

Universidad de las Ciencias Informáticas
Facultad 10



**Sistema Automatizado para la Selección de
Hardware Compatible con Software Libre**

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

Autor:

Jorge Américo Avila Cruz

Tutores:

Ing. Yaima Oval Riveron

Ing. Carlos Rafael Galán Cabello

Ciudad de la Habana, Mayo 2010

“Año 52 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 29 días del mes de Mayo del año 2010.

Ing. Carlos Rafael Galán Cabello

Ing. Yaima Oval Riveron

Firma del Tutor

Firma del Tutor

Jorge Américo Avila Cruz

Firma del Autor

***No vayas por donde te lleve el camino, ve por
donde no haya camino y deja un sendero.***

Ralph Waldo Emerson

RESUMEN

En Cuba, el proceso de importación de Hardware (entiéndase: sistemas completos, periféricos o partes a partir de ahora) tiene características especiales, debido a la persecución constante a que está sometido el país a través del embargo económico y tecnológico impuesto por E.U.; por este motivo el país está imposibilitado de adquirir el Hardware directamente de los fabricantes, teniendo que acudir a distribuidores que muchas veces no pueden satisfacer las demandas o sólo lo pueden hacer con Hardware de distintos fabricantes, lo que hace que el parque tecnológico de equipos informáticos del país sea muy heterogéneo. El presente trabajo de diploma tiene como objetivo principal desarrollar un sistema, que gestione eficientemente la información relacionada con la compatibilidad entre el Hardware y el Software Libre (SWL), con el fin de erradicar los problemas de incompatibilidades entre ambos, respaldar el proceso de informatización de la sociedad cubana, contribuir al proceso de migración hacia tecnologías libres, proporcionar un medio para que empresas, centros educacionales y de salud, así como usuarios particulares, encuentren toda la información técnica necesaria para adquirir equipos informáticos compatibles y potenciar la colaboración y el desarrollo del SWL a través del reporte de las experiencias y los conocimientos adquiridos por los usuarios antes mencionados.

Palabras Clave: Hardware, Software Libre, compatibilidad, informatización, migración, colaboración.

ÍNDICE GENERAL

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
1.1 ANÁLISIS DE SISTEMAS SIMILARES	5
1.1.1 <i>Ubuntu y Gentoo</i>	5
1.1.2 <i>Debian</i>	5
1.1.3 <i>Red Hat</i>	6
1.1.4 <i>OpenSUSE</i>	6
1.2 OTROS SISTEMAS DE INTERÉS	7
1.2.1 <i>Linux-drivers</i>	7
1.2.2 <i>Linux-laptop</i>	7
1.2.3 <i>Openprinting</i>	8
1.2.4 <i>Hardware4linux</i>	8
1.2.5 <i>Leenooks</i>	8
1.3 TENDENCIAS ACTUALES RELACIONADAS AL HARDWARE	9
1.4 CMS.....	11
<i>Joomla</i>	13
<i>Plone</i>	14
<i>Drupal</i>	14
1.4.1 <i>Fundamentación del CMS escogido</i>	15
1.5 METODOLOGÍAS DE DESARROLLO DE SOFTWARE	16
1.5.1 <i>Rational Unified Process (RUP)</i>	16
1.5.2 <i>Programación Extrema (eXtreme Programming o XP)</i>	18
1.5.3 <i>SCRUM</i>	21
1.5.4 <i>Fundamentación de la Metodología seleccionada</i>	22
1.6 LENGUAJES DE MODELADO	23
1.6.1 <i>Lenguaje Unificado de Modelado (Unified Modeling Language o UML)</i>	23
1.6.2 <i>Notación para el Modelado de Procesos del Negocio (BPMN)</i>	24
1.6.3 <i>Fundamentación del Lenguaje seleccionado</i>	25
1.7 HERRAMIENTAS CASE (COMPUTER-AIDED SOFTWARE ENGINEERING).....	25
1.7.1 <i>Visual Paradigm</i>	25
1.7.2 <i>Rational Rose</i>	26
1.7.3 <i>Fundamentación de la Herramienta seleccionada</i>	26
1.8 LENGUAJES DE PROGRAMACIÓN WEB	27
1.8.1 <i>Lenguajes de programación del lado del servidor</i>	27
<i>PHP (Hypertext Pre-procesor)</i>	27
<i>ASP (Active Server Pages)</i>	28
<i>JSP (Java Server Pages)</i>	28
1.8.2 <i>Lenguajes de programación del lado del cliente</i>	29
<i>HTML (HyperText Markup Language)</i>	29
<i>CSS (Cascading Style Sheets)</i>	30
<i>JavaScript</i>	30
1.8.3 <i>Fundamentación de los Lenguajes seleccionados</i>	31
1.9 ENTORNO DE DESARROLLO INTEGRADO (INTEGRATED DEVELOPMENT ENVIROMENT O IDE).....	31
1.9.1 <i>NetBeans IDE 6.8</i>	32
1.10 SISTEMAS GESTORES DE BASES DE DATOS	32
1.10.1 <i>Oracle</i>	33
1.10.2 <i>MySQL</i>	33
1.10.3 <i>PostgreSQL</i>	34
1.10.4 <i>Fundamentación del SGBD seleccionado</i>	35

1.11 SERVIDOR WEB	35
1.11.1 Apache.....	36
1.11.2 Internet Information Server (IIS).....	37
1.11.3 Fundamentación del Servidor Web seleccionado	37
CAPÍTULO 2: EXPLORACIÓN Y PLANIFICACIÓN.....	38
2.1 FASE DE EXPLORACIÓN	38
2.2 FASE DE PLANIFICACIÓN	46
2.2.1 Estimación de esfuerzo por HU.....	47
2.3 PLAN DE ITERACIONES.....	47
2.4 PLAN DE DURACIÓN DE LAS ITERACIONES	48
2.5 PLAN DE ENTREGAS.....	49
CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS.....	50
3.1 FASE DE IMPLEMENTACIÓN	50
3.2 FASE DE PRUEBAS	56
3.2.1 Pruebas de Aceptación.....	56
CONCLUSIONES.....	64
RECOMENDACIONES.....	65
REFERENCIAS BIBLIOGRÁFICAS.....	66
BIBLIOGRAFÍA.....	67
GLOSARIO DE TÉRMINOS Y DEFINICIONES	72
ANEXOS	74

ÍNDICE DE FIGURAS

FIGURA 1: RUP: FASES Y FLUJOS DE TRABAJO.....	17
FIGURA 2: XP.....	19
FIGURA 3: SCRUM.....	21
FIGURA 4: DIAGRAMA DE DESPLIEGUE	74
FIGURA 5: DIAGRAMA DE CLASES PERSISTENTES	74

ÍNDICE DE TABLAS

TABLA 1: HU1 CREAR USUARIO.....	38
TABLA 2: HU2 AUTENTICAR USUARIO	39
TABLA 3: HU3 MODIFICAR USUARIO	39
TABLA 4: HU4 ELIMINAR USUARIO	40
TABLA 5: HU5 CREAR REPORTE	40
TABLA 6: HU6 ACTUALIZAR REPORTE.....	41
TABLA 7: HU7 LISTAR REPORTE	41
TABLA 8: HU8 ELIMINAR REPORTE	42
TABLA 9: HU9 CREAR ARTÍCULO	43
TABLA 10: HU10 MODIFICAR ARTÍCULO.....	43
TABLA 11: HU11 LISTAR ARTÍCULOS	44
TABLA 12: HU12 ELIMINAR ARTÍCULO	44
TABLA 13: HU13 ADICIONAR HARDWARE	45
TABLA 14: HU14 ACTUALIZAR HARDWARE	45

