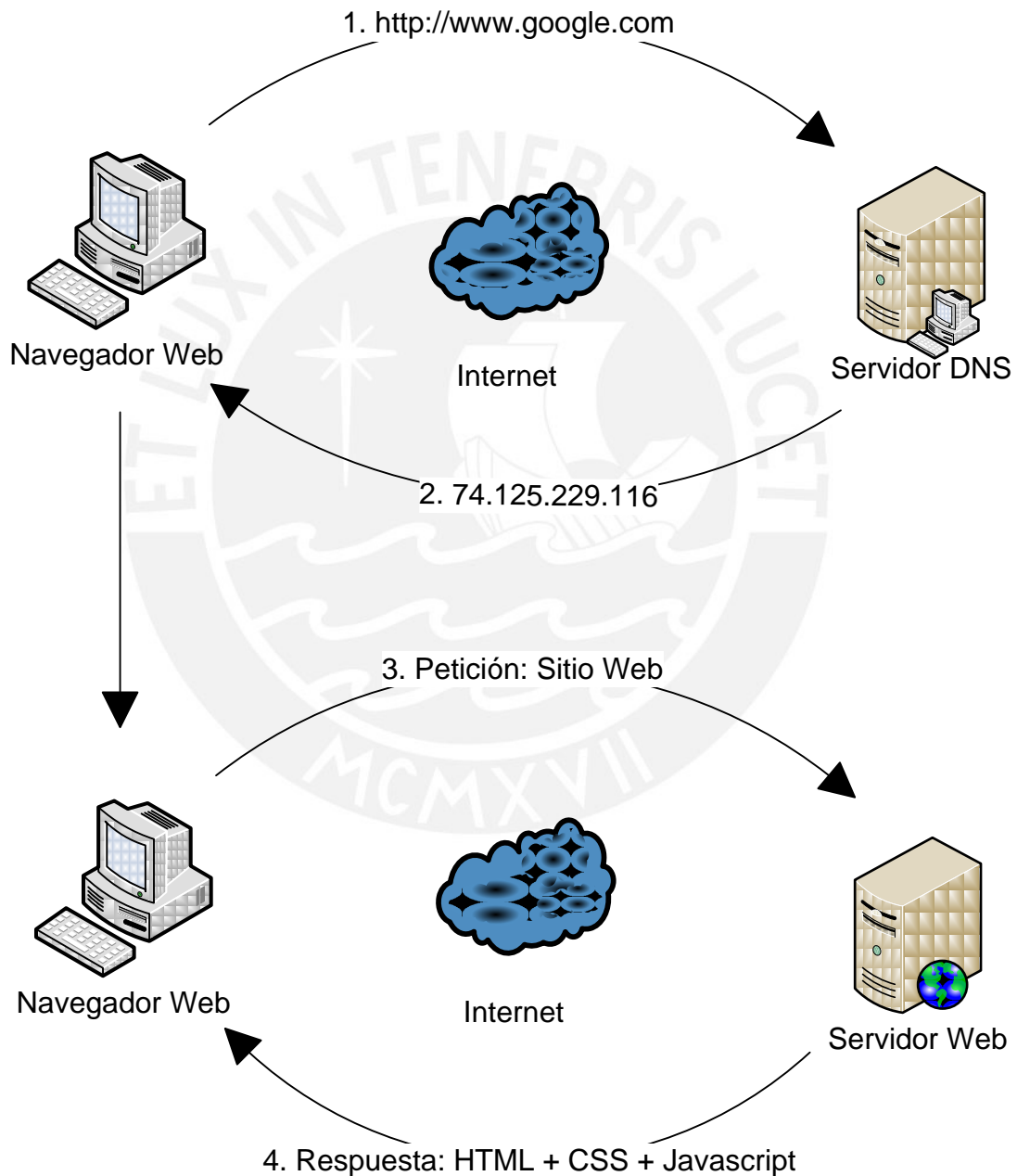


Figura 1.1 Petición de una Página Web

1.2.3. Nombres de Dominio

Los nombres de dominio tienen una estructura, por ejemplo `www.google.com`, la cual contiene tres partes principales, cada una dividida por un punto. La primera “www” indica el subdominio que por defecto siempre es “www”. También podría ser “docs” para el caso de `docs.google.com` por ejemplo. **(Network Working Group 2011: Domain Name System).**

La segunda parte es el nombre real del dominio. Para el ejemplo sería “google”. Normalmente este nombre es parecido al nombre de la entidad a la cual representa. Por último se tiene la extensión. Para el ejemplo sería “.com”, la cual indica que es una página del tipo comercial. También existen otras extensiones como “.net” y “.org”. La primera identifica sitios web relacionados al sector computacional y de redes y el segundo a organizaciones. A pesar de su significado no se restringe su utilización de acuerdo a su extensión.

Hay un caso especial para las extensiones dadas por país como las “.pe” para Perú, “.us” para Estados Unidos o “.ru” para Rusia. Estas extensiones permiten identificar los sitios web de un determinado país. Estas extensiones son controladas por una organización, que por lo general son universidades, por país. Para el caso del Perú la organización que gestiona los dominios de extensión “.pe” es la Red Científica Peruana. **(Lacnic 2011: Lacnic)**

1.2.4. Web 2.0

Las páginas web, en un inicio, eran utilizadas solamente para mostrar información. Tales sitios web se denominan estáticos ya que la información que se le presenta al visitante siempre es la misma. Un sitio web estático está compuesto por archivos HTML, CSS y Javascript que se encuentran alojados en un servidor web, los cuales son enviados a un navegador web para que éste los pueda presentar al visitante de una forma adecuada. A este esquema de trabajo se le denominó Web 1.0 y se representa en parte de la figura 1.2. **(How Stuff Works 2011: Web 1.0)**

Poco a poco fue surgiendo la necesidad de interactuar con este contenido con la finalidad de que el usuario pueda ofrecer información al sitio web, como también de

que la información pueda ser configurada dependiendo del visitante. Fue así que surgieron las páginas web dinámicas, las cuales permiten presentar información cambiante y poder ser actualizada por un visitante. Un sitio web dinámico también está compuesto por un conjunto de archivos HTML, CSS y Javascript; la diferencia radica en el procesamiento de una petición, en la cual el servidor de aplicaciones genera la información, a partir de los HTML, a ser servida por el servidor web. Normalmente el servidor de aplicaciones utiliza una base de datos como fuente de información para la generación.

Las páginas web dinámicas sentaron las bases de la web 2.0. La cual consiste en que el usuario, que antes solamente podía leer o visualizar el contenido, ahora pueda generar información para los sitios web, por ejemplo los comentarios de los lectores en un portal de noticias (O'Reilly 2008: 1 - 2). Como se puede ver en la figura 1.2, parte del concepto Web 2.0 también lo son las aplicaciones que explotan esta posibilidad de retroalimentación tales como blogs como Blogger, wikis como Wikimedia, foros como PhpBB, redes sociales como Facebook y Twitter, RSS, etc.



Figura 1.2 Esquema de la Web 2.0

(Web Social - Master en bibliotecas y servicios de información digital 2011:
Diferencias Principales entre Web 1.0 y Web 2.0)

1.2.5. WYSIWYG

Acrónimo referido a “**What You See Is What You Get**”, o en la traducción al español, “Lo que tú ves es lo que tú obtienes”. Es un término utilizado en el mundo web para describir un sistema en el que el contenido que se visualiza durante la edición es muy similar, sino igual, al contenido obtenido como resultado **(Choate 2007: 74)**.

El objetivo principal de un editor WYSIWYG es el de brindar las facilidades al usuario para que pueda modificar un texto de acuerdo a como lo haría en un editor de texto tipo Microsoft Word, con la diferencia de que se realiza directamente en un navegador web. De esta forma un usuario edita un texto aplicando formato como Negrita, Cursiva, Subrayado, Alineado, Color de Texto, Color de Fondo, etc; y puede visualizar este mismo resultado como contenido web. Esta funcionalidad brinda al usuario web actual una gran ayuda a la hora de generar contenido en el Internet **(Tomlinson 2010: 93)**.

El editor WYSIWYG, a partir del contenido generado por el usuario, lo transforma en código HTML, haciendo posible de esta forma visualizarlo como tal en la web.

Actualmente existen un conjunto de editores WYSIWYG para la web, sin embargo se indican los dos principales:

- **TinyMCE**

Es un editor WYSIWYG basado en la web que utiliza Javascript para su funcionamiento. Lanzado como código abierto bajo la licencia LGPL por Moxiecode Systems AB.

- **FCKEditor**

Es un editor WYSIWYG que se puede utilizar en páginas web. Se centra en la facilidad de integración que tiene con cualquier tipo de aplicación.

(Tomlinson 2010: 93), (Rahmel 2007: 84 - 87)

1.2.6. Wiki

Es un sistema web que permite la creación y edición de cualquier número de páginas web por medio de un grupo o conjunto de usuarios **(Choate 2007: 1)**.

Los usuarios generan páginas web utilizando un tipo especial de lenguaje de marcas conocido como WikiText, el cual presenta una estructura particular que permite generar contenido web ordenado y vistoso. Algunas wikis brindan la opción de utilizar, para la edición de texto, un editor WYSIWYG, el cual presenta en la edición la información tal cual se visualizará posteriormente **(Choate 2007: 1)**.

Las wikis ostentan un conjunto de niveles de acceso que permiten a los usuarios, de acuerdo a su rol, poder visualizar o editar solamente las páginas web para las cuales tienen permisos de lectura o edición.

Actualmente las wikis son muy utilizadas en las comunidades en línea como herramienta de documentación. Un ejemplo claro es la enciclopedia en línea llamada Wikipedia, la cual está basada en el motor de wiki Wikimedia. Esta enciclopedia permite a cualquier usuario en el mundo crear una página acerca de un concepto específico. Así también existen una serie de administradores por todo el mundo que constantemente monitorean los contenidos creados, como también los modificados, con el objetivo de salvaguardar la información correcta de la misma.

(Wikipedia 2011: Wikipedia Administration)

1.2.7. RSS

Acrónimo referido a **Really Simple Syndication**, o en la traducción al español, sindicación muy simple. Este término comprende un conjunto de feeds web, que tienen por objetivo publicar nuevos artículos, noticias productos, etc. Un feed es un flujo de información de una página web, el cual se extrae en ciertos formatos determinados.

(Boroczky y Psinas 2008: 278)

Los visitantes web se suscriben a feeds del sitio web de su preferencia y así son automáticamente notificados acerca de las actualizaciones de los mismos sin tener la necesidad de visitar el sitio web en el navegador.

1.2.8. Facebook

Es una red social, que permite a una persona tener un perfil en la web con sus características personales, media, imágenes, y lo más importante, permite tener como “amigos” a perfiles de otras personas, lo que conlleva a tener acceso a su información, contenidos y actualizaciones. **(Shih 2009: 3)**

Fue lanzada en febrero del 2004. Su tasa de crecimiento ha sido sumamente alta, contando actualmente con 600 millones de usuarios alrededor del mundo.

En el aspecto empresarial, se constituye como un medio publicitario, es así como muchas empresas tienen sus perfiles en la red social, lo cual les permite tener un contacto más cercano con sus clientes.

Entre las herramientas que brinda para su expansión en la web están los llamados social plugins, como el botón Me Gusta, la caja Me Gusta y el Feed de Actividad, los cuales se colocan en sitios webs externos y permiten una interactividad directa con algún perfil de Facebook.

1.2.9. Twitter

Es una red social y servicio de microblogging, que permite a sus usuarios leer y enviar mensajes llamados tweets. Un tweet es un artículo basado en texto de 140 caracteres que se visualiza en la página del perfil del usuario. Un usuario puede seguir y ser seguido por otros usuarios. Si se sigue a un usuario entonces se podrá visualizar sus tweets en la página de perfil; si se es seguido por otros usuarios, entonces ellos podrán visualizar nuestros tweets en sus páginas de perfil.

Twitter fue lanzado en julio del 2006. Ha tenido una tasa de crecimiento muy alta, contando actualmente con 200 millones de usuarios, que generan 190 millones de tweets al día.

Dispone de un conjunto de servicios web que permiten obtener información de los usuarios como también de los tweets con el fin de utilizarla de acuerdo a la necesidad como presentarla en sitios web, Estas funcionalidades en conjunto posicionan a Twitter como parte de la web 2.0. **(Makice 2009: 1 - 5)**

1.2.10. Sistema de Venta de Dominios

Es un sistema que se encarga de la selección, evaluación y venta de uno o muchos nombres de dominio. Se tienen muchos ejemplos de estos sistemas web, para el caso de dominios genéricos se tiene Godaddy (<http://www.godaddy.com/>) y para el caso de dominios por países se tiene el Punto Pe (<http://punto.pe/>).

1.3. Estado del Arte

Actualmente existen algunos servicios que brindan auto construcción y mantenimiento de sitios web, sin embargo limitan al usuario en varios aspectos en cuanto a estructura y libertad de diseño.

1.3.1. Servicios Existentes

1.3.1.1. Google Sites (<https://sites.google.com/>)

Servicio de construcción y mantenimiento de sitios web desarrollado por Google. Muestra un conjunto abundante de opciones para el desarrollo de páginas web, sin embargo no es muy fácil de utilizar. No se puede brindar un diseño web adecuado a un sitio web 2.0.

1.3.1.2. Negocio en Línea (<http://www.negocioenlinea.com/>)

Servicio ofrecido por la empresa Movistar con el objetivo de crear y mantener un sitio web. A pesar de que presenta un conjunto de plantillas muy diverso y una interfaz muy amigable, no se pueden cambiar la estructura de las plantillas ni los colores. Tiene componentes muy buenos para la edición de imágenes.

1.3.1.3. Yahoo Small Business (<http://smallbusiness.yahoo.com/webhosting>)

Servicio ofrecido por Yahoo, con una amplia variedad de plantillas y controles que resultan confusos al tratar de realizar la edición. Es difícil saber por qué parte empezar.

1.3.1.4. About Me (<https://about.me/>)

Servicio enfocado en perfiles de personas. Muestran una serie de plantillas que permiten configurar fácilmente la información de una persona. La restricción se da al momento de querer agregar más contenido, el cual se encuentra limitado a los bloques de contenido que ya existen por plantilla.

1.3.1.5. CMS Drupal

Drupal es un sistema gestor de contenidos como también un framework gestor de contenidos. De fácil instalación, permite obtener un sitio web operando rápidamente. A pesar de su facilidad de instalación requiere una base de conocimiento técnico para ello. Una vez instalado brinda una amplia gama de funcionalidades propias de un CMS, como administración de artículos, fotos, videos, RSS, enlaces, diseño y más. **(Tomlinson 2010: 1)**

Cuenta con un conjunto de plugings que permiten extender su funcionalidad y adecuarlo a las necesidades del usuario. También tiene un gran número disponible de plantillas que permiten personalizar el diseño del sitio web de una forma rápida y sencilla.

Sin embargo para poder instalar funcionalidades extras se necesita de conocimiento técnico. Un usuario sin estos conocimientos sería incapaz de realizar la instalación de tales funcionalidades.

También se necesita de una infraestructura web donde instalar dicho CMS.

1.3.1.6. CMS Joomla

Joomla es un sistema gestor de contenidos. Comparte muchas de sus características con Drupal como administración de artículos, fotos, videos, RSS, enlaces y diseño. El aspecto diferenciador de Joomla radica en sus plantillas utilizadas para personalizar un sitio web, las cuales son de mayor calidad que las de Drupal y de los demás gestores de contenido.

Comparte también las mismas desventajas de Drupal en cuanto a su esquema de instalación y nivel técnico necesario para ello. **(Rahmel 2007: 1 - 8)**

1.3.2. Cuadro comparativo

En la tabla 1.2 se muestra un cuadro comparativo de las características que presentan las diferentes soluciones mencionadas anteriormente.

Se marca con X aquellas opciones que cumplen con la característica.

Característica	Google Sites	Negocio en Línea	Yahoo Small Business	About Me	Drupal	Joomla
Es Servicio	X	X	X	X		
Es CMS					X	X
Requiere conocimiento técnico					X	X
Es muy fácil de usar				X		
Brinda las herramientas necesarias para personalizar el sitio web.	X		X			
Permite personalizar el contenido del sitio web arrastrándolo (Drag and Drop) y redimensionando (Resize).				X		
Permite personalizar el diseño del sitio web.			X	X		
Tiene un amplio número de plantillas para personalizar el diseño		X	X		X	X
Tiene un manejo avanzado de imágenes.		X				
Tiene Panel de Control	X	X	X			
Incluye Dominio			X			

Tabla 1.2 Cuadro comparativo de características

A través del cuadro se puede inferir, en primer lugar, que los CMSs no cuentan con las características adecuadas para facilitar la generación de un sitio web ya que se necesita de conocimiento técnico para poder realizar su instalación y configuración. En segundo lugar, que los servicios actuales no le brindan al usuario libertad de diseño total por medio de características como arrastrar contenidos y redimensionarlos, así tampoco cuentan con un amplio número de plantillas ni facilidad de uso. Y finalmente,

los servicios actuales no permiten integrar los pasos de generación de un sitio web con los de la compra de un nombre de dominio, por el cual presentar el sitio web por medio del Internet.

1.4. Plan de Proyecto

En la presente sección se determinan las actividades a realizar de acuerdo a las buenas prácticas de gestión de proyectos propuestas por el PMBOK. En la tabla 1.3 se visualizan las actividades que se ponen en práctica en el proyecto. La explicación detallada está dada en la tabla 1.4.

Áreas de Conocimiento	Procesos				
	Iniciación	Planificación	Ejecución	Monitoreo y Control	Cierre
Gestión de Alcance	Planificar el Alcance	Definición del alcance			
		Crear un WBS			
Gestión de Tiempo	Definición de las actividades		Desarrollo del cronograma	Control del cronograma	
	Establecimiento de la secuencia de actividades	Estimación de la duración de las actividades			
Gestión de la Calidad del Proyecto		Plan de Gestión de la calidad	Realizar control de calidad 1	Realizar control de calidad 2	
Gestión de Comunicaciones		Plan de gestión de la comunicación			
Gestión de Riesgos	Identificación de riesgos	Listado de riesgos	Monitoreo y control de riesgos 1	Monitoreo y control de riesgos 2	
	Análisis cualitativo de riesgos				

Tabla 1.3 Actividades de Gestión del Proyecto

Área de Conocimiento	Entradas	Herramientas y/o Técnicas	Salidas
Gestión de Alcance	<ul style="list-style-type: none"> • Plan de proyecto. • Plantillas del curso de Tesis. 	<ul style="list-style-type: none"> • Se usará como herramienta la Estructura de Descomposición del Trabajo (EDT o WBS). 	<ul style="list-style-type: none"> • Documento de Plan de Gestión del Alcance, incluirá el EDT completo.
Gestión de tiempo	<ul style="list-style-type: none"> • Plan de Proyecto. • Plan de Gestión de Alcance. • Plantillas del curso de Tesis. 	<ul style="list-style-type: none"> • Descomposición de los paquetes de trabajo definidos en el EDT, en actividades dependientes y tareas. • Utilización de un programa de gestión de proyecto (MS Project). 	<ul style="list-style-type: none"> • Plan de Gestión de tiempo, incluirá el cronograma y el diagrama de Gantt completo a nivel de las actividades del proyecto.
Gestión de calidad	<ul style="list-style-type: none"> • Plan de Proyecto. • Plan de Gestión del Alcance. • Plan de Gestión de tiempo. 	<ul style="list-style-type: none"> • Definir atributos de calidad que serán importantes para el proyecto. • Definir un checklist de elementos que deberían cumplir el software para ser considerado de calidad. • Verificación del checklist al terminar cada etapa del proyecto. 	<ul style="list-style-type: none"> • Plan de Gestión de la calidad.
Gestión de Comunicaciones	<ul style="list-style-type: none"> • Plantillas del curso de Tesis. • Plan de Proyecto. 	<ul style="list-style-type: none"> • Análisis de requisitos de comunicación que se necesita a lo largo del proyecto. • Definición de estándares de comunicación. • Uso de tecnología adecuada para mantener ordenada la comunicación. 	<ul style="list-style-type: none"> • Plan de Gestión de la comunicación.

<p>Gestión de riesgos</p>	<ul style="list-style-type: none"> • Plan de Proyecto • Plan de gestión del alcance. • Plan de gestión de la calidad • Plan de gestión de tiempo • Plan de gestión de comunicaciones • Experiencias pasadas 	<ul style="list-style-type: none"> • Se identificarán los principales riesgos encontrados después de analizar toda la información asociada al proyecto. • Se definirán acciones a tomar para reducir la probabilidad de ocurrencia de los riesgos, así como para mitigar el impacto en el alcance del proyecto. Se definirán acciones preventivas y correctivas. 	<ul style="list-style-type: none"> • Listado de Riesgos
----------------------------------	---	--	--

Tabla 1.4 Detalle de actividades para cada área de conocimiento

1.4.1. Estructura de Descomposición del Trabajo (EDT)

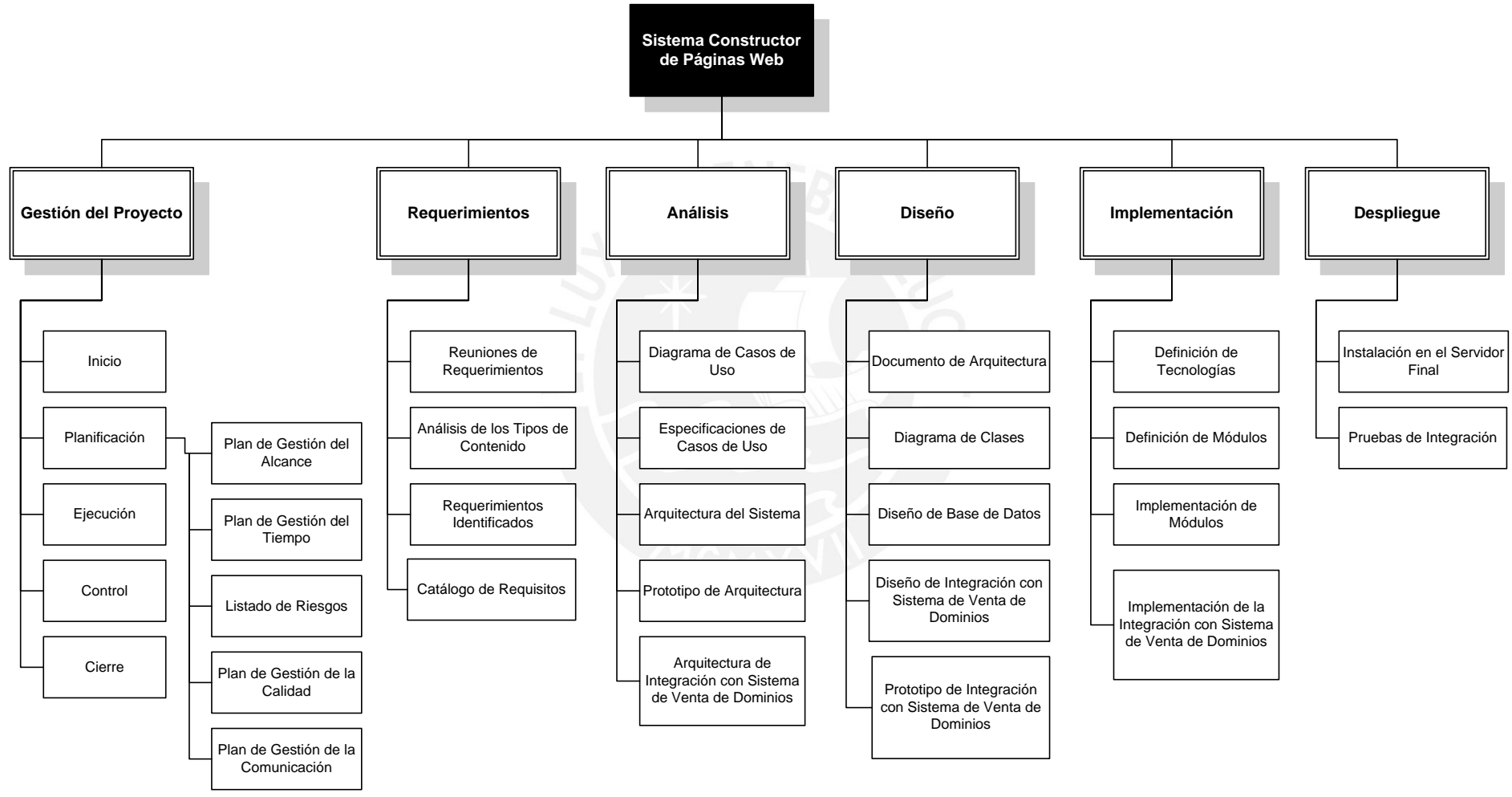


Figura 1.3 EDT del Proyecto

El EDT muestra claramente las etapas en las que se divide el proyecto. Para cada etapa muestra los entregables o paquetes de trabajo que deben ser completados para poder obtener satisfactoriamente el producto deseado.

Para el presente proyecto se definieron seis procesos principales, cinco de ellos son de construcción del producto y uno es el de gestión del proyecto. Los procesos destinados a la construcción del producto son requerimientos, análisis, diseño, implementación y despliegue. Estas etapas se definen puesto que son procesos clave en la utilización de la metodología Rational Unified Process (RUP), que se usará como base para la construcción del producto.

1.4.2. Diagrama de Gantt

Se tiene a continuación, en las figuras 1.4 y 1.5, el diagrama de Gantt desarrollado para el proyecto, el cual muestra las estimaciones de los tiempos para cada una de las actividades requeridas para la compleción del proyecto.

La estimación de los tiempos asignados a cada una de las actividades se define en función a la experiencia previa en proyectos de similar envergadura.

De acuerdo a la metodología de desarrollo, RUP, se define aproximadamente un 10% del tiempo para la fase de concepción del proyecto, 30% para la fase de elaboración del proyecto, 50% para la fase de construcción del proyecto y finalmente 10% para la fase de transición del proyecto.

(Kruchten 2003: 65 - 68)

Id	Task Name	Duración	Comienzo	Fin	Predecesoras
1	Sistema Constructor Páginas Web	150 días?	mié 6/1/11	mar 12/27/11	
2	Gestión del Proyecto	7 días?	mié 6/1/11	jue 6/9/11	
3	Gestión del Alcance	1 día	mié 6/1/11	mié 6/1/11	
4	Gestión del Tiempo	1 día	jue 6/2/11	jue 6/2/11	3
5	Gestión de Riesgos	2 días	vie 6/3/11	lun 6/6/11	4
6	Gestión de Calidad	2 días	mar 6/7/11	mié 6/8/11	5
7	Gestión de Comunicación	1 día?	jue 6/9/11	jue 6/9/11	6
8	Requerimientos	7 días	vie 6/10/11	lun 6/20/11	7
9	Realizar Reunión de Requerimientos	2 días	vie 6/10/11	lun 6/13/11	
10	Realizar Análisis de los Tipos de Contenido	2 días	mar 6/14/11	mié 6/15/11	9
11	Identificar Requerimientos	1 día	jue 6/16/11	jue 6/16/11	10
12	Realizar Catálogo de Requisitos	2 días	vie 6/17/11	lun 6/20/11	11
13	Análisis	22 días	mar 6/21/11	mié 7/20/11	12
14	Realizar Diagrama de Casos de Uso	2 días	mar 6/21/11	mié 6/22/11	
15	Realizar Especificaciones de Casos de Uso	4 días	jue 6/23/11	mar 6/28/11	14
16	Realizar Arquitectura del Sistema	2 días	mié 6/29/11	jue 6/30/11	15
17	Realizar Prototipo de Arquitectura	12 días	vie 7/1/11	lun 7/18/11	16
18	Realizar Arquitectura de Integración con Sistema de Venta de Dominios	2 días	mar 7/19/11	mié 7/20/11	17
19	Diseño	17 días	jue 7/21/11	vie 8/12/11	18
20	Realizar Documento de Arquitectura	2 días	jue 7/21/11	vie 7/22/11	
21	Realizar Diagrama de Clases	3 días	lun 7/25/11	mié 7/27/11	20
22	Realizar Diseño de Base de Datos	3 días	jue 7/28/11	lun 8/1/11	21
23	Realizar Diseño de Integración con Sistema de Venta de Dominios	3 días	mar 8/2/11	jue 8/4/11	22
24	Realizar Prototipo de Integración con Sistema de Venta de Dominios	6 días	vie 8/5/11	vie 8/12/11	23
25	Implementación	86 días	lun 8/15/11	lun 12/12/11	24
26	Definir Tecnologías	3 días	lun 8/15/11	mié 8/17/11	
27	Definir Módulos	3 días	jue 8/18/11	lun 8/22/11	26
28	Implementar Módulos	60 días	mar 8/23/11	lun 11/14/11	27
29	Implementar Integración con Sistema de Venta de Dominios	20 días	mar 11/15/11	lun 12/12/11	28
30	Despliegue	11 días	mar 12/13/11	mar 12/27/11	29
31	Instalación del Sistema	6 días	mar 12/13/11	mar 12/20/11	
32	Prueba de Integración	5 días	mié 12/21/11	mar 12/27/11	31

Figura 1.4 Cronograma de actividades del diagrama de Gantt

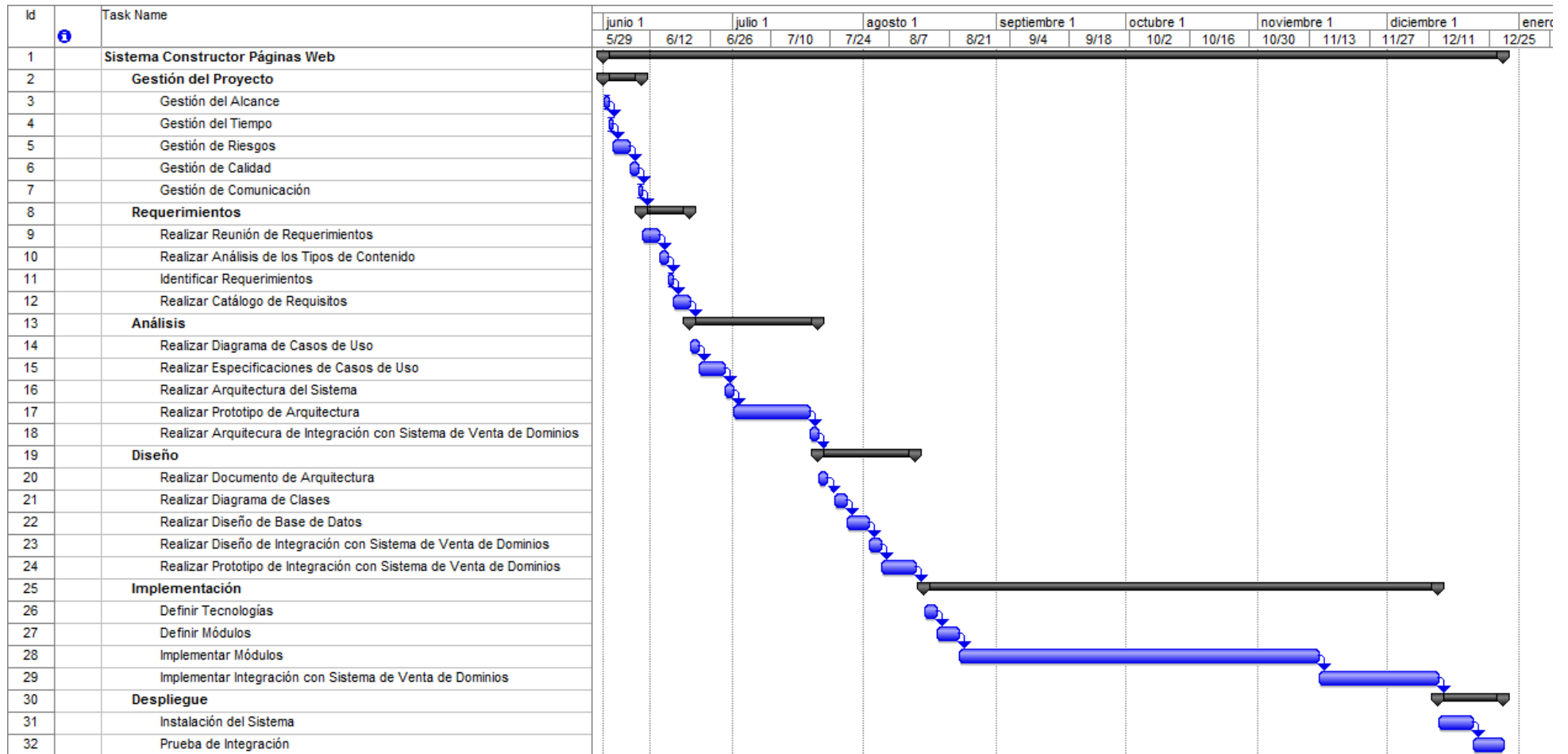


Figura 1.5 Periodos de actividades del diagrama de Gantt

1.5. Descripción y sustentación de la solución

El sistema propuesto tiene las siguientes características:

1. Selección de una plantilla para la creación de un sitio web.
2. Para la creación o edición de un sitio web:
 - a. Configurar los parámetros globales del sitio web:
 - i. Título (Html Meta tag Title)
 - ii. Descripción (Html Meta tag Description)
 - b. Crear, modificar, y eliminar páginas. Para cada página de un sitio web se permite:
 - i. Modificar el nombre.
 - ii. Cambiar el tamaño de las secciones, las cuales pueden ser Cabecera, Barra Izquierda, Cuerpo, Barra Derecha y Pie de Página.
 - iii. Crear, modificar y eliminar bloques de contenido. Los bloques pueden ser de Texto (WYSIWYG), Imagen, Galería, Video (Youtube, Vimeo y Upload), Audio, Facebook (Like Button, Activiy Feed, Like Box), Twitter (Timeline de un usuario) y RSS.
 - iv. Cada bloque tiene dos funcionalidades principales, de movimiento (Drag and Drop) y de redimensión (Javascript resize).
 - c. Cambiar de plantilla
 - d. Cambiar de formato (Agregar o quitar secciones y redimensionar)
 - e. Configurar el color de fondo del sitio web y colores de secciones
 - f. Configurar la imagen de fondo del sitio web
 - g. Configurar la familia y el tamaño de la fuente del sitio web
 - h. Configurar los colores, orientación y tamaño de la fuente del menú general del sitio web.
 - i. Guardar: Guarda todos los elementos actualizados del sitio web
 - j. Pre-visualizar: Muestra una vista previa de la página actual del sitio web
 - k. Publicar: Despliega el sitio web en su locación final

Cada sitio web cuenta con un paquete, el cual determina las funcionalidades que se pueden utilizar como también limita el número de bloques y páginas que se pueden generar para cada sitio web. Por ejemplo se tiene: Estándar, Avanzado y Premium.

Para la integración con el sistema de venta de dominios se tienen un conjunto de funcionalidades, que se inician con la creación del sitio web, luego se puede efectuar la suspensión del sitio web, como también su probable reactivación posterior, y finalmente la eliminación permanente del sitio web.

El objetivo de la solución propuesta es el de brindar una alternativa viable de servicio de desarrollo de sitios web para todas aquellas empresas unipersonales, sohos y pymes que por motivo de facilidad, tiempo o dinero; no cuentan con un sitio web, limitando de esta forma sus oportunidades de éxito a corto y mediano plazo, ya que el Internet se constituye como el medio principal de comunicación en el presente y por ende representa una ventaja competitiva tener presencia en la web.

Las ventajas principales que tiene la solución presentada son su facilidad y rapidez en la generación de un sitio web empresarial junto con su muy bajo costo, acorde al presupuesto de su sector objetivo, el cual está compuesto por personas dinámicas que necesitan un servicio de rápida y fácil adaptación a sus necesidades.

El sistema de construcción y edición de páginas web se integrará con el sistema de venta de dominios Yachay (<http://yachay.pe/>), con el objetivo de enlazar automáticamente el sitio web generado a un nombre de dominio. El sistema de venta de dominios es el encargado de la búsqueda, selección, venta y proceso de pago del dominio requerido para el despliegue del sitio web. Una vez culminado el Sistema de Construcción de Webs 2.0, podrá ser utilizado por medio de Yachay (<http://yachay.pe/index.php?tema=paginas-web>). Para realizar una prueba de la aplicación se debe buscar el enlace “Pruébanos por 30 días”.

2. Análisis

En el presente capítulo se realiza la descripción de la metodología de construcción a usar y la forma en la que se aplica a la implementación de la solución presentada. Se presentan detalladamente los requerimientos del sistema, así como también el análisis de la solución propuesta.

2.1. Metodología aplicada para el desarrollo de la solución

Para el desarrollo del producto se utiliza la metodología RUP (Rational Unified Process), porque sus características como orientado a casos de uso, centralización en la arquitectura y ser iterativo e incremental benefician de gran forma al presente proyecto. A continuación se describen las razones de su utilización.

- Al ser orientado a casos de uso permite centrar los esfuerzos en las funcionalidades principales como creación de páginas y tipos de bloque del sistema.
- Al ser centrado en la arquitectura permite evaluar correctamente las partes más importantes de la aplicación utilizando los diferentes tipos de vistas disponibles.

- Al ser iterativo e incremental asegura que la aplicación se desarrolle en un conjunto de fases que conllevan a partes funcionales del producto final, que permiten validar correctamente cada una de ellas, a diferencia del esquema en cascada.
- Es una metodología altamente conocida, y su uso es bastante extendido en muchos proyectos exitosos. Además, se cuenta con mucha documentación acerca de la misma, lo que facilita la solución de dudas y errores.
- El desarrollador cuenta con amplia experiencia en el uso de esta metodología, lo cual permite ahorro de tiempo de aprendizaje y garantiza un mayor porcentaje de éxito ya que se obtienen mejores resultados que utilizando una metodología no conocida.

(Kruchten 2003: 1 - 3)

A continuación se explica la adaptación de RUP al presente proyecto. Se especifica las fases definidas y los entregables a generar por cada una.

- **Requerimientos**
 - **Documento de Visión**

Se considera necesario para el desarrollo del proyecto ya que brinda una idea en alto nivel acerca de los requisitos del sistema, como también es necesario para entender las restricciones de diseño. Este documento servirá para que una persona pueda familiarizarse rápidamente con el sistema y su objetivo.
 - **Catálogo de Requisitos**

Compendio de los requisitos funcionales y no funcionales que se obtuvieron como resultado del estudio de las principales funcionalidades que requiere un sitio web empresarial.
- **Análisis**
 - **Especificación de Requisitos de Sistema**

Se considera necesario para poder definir claramente las funcionalidades que tendrá el sistema a partir del catálogo de requisitos.

- **Documento de Análisis**
Necesario para tener la estructura inicial de las clases y bases de datos del sistema, con los cuales poder pasar a la etapa de diseño.
- **Diseño**
 - **Documento de Arquitectura**
Se considera de suma importancia para entender el esquema sobre el cual será basado el sistema, como también el de su integración con el sistema de venta de dominios. Con la ayuda de las cinco vistas arquitecturalmente significativas se podrá aclarar la base sobre la cual se sienta el sistema y su integración posterior.
 - **Diagrama de Base de Datos o Modelado de Datos**
Esencial para poder diseñar la estructura de la base de datos final del sistema.
 - **Prototipo de Arquitectura**
Se utiliza con el fin de establecer la estructura de funcionamiento base del constructor de sitios web.
- **Implementación**
 - **Software desarrollado**
El sistema completamente implementado.
 - **Plan de pruebas**
Este documento permite planificar todas las pruebas a realizar en el sistema, tanto para su desarrollo, como también para su integración con el sistema de venta de dominios.
- **Despliegue**
 - **Software instalado**
El sistema correctamente desplegado en el servidor final.
 - **Realización de pruebas de integración**
Permite comprobar si el sistema constructor de sitios web está correctamente integrado con el sistema de venta de dominios.

2.2. Identificación de Requerimientos

La presente sección tiene por objetivo el análisis de requerimientos para el sistema propuesto.

Se procederá a listar todos los requerimientos encontrados en el sistema y que forman parte de la solución. Luego se identificarán los casos de uso asociados a los requerimientos encontrados.