TABLA 15: HU 15 LISTAR HARDWARE	46
TABLA 16: HU16 ELIMINAR HARDWARE	46
TABLA 17: ORGANIZACIÓN POR MÓDULOS Y ESTIMACIÓN DE ESFUERZO POR HU	47
TABLA 18: PLAN DE DURACIÓN DE LAS ITERACIONES	49
TABLA 19: PLAN DE ENTREGAS	49
TABLA 20: TAREA 1 CREAR USUARIO	50
TABLA 21: TAREA 2 AUTENTICAR USUARIO	51
TABLA 22: TAREA 3 MODIFICAR USUARIO	51
TABLA 23: TAREA 4 ELIMINAR USUARIO	51
TABLA 24: TAREA 5 CREAR REPORTE	52
TABLA 25: TAREA 6 ACTUALIZAR REPORTE.....	52
TABLA 26: TAREA 7 LISTAR REPORTES	52
TABLA 27: TAREA 8 ELIMINAR REPORTE	53
TABLA 28: TAREA 9 CREAR ARTÍCULO	53
TABLA 29: TAREA 10 MODIFICAR ARTÍCULO.....	53
TABLA 30: TAREA 11 LISTAR ARTÍCULOS	54
TABLA 31: TAREA 12 ELIMINAR ARTÍCULO	54
TABLA 32: TAREA 13 ADICIONAR HARDWARE	54
TABLA 33: TAREA 14 ACTUALIZAR HARDWARE	55
TABLA 34: TAREA 15 LISTAR HARDWARE.....	55
TABLA 35: TAREA 16 ELIMINAR HARDWARE	56
TABLA 36: CP CREAR USUARIO	57
TABLA 37: CP AUTENTICAR USUARIO	57
TABLA 38: CP MODIFICAR USUARIO	58
TABLA 39: CP ELIMINAR USUARIO	58
TABLA 40: CP CREAR REPORTE	59
TABLA 41: CP ACTUALIZAR REPORTE	59
TABLA 42: CP LISTAR REPORTES	60
TABLA 43: CP ELIMINAR REPORTE	60
TABLA 44: CP CREAR ARTÍCULO	60
TABLA 45: CP MODIFICAR ARTÍCULO	61
TABLA 46: CP LISTAR ARTÍCULOS.....	61
TABLA 47: CP ELIMINAR ARTÍCULO.....	62
TABLA 48: CP ADICIONAR HARDWARE.....	62
TABLA 49: CP ACTUALIZAR HARDWARE.....	63
TABLA 50: CP LISTAR HARDWARE	63
TABLA 51: CP ELIMINAR HARDWARE	63

INTRODUCCIÓN

El mundo actual está enfrentando un acelerado proceso científico y tecnológico desde hace varias décadas, donde las Tecnologías de la Información y las Comunicaciones (TIC) están desempeñando un papel fundamental, que traspasa todas las esferas de la sociedad a nivel global de forma irreversible. Ya no es posible concebir la vida en el planeta separada de la tecnología.

En medio de este fenómeno sociocultural emergen una serie de fabricantes privados de hardware de todo tipo que explotan estas características de la nueva sociedad de la información y el consumo hasta niveles exorbitantes, pues poseen los derechos de explotación de estas tecnologías. Como consecuencia crean una dependencia muy fuerte con países que se vuelven cada vez más vulnerables y dependientes a sus políticas comerciales, debido fundamentalmente a sus impulsos consumistas y a la privatización del derecho de autor.

Como alternativa al software propietario imprescindible para operar estos tipos de hardware, surge hace ya algunos años el Movimiento del Software Libre, encabezado por Richard Stallman, que en 1983 fundó el proyecto GNU para dar libertad a los usuarios de reemplazar el software propietario por Software Libre, defendiendo sus libertades para la libre cooperación y promoviendo un marco de trabajo colaborativo para el desarrollo del software.

Sería absurdo negar las posibilidades que brindan estas tecnologías y las ventajas que ofrecen para los pueblos con menos recursos. Cuba como país socialista y soberano en su lucha constante por liberarse de estas situaciones de monopolio, está inmersa en un proceso de informatización de su sociedad para llevar las TIC a todos los rincones de su territorio y ponerlas a disposición de sus ciudadanos. Aparejado a este proceso de informatización se está ejecutando un proceso de migración hacia tecnologías libres que como premisa fundamental se propone alcanzar una independencia tecnológica que respalde el desarrollo sustentable y el libre acceso al conocimiento de la sociedad cubana.

Por motivo de este proceso de informatización, la cantidad y los volúmenes de importaciones de hardware hacia el país han aumentado considerablemente, sobre todo hacia el sector académico y el empresarial, pero como consecuencia del injusto

embargo económico y tecnológico impuesto a Cuba por Estados Unidos, el país está imposibilitado de adquirir este hardware directamente de los fabricantes, por lo que tiene que acudir a distribuidores que muchas veces no poseen las cantidades requeridas o sólo pueden cubrir las demandas con hardware de diferentes fabricantes.

Un ejemplo concreto de esto es, la Universidad de las Ciencias Informáticas (UCI), un centro de nuevo tipo que vincula docencia e investigación a la producción y que aunque es la universidad más joven del país posee grandes logros en el campo del desarrollo de software y la investigación y aplicación del SWL, dedicando toda una facultad -la denominada Facultad 10- al estudio de las tecnologías libres. La UCI actualmente dispone de una cantidad que sobrepasa las 5000 computadoras y una amplia variedad de accesorios informáticos, los cuales provienen de una gran diversidad de fabricantes.

Todo esto resulta en un proceso de migración que se ve frenado en muchos casos porque el hardware no es multiplataforma o los drivers que usa para su desempeño están bajo una licencia privativa, lo que provoca gran cantidad de incompatibilidades que atentan contra la popularización y el desarrollo del SWL. Creando, además de la resistencia al cambio que conlleva todo proceso de migración, un rechazo adicional por parte de los usuarios que no pueden disponer a plenitud de todas las capacidades de sus equipos.

Actualmente el proceso de selección del hardware que se desea importar al país es poco eficiente, debido a la inexistencia de una fuente de información concreta y propia sobre la compatibilidad de hardware con software libre. Esta selección se realiza de forma empírica y manual, lo que provoca que se emplee un tiempo excesivo buscando información en Internet, que muchas veces resulta poco confiable. Además, es de especial interés encontrar una solución a estos problemas con el fin de evitar los gastos innecesarios en equipos que no resulten ser los más óptimos.

Dada la situación problemática anteriormente planteada el **problema científico** es: ¿Cómo automatizar los procesos de gestión de la información relacionada con la compatibilidad del Hardware y el Software Libre?

En la presente investigación se tiene como **Objeto de estudio** los sistemas automatizados sobre compatibilidad de Hardware con SWL, siendo el **Campo de**

Acción los sistemas automatizados para gestionar información sobre compatibilidad de Hardware con Software Libre.