2.2.1. Lista de requerimientos

El objetivo del presente proyecto es el de desarrollar una solución que permita satisfacer el problema de obtener sitios web altamente personalizables de manera rápida y sencilla. Para ello es necesario definir un conjunto de requerimientos que permitan detallar las funcionalidades del producto final, así como implementar un sistema que se ajuste a las necesidades actuales del mercado de construcción de sitios web.

Los requerimientos se obtuvieron por medio del análisis de las funcionalidades encontradas en las soluciones actuales que fueron profundizadas en la sección del Estado del Arte del capítulo 1. Así también, se obtuvo un importante aporte de las áreas de Ingeniería de Software y Planeamiento de la Red Científica Peruana, que proporcionaron los requerimientos base a partir de un conjunto de reuniones llevadas a cabo con anterioridad.

En la tabla 2.1 se visualizan los requerimientos funcionales con los que cuenta el sistema y la tabla 2.2 muestra todos los requerimientos no funcionales del mismo.

N°	Requerimiento	Prioridad
Módulo de Configuración Sitio Web		
1	Permite crear un nuevo sitio web por medio de una url, un usuario y un paquete.	1
2	Permite seleccionar una plantilla base para un nuevo sitio web.	1
3	Permite administrar la información general de un sitio web: Título y Descripción.	1
4	Permite cambiar de plantilla a un sitio web existente.	3
5	Permite cambiar de formato (disposición de secciones) a un sitio web existente.	2
6	Permite modificar el color de fondo de un sitio web.	2
7	Permite modificar la imagen de fondo de un sitio web.	2
8	Permite modificar la familia de fuente de un sitio web.	2
9	Permite modificar el tamaño de fuente de un sitio web.	2
10	Permite modificar los colores, orientación, y tamaño de fuente del menú del sitio web.	2
11	Permite guardar todos los cambios realizados al sitio web.	1
12	Permite pre-visualizar la página seleccionada de un sitio web.	1
13	Permite publicar el sitio web a su locación final.	1
Módulo de Páginas		
14	Permite agregar y eliminar páginas a un sitio web.	1
15	Permite modificar el nombre de una página de un sitio web.	1
16	Permite modificar el tamaño vertical de las secciones de una página.	2
17	Permite modificar el color de las secciones de una página.	2
18	Permite limitar el número de páginas de acuerdo al paquete del sitio web.	2
Módulo de Bloques de Contenido		
19	Permite agregar y eliminar bloques de contenido a una página del sitio web.	1
20	Permite crear bloques de tipo Texto por medio de un editor WYSIWYG.	1
21	Permite crear bloques de tipo Imagen por medio de la carga de un archivo de imagen.	1
22	Permite crear bloques de tipo Galería por medio de la carga de un conjunto de archivos de imagen.	1
23	Permite crear bloques de tipo Video Youtube por medio de una url de un video.	2
24	Permite crear bloques de tipo Video Vimeo por medio de una url de un video.	2
25	Permite crear bloques de tipo Video Carga por medio de la carga de un archivo de video.	2

26	Permite crear bloques de tipo Audio por medio de la carga de un archivo de audio.	2
27	Permite crear bloques de tipo Facebook Botón Me Gusta por medio de una url.	2
28	Permite crear bloques de tipo Facebook Feed de Actividad por medio de una url que sea una página de Facebook.	2
29	Permite crear bloques de tipo Facebook Caja Me Gusta por medio de una url.	2
30	Permite crear bloques de tipo Twitter por medio de un usuario twitter que obtenga los cinco últimos tweets del mismo.	2
31	Permite crear bloques de tipo RSS por medio de una url.	2
32	Permite redimensionar los bloques existentes en cualquier dirección.	1
33	Permite mover cualquier bloque existente a cualquier posición interior de una página.	1
34	Permite limitar el número de bloques de acuerdo al paquete del sitio web.	2
Módulo de Mantenimiento de Sitios Web		
Integración con el Sistema de Venta de Dominios		
35	Permite a un sistema externo crear un sitio web.	1
36	Permite a un sistema externo suspender un sitio web.	1
37	Permite a un sistema externo activar un sitio web.	1
38	Permite a un sistema externo eliminar un sitio web.	1

Tabla 2.1 Lista de requerimientos funcionales

N°	Requerimiento	Prioridad
1	El sistema se desarrolla sobre una plataforma web.	1
2	El sistema deberá estar disponible a cualquier hora del día, durante todos los días.	1
3	El sistema es intuitivo y de fácil uso para los usuarios.	1
4	El sistema se despliega sobre una infraestructura tipo nube.	3

Tabla 2.2 Lista de requerimientos no funcionales

Prioridad

Valores	Descripción
1	Alta
2	Media
3	Baja

2.2.2. Casos de Uso

Una vez definidos los principales requerimientos del sistema, se pueden definir los casos de uso que se establecen como la base para el análisis del sistema de acuerdo a la metodología RUP. Los principales casos de uso se muestran a continuación:

1. Administrar información sitio web
2. Administrar plantillas
3. Administrar formatos
4. Personalizar sitio web
5. Personalizar menú de sitio web
6. Guardar sitio web
7. Pre-visualizar página sitio web
8. Publicar sitio web
9. Administrar páginas
10. Administrar secciones
11. Administrar bloques
12. Administrar bloques tipo Texto
13. Administrar bloques tipo Imagen
14. Administrar bloques tipo Galería
15. Administrar bloques tipo Video
16. Administrar bloques tipo Audio
17. Administrar bloques tipo Facebook
18. Administrar bloques tipo Twitter
19. Administrar bloques tipo RSS
20. Crear sitio web externamente
21. Suspende sitio web externamente
22. Activar sitio web externamente
23. Eliminar sitio web externamente

Para poder verificar que todos los requerimientos exigibles previamente identificados han sido correctamente cubiertos por los casos de uso definidos se muestra la tabla 2.3 que visualiza una matriz de trazabilidad de los casos de uso con los requerimientos.

Caso de Uso	Requerimiento	
	N°	Descripción
Seleccionar Plantilla Sitio Web	1	Permite crear un nuevo sitio web por medio de una url, un usuario y un paquete.
Seleccionar Plantilla Sitio Web	2	Permite seleccionar una plantilla base para un nuevo sitio web.
Administrar información sitio web	3	Permite administrar la información general de un sitio web: Título y Descripción.
Administrar plantillas	4	Permite cambiar de plantilla a un sitio web existente.
Administrar formatos	5	Permite cambiar de formato (disposición de secciones) a un sitio web existente.
Personalizar sitio web	6	Permite modificar el color de fondo de un sitio web.
Personalizar sitio web	7	Permite modificar la imagen de fondo de un sitio web.
Personalizar sitio web	8	Permite modificar la familia de fuente de un sitio web.
Personalizar sitio web	9	Permite modificar el tamaño de fuente de un sitio web,
Personalizar menú de sitio web	10	Permite modificar los colores, orientación, y tamaño de fuente del menú del sitio web.
Guardar sitio web	11	Permite guardar todos los cambios realizados al sitio web,
Pre-visualizar página sitio web	12	Permite pre-visualizar la página seleccionada de un sitio web.
Publicar sitio web	13	Permite publicar el sitio web a su locación final.
Administrar páginas	14	Permite agregar y eliminar páginas a un sitio web.
Administrar páginas	15	Permite modificar el nombre de una página de un sitio web.
Administrar secciones	16	Permite modificar el tamaño vertical de las secciones de una página.
Administrar secciones	17	Permite modificar el color de las secciones de una página.
Administrar páginas	18	Permite limitar el número de páginas de acuerdo al paquete del sitio web.
Administrar bloques	19	Permite agregar y eliminar bloques de contenido a una página del sitio web.
Administrar bloques tipo Texto	20	Permite crear bloques de tipo texto por medio de un WYSIWYG.
Administrar bloques tipo Imagen	21	Permite crear bloques de tipo imagen por medio de la carga de un archivo de imagen.

Administrar bloques tipo Galería	22	Permite crear bloques de tipo galería por medio de la carga de un conjunto de archivos de imagen.
Administrar bloques tipo Video	23	Permite crear bloques de tipo Video Youtube por medio de una url de un video.
Administrar bloques tipo Video	24	Permite crear bloques de tipo Video Vimeo por medio de una url de un video.
Administrar bloques tipo Video	25	Permite crear bloques de tipo Video Carga por medio de la carga de un archivo de video.
Administrar bloques tipo Audio	26	Permite crear bloques de tipo Audio por medio de la carga de un archivo de audio.
Administrar bloques tipo Facebook	27	Permite crear bloques de tipo Facebook Botón Me Gusta por medio de una url.
Administrar bloques tipo Facebook	28	Permite crear bloques de tipo Facebook Feed de Actividad por medio de una url que sea una página de Facebook.
Administrar bloques tipo Facebook	29	Permite crear bloques de tipo Facebook Caja Me Gusta por medio de una url.
Administrar bloques tipo Twitter	30	Permite crear bloques de tipo Twitter por medio de un usuario twitter que obtenga los cinco últimos tweets del mismo.
Administrar bloques tipo RSS	31	Permite crear bloques de tipo RSS por medio de una url.
Administrar bloques	32	Permite redimensionar los bloques existentes en cualquier dirección.
Administrar bloques	33	Permite mover cualquier bloque existente a cualquier posición interior de una página.
Administrar bloques	34	Permite limitar el número de bloques de acuerdo al paquete del sitio web.
Crear sitio web externamente	35	Permite a un sistema externo crear un sitio web.
Suspender sitio web externamente	36	Permite a un sistema externo suspender un sitio web.
Activar sitio web externamente	37	Permite a un sistema externo activar un sitio web.
Eliminar sitio web externamente	38	Permite a un sistema externo eliminar un sitio web.

Tabla 2.3 Matriz de trazabilidad de requerimientos con casos de uso

2.3. Análisis de la solución

En la presente sección se realizará el análisis de la viabilidad del proyecto, para lo cual se analizarán los factores técnicos y económicos relacionados con el proyecto. Así como las restricciones de costo y tiempo determinadas.

Posteriormente, se mostrará por qué es ventajoso desarrollar el presente proyecto mediante un análisis de viabilidad en base a los factores mencionados con anterioridad. Por último, se detallará el diagrama de clases de análisis de la solución.

2.3.1. Análisis de factores técnicos

Para la realización del proyecto se necesita una plataforma web que utilice un motor de base de datos estable. También se necesita de un conjunto de tecnologías que permitan una interfaz sumamente amigable e intuitiva al usuario.

Todas estas herramientas mencionadas son bastante comunes en la actualidad, y se tiene toda una gama de ellas que pueden ser de código abierto como también de código privativo.

- Para el caso de plataforma web se cuenta con las tecnologías JSP de Java y PHP, ambas de código abierto y libre utilización.
- Para el caso de base de datos se tienen libres como MySQL y PostgreSQL.
- Para el caso de las interfaces amigables e intuitivas se cuenta con un par de combinaciones también libres: HTML + Javascript + CSS y HTML + Flash.

Además, se cuenta con abundante documentación de todas estas herramientas diseminada por todo el Internet.

Por estas razones se puede concluir que en cuanto al aspecto técnico, el proyecto es viable y no presenta problemas ni complicaciones en este sentido.

2.3.2. Análisis de factores económicos

En cuanto al análisis económico, es necesario mencionar que no se considerarán costos de licencias. Así es como la mayor parte de costos del proyecto son los costos de construcción del mismo, es decir el costo de horas hombre necesario para construir el producto. También se consideran un conjunto de gastos no laborales que se dan a

lo largo del proyecto como consumo de servicios de luz, agua, teléfono e Internet, gastos de movilidad, materiales y otros adicionales. A continuación se visualiza en la tabla 2.4 la estimación de los costos.

Conceptos laborales

Fase	Costo Hora (S/.)	Horas	Costo(S/.)
Gestión de proyecto	25	56	1,400
Requerimientos	15	56	840
Análisis	20	176	3,520
Diseño	20	136	2,720
Implementación	20	688	13,760
Despliegue	15	88	1,320
Total		1200	23,560

Conceptos no laborales

Concepto	Costo(S/.)
Movilidad (gastos en pasajes)	300
Servicios (luz, agua, teléfono, Internet)	500
Materiales (útiles de escritorio)	300
Otros gastos	200
Total	1,300

Inversión estimada total del proyecto (S/.)	24,860
--	---------------

Tabla 2.4 Estimación del presupuesto para el proyecto

2.3.3. Restricciones de costo y tiempo

Tomando en cuenta el análisis económico anterior, el factor costo no se establece como una restricción real del sistema, debido a que la mayor parte del gasto invertido en el proyecto tiene como base las horas hombre del desarrollador, lo cual no implica un desembolso real de dinero.

En cuanto al tiempo, sí hay una restricción real en la duración del proyecto. De acuerdo al Diagrama de Gantt mostrado en la sección 1.4.2, la duración máxima del proyecto es de 150 días. Además se tiene como restricción el finalizar con

aproximadamente el 80% del proyecto para el mes de octubre del 2011. En función a la planificación dada se debe cumplir con dicha restricción.

2.3.4. Análisis de viabilidad

En base a lo presentado en la sección 2.3.1 Análisis de los factores técnicos, se puede concluir que los factores técnicos no representan una dificultad real para el correcto desarrollo del proyecto, ya que las tecnologías necesarias para el mismo están a plena disposición.

En base a lo presentado en la sección 2.3.2 Análisis de los factores económicos, se puede concluir que la inversión total del proyecto sería de S/. 24,860. Si analizamos detalladamente la composición de esta cantidad, se puede apreciar que más del 95% de la misma corresponde al costo de las horas hombre del desarrollador. Por ello el costo depende en gran medida del esfuerzo invertido para el desarrollo del mismo y no de otros gastos que puedan representar una amenaza real.

En base al análisis presentado en la sección 2.3.3 Restricciones de costo y tiempo, las restricciones para el presente proyecto no son lo suficientemente significativas.

El proyecto como tal ofrece una alternativa de solución para el problema de empresas unipersonales, sohos y pymes en la construcción de un sitio web 2.0 de manera fácil y rápida. Actualmente, las soluciones existentes no brindan una adecuada satisfacción a esta necesidad. Es por ello que el desarrollo de este proyecto brinda una herramienta de gran utilidad a este sector del mercado, la cual permitirá introducir a este sector al mundo del Internet y por ende ampliar sus oportunidades de supervivencia y éxito en el mundo empresarial peruano y mundial.

Finalmente, se puede concluir por todo lo expuesto anteriormente que el proyecto es viable, debido a que no existen mayores dificultades para su desarrollo, y el beneficio del mismo es muy alto.

2.3.5. Clases de Análisis

En función al análisis realizado en la sección 2.2 Identificación de Requerimientos, se define un conjunto de entidades que necesita la solución para poder satisfacer completamente los requerimientos establecidos.

Las entidades se presentan mediante un diagrama de clases de análisis en la figura 2.1, el cual es una herramienta muy útil para describir y comprender el comportamiento y relaciones de estas entidades para poder soportar los requerimientos de la solución.

Con el propósito de aclarar el diagrama de clases, se especifica que una página puede tener 3 secciones como mínimo y como máximo 5. Las posibles secciones son:

- Cabecera (Obligatoria)
- Barra Lateral Izquierda
- Cuerpo (Obligatoria)
- Barra Lateral Derecha
- Pie de Página (Obligatoria)

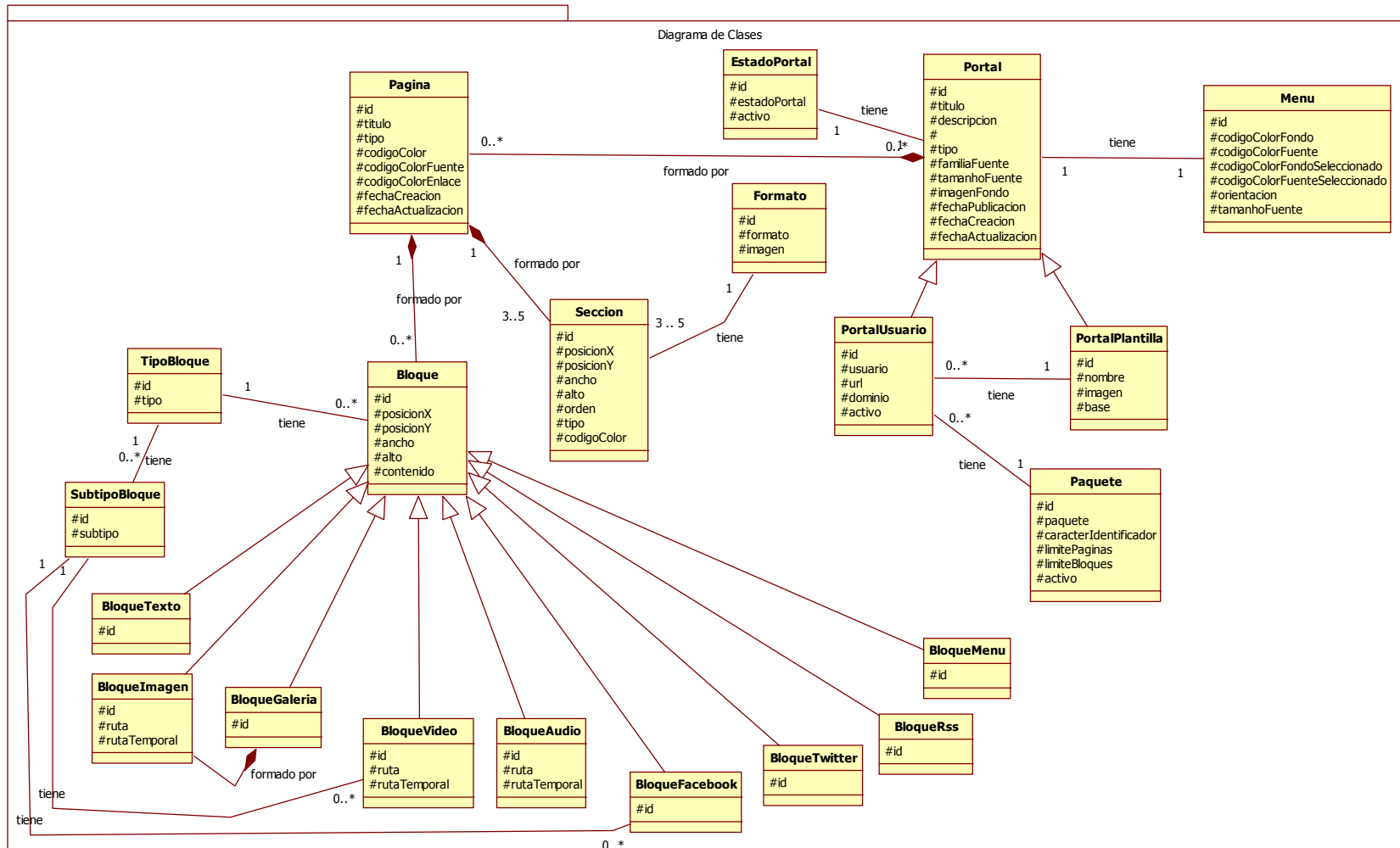


Figura 2.1 Diagrama de clases de análisis

3. Diseño

En el presente capítulo se plantea la arquitectura de la solución, la cual representa los componentes e interfaces que forman parte del diseño a alto nivel de la aplicación. Así también se describe cómo dicha arquitectura soporta los requerimientos planteados.

Más adelante se presenta el diseño de la interfaz gráfica, o medio de interacción entre el usuario y la aplicación, así como también se describen los criterios utilizados para su desarrollo.

3.1. Arquitectura de la solución

En el presente punto se describe la representación de la arquitectura, así como las metas que ésta debe tener. También, con el objetivo de comprender con mayor facilidad el diseño de la solución, se presentan las diferentes vistas arquitectónicas de la aplicación.

3.1.1. Representación de la arquitectura

La solución utiliza un conjunto de estilos arquitectónicos que se describen a continuación:

- Orientado a Objetos: Las entidades del negocio, las entidades gestoras y entidades auxiliares se implementan bajo el paradigma de programación de orientación a objetos.
- Orientado a Eventos: Los controles de interacción con el usuario se implementan utilizando eventos para brindar la mejor experiencia de uso.
- Aplicación web de 3 capas: La aplicación se desarrolla bajo una estructura de 3 capas principales.
 - Capa Externa: Representada por el navegador web, como Mozilla Firefox o Internet Explorer, el cual se encarga de proporcionar la interfaz adecuada para que sea posible realizar peticiones a la capa intermedia.
 - Capa Intermedia: Representada por un servidor web el cual brinda soporte a la tecnología de servidor PHP. Se encarga de realizar las consultas y/o actualizaciones en la capa interna, con el fin de responder a los requerimientos de la capa externa.
 - Capa Interna: Representada por algún sistema de almacenamiento como una base de datos relacional la cual en este caso es MySQL. Cumple la misión de guardar y mantener la información proporcionada por la capa intermedia, la cual a su vez es proporcionada por la capa externa.

3.1.2. Metas y Restricciones de la Arquitectura

Se presentan las metas y restricciones sobre las cuales se rige la arquitectura de la solución:

- La solución se implementa como una aplicación web, a la cual se accede por medio del Internet desde cualquier parte del mundo.
- Para que el usuario disponga del 100% de funcionalidades del sistema deberá hacer uso del navegador Google Chrome versión 5.0 o superior y Mozilla Firefox versión 3.6 o superior. El navegador Internet Explorer, versión 7.0 o superior, también es soportado, sin embargo solo cubre un 95% del total de funcionalidades de la aplicación.
- Es necesario que los usuarios cuenten con acceso a Internet para poder utilizar la solución.

- La validación de datos se realiza a nivel de cliente y servidor.

3.1.3. Vista Lógica

La vista lógica de la aplicación de Construcción de Páginas Web se compone de las capas principales que forman el patrón MVC (Modelo Vista Controlador), ya que es un patrón de diseño sumamente útil para una aplicación web, que permite separar la presentación de la lógica de negocio de una manera limpia y sencilla. A su vez las capas se componen de paquetes que contienen las clases de la aplicación. En la figura 3.1 se visualiza esta representación.

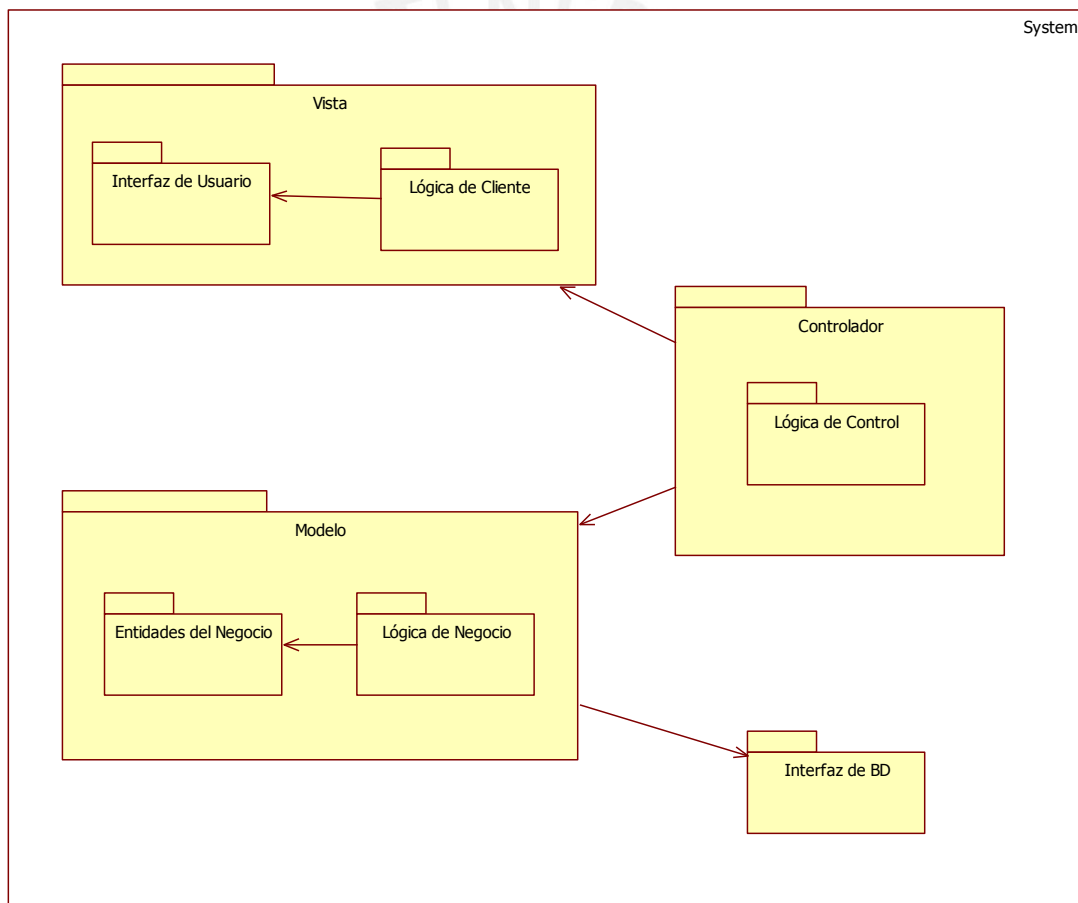


Figura 3.1 Vista Lógica de la solución

- **Lógica de Control:** Contiene a todos los elementos encargados del control de la aplicación. Dirige el funcionamiento principal de la solución.
- **Entidades del Negocio:** Son las clases que representan las abstracciones de la solución como por ejemplo Página.

- **Lógica de Negocio:** Son los elementos encargados de manejar las entidades del negocio para producir un resultado.
- **Interfaz de BD:** Contiene todas las clases relacionadas al acceso e interacción con la base de datos.
- **Lógica del Cliente:** Son las clases que brindan funcionalidad en el lado del cliente por medio de la interfaz, a través de la información suministrada por el controlador y el modelo. Se compone principalmente de clases Javascript.
- **Interfaz de Usuario:** Representa aquellos elementos que generan la presentación con la cual el usuario interactúa. Se compone principalmente de archivos HTML que brindan la estructura de la presentación, y archivos CSS que brindan el diseño estético.

3.1.4. Vista de Despliegue

La vista de despliegue presenta las configuraciones físicas de red sobre las cuales se despliega el Sistema de Construcción de Páginas Web. En la figura 3.2 se muestra el diagrama indicado.

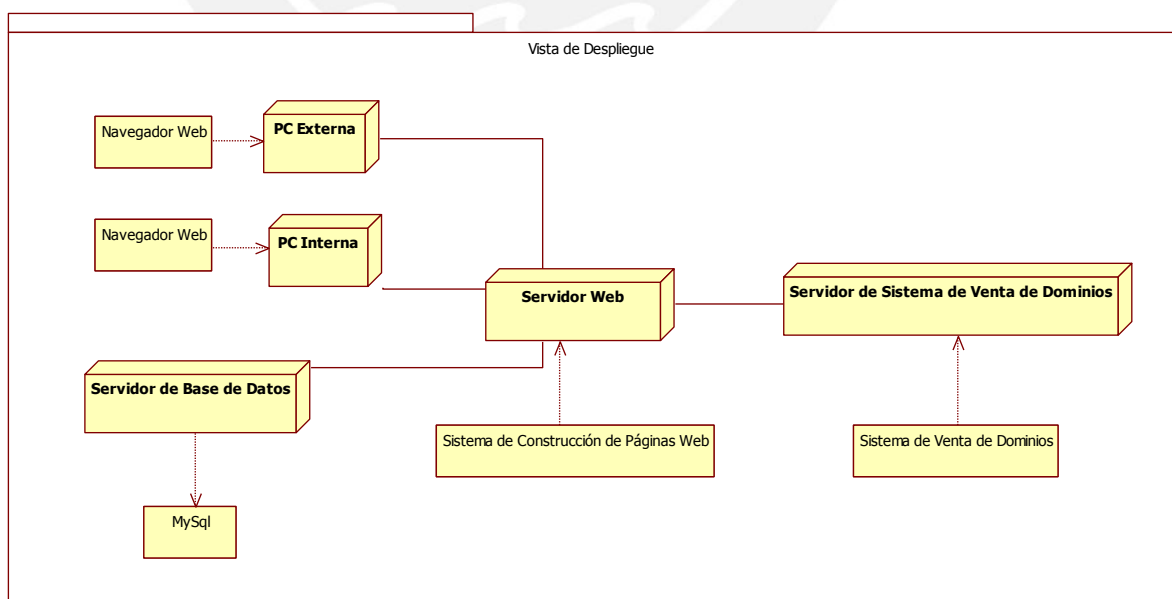


Figura 3.2 Vista de Despliegue de la solución

- **PC Externa:** Representa a un computador externo, es decir fuera de la red local, por medio del cual los usuarios pueden acceder a la aplicación.
- **PC Interna:** Representa a un computador interno, es decir que se encuentra en la red local, por medio del cual los usuarios pueden acceder a la aplicación, para la fase de pruebas por ejemplo.
- **Servidor Web:** Representa a un servidor que se encuentra a la espera de peticiones por parte de los diferentes clientes para luego responder a ellas de forma adecuada, enviando los archivos necesarios para que en el navegador del cliente se visualice la interfaz de la aplicación.
- **Servidor de Sistema de Venta de Dominios:** Representa el servidor al cual se integra el Sistema de Construcción de Páginas Web, el cual se encarga de responder a las peticiones que se realizan al Sistema de Venta de Dominios. Este sistema es el encargado de la selección, compra y activación del dominio en el cual se despliega un sitio web generado por el Sistema de Construcción de Páginas Web.
- **Servidor de Base de Datos:** Representa al servidor que contiene a la base de datos de la aplicación, en la cual se encuentra toda la información correspondiente a los sitios web generados por los usuarios.

3.1.5. Vista de Implementación

La vista de implementación presenta los elementos físicos del sistema por medio del uso de componentes así como la relación entre ellos. A continuación, en la figura 3.3 se muestra el diagrama de implementación de la solución.

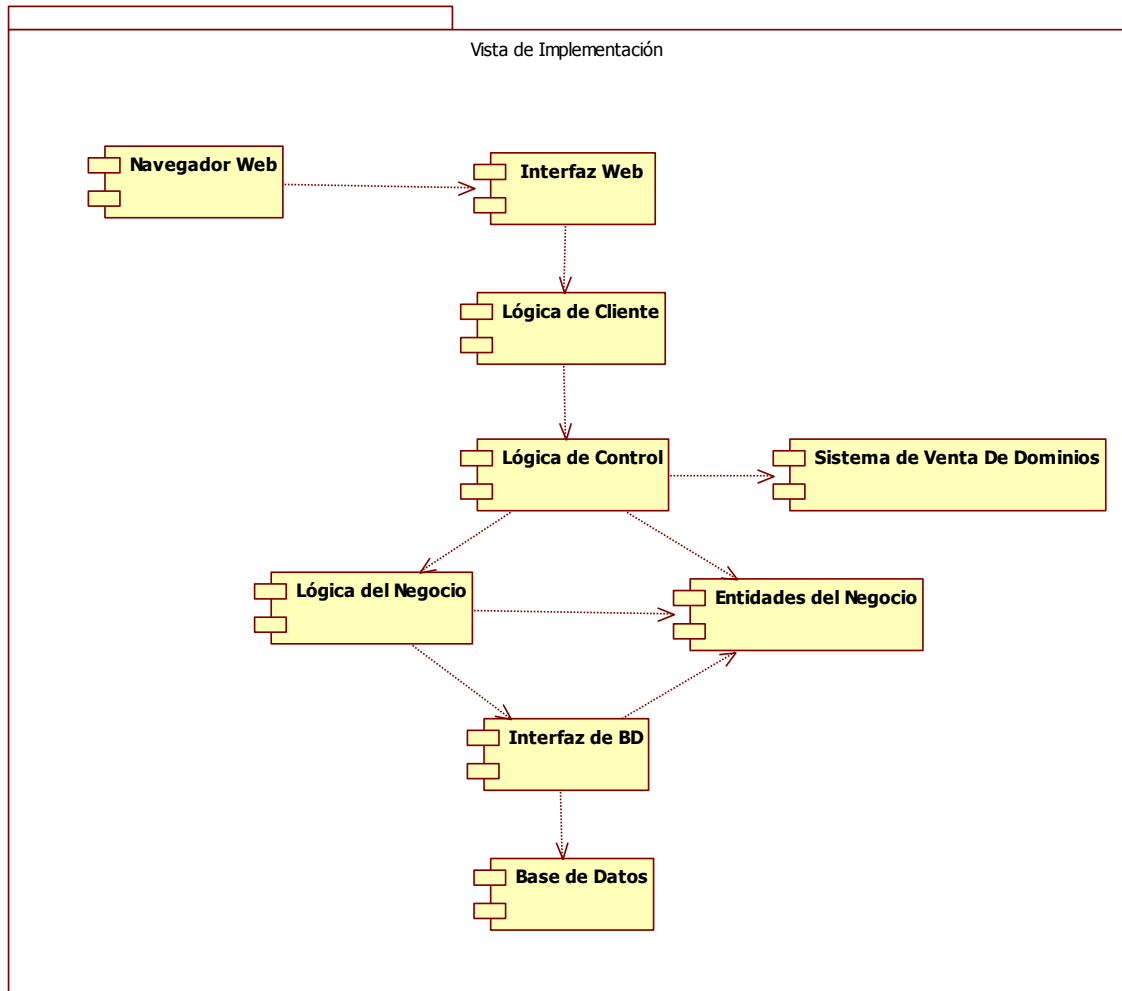


Figura 3.3 Vista de Implementación de la solución

- **Navegador Web:** Componente que representa la interfaz de usuario a través de la cual se realiza peticiones a la aplicación.
- **Interfaz Web:** Componente que representa a los elementos visuales del sistema. Por medio de este componente el usuario interactúa con la aplicación.
- **Lógica de Cliente:** Componente que se encarga de gestionar la petición del usuario generada por un evento en el cliente antes de enviar la petición al servidor web. Se compone principalmente de clases Javascript. En la figura 3.4 se muestra el diagrama de clases Javascript, el cual describe el comportamiento de las entidades y clases gestoras a nivel del lenguaje cliente.

- **Lógica de Control:** Componente que se encarga de gestionar la petición del usuario una vez recibida en el servidor web.
- **Lógica del Negocio:** Componente que contiene toda la implementación de las diferentes funcionalidades propias del negocio en la aplicación.
- **Entidades del Negocio:** Componente que contiene las entidades propias del Sistema de Construcción de Páginas Web, tales como Página y Bloque.
- **Interfaz de la BD:** Componente que se encarga de manejar la interacción de la aplicación con la base de datos.
- **Base de Datos:** Componente que representa la información relacionada a la base de datos MySQL.
- **Sistema de Venta de Dominios:** Sistema externo que se comunica con el Sistema Constructor de Sitios Web. La comunicación se da por medio de un conjunto de servicios web que implementa la aplicación solución. Estos servicios web permiten la creación, suspensión, reactivación y eliminación de un sitio web.

La conexión concretamente se realiza por medio de la implementación de un web service (servicio web), que está formado por los métodos: crearPortal, suspenderPortal, activarPortal y eliminarPortal. Este web service es implementado por el Sistema de Construcción de Sitios Web 2.0, y es consumido por el Sistema de Venta de Dominios. De esta forma cuando por ejemplo un usuario compra un dominio .pe y desea tener su sitio web, entonces este sistema invoca el servicio de creación del sitio web. De la misma forma, si este usuario no cumple con el pago del sitio web, entonces el sistema de venta de dominios invoca el servicio de suspensión del sitio web. Para más detalle acerca del funcionamiento de este servicio, consultar en el Anexo ERS, el módulo de Mantenimiento de Sitios Web.

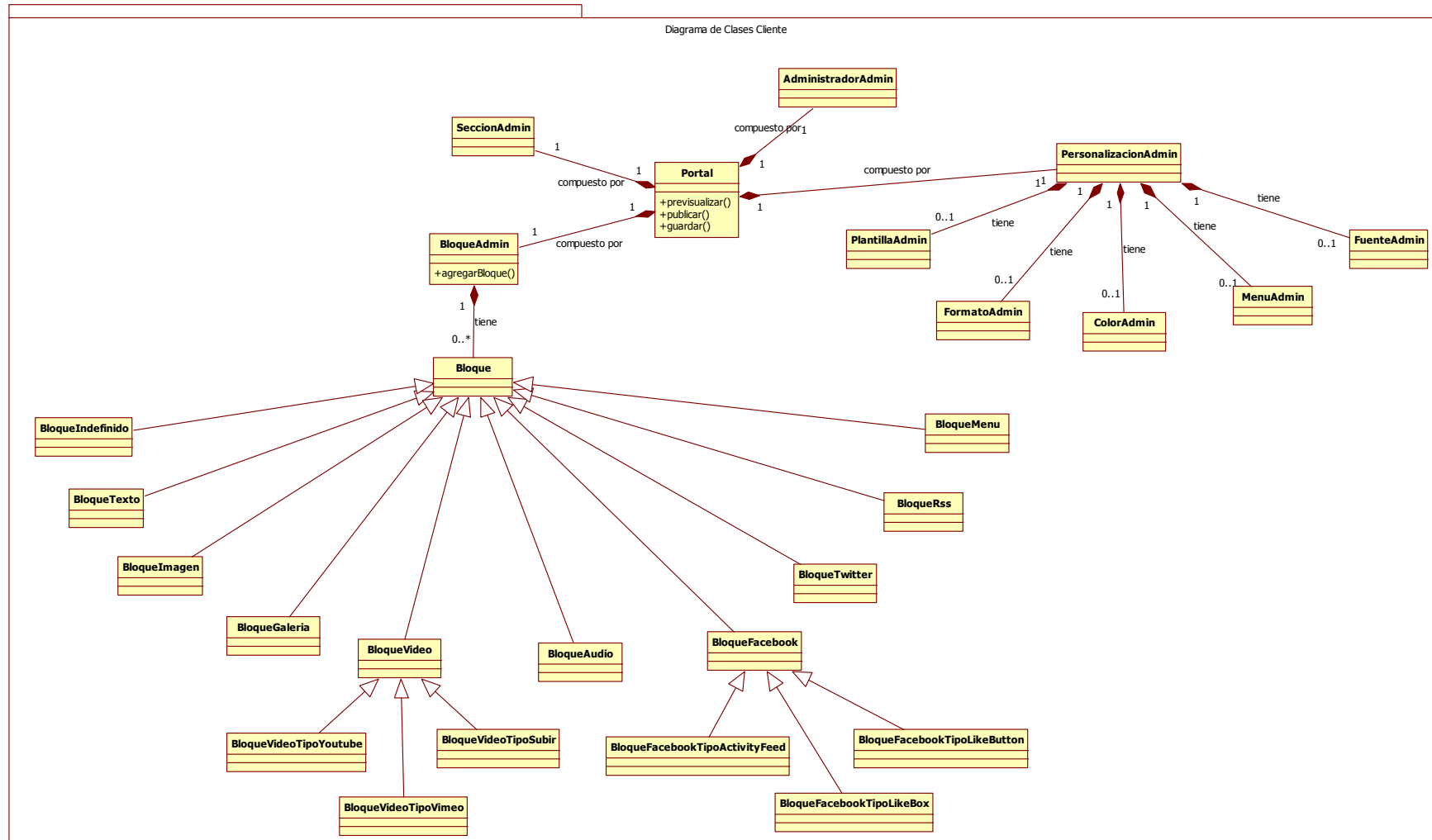


Figura 3.4 Diagrama de Clases en Cliente

3.2. Diseño de Interfaz Gráfica

En la presente sección se describe el diseño de las interfaces gráficas con que cuenta la solución planteada. Estos diseños se desarrollan tomando en cuenta criterios importantes como navegabilidad, usabilidad y facilidad de acceso. De esta forma se espera que el usuario disponga de una interfaz amigable e intuitiva.

Se habla acerca de navegabilidad y usabilidad debido a que la creación y edición de un sitio web se brinda en una sola pantalla. No se tiene que navegar a pantallas externas. En la misma pantalla se puede agregar nuevas páginas para el sitio web, agregar bloques para cada una de estas páginas, personalizar estos bloques, personalizar la misma página e incluso cambiar la plantilla base del sitio web. De la misma forma los controles se visualizan de la forma más clara posible. Todas estas características se dan con el objetivo de brindar la mejor experiencia de utilización al usuario.