Para el desarrollo de la investigación se trazó como **Objetivo General** Desarrollar un sistema automatizado que facilite la información necesaria sobre la compatibilidad de Hardware con SWL. Del mismo, se pueden derivar los siguientes **Objetivos Específicos**:

- ❖ Sistematizar los referentes teóricos del tema.
- ❖ Determinar y describir las funcionalidades necesarias para elaborar un sistema que favorezca la gestión eficiente de la información.
- ❖ Desarrollar un sistema que permita gestionar una lista de compatibilidad de Hardware con SWL.

Se parte de la siguiente **Idea a Defender**:

Se logrará realizar una correcta selección de Hardware compatible con una distribución de SWL si se desarrolla un sistema automatizado que gestione eficientemente la información actualizada.

Para dar cumplimiento a los objetivos de la investigación se definieron una serie de **Tareas a desarrollar**, tales como:

1. Análisis de sistemas similares a nivel mundial.
2. Análisis y selección de las herramientas necesarias para el desarrollo de la aplicación.
3. Análisis de las Metodologías de Desarrollo de software existentes.
4. Definición de los requisitos funcionales del sistema.
5. Realización de las actividades de implementación.
6. Realización de pruebas funcionales al sistema.

Para el desarrollo de esta investigación se usaron métodos empíricos y teóricos. A continuación se definirá cada uno de estos:

Entre los **Métodos empíricos** se encuentra:

- ❖ **Observación**: se hizo efectiva durante el estudio de los diferentes sistemas identificados, lo que permitió determinar las principales funcionalidades con las que debe contar el sistema que se desea desarrollar y las posibles mejoras a realizarle.

Entre los **Métodos teóricos** se encuentran:

- ❖ **Histórico-Lógico:** este método fue de gran utilidad para conocer el proceso de desarrollo y la interacción del Hardware y el Software Libre desde su surgimiento hasta las tendencias actuales.
- ❖ **Analítico-Sintético:** al analizar y sistematizar la información consultada durante el desarrollo de la investigación para definir ideas concretas que se apliquen a la propuesta de solución.

El presente trabajo de diploma consta de 3 capítulos bien estructurados y de forma organizada para un buen entendimiento del lector.

El primer capítulo presenta un análisis crítico de algunos de los sistemas identificados en Internet que poseen características similares o que pretenden desempeñar una labor análoga al sistema que se desea desarrollar, con el fin de identificar funcionalidades que sean convenientes implementar, así como mejoras potenciales que se les puedan agregar a la solución propuesta. También tiene como objetivo seleccionar las técnicas, tecnologías y herramientas necesarias para desarrollar el sistema que se desea realizar, haciendo un estudio de las más usadas a nivel mundial y enfatizando en las que son libres, que se adapten a las necesidades y características específicas del sistema en cuestión.

El segundo capítulo presenta la descripción de las Historias de Usuario identificadas, que no son más que la forma en que se definen los requisitos funcionales y no funcionales, que debe satisfacer el sistema a desarrollar, en la metodología XP. Una explicación detallada del proceso de planificación, donde se utilizaron estas Historias de Usuario para realizar el análisis y la estimación del tiempo de desarrollo de cada una, a través de un proceso iterativo que se ajusta en todo momento a la velocidad de trabajo del equipo de desarrollo.

El tercer capítulo presenta la descripción de las Tareas de Implementación y las Pruebas de Aceptación aplicadas al sistema, realizadas durante la Fase de Iteraciones del proyecto. Cada iteración abarcó un ciclo completo de desarrollo (diseño, implementación y pruebas), generando un entregable funcional del software, que implementaba las Historias de Usuario pertenecientes a cada iteración y mostrando la medida del avance del proyecto, puesto que una iteración terminada sin errores es una medida real del avance.

CAPÍTULO FUNDAMENTACIÓN TEÓRICA

1.1 Análisis de Sistemas Similares

Actualmente existen diversas distribuciones de GNU/Linux que son utilizadas en mayor o menor medida a nivel mundial, generalmente cada una de estas posee una Lista de Compatibilidad de Hardware o Hardware Compatibility List (HCL por sus siglas en inglés) que recoge en alguna medida la información relacionada con el nivel de soporte que brindan algunas personas, grupos o los propios fabricantes en relación al desempeño del Hardware ejecutando SWL.

1.1.1 Ubuntu y Gentoo

La Lista de Compatibilidad de Hardware de las distribuciones Ubuntu y Gentoo es uno de los buenos ejemplos que existen en la red. Posee una sencilla arquitectura que permite una navegación bastante rápida y eficaz. Brinda la posibilidad de escribir revisiones a un hardware en específico lo que facilita que haya una retroalimentación proveniente de las experiencias particulares de los usuarios de estos sistemas operativos.

Consta de una oportunidad de búsqueda por varios criterios (categorías, fabricante, sistema operativo) que facilita mucho el trabajo con la gran cantidad de información que posee y optimiza los tiempos de búsqueda disminuyendo la congestión en la BD. En sentido general el sistema posee muchas características que son de interés por su posible aplicación al sistema que se desea desarrollar, aunque tiene como deficiencia que toda la valiosa información que posee está en inglés por lo que constituye un obstáculo a la hora de acceder a esta.

1.1.2 Debian

La Lista de Compatibilidad de Hardware de la distribución Debian posee una funcionalidad muy interesante: realiza un test online del funcionamiento de los

dispositivos PCI¹ del ordenador, ejecutando el comando "lspci -n" y copiando el resultado en un campo que se muestra en pantalla, aunque no chequea controladores de X, ISA, USB, entre otros.

Soporta varios lenguajes, entre ellos el español. Posee gran cantidad de información, organizada por fabricantes pero con una arquitectura que no favorece el acceso a la misma. Este sitio constituye el esfuerzo particular de un desarrollador japonés llamado Kenshi Muto por lo que su actualización es lenta y carece de otras funcionalidades que pueden brindarle mayor valor al sistema. El diseño visual está un poco descuidado por lo que no brinda mucha ayuda para la solución que se piensa construir.

1.1.3 Red Hat

Este es un sistema totalmente comercial que brinda una lista de hardware certificado bajo el uso de esta distribución, por lo que es bastante cómodo comprobar si el hardware que se desea adquirir está incluido en esta lista, siempre y cuando se esté utilizando esta distribución.

Este es un sitio más orientado al comercio electrónico que a la promoción del hardware compatible con SWL, por lo tanto carece de funcionalidades que permitan a los usuarios intercambiar experiencias relacionadas al funcionamiento del hardware y este sistema operativo, sin embargo la información es muy confiable por el alto grado de profesionalismo y legalidad en que se ampara el sistema. Además, toda su información está en inglés, lo que dificulta la comprensión de algunos aspectos técnicos.

1.1.4 OpenSUSE

Es un sistema muy interesante por sus características, aunque recoge solamente la información relacionada con esta distribución. Es colaborativo, lo que permite una retroalimentación entre sus usuarios y que compartan experiencias y posibles soluciones a problemas que hayan presentado con el hardware anteriormente.

Soporta varios lenguajes, entre ellos el español por lo que es muy útil para facilitar el acceso a su información. El contenido está muy bien estructurado y se puede acceder a la información que se desee consultar, ya sea sobre componentes específicos o

¹ **Peripheral Component Interconnect** ("Interconexión de Componentes Periféricos") consiste en un bus de ordenador estándar para conectar dispositivos periféricos directamente a su placa base.

sistemas completos. Además, provee guías para la configuración y el uso del hardware usando este sistema operativo, muy útiles para sus usuarios.