A continuación se detallan los diseños más importantes utilizados en el desarrollo de la aplicación.

3.2.1. Pantalla de selección de plantilla de un sitio web

Es la primera pantalla que el usuario visualiza tras la creación de un sitio web. En esta pantalla se permite elegir una de las 20 plantillas disponibles en el sistema. El usuario debe seleccionar una plantilla para poder continuar con el proceso de creación. En la figura 3.5 se puede visualizar la pantalla de selección de plantilla del Sistema de Construcción de Páginas Web.

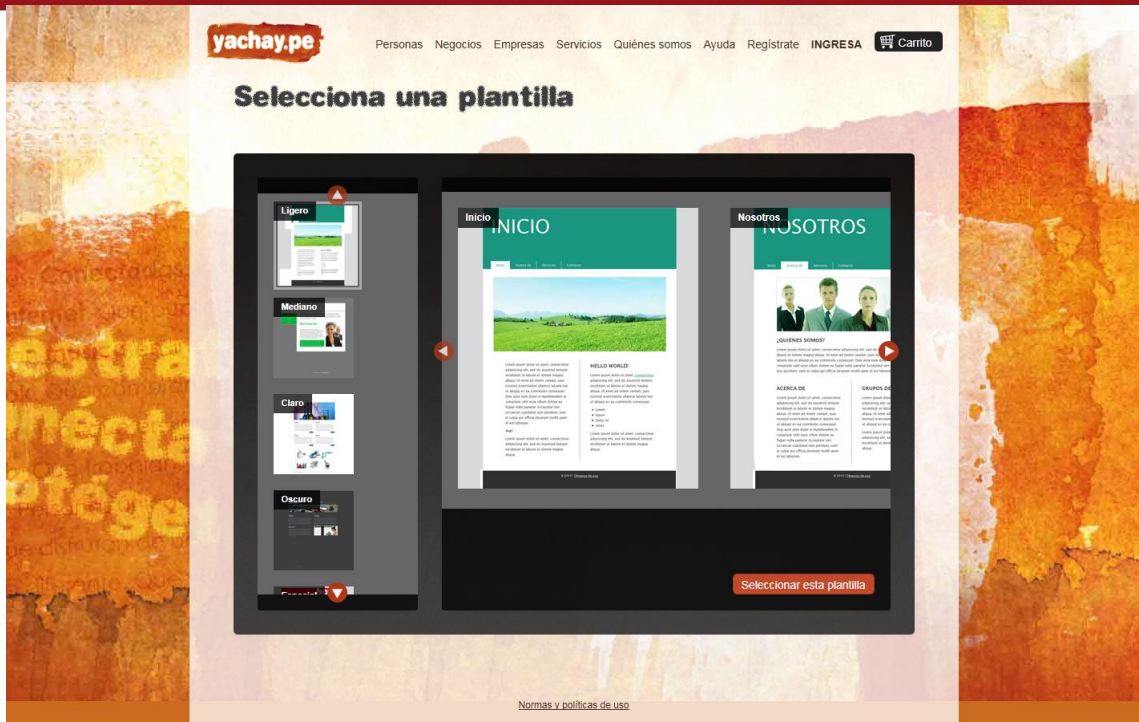


Figura 3.5 Pantalla de Selección de Plantillas

Al seleccionar cualquiera de las plantillas de la barra lateral izquierda, en la parte derecha se actualizan las imágenes correspondientes a la plantilla seleccionada. Tal hecho se puede observar en la figura 3.6.

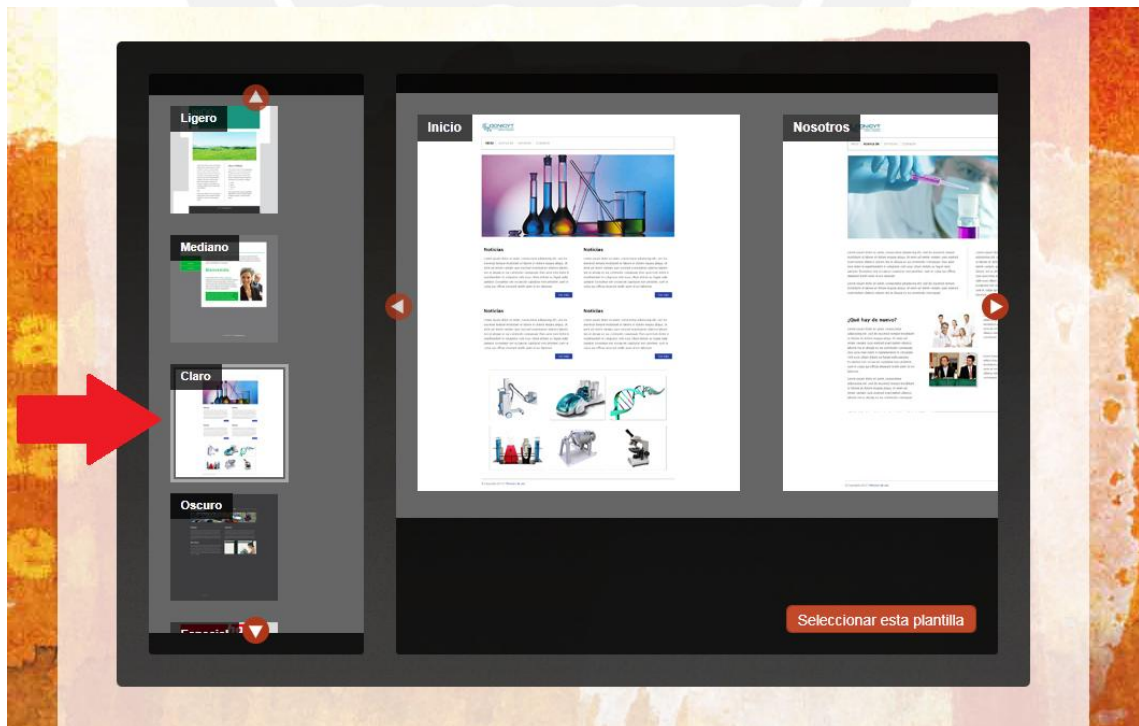


Figura 3.6 Detalle de Selección de Plantilla

3.2.2. Pantalla principal de edición de un sitio web

Una vez elegida la plantilla para el sitio web se visualiza la pantalla principal de edición. En esta pantalla es en donde se realiza todo el trabajo de edición del sitio web. Contiene un conjunto de herramientas y menús que ponen a disposición del usuario las diferentes funcionalidades del sistema. En la figura 3.7 se visualiza la pantalla principal de edición.

A continuación se detallan las partes más importantes de la pantalla principal:

3.2.2.1. Menú Principal

Este menú se encuentra en la parte superior de la pantalla y contiene las funcionalidades principales del sistema. En las figuras 3.8 y 3.9 se muestran las partes del menú principal a detalle.



Figura 3.7 Pantalla Principal de Edición



Figura 3.8 Menú Principal Izquierda

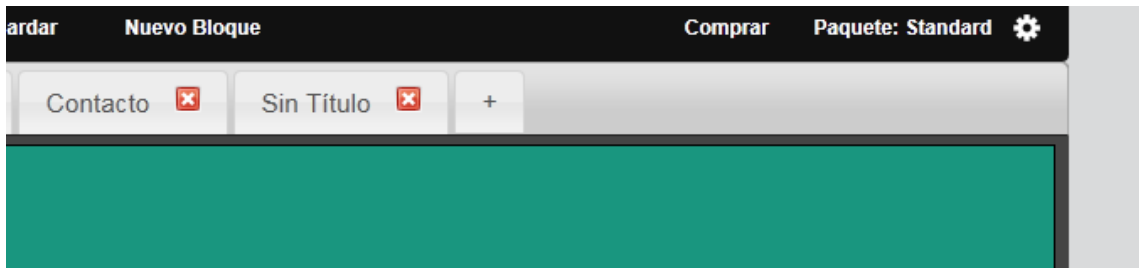


Figura 3.9 Menú Principal Derecha

3.2.2.2. Menú de Tipos de Bloque

En el menú de tipos de bloque se encuentran los diferentes tipos de bloques que se pueden crear en el sitio web. Este menú se visualiza al seleccionar la opción Nuevo Bloque del menú principal. En la figura 3.10 se muestra el menú de tipos de bloque.

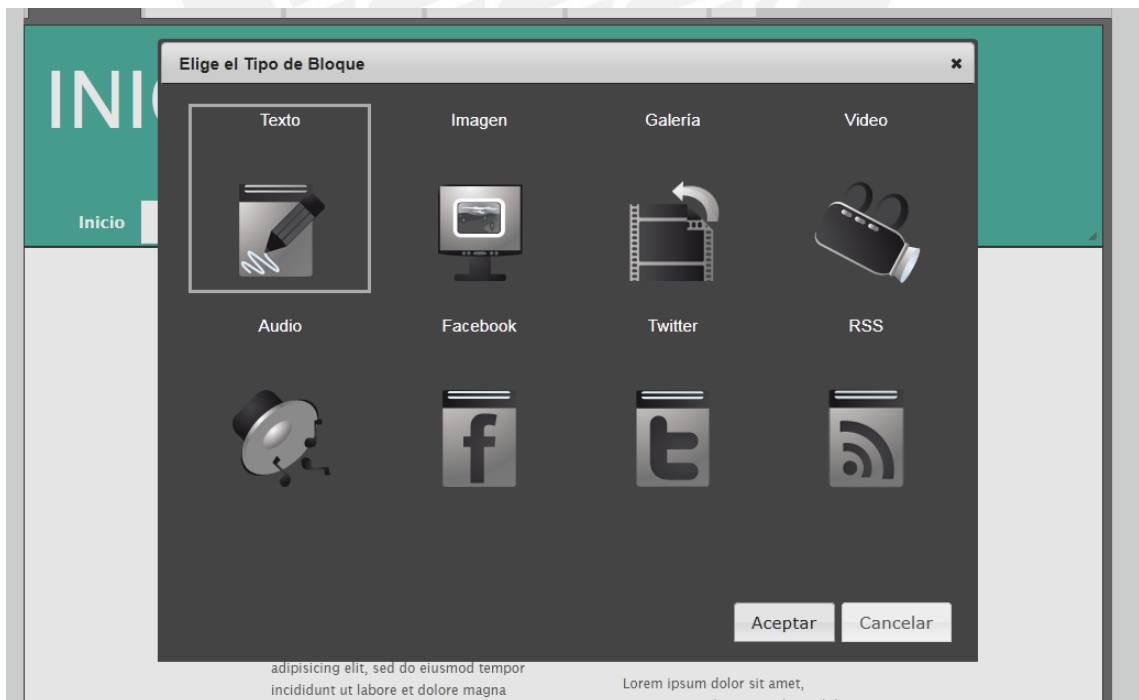


Figura 3.10 Menú de Tipos de Bloque

3.2.2.3. Menú de Personalización

En este menú se presentan las diferentes personalizaciones que se pueden aplicar a un sitio web como cambio de plantilla, formato, colores, menú del sitio web y fuente. Dicho menú se visualiza al seleccionar la opción de Tuerca del menú principal. En la figura 3.11 se muestra el menú de personalización.

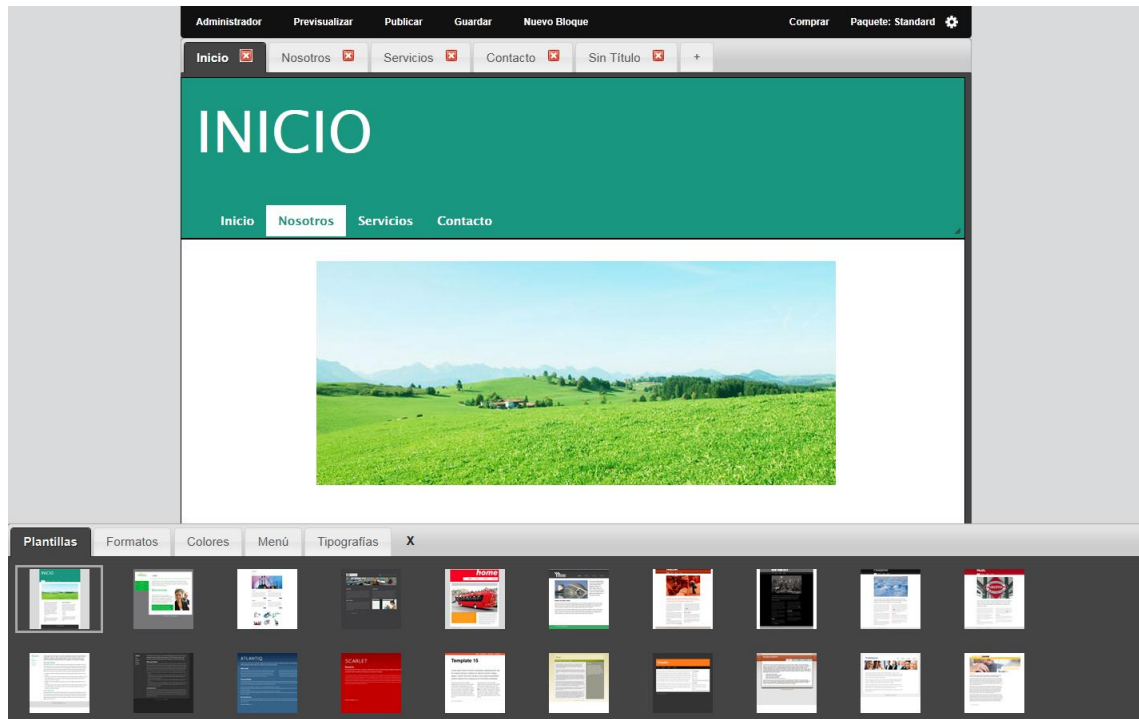


Figura 3.11 Menú Personalización

4. Construcción

En este capítulo se presentan las tecnologías, frameworks y patrones de diseño utilizados en la construcción de la solución, así como la justificación de su uso.

Más adelante se detalla la estrategia de pruebas que se utiliza, así como también los tipos de pruebas realizados, catálogo de pruebas y reporte de ejecución de las pruebas.

4.1. Construcción

En la presente sección se detallan todos los elementos involucrados en la construcción de la solución. De esta forma se describe y justifica la selección de tecnologías y frameworks empleados en el desarrollo de la aplicación, así como también el lenguaje de programación, plataforma y patrones.

4.1.1. Tecnologías

En este punto para cada una de las posibles tecnologías a utilizar se presenta un conjunto de opciones y de acuerdo a sus funcionalidades se justifica la elección de una para el proyecto actual.

4.1.1.1. Lenguajes de Programación

En esta sección se describen algunos lenguajes de programación y a través de su comparación se elige la mejor alternativa a utilizar para la solución. Es necesario especificar que los lenguajes propuestos deben ser de código abierto, es decir sus licencias deben ser de libre utilización y no deben conllevar a un pago.

Entre los lenguajes analizados se tienen los siguientes:

- **Java**

Es un lenguaje estático de programación orientado a objetos y desarrollado por Sun Microsystems. El lenguaje utiliza mucha de la sintaxis de C y C++, pero tiene una base de orientación a objetos mucho más sólida. Otra diferencia con respecto a sus antecesores es que elimina herramientas de bajo nivel que ocasionan muchos errores y problemas como el manejo de punteros a memoria. **(Barker 2005: 1 - 3)**

Entre sus características principales se tiene:

- Java es un lenguaje compilado e interpretado a la vez, ya que se compila a un código intermedio conocido como bytecode, el cual en tiempo de ejecución es interpretado por la máquina virtual de Java. A su vez esta máquina virtual se puede implementar para los diferentes sistemas operativos, es por eso que se afirma que Java es un lenguaje multiplataforma.
- Para el caso de desarrollo web dinámico Java cuenta con un par de posibilidades importantes JSP (Java Server Pages) y JSF (Java Server Faces).
- En el mundo de Java existen un amplio número de herramientas y frameworks que agilizan y facilitan el desarrollo de sistemas como Struts, Spring, Webwork y Hibernate.
- Java a comparación de sus antecesores es un lenguaje más compacto y eficiente sin embargo a comparación de los lenguajes dinámicos se considera un lenguaje muy verboso y largo de codificar.
- Java es un lenguaje multipropósito ya que se pueden codificar desde scripts para tareas simples hasta enormes sistemas de escritorio y web.

- **PHP**

Su nombre es un acrónimo recursivo que significa “PHP Hypertext Preprocessor”. Es un lenguaje dinámico de programación de scripts originalmente diseñado para desarrollo web para producir páginas web dinámicas.

PHP fue originalmente creado por Rasmus Lerdorf en 1995. Desde entonces ha sufrido una evolución, primero con PHP 3 a manos de Zeeb Suraski y Andi Gutmans y más tarde con PHP 4 y PHP 5.

(PHP – Manual 2011: History of PHP)

Las primeras versiones de PHP (1, 2 y 3) no implementan el paradigma de la programación orientada a objetos. Sin embargo su sucesor, PHP 4, tiene la implementación básica, la cual fue ampliamente mejorada en PHP 5. Por otro lado, al igual que otros lenguajes, su sintaxis esta también basada en C y C++, pero con algunas diferencias correspondientes a un lenguaje dinámico.

(Gilmore 2008: 1 - 10)

Entre sus características principales se tiene:

- PHP es un lenguaje interpretado, es decir el código PHP es directamente leído, transformado a optcode y ejecutado por un servidor web que tenga instalado el módulo de PHP.
- Desde sus inicios PHP se desarrolla primordialmente como un lenguaje de desarrollo web, es por ello que también se le conoce como lenguaje de plantillas.
- En el mundo de PHP existen un gran número de herramientas y frameworks, sobre todo de código abierto, que agilizan enormemente el desarrollo, como por ejemplo Symfony Framework, Zend Framework, CakePhp, Drupal, Joomla, y más.
- PHP es un lenguaje de desarrollo rápido debido a su naturaleza dinámica. La programación es menos estricta que en el paradigma estático, ya que por ejemplo no se tienen que declarar variables. Sin embargo debido a este hecho se piensa que promueve el desorden y el código complejo de mantener.
- PHP se ha transformado en un lenguaje multipropósito en el que se pueden codificar desde pequeños scripts, hasta grandes sistemas web y de escritorio (PHP-GTK).

- **Python**

Es un lenguaje de programación interpretado de alto nivel, cuya filosofía de diseño enfatiza la legibilidad del código. Python soporta múltiples paradigmas, principalmente el orientado a objetos, el imperativo, y en menor grado el estilo de programación funcional. **(Python 2011: About)**

Python fue concebido a finales de los 80s y su implementación se inicia en 1989 por Guido van Rossum en CWI (Holanda). Luego llega Python 2.0 en el 2000 con muchas funcionalidades nuevas como un recolector de basura y soporte de Unicode. En el 2008 se lanza Python 3.0, tras un largo periodo de pruebas.

(Summerfield 2010: 1 - 3)

Entre sus características principales se tiene:

- Python, como la gran mayoría de lenguajes dinámicos, es interpretado.
- En el desarrollo web, Python ofrece muchas posibilidades ya que puede ser utilizado con un servidor web que cuente con el módulo de Python.
- El entorno de Python es extenso ya que cuenta con herramientas y frameworks que agilizan el desarrollo, entre los principales se destaca su propia librería estándar y el framework de facto para el desarrollo web, Django.
- Por su naturaleza dinámica acelera y agiliza el desarrollo, cuando es utilizado.
- Desde sus inicios se desarrolla como un lenguaje multipropósito, es decir que permita ejecutar cualquier tipo de tareas.

En la tabla 4.1 se muestran las ventajas y desventajas que cada lenguaje de programación presenta para el presente proyecto.

Característica	Java	PHP	Python
Multiplataforma	Sí	Sí	Sí
Nivel de Aprendizaje	Intermedio	Fácil / Intermedio	Intermedio
Software Libre	Sí	Sí	Sí
Documentación	Amplia	Amplia	Regular

Nivel de Experiencia	Intermedio	Intermedio / Avanzado	Básico
Utilizado en Empresa de Sistema de Dominios	Sí	Sí	No
Facilidad de Despliegue	Regular	Fácil	Regular
Agilidad de Desarrollo	Regular	Muy Buena	Buena

Tabla 4.1 Comparación de lenguajes de programación

A partir de los lenguajes de programación presentados se puede observar que tienen muchas características similares en cuanto a los criterios evaluados. Sin embargo se opta por elegir PHP como lenguaje de desarrollo de la solución debido a la experiencia previa del desarrollador con este lenguaje, a la agilidad que presenta el lenguaje en el desarrollo de sistemas web y a que el Sistema de Venta de Dominios también se encuentra implementado en este lenguaje de programación.