1.2 Otros sistemas de interés

Existen otros sistemas similares a los anteriormente analizados que aunque no son una HCL como tal, de una distribución específica, sí se dedican a registrar compatibilidad de hardware con SWL, ya sea especializándose en periféricos o dispositivos de la PC, además poseen una serie de características muy interesantes que pueden servir como punto de partida para la aplicación que se desea construir.

1.2.1 Linux-drivers

Este portal informativo aunque no maneja una HCL específicamente, posee una base de datos muy amplia dividida por categorías, que brindan una descripción además del link donde se puede hallar la información precisa sobre disímiles tipos de hardware, dígame Notebooks, PDAs, teléfonos móviles, cámaras digitales, webcams, scanner e impresoras, y una amplia referencia a sitios donde se puede encontrar información relacionada con los drivers necesarios para el correcto funcionamiento de un hardware, cosa prácticamente atípica en relación a los otros sistemas analizados.

La mayor restricción que posee este portal informativo es que su información está en inglés y otros idiomas extranjeros que dificultan el acceso a la misma. Además, arquitectónicamente no está bien estructurado.

1.2.2 Linux-laptop

Este sistema posee una amplia base de datos organizada en una gran cantidad de fabricantes, que al seleccionar uno de ellos brinda una lista de los distintos modelos de laptops del fabricante y una descripción de las configuraciones específicas realizadas a las distribuciones de Linux que han reportado ese hardware.

Estos reportes están hechos en idiomas extranjeros mayormente, por lo que se dificulta su aprovechamiento. Hay que señalar que el sistema posee una estructura que dificulta el acceso a la información y que el diseño visual del sistema está realmente pobre, dando la impresión al usuario de que es un sitio poco profesional.

1.2.3 Openprinting

Este sitio es una iniciativa de The Linux Foundation para ayudar a los usuarios que usan sistemas operativos libres y que buscan información relacionada con las configuraciones y los drivers necesarios para el correcto funcionamiento de estos equipos.

Permite también que los usuarios colaboren a través de sus experiencias, reportando los problemas que han tenido con el hardware que usan y las soluciones (si las tienen) que le han dado al problema. Además organizan proyectos colaborativos de desarrollo de drivers y otras soluciones para el correcto desempeño de estos equipos, todos con el ánimo de compartir el conocimiento y los resultados de forma libre. La información almacenada en su base de datos está en inglés por lo que dificulta en alguna medida su acceso y comprensión.

1.2.4 Hardware4linux

Este sitio es una iniciativa particular, está implementado en el framework² Django, con un diseño sencillo que permite acceder a sus distintas páginas de forma rápida. Su objetivo es proporcionar la información relacionada con la compatibilidad de hardware con SWL así como permitir a los usuarios de cualquiera de las distribuciones reportar sus experiencias con el hardware en general. Ofrece un software de forma libre que se instala en la máquina y que ayuda a realizar reportes del hardware donde se está ejecutando, a través de un sistema de rating establecido y explicado por el sitio.

Posee gran cantidad de información en su base de datos aunque hace ya algunos meses que no se ha actualizado. Carece de un sistema de búsqueda por distintos criterios lo que dificulta encontrar la información deseada. Toda la información, tanto del sitio como de la base de datos está en inglés lo que dificulta en alguna medida el acceso a la misma. En general el sitio es un buen ejemplo y puede servir como punto de partida para desarrollar la aplicación que se desea implementar.

1.2.5 Leenooks

Este sitio es otro de los ejemplos interesantes que existen en Internet por ser totalmente colaborativo, es una iniciativa particular también, implementado en un framework para proyectos colaborativos. Se dedica a catalogar las incompatibilidades

² Es una estructura conceptual y tecnológica de soporte definida, normalmente con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado.

de hardware de todo tipo que no funcionan correctamente con SWL o que requieren drivers propietarios para su desempeño.

Está diseñado de forma sencilla, dando la impresión de ser poco profesional y poco confiable, pero el diseño no tiene nada que ver con la información que tiene almacenada ni la forma de manejarla, pues posee una extensa base de datos organizada por tipos de hardware que facilita en alguna medida el acceso a la misma, sería muy útil si posibilitara una búsqueda por diferentes criterios. Posee también una amplia variedad de referencias a otros sitios y HCL que sirven de apoyo si la información que se desea constatar no existe en la base de datos. Toda la información está en inglés por lo que se dificulta el acceso a la misma.

1.3 Tendencias actuales relacionadas al hardware

Actualmente el tema del hardware ha suscitado una polémica muy grande a nivel mundial, debido al auge y la aceptación que está teniendo el SWL en los usuarios, no sólo independientes sino también en compañías que lo adoptan como plataforma de desarrollo. Cada día son más las personas y las instituciones que se suman al desarrollo de SWL y como consecuencia comenzaron a darse situaciones que refutaban la opinión generalizada de que el hardware funcionaba perfectamente tanto en el software propietario como en el libre.

Mucha de esta responsabilidad la tienen un grupo de compañías privadas que no les interesa para nada la libertad del software y que buscaron la forma de controlar esto mediante la fabricación de hardware con mecanismos restrictivos para que no realicen algunos tipos de acciones o no permitan que el software varíe ni evolucione, por lo tanto que no permita el desempeño correcto del SWL.

Los casos de Sony, Microsoft, Atari y BBC son ejemplos de compañías que defienden y aplican el Digital Right Management (DRM por sus siglas en inglés), tanto en el hardware como a los contenidos digitales que crean y lo más penoso y preocupante es que estos mecanismos de restricción están siendo incluidos en todo tipo de dispositivos digitales sin informar a quienes los adquieren, bajo el nombre de "Trusted Computing" o computación confiable como engañosamente se le quiere hacer ver a las personas. Se pueden encontrar en computadoras, teléfonos, televisores, radios, juguetes, fotocopiadoras, impresoras y muchos más.

La idea técnica detrás de la “computación traidora” como la han denominado los defensores del SWL es que la PC o cualquier equipo electrónico incluya un dispositivo de cifrado y firma digital, y las claves se mantengan secretas. Los programas privativos usan este dispositivo para controlar qué otros programas pueden ejecutarse, a qué documentos o datos se puede acceder y a qué programas se los puede transferir. Esos programas continuamente descargarán nuevas reglas de autorización a través de Internet, e impondrán dichas reglas automáticamente a su trabajo. Si no permite al ordenador obtener las nuevas reglas periódicamente de Internet, algunas capacidades dejarán automáticamente de funcionar.

Como la mayoría del hardware, la computación traidora puede usarse para propósitos que no son dañinos. Pero estos usos pueden implementarse mediante otros métodos, sin hardware de computación traidora. La principal diferencia que hace a la computación traidora tan dañina para los usuarios es la sucia consecuencia: adaptando a su ordenador a trabajar contra usted. (Stallman, 2007)

Otro caso que hay que señalar es el que ha sido denominado por la Free Software Foundation (FSF por sus siglas en inglés) como “tivoización” del hardware, que no es más que la utilización de un sistema operativo libre en hardware restrictivo, o sea que cualquier actualización o modificación que se le realice al software el hardware la rechazará dejándola inoperativa, por lo tanto se arruina el propósito de la licencia General Public License (GPL por sus siglas en inglés) de mantener las libertades del software.

Esto sucedió en el año 2006 cuando la compañía TiVo con su tecnología del mismo nombre, que captura y almacena la programación de la televisión en un disco duro para su posterior reproducción a placer del usuario, comenzó a usar GNU/Linux y software libre bajo la licencia GPLv2.0 pero repentinamente hizo una jugada que tomó por sorpresa al propio Richard Stallman, utilizó un hardware restrictivo que impide que los usuarios puedan modificar el código fuente e instalar sus propias aplicaciones, de ahí surge el fenómeno de la “tivoización” del hardware, que explota una brecha legal en la licencia que aplica y que fue corregida con la GPLv3.0 presentada en el 2007.

Afortunadamente estas situaciones están ocurriendo cada vez menos, pues muchas de las grandes compañías fabricantes de hardware como Sun Microsystem, Hp, IBM, Dell y otras, están comenzando a certificar sus equipos para que funcionen

perfectamente con distribuciones de GNU/Linux. Se impulsan iniciativas como la auspiciada por la compañía Novell que agrupó alrededor de 200 desarrolladores con el objetivo de mejorar el soporte para este sistema operativo, ofreciendo a las compañías fabricantes de hardware interesadas, facilidades para desarrollar drivers que se incluyen en el kernel³ de sus distribuciones y que posteriormente si lo desean reciben capacitación para mantenerlos por sí mismos.

1.4 CMS

Los CMS (*Content Management System*) son software que se utilizan principalmente para facilitar la gestión de aplicaciones Web, ya sea en Internet o en una Intranet, y por eso también son conocidos como Gestores de Contenido Web (*Web Content Management* o WCM).

Un Sistema de Gestión de Contenido facilita la gestión de contenidos en todos sus aspectos: creación, mantenimiento, publicación y presentación, son sistemas muy utilizados actualmente en el mundo del desarrollo web, que permiten crear y mantener un sitio web con facilidad, encargándose de los trabajos más tediosos que hasta ahora ocupaban el tiempo de los administradores de las webs, gracias a la gran cantidad de funcionalidades que proveen para el desarrollo de aplicaciones de cualquier tipo que manejen contenido dinámico, y sobre todo de una forma relativamente fácil. En cualquier entorno virtual ésta es una característica muy importante.

Su gran diferencia radica en separar el contenido actual de un sitio web y almacenarlo en una base de datos, para poder ser administrado posteriormente de manera ágil y sencilla. Permite manipular textos, diseño, publicación de contenidos privados, enlaces, menús y mucho más. Un sistema de administración de contenido web reduce costos, distribución, mantenimiento de contenido y ayuda a superar barreras potenciales de la comunicación web.

Algunas de las ventajas que proporcionan son:

- **Inclusión de nuevas funcionalidades en el web.** Esta operación puede implicar la revisión de multitud de páginas y la generación del código que aporta las funcionalidades. Con un CMS eso puede ser tan simple como incluir un módulo realizado por terceros, sin que eso suponga muchos cambios en la

³ Es un software que actúa de sistema operativo, en este caso el núcleo de Linux.

web. El sistema puede crecer y adaptarse a las necesidades futuras.

- **Mantenimiento de gran cantidad de páginas.** En una web con muchas páginas hace falta un sistema para distribuir los trabajos de creación, edición y mantenimiento con permisos de acceso a las diferentes áreas. También se tienen que gestionar los metadatos de cada documento, las versiones, la publicación y caducidad de páginas y los enlaces rotos, entre otros aspectos.
- **Reutilización de objetos o componentes.** Un CMS permite la recuperación y reutilización de páginas, documentos, y en general de cualquier objeto publicado o almacenado.
- **Páginas interactivas.** Las páginas estáticas llegan al usuario exactamente como están almacenadas en el servidor web. En cambio, las páginas dinámicas no existen en el servidor tal como se reciben en los navegadores, sino que se generan según las peticiones de los usuarios. De esta manera cuando por ejemplo se utiliza un buscador, el sistema genera una página con los resultados que no existían antes de la petición. Para conseguir esta interacción, los CMS conectan con una base de datos que hace de repositorio central de todos los datos de la web.
- **Cambios del aspecto de la web.** Si no hay una buena separación entre contenido y presentación, un cambio de diseño puede comportar la revisión de muchas páginas para su adaptación. Los CMS facilitan los cambios con la utilización, por ejemplo, del estándar Cascading Style Sheets (CSS u hojas de estilo en cascada) con lo que se consigue la independencia de presentación y contenido.
- **Consistencia de la web.** La consistencia en un web no quiere decir que todas las páginas sean iguales, sino que hay un orden (visual) en vez de caos. Un usuario nota enseguida cuándo una página no es igual que el resto de las de la misma web por su aspecto, la disposición de los objetos o por los cambios en la forma de navegar. Estas diferencias provocan sensación de desorden y dan a entender que el web no lo han diseñado profesionales. Los CMS pueden aplicar un mismo estilo en todas las páginas con el mencionado CSS, y aplicar una misma estructura mediante patrones de páginas.

- **Control de acceso.** Controlar el acceso a una web no consiste simplemente en permitir la entrada a la web, sino que comporta gestionar los diferentes permisos a cada área de la web aplicados a grupos o individuos. (CMS Hispano, 2007)

Existe una gran variedad de CMS que son desarrollados y utilizados en la actualidad, aquí se hace un análisis de algunos de los más importantes:

Joomla

Es uno de los mejores CMS de Código Abierto. Es usado para crear desde sitios simples hasta complejas aplicaciones corporativas, es fácil de instalar aunque no se tengan conocimientos de usuario avanzado, simple de manejar y confiable. Si se requieren de funciones especializadas, estas se pueden crear fácilmente puesto que es altamente extensible y además posee miles de módulos y extensiones disponibles para su uso creados por otros desarrolladores y puestos a disposición de la comunidad.

Posee una creciente comunidad de más de 200 000 miembros entre usuarios y contribuyentes que trabajan a diario por ofrecer un software más completo y seguro, por lo que se augura un futuro próspero e inmediato, también se facilita mucho la actualización del código ya que las vulnerabilidades son rápidamente erradicadas y distribuidas. Debido a que está basado en PHP y MySQL se construyen poderosas aplicaciones en una plataforma abierta (Linux, Mac, Windows) que cualquiera puede usar, compartir y brindar soporte. Permite acceder a lo más íntimo del código fuente facilitando la personalización a través de la creación de módulos novedosos y con mucha versatilidad en los diseños alcanzados.

Usa títulos independientes, etiquetas diferenciadas y URLs⁴ amigables por cada página lo que favorece el posicionamiento de los sitios en internet. Es notable la rapidez con que carga las páginas gracias al tratamiento que da al sistema caché y el cumplimiento de los estándares del W3C⁵ debido a que el sistema de plantillas permite marcar semánticamente los documentos separando el contenido de la presentación.

⁴ **Uniform Resource Locator** es localizador uniforme de recurso y se refiere a la dirección única que identifica a una página web en Internet.

⁵ El **Consortio World Wide Web** (W3C) es un consorcio internacional donde las organizaciones miembro, personal a tiempo completo y el público en general, trabajan conjuntamente para desarrollar estándares Web.

Es libre y constituye una interesante solución cuando se desean ampliar las posibilidades y características de un sitio web.

Plone

Es un CMS que corre sobre el servidor de aplicaciones Zope distribuido bajo la licencia GNU/GPL. Plone es multiplataforma corriendo tanto en código abierto o privativo como Linux, Windows, Mac OS X, Solaris o BSD. Fácil de instalar ya que con un solo clic está operativo en minutos. Sumamente flexible al implementar estándares que facilitan la accesibilidad y la usabilidad usando lenguajes de programación abiertos a diversas plataformas.