4.1.1.2. Servidor Web

En la actualidad existen una serie de servidores web que hacen posible el funcionamiento de aplicaciones en el lenguaje de programación PHP, como por ejemplo Apache, Internet Information Services (IIS), Nginx y Lighttpd.

(Shklar and Rosen 2009: 121)

Apache es un servidor web cuya historia inicia con los orígenes del Internet. En el 2009 se consolida como el primer servidor web utilizado en más de 100 millones de sitios web. Es desarrollado y mantenido por toda una comunidad libre de desarrolladores que son auspiciados por la Fundación de Software Apache. Está disponible para un gran número de sistemas operativos, sin embargo los que le ofrecen un soporte más amplio son la familia de sistemas operativos Linux.

(Apache Software Foundation 2011: Apache HTTP Server)

El servidor Apache forma parte de la combinación LAMP que significa Linux + Apache + MySQL + PHP. Este combo de tecnologías es uno de los más utilizados en la actualidad para las aplicaciones web debido en parte a que el servidor Apache se encuentra disponible en un gran número de servicios de hosting. De esta forma

Apache se convierte en una opción sumamente robusta y estable que presenta un gran nivel de integración con PHP.

(Gerner, Naramore, Owens and Warden 2006: 1 - 2)

Internet Information Services o IIS es un servidor web desarrollado por Microsoft Corporation para dar soporte a sus diferentes soluciones para el desarrollo web. En sus inicios solo brindaba soporte para desplegar aplicaciones ASP.NET, sin embargo en la actualidad también brinda soporte para desplegar aplicaciones PHP.

Tiene un conjunto de características que lo sitúan como una buena alternativa como, modularidad, extensibilidad, integración con la plataforma Microsoft, avanzadas herramientas de administración, compresión y caching dinámico, y seguridad avanzada.

(The Official Microsoft IIS Site 2011: IIS)

Estos dos servidores web presentados, son los más importantes, en la actualidad, para el desarrollo web. Entre ellos se elige al servidor web Apache debido a su gran integración con el lenguaje de programación PHP. La gran desventaja del servidor web IIS de Windows es que necesita el sistema operativo Windows Server como base, el cual, para poder ser utilizado, necesita del pago de una licencia.

4.1.1.3. Tecnologías de Presentación

Para poder obtener una aplicación que tenga muchas características parecidas a las aplicaciones de escritorio, como interfaces sumamente amigables e intuitivas, se debe elegir alguna de las opciones que existen actualmente. Así se obtiene una RIA (Rich Internet Application), o aplicación rica de Internet, la cual permite una interacción fácil, intuitiva y sólida entre el usuario y la aplicación. Se encuentran un conjunto de tecnologías que permiten que una aplicación sea denominada RIA como: HTML + Javascript + CSS, HTML + Flash y HTML + Silverlight.

(W3C: RIA 2011)

Es necesario mencionar que no se toma en cuenta la combinación HTML + Silverlight debido a que Microsoft Silverlight es una tecnología que tiene gran integración con los productos Microsoft y al igual que ellos necesita del pago de una licencia para ser utilizado.

- **HTML + Javascript + CSS**

Esta combinación de tecnologías es la más utilizada en las aplicaciones y sitios web en la actualidad. Se describe cada una de ellas.

HTML (Hypertext Markup Language), es la tecnología de presentación sobre la que se basa el Internet desde sus inicios. Constituye la capa de contenido de una aplicación web ya que en ella se sitúa la información que se desea compartir a través de la web. Esta información se estructura mediante etiquetas, rodeadas por corchetes de apertura y de cierre de la etiqueta ('<' y '>').

En su quinta edición (HTML 5) la tecnología trae muchas mejoras y adelantos que no tenía antes como manejo nativo de audio y video, un elemento nativo para las imágenes (canvas), almacenamiento nativo de datos, locación geográfica, mensajes, workers y sockets.

(Lawson and Sharp 2011: 1 - 3)

Javascript es un lenguaje de programación interpretado que se encarga de la interacción en tiempo real del usuario con una aplicación web. Constituye la capa de comportamiento de una aplicación web ya que ella dirige las interacciones posibles del usuario en el navegador. Esta capa es la que permite que las aplicaciones web tengan una interacción y apariencia parecidas a las aplicaciones de escritorio.

(Olsson and O'Brien 2008: 1 - 2)

CSS (Cascading Style Sheets), es la tecnología de presentación estética de una aplicación web. Constituye la capa de presentación ya que controla los estilos, formas, diseños y colores en los cuales se presentará un sitio web al usuario.

En su tercera edición (CSS3) trae muchas mejoras importantes, como la utilización de módulos para la división de grandes documentos. Así como la adición de nuevos elementos para que la estilización de las partes de un sitio web pueda ser más detallada y específica, por ejemplo tags para bordes (border-color), tags para fondo (background-size), tags para efectos de texto (text-shadow), así como también nuevos selectores de atributos como [att^=val].

Estas tres capas componen la presentación de un sitio o aplicación web. Así se puede decir que se cuenta con archivos .html, .js y .css entre los archivos de presentación de un proyecto web. La ventaja de realizar esta separación es la de modificar o reemplazar cualquiera de las capas sin tener que cambiar las otras. Otra de las ventajas de utilizar esta combinación de tecnologías es que tienen gran integración con otros proyectos que también se desarrollan en ellas como Google Maps, librerías Javascript, librerías CSS, etc.

(Olsson and O'Brien 2008: 1 - 3)

Otro resultado de la utilización de estas tres tecnologías es AJAX (asynchronous JavaScript and XML), técnica web utilizada para hacer peticiones de contenido al servidor y mostrar sus resultados al usuario en tiempo real, sin tener que recargar la aplicación como se hacía en el pasado. Esta técnica tiene por objetivo conseguir una mejor experiencia de utilización para el usuario.

(Zakas 2009: 567)

- **HTML + Flash**

Adobe Flash es una plataforma multimedia que se utiliza para brindar animación, video, e interactividad en páginas y sistemas web. Frecuentemente es usado para anuncios, juegos y animaciones. Es así que se considera como una herramienta para producir aplicaciones ricas de Internet (RIAs).

(Keefe 2008: 1)

Flash manipula vectores y gráficos de trama para ofrecer animación de texto, dibujos e imágenes. También puede capturar los eventos de entrada generados por el ratón, teclado, micrófono, cámara y más. Otra de sus características es que permite ofrecer la transmisión bidireccional de audio y video.

Flash contiene un lenguaje de programación orientado a objetos interno denominado ActionScript el cual soporta automatización por medio del lenguaje Javascript Flash (JFSL).

Una desventaja de Flash es que para poder ser visualizado en los navegadores necesita que un programa cliente sea instalado, denominado Adobe Flash

Player. Este programa es gratuito para los navegadores, teléfonos móviles y otros dispositivos electrónicos.

Para utilizar Flash en una página web, se utiliza un tag HTML que permite cargar el archivo de tipo Flash y de esta forma poder visualizar el contenido deseado.

(Allen and Arnold 2008: 1)

A continuación en la tabla 4.2 se muestra una comparación a detalle de las dos alternativas presentadas.

Característica	HTML + Javascript + CSS	HTML + Flash
Rendimiento	Muy Bueno (Nativo)	Bueno
Estabilidad	Muy Bueno (Nativo)	Bueno
Documentación	Muy Buena	Buena
Nivel de Experiencia	Intermedio	Básico
Orientado a SEO	Muy Bueno	Regular
Necesita Extensión	No (Nativo)	Sí
Portable	Muy Bueno	Regular

Tabla 4.2 Comparación de tecnologías de presentación

De acuerdo a la tabla 4.2 se puede observar claramente que la mejor opción a utilizar es la combinación de tecnologías HTML + Javascript + CSS. Debido a que son tecnologías nativas al navegador, su rendimiento es mayor que el de Flash, el cual necesita de una extensión para poder funcionar.

Otro de los factores clave para no seleccionar Flash es el hecho de que no es amigable con respecto al SEO (Search Engine Optimization), ya que los motores de búsqueda comprenden muy poco acerca de los archivos Flash embebidos en una página web. Un escenario muy distinto se produce con el contenido HTML, el cual es totalmente leído e indexado por los motores de búsqueda. Así aquellas páginas basadas en HTML tienen mejores puestos en los resultados de las búsquedas.

Por último hay algunas plataformas que no soportan Flash, como por ejemplo la familia de dispositivos móviles Apple, como el iPad y el iPhone. Esta limitación no se presenta con la primera opción de tecnologías debido a que son nativas al navegador.

(Jerkovic 2010: 154)

4.1.1.4. Gestor de Base de Datos

Un gestor de base de datos es un programa que se encarga de la creación, mantenimiento y uso de la base de datos. Una base de datos es un conjunto de registros y archivos que permiten guardar información para luego recuperarla en el futuro.

En la actualidad existen una gran cantidad de gestores de bases de datos como Oracle, MySQL, DB2, etc. Sin embargo para la evaluación se utiliza gestores de libre disponibilidad: MySQL y PostgreSQL.

- **MySQL**

Es un gestor de base de datos relacional que ejecuta un servidor el cual provee de acceso multiusuario a un número de bases de datos. Es el gestor de base de datos más utilizado a nivel mundial ya que muchos de los proyectos más populares como Joomla, Drupal, Wordpress, Google y Facebook, lo utilizan a gran escala.

MySQL provee un gran número de funcionalidades como soporte multiplataforma, procedimientos almacenados, disparadores, schemas, motores de almacenamiento independientes, soporte de Unicode, soporte de transacciones, y más.

Entre las características en las que MySQL sobresale se tiene, velocidad al ser la base de datos más rápida actualmente, facilidad de uso en comparación a otras bases de datos, capacidad de soportar múltiples conexiones al mismo tiempo, seguridad, conectividad, portabilidad y disponibilidad.

Para el presente proyecto una de las características más importantes a considerar es el hecho de que MySQL pueda soportar un gran número de usuarios al mismo tiempo. **(DuBois 2009: 1 - 2)**

- **PostgreSQL**

Es un gestor de base de datos orientado a objetos, el cual se encuentra disponible para muchas plataformas como Windows, Linux y Mac. Es el gestor de código abierto más avanzado en la actualidad ya que tiene muchas características de los gestores de licencia privada.

Entre sus características principales tiene manejo de índices, disparadores, control de concurrencia multi versión, objetos definidos por el usuario, herencia, soporte de transacciones, soporte de Unicode, respaldo en línea, recuperación por puntos en tiempo, replicación binaria, y más.

PostgreSQL sobresale principalmente en robustez, escalabilidad y configurabilidad. Así como en el manejo de transacciones, las cuales, ejecuta mucho más rápido que MySQL, sin embargo este último le gana en cuanto a velocidad de lectura de registros.

(PostgreSQL 2011: PostgreSQL)

A continuación en la tabla 4.3 se muestra una comparación a detalle de las dos opciones presentadas.

Característica	MySQL	PostgreSQL
Rendimiento	Muy Bueno	Muy Bueno
Estabilidad	Muy Buena	Muy Buena
Documentación	Muy Buena	Buena
Nivel de Experiencia	Intermedio	Básico
Multiplataforma	Sí	Sí
Transacciones	Sí	Sí
Velocidad de Inserción	Buena	Muy Buena
Velocidad de Lectura	Muy Buena	Buena
Escalabilidad	Muy Buena	Muy Buena
Seguridad	Buena	Muy Buena
Complejidad	Intermedia	Alta

Tabla 4.3 Comparación de gestores de bases de datos

De acuerdo a la tabla 4.3, se puede apreciar que ambas opciones son muy parecidas, sin embargo debido a su mejor documentación, a la experiencia del desarrollador, a su

velocidad de lectura y a la complejidad que presenta el gestor de base de datos, se elige MySQL.

4.1.1.5. Entorno de Desarrollo

A partir del lenguaje de programación elegido se plantea la posibilidad del uso de tres entornos de desarrollo de software libre: Netbeans, Eclipse y Zend Studio

A continuación se muestra la descripción de cada uno.

- **Netbeans**

Es un entorno de desarrollo integrado y software libre, el cual permite escribir, compilar y ejecutar aplicaciones fácilmente. Su desarrollo está dado utilizando el lenguaje de programación Java.

Actualmente tiene soporte para una gran cantidad de lenguajes de programación como Ruby, Java, PHP, C++, etc.

Para el caso particular de la versión para PHP trae consigo un conjunto de herramientas que ayudan al desarrollo como auto completación de código, soporte para los frameworks Zend y Symfony, integración con MySQL, etc.

Es de fácil instalación, su interfaz es bastante intuitiva, aunque su apariencia visual no es muy atractiva. Debido a su madurez como IDE (entorno de desarrollo integrado) es un producto bastante estable y robusto que permite el desarrollo de aplicaciones de una forma fácil y sencilla.

(Netbeans 2011: Netbeans)

- **Eclipse**

Eclipse IDE es un entorno de desarrollo integrado y software libre, el cual permite el desarrollo de aplicaciones en casi cualquier lenguaje de programación, incluyendo Java, C/C++, Python, Ruby, y más.

Se basa sobre el fundamento de ser una herramienta universal de código abierto y que sirva para cualquier tarea de codificación requerida.

Se puede agregar nuevas funcionalidades al IDE por medio de plugins, los cuales son componentes que se integran al entorno de desarrollo y brindan nuevas opciones como reconocimiento de código Javascript, de código de hojas de estilo (CSS), soporte para Subversion, etc. Sin embargo toda esta

personalización no es muy sencilla y requiere que el usuario tenga cierto grado de familiaridad y manejo del entorno para poder realizarla.

La interfaz gráfica que ofrece es muy intuitiva, así como su apariencia visual bastante atractiva. Por su madurez como entorno y sus múltiples versiones Eclipse se convierte en una opción estable y robusta para el desarrollo de aplicaciones de diferente tipo.

(Eclipse 2011: Eclipse)

- **Zend Studio**

Es el entorno de desarrollo integrado elaborado por Zend, la compañía de PHP. Es un entorno enfocado en el desarrollo de aplicaciones y sistemas en el lenguaje PHP, el cual cuenta con una gran gama de herramientas y funcionalidades que aceleran y facilitan el desarrollo.

Entre sus funcionalidades principales se encuentran auto completión de código avanzada especial para el lenguaje PHP, soporte para reconocer la sintaxis del lenguaje Javascript, como también de las hojas de estilo en cascada utilizadas para el diseño de una aplicación web. También trae una muy buena integración con el Zend Framework, desarrollado por la misma compañía, como también con otros productos como el Zend Server, Zend Guard, Zend PHP Cloud Platform y más.

La interfaz gráfica de Zend Studio es muy intuitiva y agradable, al igual que su apariencia visual. El producto cuenta con múltiples versiones a lo largo de su vida de desarrollo, así se convierte en un entorno muy robusto y estable que representa una gran alternativa en el desarrollo de aplicaciones en PHP. Sin embargo tiene una gran desventaja, no es software libre, es decir se tiene que hacer el pago de una licencia para poder utilizarlo. La ventaja de este punto es que la compañía ofrece un gran soporte al entorno con frecuentes actualizaciones y mejoras.

(Zend Studio 2011: Zend Studio)

En la tabla 4.4 se resumen las ventajas y desventajas que presentan los entornos de desarrollo descritos anteriormente.

Característica	Netbeans	Eclipse	Zend Studio
Usabilidad	Muy buena	Muy buena	Muy buena
Estabilidad	Buena	Muy buena	Muy buena
Software Libre	Sí	Sí	No
Documentación	Buena	Buena	Muy buena
Nivel de Experiencia	Intermedio	Intermedio / Avanzado	Básico
Soporte de Frameworks	Symfony, Zend	Symfony, Zend, Cake, Code Igniter	Zend
Versatilidad de Extensiones	Regular	Muy Buena	Buena

Tabla 4.4 Comparación de entornos de desarrollo

En función a la información presentada se puede observar que una de las mejores opciones a utilizar es el entorno Zend Studio, a pesar de la poca experiencia que tiene el desarrollador con este entorno. Sin embargo, debido a los lineamientos del proyecto se debe escoger una herramienta de software libre. Es por ello que se escoge el entorno Eclipse que cuenta con un conjunto de extensiones que satisfacen las necesidades de funcionalidades para el desarrollo del proyecto. Además el desarrollador cuenta con amplia experiencia previa con este entorno.

4.1.2. Patrones de Diseño

A continuación de describen y comparan dos de los patrones de diseño más conocidos.

4.1.2.1. Modelo Vista Controlador

Es el patrón de diseño de tipo arquitectura altamente recomendado para aplicaciones web. Permite organiza en tres capas las diferentes partes de una aplicación. Esta separación permite que disminuya la duplicación de código, centraliza el control y permite que la aplicación sea más extensible.

Estas tres capas son las siguientes:

- **Modelo**

Contiene a las entidades del negocio, así como su funcionalidad. Accede a la capa de acceso a datos.

- **Controlador**

Es el que dirige el flujo de una aplicación. Recibe los eventos de entrada y se encarga de realizar peticiones al modelo para presentar su información por medio de la vista.

- **Vista**

Es la encargada de mostrar la información facilitada por el controlador, a través de la interfaz gráfica de usuario. Así también notifica al controlador cuando se produce un evento.

(Zaninotto and Potencier 2007: 13 - 16)

En la figura 4.1, se visualiza la relación entre los diferentes componentes del patrón.

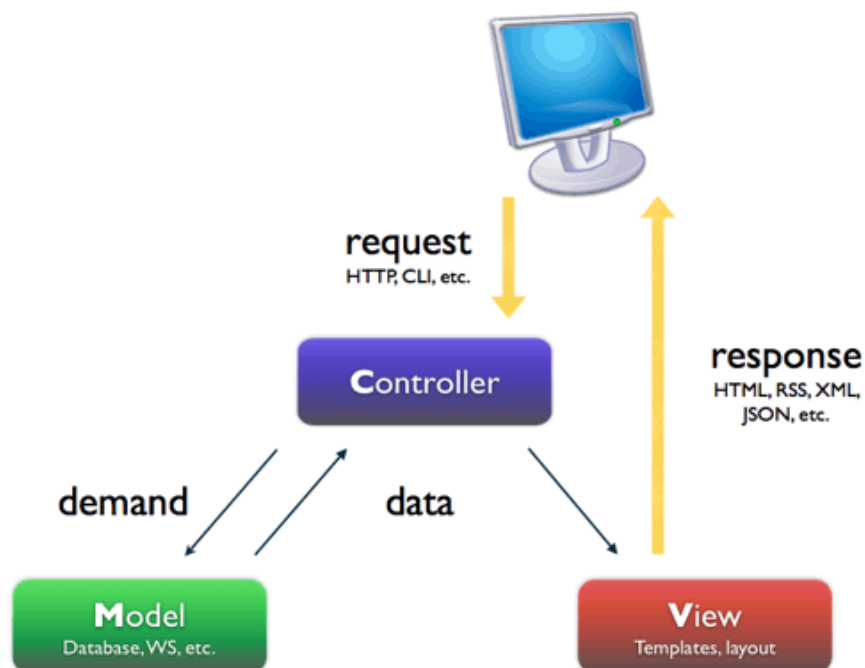


Figura 4.1 Modelo Vista Controlador

(Zaninotto and Potencier 2007: 15)

4.1.2.2. Modelo Vista Presentador

Es el patrón de diseño de tipo arquitectura derivado del patrón MVC. Se centra en que la vista tenga la menor cantidad de código posible y cuya lógica sea administrada por una clase “Presentador”, que no dependa de los componentes de la interfaz gráfica, y de esta forma, sea más fácil de probar.