Se está convirtiendo en el inicio de una nueva generación de sistemas de gestión de contenidos para la Web, soportando todas las tecnologías actuales, XHTML, CSS, generador de contenidos de forma amigable a los robots de los buscadores y accesibles dejando atrás en estos aspectos a otros CMS que anteriormente han sido sistemas de gestión de contenidos de código abierto muy populares. (Developers4Web, 2005)

Proporciona un sistema para la administración Web que de forma sencilla permite agregar, actualizar y almacenar contenido, generalmente es ideal para los proyectos de grupos, comunidades, sitios web, extranets e intranets, así como para la colaboración entre entidades que se encuentran físicamente distantes.

Drupal

Drupal es un sistema manejador de contenido altamente modular y muy configurable. Es un programa de código abierto, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la web, y un énfasis especial en la usabilidad y consistencia de todo el sistema. El diseño de Drupal es especialmente idóneo para construir y gestionar comunidades en Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar muchos tipos diferentes de sitios web. (Drupal Hispano, 2005)

Drupal permite publicar artículos, imágenes, u otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs y administración de usuarios y permisos. Es

un sistema dinámico, en lugar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido y todas las configuraciones son almacenados en una base de datos que se editan utilizando un entorno web. El programa dispone de innumerables características que lo hacen atractivo como gestor de contenido, de ahí la amplia comunidad de usuarios que lo utilizan y lo mantienen. Además de las funciones mencionadas, otras características interesantes son: agregador de noticias (RSS⁶) que permite leer y publicar enlaces a noticias externas y la posibilidad de que cada usuario disponga de un blog personal.

Es altamente personalizable, debido a que permite separar el contenido de la presentación de acuerdo a las preferencias del usuario a través del sistema de plantillas, facilitando el control y los cambios de apariencia del sitio web. Se puede integrar a un servidor LDAP⁷ para brindar servicios de intranet o de autenticación externa como Jabber⁸ o Blogger⁹. Incorpora una capa de abstracción de Base de Datos (BD) que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos, corre en servidores web como IIS o Apache y en sistemas operativos como Windows, Solaris, Mac OS X, Linux y BSD. Tiene un desempeño ideal para entornos de acceso heterogéneo, implementando mecanismos de cache que elimina consultas innecesarias a la BD aumentando el rendimiento y aligerando el estrés del servidor.

1.4.1 Fundamentación del CMS escogido

De los CMS analizados se escoge Drupal porque la gran cantidad de módulos disponibles que existen proveen un amplio surtido de características. Además brinda la posibilidad de asignar permisos a un rol y agrupar los usuarios por roles facilitando los procesos en la capa de negocio, seguir y auditar continuas actualizaciones del contenido y comentar dichos cambios, mostrar informes a los administradores de popularidad de los diferentes contenidos y de cómo navegan los usuarios en el sitio, y la útil posibilidad de administrar y configurar el sitio a través de un navegador sin necesidad de instalar software adicional; cosas que serán explotadas en alto grado y muy importantes para la necesaria retroalimentación y la optimización de la aplicación que se desea realizar.

⁶ **Sindicación Realmente Simple** es un formato para la sindicación de contenido de páginas web.

⁷ **Protocolo Ligero de Acceso a Directorios** es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

⁸ Protocolo abierto basado en el estándar XML para el intercambio en tiempo real de mensajes y presencia entre dos puntos en Internet.

⁹ Es un servicio para crear y publicar una bitácora en línea.

También hay que señalar la influencia que tuvo a la hora de tomar una decisión la experiencia previa que existe en la UCI y particularmente en la Facultad 10 donde existe un proyecto dedicado a la producción de portales que se basa principalmente en el estudio y aplicación de este CMS y que posee un capital humano capacitado y con la experiencia suficiente para servir de apoyo a la hora de buscar asesoramiento en el tema.

1.5 Metodologías de desarrollo de software

Una metodología no es más que un conjunto de actividades necesarias para transformar los requisitos de los usuarios en un sistema software. Las metodologías tienen como fin hacer del proceso de desarrollo de software un proceso más predecible y eficiente al imponer un proceso disciplinado sobre el desarrollo del software.

Hoy en día existen numerosas propuestas metodológicas que inciden con distintos énfasis en las diferentes etapas del proceso de desarrollo de software, por lo que se debe tener presente cuál metodología usar en dependencia de las necesidades del proyecto software que se va a realizar. A continuación se realiza un estudio de algunas de las metodologías más utilizadas dentro del mundo del desarrollo de software.

1.5.1 Rational Unified Process (RUP)

RUP es un proceso de desarrollo de software y unido a UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Está enfocado hacia el cliente. No es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización. La metodología RUP es más apropiada para proyectos grandes, dado que requiere un equipo de trabajo capaz de administrar un proceso complejo en varias etapas.

Describe como diseñar una arquitectura flexible, que se acomode a los cambios, comprensible intuitivamente y promueve una más efectiva reutilización de software. Provee un enfoque sistemático para definir una arquitectura utilizando componentes nuevos y preexistentes. Muestra como modelar software visualmente para capturar la estructura y comportamiento de arquitecturas y componentes. Las abstracciones

visuales ayudan a comunicar diferentes aspectos del software; comprender los requerimientos, ver como los elementos del sistema se relacionan entre sí, mantener la consistencia entre diseño e implementación y promover una comunicación precisa. (Tecnologías Racionales, 2007)

En la siguiente figura se muestran las fases y los flujos de trabajos de RUP y cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto.

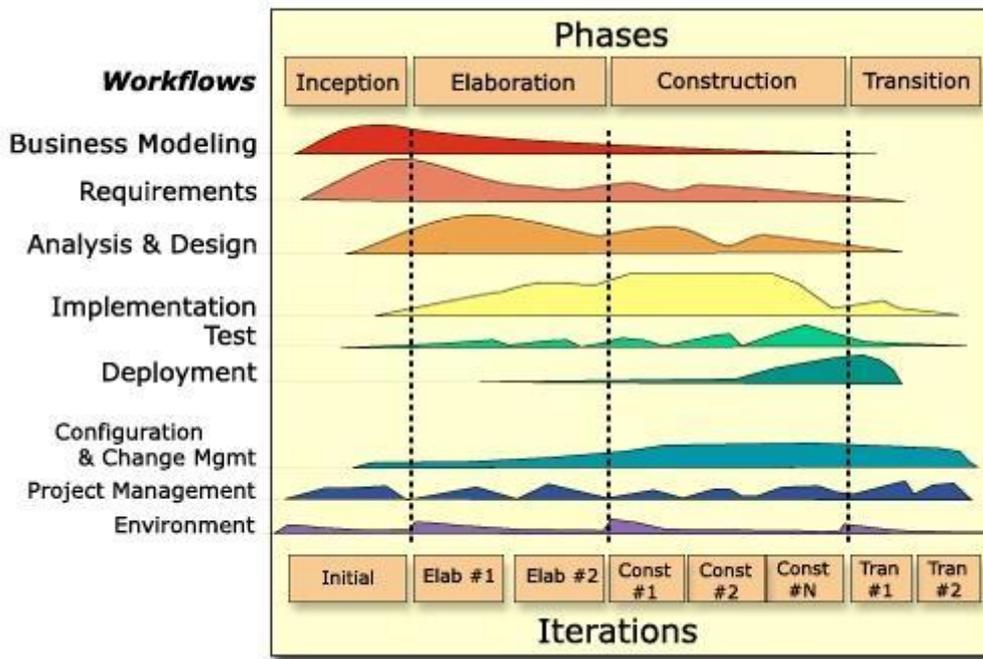


Figura 1: RUP: Fases y Flujos de trabajo

Las primeras iteraciones en las fases de *Inicio* y *Elaboración* se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una línea base de la arquitectura.