Está compuesta por cuatro capas:

- **Modelo**
Compuesta por las entidades de la aplicación.
- **Interfaz de Vista**
Medio por el cual el presentador se comunica con la vista.
- **Vista**
Se encarga de la presentación de los datos. Mantiene una referencia a su presentador, al cual, le delega la responsabilidad del manejo de eventos.
- **Presentador**
Contiene la lógica para manejar los eventos cuando se presenten. También manipula el estado de la vista mediante una referencia a la interfaz de vista. Además, recurre al modelo para conocer la forma de responder a los eventos generados por la vista.

En la tabla 4.5 se muestra una comparación a detalle de las dos alternativas presentadas.

Característica	MVC	MVP
Clara separación de Conceptos	Sí	Sí
Reutilización de Código	Sí	Sí
Documentación	Muy Buena	Regular
Nivel de Experiencia	Intermedio	Bajo
Flexibilidad	Sí	Sí
Manejo de Vistas Múltiples	Fácil	Regular
Complejidad	Intermedia	Alta

Tabla 4.5 Comparación de patrones de diseño

De acuerdo a la tabla 4.5, se observa que debido a la amplia documentación, al nivel de experiencia y a la complejidad presentada, la mejor opción es la del patrón Modelo Vista Controlador.

4.1.3. Frameworks

Los frameworks son importantes herramientas para agilizar y facilitar el desarrollo. A continuación se escogen los frameworks de acuerdo a las tecnologías escogidas anteriormente.

4.1.3.1. Framework de PHP

Una vez elegido el lenguaje de programación servidor, el cliente y el framework del lenguaje cliente, también se debe elegir una herramienta que facilite el desarrollo a nivel de lenguaje servidor.

En el mundo PHP existe una amplia variedad de frameworks con diferentes características. Tomando en cuenta las características del proyecto, se proponen tres opciones a evaluar: Code Igniter, Yii y Symfony.

- **Code Igniter (V2.0)**

Es un framework libre desarrollado por EllisLab, cuya meta es permitir que los desarrolladores realicen proyectos más rápido que escribiendo todo desde cero, por medio de un conjunto de librerías y funcionalidades comunes en el desarrollo web.

Code Igniter está basado en el popular patrón de diseño Modelo Vista Controlador, el cual permite una clara separación entre la información, la lógica de negocio y la presentación.

La ventaja de este framework es que es uno de los más rápidos que existen, permitiendo que no haya mucha diferencia en utilizarlo o utilizar archivos planos de php, es decir es muy ligero.

(Code Igniter 2011: Code Igniter)

- **Yii (V1.1)**

Es un framework de PHP diseñado para brindar un muy alto rendimiento. Está desarrollado basado en componentes, los cuales permiten realizar aplicaciones permitiendo la máxima reutilización y acelerando significativamente el proceso de desarrollo.

Yii ofrece toda una gama de funcionalidades, como utilizar el patrón de diseño Modelo Vista Controlador, una capa de acceso a la base de datos (DAO, Data Access Object), integración con jQuery, autenticación y autorización, internacionalización y localización, manejo de formularios y validación, manejo de cache, generación automática de código, etc.

Para el presente proyecto este framework ofrece como principal fortaleza su escalabilidad y muy alto rendimiento, los cuales se mantienen aún cuando aumenta el flujo de usuarios.

(Winesett 2010: 1 - 2)

- **Symfony (V1.4)**

Es un framework de PHP que implementa el patrón de diseño Modelo Vista Controlador. Su meta principal es la de acelerar el proceso de creación y mantenimiento de aplicaciones web y reemplazar las tareas de codificación repetitivas.

Symfony ofrece una serie de funcionalidades como, integración nativa con dos ORMs (Object Relational Mapping) populares las cuales son Propel y Doctrine, fácil instalación y configuración, manejo avanzado de formularios y validación, internacionalización y localización, enrutamiento y urls inteligentes, autenticación y autorización, y manejo de cache avanzado.

Symfony es un framework listo para empresas ya que sigue un conjunto de lineamientos de seguridad, arquitectura y estabilidad necesarios para proyectos largos y de envergadura. Así también está basado en muchos conceptos de desarrollo rápido como RAD (Rapid Application Development) y DRY (Don't repeat yourself).

(Zaninotto and Potencier 2007: 1 - 3)

A continuación en la tabla 4.6 se muestra una comparación a detalle de las tres alternativas presentadas.

Característica	Code Igniter	Yii	Symfony
MVC	Sí	Sí	Sí
Rendimiento	Bueno	Muy Bueno	Regular
Estabilidad	Muy Buena	Muy Buena	Muy Buena
Documentación	Buena	Muy Buena	Excelente
Nivel de Experiencia	Básico	Intermedio	Avanzado
Escalabilidad	Bueno	Muy Bueno	Bueno
Integración con otras librerías	Regular	Bueno	Bueno
Integración nativa con ORM	No	No	Sí
Versatilidad de Componentes	Regular	Bueno	Muy Bueno
Dificultad de Configuración	Baja	Media	Alta

Tabla 4.6 Comparación de frameworks de PHP

De acuerdo a la tabla 4.6, se puede observar que Symfony ofrece para la mayoría de los criterios, los mejores resultados. Sin embargo para el presente proyecto no se van a utilizar de forma intensa componentes como formularios y validación ya que la mayor parte de interacción con el usuario se da a través de Javascript. Lo que sí es necesario es que el framework brinde un alto rendimiento y escalabilidad para soportar un gran número de usuarios concurrentes.

Así, a pesar de que Symfony parece una mejor opción para proyectos generales, por sus múltiples funcionalidades y excelente documentación, para este proyecto en particular se elige el framework Yii debido a sus especificaciones base.

4.1.3.2. Framework de Javascript

Una vez elegida la combinación HTML + Javascript + CSS como tecnologías de presentación, es necesario elegir un framework de Javascript que soporte y agilice el desarrollo del Sistema de Construcción de Sitios Web.

El problema principal que resuelven estos frameworks es el hecho de que el javascript nativo tiene diferencias de navegador a navegador, entonces es necesaria una herramienta que oculte estas diferencias brindando una interfaz única para el desarrollo. Otro de los objetivos es el de ofrecer un conjunto de funciones ya implementadas que son comunes en el desarrollo web, así como también lo ayudan y agilizan.

Entre las principales opciones de frameworks libres de Javascript se tiene, jQuery, Dojo Toolkit y MooTools.

- **jQuery**

Es una librería de Javascript inter navegador, es decir que soporta un conjunto de navegadores, diseñada para simplificar el desarrollo por el lado del cliente. Es el framework de Javascript más popular en la actualidad, ya que se utiliza en el 46% de los 10,000 sitios web más visitados hoy en día.

jQuery ofrece una amplia gama de funcionalidades como selección de elementos del DOM (Document Object Model), modificación del DOM, manejo simplificado de eventos, manipulación de CSS, efectos y animaciones, ajax, extensibilidad a través de plugins y utilitarios como la práctica función “each”.

Para el presente proyecto las funcionalidades más beneficiosas que jQuery puede brindar son la selección de elementos del DOM, la cual se da por medio del estándar establecido por CSS3. De esta forma provee un conjunto de herramientas poderosas para poder obtener información como también modificar los elementos de un documento HTML.

Otra de las funcionalidades prioritarias para el proyecto es el manejo de efectos y animaciones. Alrededor del mundo de jQuery existen un sin número de plugins los cuales permiten realizar efectos que antes solo se tenían pensados para sitios web construidos a base de Flash. Se tiene efectos como el arrastrar objetos en el navegador (drag and drop), y redimensionar elementos (resize).

(Chaffer and Swedberg 2009: 1 - 3)

- **Dojo Toolkit**

Es una librería de Javascript modular, diseñada con el objetivo de facilitar el desarrollo rápido entre navegadores. La librería es soportada por la fundación Dojo, la cual es una organización sin fines de lucro, cuyo objetivo principal es el de promover la utilización de la librería.

Dojo ofrece un gran conjunto de componentes listos para utilizar como menús, tabs, tablas ordenadas, gráficos dinámicos, animaciones y efectos, herramientas de árbol que soportan arrastrar, formularios y componentes de validación.

A pesar de la gran cantidad y calidad de sus componentes Dojo tiene dos debilidades. Su documentación es de baja calidad ya que es confusa y en muchas ocasiones incompleta. Así también, se considera a muchos de sus componentes poco flexibles, debido a que no se pueden adaptar a situaciones parecidas ya que no cuentan con opciones de configuración adecuadas.

(Russell 2008: 1 - 3)

- **MooTools**

Significa My Object Oriented Tools o Mis herramientas orientadas a objetos. Es un framework ligero para Javascript. Se centra en ofrecer modularidad y reusabilidad, a través de mejoras que le hace al mismo lenguaje Javascript, es decir a sus funciones, para agilizar y facilitar el desarrollo por el lado del cliente.

MooTools ofrece un conjunto de ventajas como extensibilidad y configurabilidad que permiten al desarrollador escoger solo las herramientas que va a utilizar. También ofrece un componente de efectos muy avanzado, que implementa transiciones optimizadas como flexibilización de ecuaciones, las cuales son utilizadas por desarrolladores de Flash por su calidad.

Una de las características peculiares de MooTools es que está basado en Prototype, una de las primeras librerías de Javascript disponibles para el desarrollo. Es por ello que MooTools modifica o mejora muchas funciones nativas de Javascript. La ventaja de ello es que permite tener funciones poderosas nativas en el lenguaje. La desventaja radica en que si se desea utilizar librerías externas que utilizan estas funciones, resultará en conflictos ya que estas funciones han sido modificadas.

(Newton 2008: 1 - 2)

A continuación en la tabla 4.7 se muestra una comparación a detalle de las tres alternativas presentadas.

Característica	jQuery	Dojo Toolkit	MooTools
Rendimiento	Muy Bueno	Muy Bueno	Excelente
Estabilidad	Muy Buena	Muy Buena	Muy Buena
Documentación	Muy Buena	Regular	Buena
Nivel de Experiencia	Intermedio	Básico	Básico
Drag and Drop	Sí	Sí	Sí
Redimensión de elementos	Sí	Sí	Sí

Soporte de Selectores	Sí	No	Sí
Soporte de Ajax	Muy Bueno	Bueno	Muy Bueno
Compatibilidad con otras librerías	Muy Buena	Muy Buena	Mala
Soporte de Navegadores	Muy Bueno	Muy Bueno	Muy Bueno

Tabla 4.7 Comparación de frameworks Javascript

Como se aprecia en la tabla 4.7, para algunas características como soporte de navegadores y estabilidad, las tres alternativas muestran buenos resultados. Sin embargo, para otras como compatibilidad, soporte de selectores y documentación se puede observar la clara superioridad que demuestra jQuery.

Las características presentadas como soporte de selectores y soporte de ajax son muy importantes para el desarrollo del proyecto ya que permiten realizar un manejo fácil y sencillo de los elementos del DOM en la programación cliente, así como el intercambio de información transparente con el servidor que permite ajax. Estas características, apoyadas por una gran documentación y compatibilidad con otras librerías es que permiten que la elección sea clara. Así, se elige jQuery, como el framework Javascript, para el desarrollo del proyecto.

4.1.3.3. Framework ORM (Object Relational Mapping)

Una vez escogido el framework de desarrollo PHP, se debe escoger el framework necesario para el mapeo objeto relación, el cual permite una mayor velocidad de desarrollo en los procesos relacionados con la base de datos.

En el mundo PHP existen dos ORMs principales: Propel y Doctrine.

- **Propel (V1.3)**

Es una ORM basada en el proyecto Apache Torque de lenguaje Java. Una de las primeras ORMs para PHP. Mediante un conjunto de patrones de diseño, permite acceder a la base de datos por medio de un conjunto de objetos, los cuales implementan una interfaz sencilla para obtener y guardar información.

Brinda un mapeo de la base de datos hacia clases PHP, de acuerdo a una definición del modelo de datos, dada en XML. Este mapeo lo realiza

generando clases PHP las cuales reflejan a las tablas de la base de datos. Así, cada vez que el modelo cambia, Propel regenera las clases de PHP para que se adecúen a estos cambios.

Contiene una DBAL (Database abstraction layer), la cual permite que se cambie de base de datos sin ningún problema, ya que toda la interacción con la misma se abstrae y se oculta al programador para que tenga una sola interfaz de comunicación.

Implementa el patrón de diseño Active Record o Registro Activo, el cual permite que la interfaz de un objeto tenga funciones como insertar, actualizar, y eliminar las cuales actualizan directamente la información de tal objeto con la del registro que le corresponde en base de datos.

(Propel 2011: Propel)

- **Doctrine (V1.2)**

ORM basada en el popular proyecto de Java, Hibernate. Doctrine provee mapeo de las tablas de base de datos a objetos PHP con el objetivo de recuperar y guardar información de forma fácil y sencilla.

Al igual que Propel, Doctrine implementa un conjunto de patrones de diseño que rigen sus funcionalidades como DBAL que brinda una interfaz única sin importar la base de datos sobre la que se trabaje. También implementa Active Record para facilitar la interacción de los objetos con las base de datos por medio de funciones como insertar y actualizar.

A pesar de ser similares en algunos aspectos, Doctrine se diferencia en otros, por ejemplo presenta una funcionalidad muy importante para ejecutar consultas, DQL (Doctrine Query Language), el cual permite obtener información de la base de datos sin tener que escribir repetitivo código SQL. Esta funcionalidad fue basada en el HQL de Hibernate.

Otra de las ventajas de Doctrine sobre Propel es que no regenera las clases mapeadas de PHP, sino en su lugar las reconfigura, lo cual brinda mucha flexibilidad al momento de realizar cambios en el desarrollo.

(Doctrine 2011: Doctrine)

A continuación en la tabla 4.8 se muestra una comparación a detalle de las dos opciones presentadas.

Característica	Propel	Doctrine
Rendimiento	Muy Bueno	Muy Bueno
Estabilidad	Muy Buena	Muy Buena
Documentación	Buena	Muy Buena
Nivel de Experiencia	Básico	Intermedio
Lenguaje de Consultas	No	Sí
DBAL	Sí	Sí
Active Record	Sí	Sí
Transacciones ACID	Sí	Sí
Comportamientos	Sí	Sí

Tabla 4.8 Comparación de ORMs de PHP

En función a la tabla 4.8 y a lo explicado anteriormente se elige como ORM a Doctrine debido a su escalabilidad, característica necesaria para el presente proyecto y a la implementación del DQL, la cual agiliza el desarrollo de las consultas. Así también debido a su flexibilidad al momento de regenerar el modelo.

4.2. Pruebas

En este punto se presenta la estrategia de pruebas a utilizar, que tienen lugar para comprobar el correcto funcionamiento del Sistema de Construcción de Sitios Web. También se incluye una parte principal del catálogo de pruebas con el objetivo de mostrar los resultados obtenidos.

4.2.1. Estrategia de Pruebas

Para el presente proyecto se aplica una estrategia basada en un esquema de pruebas incrementales, el cual aborda las pruebas desde los componentes atómicos o componentes base hasta los componentes más complejos, de acuerdo a la planificación efectuada para el desarrollo del proyecto.

4.2.2. Tipos de Pruebas

A continuación se presentan los tipos de pruebas utilizados para la validación de la solución.

4.2.2.1. Pruebas Unitarias

Permiten probar los diferentes componentes que forman parte del sistema, de manera independiente, para verificar su correcto funcionamiento. Este tipo de pruebas consiguen una gran cobertura de manera granular, ayudando a mejorar el diseño y disminuyendo la necesidad de depuración. La documentación de las pruebas unitarias constituye un elemento muy importante ya que permite conocer el objetivo y los diferentes casos de utilización de un componente. Sin embargo, al modificar un componente, estas pruebas no son suficientes para verificar que el cambio no impacta en otra parte del sistema. Esa es la razón de ser de las prueba de integración.

4.2.2.2. Pruebas de Integración

Permiten probar la combinación de diferentes partes del sistema con el objetivo de determinar si funcionan correctamente integradas. Luego de realizar las pruebas unitarias sobre los componentes individuales, las pruebas de integración permiten probar el buen funcionamiento del sistema cuando existe una transferencia de datos entre los componentes. A pesar de probar la correcta interacción entre los componentes, para brindar un diagnóstico general del sistema se necesitan las pruebas de casos de uso.

4.2.2.3. Pruebas de Casos de Uso

Una vez realizadas las pruebas unitarias y de integración, se efectúan las pruebas de casos de uso, las cuales permiten verificar la correcta implementación de los flujos básico y alternativos de todos los casos de uso presentes en la solución. Después de realizar estas pruebas se puede pasar a las pruebas de aceptación.

4.2.2.4. Pruebas de Aceptación

Pruebas realizadas por el usuario final con el objetivo de validar que el sistema cumpla con el funcionamiento esperado. Estas pruebas son las últimas en realizarse y marcan el fin de la fase de pruebas de sistema.

4.2.3. Catálogo de Pruebas

A continuación se detallan las pruebas para los módulos del sistema.

4.2.3.1. Módulo de Configuración Sitio Web

Se presentan los casos de prueba para el módulo de configuración de sitio web.

- **Administrar Información Sitio Web**

a) Campo de entrada: Título

Clases de Equivalencia	Valores	Identificador
CV	Caracteres alfanuméricos	1
CNV	Caracteres no alfanuméricos	2
CNV	Vacío	3

b) Campo de entrada: Descripción

Clases de Equivalencia	Valores	Identificador
CV	Caracteres alfanuméricos	4
CNV	Caracteres no alfanuméricos	5
CV	Vacío	6

c) Campo de entrada: E-mail

Clases de Equivalencia	Valores	Identificador
CV	Email correcto	7
CNV	Email incorrecto	8
CV	Vacío	9

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1, 4, 7	Título, desc, jonathan.nieto@pucp.edu.pe	Se cierra el formulario y se actualiza la información.	OK
2	1, 6, 9	Título, "" ""	Se cierra el formulario y se actualiza la información.	OK

3	2, 4, 7	Título<>?, desc, jonathan.nieto@pucp.edu.pe	Se visualiza el mensaje: Los caracteres no alfanuméricos nos son permitidos en el Título.	NO OK
4	3, 4, 7	"" desc, jonathan.nieto@pucp.edu.pe	Se visualiza el mensaje: Favor de ingresar Título.	NO OK
5	1, 5, 7	Título, desc<>?, jonathan.nieto@pucp.edu.pe	Se visualiza el mensaje: Los caracteres no alfanuméricos nos son permitidos en la Descripción.	NO OK
6	1, 4, 8	Título, desc, jonathan.nieto@	Se visualiza el mensaje: El E-mail no es válido.	NO OK

4.2.3.2. Módulo de Bloques

Se presentan los casos de prueba para el módulo de bloques del sistema.

- **Administrar Bloques Tipo Texto**
 - a) Campo de entrada: Texto

Clases de Equivalencia	Valores	Identificador
CV	Texto HTML	1
CNV	Vacío	2

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1	Hola, este es mi bloque de texto.	Se cierra el formulario y se actualiza la información.	OK
2	2	""	Se visualiza el mensaje: Favor de ingresar Contenido.	NO OK

- **Administrar Bloques Tipo Imagen**
 - a) Campo de entrada: Imagen

Clases de Equivalencia	Valores	Identificador
CV	Imagen correcta	1
CNV	Imagen Formato Incorrecto	2
CNV	Imagen Peso Incorrecto	3
CNV	Imagen Dimensión Incorrecta	4

CNV	Vacío	5
-----	-------	---

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1	Un archivo imagen correcto	Se cierra el formulario y se actualiza la información.	OK
2	2	Un archivo imagen de extensión .bmp	Se visualiza el mensaje: El tipo de imagen no es permitido. Favor de subir imágenes con extensión .jpg, .gif o .png	NO OK
3	3	Un archivo imagen mayor a 2MB.	Se visualiza el mensaje: La imagen pesa más de 2MB. Por favor escoge un archivo menos pesado.	NO OK
4	4	Un archivo imagen con dimensiones mayores a 980px x 500px.	Se visualiza el mensaje: La imagen mide más de 980px x 500px. Favor de subir otra imagen.	NO OK
5	5	Ningún archivo.	Se visualiza el mensaje: Favor de elegir una imagen para subir.	NO OK

- **Administrar Bloques Tipo Galería**

Se presenta igual que para el caso de imagen pero para 6 componentes de imagen.