Las fases definidas por RUP y los hitos de cada fase son:

- Inicio: Se describe el negocio y se determina el alcance del proyecto.
- Elaboración: Se define la línea base de la arquitectura.
- Construcción: Se obtienen uno o varios releases¹⁰ que han pasado por pruebas y la documentación de todo el proceso.
- Transición: Se obtiene un producto final listo para su uso. Puede implicar reparación de errores.

¹⁰ Son porciones entregables de un proyecto que van siendo liberadas a medida que avanza el desarrollo.

Los flujos de trabajo definidos por RUP son:

- Modelación del Negocio: Se describen y se detallan las necesidades del negocio.
- Requerimientos: Se traducen las necesidades del negocio a un sistema automatizado.
- Análisis y Diseño: Los requerimientos se trasladan a la arquitectura de software.
- Implementación: Se comienza a crear el software ajustándose a la arquitectura para que tenga el comportamiento esperado.
- Prueba: Se verifica si el comportamiento logrado es correcto y si el desarrollo ha sido acorde a la arquitectura.
- Administración de proyecto: Se administran horarios y recursos.
- Administración de Configuración y Cambios: Se controlan las versiones del proyecto.
- Ambiente: Actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto.

1.5.2 Programación Extrema (eXtreme Programming o XP)

Es una metodología de desarrollo ágil utilizada en proyectos de corto plazo. Se basa en pruebas unitarias, que se realizan a los principales procesos y a posibles fallas que pudieran ocurrir, también en la reutilización de código para lo cual se crean patrones o modelos estándares.

Para desarrollar un proyecto usando XP, el cliente junto al equipo de desarrollo definen qué es lo que va a hacer el sistema, para esto siguen una serie de pasos, entre los que se encuentra la identificación de las “Historias de Usuario” (HU).

Las Historias de Usuarios sustituyen a los documentos de especificación funcional, y a los Casos de Uso. Estas “historias” son escritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar. La diferencia más importante entre estas historias y los tradicionales documentos de especificación funcional se encuentra en el nivel de detalle requerido. Las historias de usuario deben tener el detalle mínimo como para que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. Cuando llegue el momento de la implementación, los desarrolladores dialogarán directamente con el

cliente para obtener todos los detalles necesarios. Las historias de usuarios deben poder ser programadas en un tiempo entre una y tres semanas. Si la estimación es superior a tres semanas, debe ser dividida en dos o más historias. Si es menos de una semana, se debe combinar con otra historia. (Joskowicz, 2008)

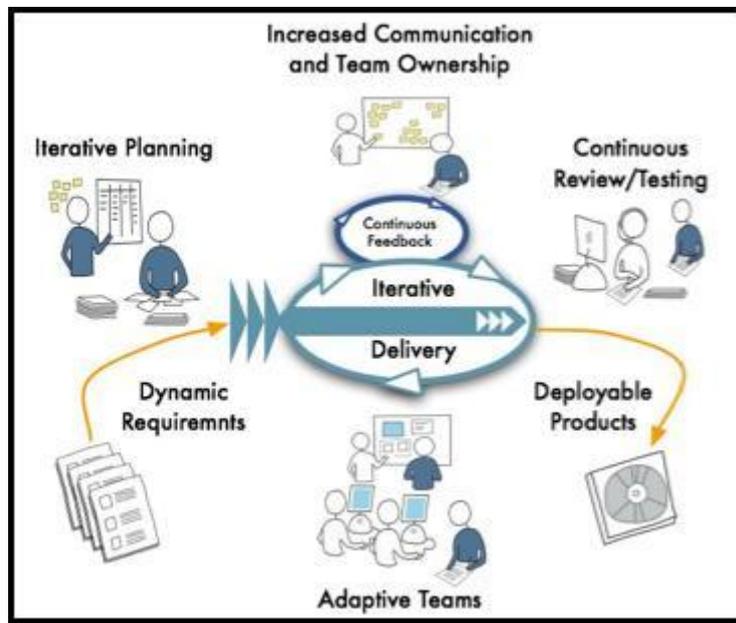


Figura 2: XP

La XP empieza con cuatro valores: Comunicación, Retroalimentación, Simplicidad y Coraje:

- Comunicación: Se nutre de la comunicación directa entre personas. Cuando dos o más personas se comunican directamente pueden no solo interpretar las ideas de la otra persona, sino que también aprecian el lenguaje corporal, miradas, etc. Diferente de lo que ocurre en una conversación mediante el correo electrónico, hay muchos factores que hacen de esta una comunicación mucho menos efectiva.
- Retroalimentación: La agilidad se define, entre otras cosas, por la capacidad de respuesta ante los cambios que se van haciendo necesarios a lo largo del camino. Por este motivo uno de los valores que nos hace más ágiles es el continuo seguimiento o retroalimentación que recibimos a la hora de desarrollar en un entorno ágil de desarrollo. Esta retroalimentación se toma del cliente, de los miembros del equipo, en general de todo el entorno en el que se mueve un equipo de desarrollo ágil.
- Simplicidad: Dado que no se puede predecir cómo va a ser el software que se está desarrollando en el futuro; un equipo de programación extrema intenta

mantener el software lo más sencillo posible. Esto quiere decir que no se va a invertir ningún esfuerzo en hacer un desarrollo que en un futuro pueda llegar a tener valor.

- Coraje: El coraje es un valor muy importante dentro de la programación extrema. Un miembro de un equipo de desarrollo extremo debe de tener el coraje de exponer sus dudas, miedos, experiencias sin "adornar" éstas de ninguna manera. Esto es muy importante ya que un equipo de desarrollo extremo se basa en la confianza entre sus miembros.

Construye sobre ellos una docena de prácticas que los proyectos XP deben seguir. Muchas de estas prácticas son técnicas antiguas, tratadas y probadas, aunque a menudo olvidadas por muchos, incluyendo la mayoría de los procesos planeados. Además de renovar estas técnicas, XP las combina en un todo correlativo dónde cada una refuerza a las demás. Construye un proceso de diseño evolutivo que se basa en refactorar un sistema simple en cada iteración. Todo el diseño se centra en la iteración actual y no se hace nada para iteraciones futuras. El resultado es un proceso de diseño disciplinado, lo que es más, combina la disciplina con la adaptabilidad de una manera que indiscutiblemente la hace la más desarrollada entre todas las metodologías ágiles.

Proporciona una serie de ventajas que se ven reflejadas en:

- Código altamente optimizado, ya que es escrito para una arquitectura concreta. Lo que, generalmente repercute en un ahorro de hardware y en ahorro de tiempo de ejecución.
- Altas posibilidades de tener éxito con el proyecto, ya que el propio cliente participa continuamente y ante cualquier cambio de última hora, se vuelve a replantear los objetivos.
- Realización de pruebas continuamente, encontrando los errores cuando se comenten, no cuando el cliente prueba el producto. Lo que repercute en las inversiones, si se cometen errores al principio, no es lo mismo tirar el trabajo de un par de días, que el de varias semanas.
- Humanización del trabajado, trata y entiende al trabajador como es debido, respetando su vida fuera del trabajo y creando un ambiente de trabajo saludable para su mejor desempeño.