- **Administrar Bloques Tipo Video Youtube**

a) Campo de entrada: Youtube Url

Clases de Equivalencia	Valores	Identificador
CV	Url válida de Youtube	1
CNV	Url no válida de Youtube	2
CNV	Vacío	3

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
----	-------------------	---------	--------------------	------

1	1	http://www.youtube.com/watch?v=7QU1nvuxaMA	Se cierra el formulario y se actualiza la información.	OK
2	2	http://www.facebook.com/watch?v=7QU1nvuxaMA	Se visualiza el mensaje: Favor de ingresar una url de video válida.	NO OK
3	3	""	Se visualiza el mensaje: Favor de ingresar la url.	NO OK

- **Administrar Bloques Tipo Video Vimeo**

a) Campo de entrada: Vimeo Url

Clases de Equivalencia	Valores	Identificador
CV	Url válida de Vimeo	1
CNV	Url no válida de Vimeo	2
CNV	Vacío	3

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1	http://www.vimeo.com/22046322	Se cierra el formulario y se actualiza la información.	OK
2	2	http://www.youtube.com/22046322	Se visualiza el mensaje: Favor de ingresar una url de video válida.	NO OK
3	3	""	Se visualiza el mensaje: Favor de ingresar la url.	NO OK

- **Administrar Bloques Tipo Video Subir**

a) Campo de entrada: Video

Clases de Equivalencia	Valores	Identificador
CV	Video correcto	1
CNV	Video Formato Incorrecto	2
CNV	Video Peso Incorrecto	3
CNV	Vacío	4

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1	Un archivo video correcto	Se cierra el formulario y se actualiza la información.	OK
2	2	Un archivo video de extensión .mp6	Se visualiza el mensaje: El tipo de archivo no es permitido. Favor de subir archivos con extensión .mpg-1, .mpg-2, .avi, .mpg4, .divx, .xvid, .mpg, .mpeg o mp3	NO OK
3	3	Un archivo video mayor a 5MB.	Se visualiza el mensaje: El video pesa más de 5MB. Por favor escoge un archivo menos pesado.	NO OK
4	4	Ningún archivo.	Se visualiza el mensaje: Favor de elegir un video para subir.	NO OK

- **Administrar Bloques Tipo Audio**

a) Campo de entrada: Audio

Clases de Equivalencia	Valores	Identificador
CV	Audio correcto	1
CNV	Audio Formato Incorrecto	2
CNV	Audio Peso Incorrecto	3
CNV	Vacío	4

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1	Un archivo audio correcto	Se cierra el formulario y se actualiza la información.	OK
2	2	Un archivo video de extensión .wav	Se visualiza el mensaje: El tipo de archivo no es permitido. Favor de subir archivos con extensión .mp3 o .wma	NO OK
3	3	Un archivo audio mayor a 4MB.	Se visualiza el mensaje: El audio pesa más de 4MB. Por favor escoge un archivo menos pesado.	NO OK
4	4	Ningún archivo.	Se visualiza el mensaje: Favor de elegir un audio para subir.	NO OK

- **Administrar Bloques Tipo Facebook**

- a) Campo de entrada: Tipo

Clases de Equivalencia	Valores	Identificador
CV	Tipo Seleccionado	1
CNV	Tipo no Seleccionado	2

- b) Campo de entrada: Página Url

Clases de Equivalencia	Valores	Identificador
CV	Url válida	3
CNV	Url no válida	4
CNV	Vacío	5

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1, 3	Tipo Seleccionado, http://www.facebook.com/cocacola	Se cierra el formulario y se actualiza la información.	OK
2	1, 4	Tipo Seleccionado, http://face/cocacola ,	Se visualiza el mensaje: La url no es válida	NO OK
3	1, 5	Tipo Seleccionado, ""	Se visualiza el mensaje: Favor de ingresar la url.	NO OK
4	2, 3	"" http://www.facebook.com/cocacola	Se visualiza el mensaje: Favor de seleccionar un tipo.	NO OK

- **Administrar Bloques Tipo Twitter**

- a) Campo de entrada: Usuario Twitter

Clases de Equivalencia	Valores	Identificador
CV	Usuario Twitter Correcto	1
CNV	Usuario Twitter Incorrecto	2
CNV	Vacío	3

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1	Php	Se cierra el formulario y se actualiza la información.	OK
2	2	Aaaaaaaaaaaaaa	Se visualiza el mensaje: Favor de ingresar un usuario Twitter correcto.	NO OK
3	3	""	Se visualiza el mensaje: Favor de ingresar el usuario Twitter.	NO OK

- **Administrar Bloques Tipo RSS**
 - a) Campo de entrada: Url RSS

Clases de Equivalencia	Valores	Identificador
CV	Url RSS correcta	1
CNV	Url RSS Incorrecta	2
CNV	Vacío	3

Casos de Prueba:

CP	Clase Equivalente	Valores	Resultado Esperado	Obs.
1	1	http://elcomercio.pe/feed/portada.xml	Se cierra el formulario y se actualiza la información.	OK
2	2	http://elcomercio/feed	Se visualiza el mensaje: Favor de ingresar una url rss correcta.	NO OK
3	3	""	Se visualiza el mensaje: Favor de ingresar la url.	NO OK

4.2.4. Resultado Pruebas

Para el desarrollo de las pruebas, mencionadas en esta sección, se asignaron a tres personas de tres áreas diferentes de la Red Científica Peruana: Planeamiento, Servicio de Atención al Cliente (SAC) y Contenidos.

Cada una de las tres personas realizó todas las pruebas comprendidas en el Catálogo de Pruebas presentado en esta sección, validaron cada uno de los puntos del

Checklist de Validación (Anexo 6) y adicionalmente completaron una tabla de ponderaciones de las Características de Usabilidad de la Aplicación.

En la tabla 4.9 se presenta el resultado de las pruebas comprendidas en el Catálogo de Pruebas.

Área	Pruebas Erróneas	Pruebas Satisfactorias
Planeamiento	0	37
SAC	0	37
Contenidos	0	37
Total	0	111

Tabla 4.9 Resultado pruebas de Catálogo de Pruebas

Como se puede apreciar a partir de la tabla 4.9 no se presentaron ninguna clase de errores en las pruebas correspondientes al Catálogo de Pruebas.

En la tabla 4.10 se presenta el resultado de la validación de cada uno de los puntos del Checklist de Validación presentado en el Anexo 6.

Área	Pruebas Erróneas	Pruebas Satisfactorias
Planeamiento	0	166
SAC	0	166
Contenidos	0	166
Total	0	498

Tabla 4.10 Resultado pruebas de Checklist de Validación

Como se puede apreciar a partir de la tabla 4.10 todos los puntos del Checklist de Validación fueron revisados y se cumplieron satisfactoriamente.

En la tabla 4.11 se presenta el resultado de las ponderaciones de las Características de Usabilidad de la Aplicación. La escala presentada se da según los valores:

- 5: Muy de acuerdo
- 4: De acuerdo
- 3: Medianamente de acuerdo
- 2: En desacuerdo
- 1: Muy en desacuerdo

Característica / Área	Planeamiento	SAC	Contenidos	Promedio
Interfaz intuitiva y sencilla	5	5	4	4.67
Rapidez en creación / edición (10 mins)	5	5	5	5.00
Compatibilidad Navegadores (5)	5	3	4	4.00
Facilidad en el manejo de páginas	5	5	5	5.00
Facilidad en el manejo de contenidos. (Bloques y Secciones)	5	4	4	4.33
Facilidad en la personalización de sitio web (Menú Personalización)	5	5	4	4.67
Promedio	5.00	4.50	4.33	4.61

Tabla 4.11 Resultado ponderaciones Características Usabilidad

A partir del valor final de **4.61** de la tabla 4.11 se puede decir que la aplicación es sencilla, intuitiva, rápida y fácil de utilizar. En resumen se puede decir que es usable y así cumple con los objetivos establecidos inicialmente para la aplicación.

Es necesario mencionar que para llegar a este resultado idóneo presentado en cada una de las tablas se tuvieron que efectuar tres iteraciones de corrección de errores para el Catálogo de Pruebas, Checklist de Validación y Características de Usabilidad de la Aplicación. En cada una se corrigieron, progresivamente, una serie de errores y defectos hasta llegar al estado actual.

5. Observaciones, conclusiones y recomendaciones

En este último capítulo, se describen los puntos vistos en el proyecto a los que se desea dar énfasis. También se presentan las conclusiones obtenidas en base al trabajo realizado. Por último, se dan consejos acerca del uso y ampliación del proceso seguido y el trabajo resultado.

5.1. Observaciones

- En la actualidad, como se menciona en la descripción del problema, gran parte del crecimiento del país se debe a empresas unipersonales, SOHOs y pymes, las cuales tendrían un mayor crecimiento económico si es que utilizarasen las tecnologías de la información. Éste es el objetivo principal que se quiere satisfacer con la solución propuesta. Por otro lado se sabe que la actualización de páginas web es un factor cada vez más determinante en el éxito de una de estas empresas.
- La construcción de la solución se basa totalmente en decisiones meditadas y razonadas, vistas desde el punto de la ingeniería, lo que representa un reto importante de investigación, documentación, estudio y aprendizaje. La

utilización de nuevas tecnologías, no solo se debe a su aspecto innovador, sino fundamentalmente por su aporte al cumplimiento de las funcionalidades del Sistema de Construcción de Sitios Web. Como se muestra en el documento, todas las alternativas investigadas tienen ventajas y desventajas frente a otras, pero es a través de la investigación y comparación de estas alternativas que se puede obtener la conclusión, y decisión de utilizar determinada herramienta por sobre las demás. El producto indudable de este proceso de selección es un sistema de calidad.

- La utilización de una guía de gestión de proyectos como PMBOK, acompañada por una metodología de desarrollo de software como RUP, permiten un progreso meditado y ordenado de los procesos y actividades. De la misma forma los documentos y modelos propuestos por las metodologías facilitan la documentación y el control de todas las actividades realizadas en el proyecto.
- Un prototipo de arquitectura desarrollado de forma adecuada, permite mitigar riesgos que se puedan encontrar en cuanto a la utilización de nuevas tecnologías y librerías. De no ser así, el proyecto podría tener problemas si lo planeado no se puede realizar con las tecnologías propuestas.
- La realización de pruebas unitarias, de integración y aceptación, brinda la posibilidad de obtener un producto que cumpla con los requisitos especificados por el usuario, reduciendo la posibilidad de errores y el correspondiente retraso en el proyecto.
- El Sistema de Construcción de Sitios Web se puede utilizar para construir prácticamente cualquier tipo de sitio web independientemente del tipo de negocio del usuario. Por medio de las diferentes opciones de configuración y de diseño se puede crear un sitio web de acuerdo a los lineamientos y preferencias del usuario, tanto en cuestión de contenido como de diseño.

5.2. Conclusiones

- Se cumple con el objetivo de desarrollar un sistema de construcción de sitios web, así como, de integrar este sistema con otro de venta de dominios.

- El tiempo invertido en las etapas de análisis y diseño minimiza riesgos en la implementación y reduce pérdidas de tiempo, debido a que una acertada definición de funcionalidades permite que los cambios posteriores sean mínimos. Por ejemplo si se hubiese realizado un análisis rápido y no concienzudo del diagrama de clases, se habrían presentado muchos problemas en la fase de construcción de la jerarquía de bloques, debido a la complejidad presentada.
- La selección de una arquitectura adecuada permite el desarrollo de un sistema de manera rápida y organizada. Para el caso, se logra que el acceso a datos, la lógica de negocio y la interfaz gráfica sean componentes independientes, lo cual permite un sistema flexible y de fácil mantenimiento. De esta forma si se desea por ejemplo cambiar base de datos por algún motivo, este cambio ocasionaría un impacto mínimo en los componentes.
- La utilización de una ORM permite agilizar el desarrollo y acortar los tiempos de construcción del proyecto, ya que por medio del mapeo entre tablas de la base de datos y clases en la programación, así como de la automatización de las operaciones básicas de inserción, actualización y selección; permiten que la interacción con la base de datos, por medio del lenguaje de programación, no sean muy complicadas ni requieran de arduo trabajo.
- La utilización de web services para que el Sistema de Venta de Dominios pueda interactuar con el Sistema de Construcción de Sitios Web es una muy buena solución de integración entre dos sistemas independientes, ya que permite que su desarrollo sea por separado y sólo a través de un conjunto de métodos interactuar fácilmente.

5.3. Recomendaciones y Trabajos Futuros

- Para satisfacer completamente a los usuarios finales se debe probar el sistema en varios navegadores y versiones de estos, con el objetivo de especificar desde qué versiones y qué navegadores se pueden utilizar para navegar en el sistema. Como se señala en el documento actualmente solo se soportan Internet Explorer desde la versión 7.0, Mozilla Firefox desde la versión 3.6 y Google Chrome desde la versión 5.0. Se debería ampliar en el futuro esta lista de navegadores soportados.

- Se recomienda implementar, en el futuro, un módulo de ayuda y soporte técnico, que permita a los usuarios registrar los problemas que se presentan durante el uso de la aplicación, de modo que se brinde información adecuada sobre las posibles soluciones.
- La solución cuenta con 15 plantillas como base para realizar un sitio web. Este número se puede ampliar en beneficio del usuario. También se puede categorizar por rubros las diferentes plantillas para que la selección inicial sea más sencilla.
- La solución desarrollada, presenta un gran potencial futuro, ya que todavía existen toda una gama de funcionalidades que se le pueden agregar a la solución para que sea más potente aún, como también, para que pueda satisfacer de mejor forma las demandas actuales de un sitio web. Se puede agregar por ejemplo un bloque de Google Analytics para tener un conjunto de cálculos y métricas de las visitas al sitio web. También se le puede incluir un bloque de formulario de contacto, el cual estaría direccionado al correo electrónico del propietario del sitio web. Así también, podría tener un menú con las estadísticas de los bloques consumidos, los tipos de estos bloques, y de igual forma el número de los bloques utilizados por página. Se darían todas estas adiciones con el objetivo de brindar una mejor experiencia de utilización al usuario.

Bibliografía

- Kevin Potts
2008 Web Design and Marketing Solutions for Business Websites
New York, USA: Springer-Verlag, Inc
- Bud E. Smith, Ryan C. Williams
2009 Google Business Solutions All-in-One For Dummies
Indianapolis, Indiana: Wiley Publishing, Inc
- Mick Spillane
2011 How to Open & Operate a Financially Successful Notary Business
Ocala, Florida: Atlantic Publishing Group, Inc
- Britannica Online Encyclopedia
2011 Website
Consulta: 6 de junio de 2011
<<http://www.britannica.com/EBchecked/topic/690679/Web-site>>
- W. Jason Gilmore
2008 Beginning PHP and MySQL: From Novice to Professional (Third Edition)
EEUU, New York: Apress
- Bruce Lawson and Remy Sharp
2011 Introducing HTML 5
Collingwood, Australia: New Riders
- Tommy Olsson and Paul O'Brien
2008 The Ultimate CSS Reference
Collingwood, Australia: Site Point
- Network Working Group
2011 Domain Name System
Consulta: 6 de junio de 2011
<<http://tools.ietf.org/html/rfc1035>>
- Lacnic
2011 Lacnic
Consulta: 6 de junio de 2011
<<http://www.lacnic.net/>>
- How Stuff Works
2011 Web 1.0
Consulta: 6 de junio de 2011
<<http://computer.howstuffworks.com/web-101.htm>>
- Tim O'Reilly
2008 What Is Web 2.0: Design Patterns and Business Models for the Next
Generation of Software
Sebastopol (CA) USA: Tim O'REILLY

- Web Social - Master en bibliotecas y servicios de información digital
 2011 Diferencias Principales entre Web 1.0 y Web 2.0
 Consulta: 6 de junio de 2011
 <<http://mariawebsoc.blogspot.com/2012/04/analisis-de-la-lectura-3-key.html>>
- Mark S. Choate
 2007 Professional Wikis (Dec 2007)
 Indianapolis, Indiana: Wiley Publishing, Inc
- Todd Tomlinson
 2010 Beginning Drupal 7
 New York: Apress, Inc
- Dan Rahmel
 2007 Professional Joomla
 Indianapolis, Indiana: Wiley Publishing, Inc
- Wikipedia
 2011 Wikipedia Administration
 Consulta: 10 de octubre de 2011
 <<http://en.wikipedia.org/wiki/Wikipedia:FAQ/Administration>>
- Tim Boronczyk with Martin E. Psinas
 2008 PHP and MySQL: Create-Modify-Reuse
 Indianapolis, Indiana: Wiley Publishing, Inc
- Clara Shih
 2009 The Facebook Era
 Indianapolis, Indiana: Prentice Hall
- Kevin Makice
 2009 Twitter API: Up and Running
 Indianapolis, Indiana: Prentice Hall
- Philippe Kruchten
 2003 The Rational Unified Process, An Introduction (3rd Edition)
 California: Addison Wesley
- Jacquie Barker
 2005 Beginning Java Objects, Second Edition
 EEUU, New York: Apress
- PHP - Manual
 2011 PHP: History of PHP
 Consulta: 09 de setiembre de 2011
 <<http://www.php.net/manual/en/history.php>>
- Python
 2011 About
 Consulta: 09 de setiembre de 2011
 <<http://www.python.org/about/>>

- Mark Summerfield
2010 Programming in Python 3 (Second Edition)
California: Addison Wesley
- Leon Shklar and Rich Rosen
2009 Web Application Architecture: Principles, Protocols and Practices
Indianapolis, Indiana: Wiley Publishing, Inc
- Apache Software Foundation
2011 Apache HTTP Server
Consulta: 16 de agosto de 2011
<http://httpd.apache.org/ABOUT_APACHE.html>
- Jason Gerner, Elizabeth Naramore, Morgan L. Owens, Matt Warden
2006 Professional LAMP
Indianapolis, Indiana: Wiley Publishing, Inc
- The Official Microsoft IIS Site
2011 IIS
Consulta: 16 de agosto de 2011
<<http://www.iis.net/>>
- W3C: RIA
2011 Accessible Rich Internet Applications (WAI-ARIA) 1.0
Consulta: 09 de setiembre de 2011
<<http://www.w3.org/TR/wai-aria/>>
- Nicholas C. Zakas
2009 Professional JavaScript for Web Developers, 2nd Edition
Indianapolis, Indiana: Wiley Publishing, Inc
- Matthew Keefe
2008 Flash and PHP Bible
Indianapolis, Indiana: Wiley Publishing, Inc
- Chris Allen and Wade Arnold
2008 Open Source Flash Development
EEUU, New York: Apress
- John I. Jerkovic
2010 SEO Warrior
Sebastopol, CA: O'Reilly Media, Inc.
- Paul DuBois
2009 MySQL (Fourth Edition)
California: Addison Wesley
- PostgreSQL
2011 PostgreSQL
Consulta: 28 de agosto de 2011
<<http://www.postgresql.org/about/>>

- Netbeans
2011 Netbeans
Consulta: 16 de agosto de 2011
<<http://netbeans.org/>>
- Eclipse
2011 Eclipse
Consulta: 16 de agosto de 2011
<<http://www.eclipse.org/>>
- Zend Studio
2011 Zend Studio
Consulta: 16 de agosto de 2011
<<http://www.zend.com/products/studio/>>
- François Zaninotto and Fabien Potencier
2007 The Definitive Guide to symfony
EEUU, New York: Apress
- Code Igniter
2011 Code Igniter
Consulta: 28 de agosto de 2011
<<http://codeigniter.com/>>
- Jeffery Winesett
2010 Agile Web Application Development with Yii 1.1 and PHP5
Olton Birmingham, UK: Packt Publishing Ltd.
- Jonathan Chaffer and Karl Swedberg
2009 Learning jQuery 1.3
Olton Birmingham, UK: Packt Publishing Ltd.
- Matthew A. Russell
2008 Dojo: The Definitive Guide
Sebastopol (CA) USA: Tim O'REILLY
- Aaron Newton
2008 MooTools Essentials
EEUU, New York: Apress
- Propel
2011 Propel
Consulta: 28 de agosto de 2011
<<http://www.propelorm.org/>>
- Doctrine
2011 Doctrine
Consulta: 28 de agosto de 2011
<<http://www.doctrine-project.org/>>