1.5.3 SCRUM

Scrum es un proceso en el que se aplican de manera regular un conjunto de mejores prácticas para trabajar en equipo y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. En Scrum se realizan entregas parciales y regulares del resultado final del proyecto, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad y la productividad son fundamentales. Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto. (Proyectos Ágiles, 2007)

Un proyecto que aplique Scrum ejecuta iteraciones de 15 días a un mes de duración, donde cada iteración produce un resultado concreto entregable al cliente, debe realizarse con el mínimo esfuerzo.

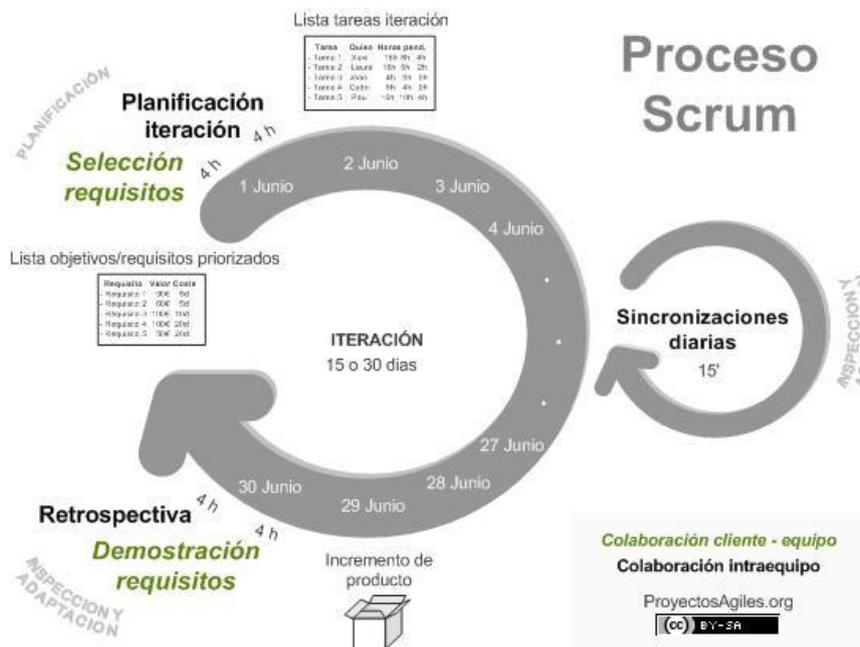


Figura 3: SCRUM

Scrum proporciona una serie de ventajas muy interesantes tanto para los desarrolladores como para los clientes, entre ellas se pueden mencionar:

- Gestión periódica de los intereses del cliente a través de la lista de requerimientos, donde se reflejan las expectativas esperadas con cada requisito y el valor que aporta. Además, luego de cada iteración se demuestra al cliente los resultados obtenidos.
- Se obtienen resultados anticipados al priorizar los requisitos más importantes y que aportan más valor al cliente, por lo que este puede comenzar a recuperar la inversión más rápido, sacar el producto (parcialmente) al mercado antes que los competidores y autofinanciarse.
- Retroalimentación continua: el cliente está en constante intercambio con el equipo de desarrollo por lo que puede replanificar los requisitos antes de comenzar cada iteración, en función de nuevas necesidades, cambios en el mercado o de la eficiencia del equipo de desarrollo.
- Gestión temprana de los riesgos, al dividir los requerimientos por iteraciones aísla los posibles errores y se pueden manejar mejor por separado, como son iteraciones cortas los errores se detectan rápidamente y pueden ser mitigados sin necesidad de mucho esfuerzo.
- Mayor productividad y calidad al optimizar el proceso de trabajo en cada iteración, al finalizar una iteración se analiza el proceso de desarrollo y se identifican las tareas que se deben priorizar, las decisiones más oportunas y mejores formas para organizar próximas iteraciones, lo que propicia la calidad del producto de forma constante y objetiva.
- Alta motivación y compromiso con el proyecto, el equipo de desarrollo es quien conoce sus limitaciones y potencialidades así que es quien decide qué requerimientos satisfacer en cada iteración y se planifica para ello, además, luego de cada iteración se puede apreciar el resultado alcanzado.

1.5.4 Fundamentación de la Metodología seleccionada

Teniendo en cuenta las características de las metodologías de desarrollo antes analizadas, se pudo apreciar que aunque se centran en la producción de software y que el proceso se establece para aumentar la calidad del producto y la eficiencia de los desarrolladores, poseen sus diferencias. Por lo que se decide usar XP como metodología de desarrollo ya que está diseñada para equipos de trabajo pequeños, en este caso de un solo desarrollador, para proyectos cortos con iteraciones igualmente cortas, que se corresponden con el proyecto de software en cuestión.

Brinda una serie de características que recuerdan mucho su forma de trabajar a la forma en que el SWL es desarrollado, sobre todo los ambientes colaborativos en que se desarrolla y la aplicación de estándares abiertos y patrones de codificación que independiza en gran medida el código de su autor; además posee una buena documentación y existe en la universidad un personal calificado con amplios conocimientos sobre esta metodología, también influyó mucho a la hora de tomar esta decisión, el corto tiempo del que se dispone para desarrollar la solución del problema y la rapidez con que se obtienen buenos resultados de la aplicación de esta metodología.

1.6 Lenguajes de Modelado

Los lenguajes de modelado constituyen una herramienta y un mecanismo para facilitar la comprensión de los complejos sistemas informáticos que se desarrollan en la actualidad. Debido a que estos modelos proporcionan un mayor nivel de abstracción, permiten trabajar con sistemas más complejos, facilitando el proceso de codificación e implementación del sistema de forma distribuida y en distintas plataformas puesto que son independientes del lenguaje de programación que se esté utilizando.

1.6.1 Lenguaje Unificado de Modelado (Unified Modeling Language o UML)

UML es el lenguaje de modelado de sistemas de software más utilizado en la actualidad para especificar, construir, visualizar y documentar artefactos de un sistema orientado a objetos. Proporciona una forma estándar de representar los planos de un sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

Brinda una notación gráfica que permite representar todas las fases de un proyecto de desarrollo: desde el análisis con los casos de uso, el diseño con los diagramas de clases y objetos, hasta la implementación con el diagrama de despliegue. Es un lenguaje independiente del lenguaje de implementación de forma tal que los modelos realizados se puedan implementar en cualquier lenguaje, principalmente los que son orientados a objetos.

Dentro de las ventajas que proporciona este lenguaje de modelado se puede destacar

las siguientes:

- Permite modelar sistemas utilizando técnicas orientadas a objetos.
- Permite especificar todas las decisiones de análisis, diseño e implementación, construyendo así modelos precisos y completos.
- Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).
- Permite documentar todos los artefactos de un proceso de desarrollo.
- Cubre las cuestiones relacionadas con el tamaño propio de los sistemas complejos y críticos.
- Existe un equilibrio entre expresividad y simplicidad, pues no es difícil de aprender ni de utilizar.
- UML es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

1.6.2 Notación para el Modelado de Procesos del Negocio (BPMN)

BPMN es un nuevo estándar para modelar flujos de procesos de negocio y servicios web, su principal objetivo es especificar una notación comprensible para todos los usuarios del negocio, desde los analistas del negocio que modelan los procesos hasta los desarrolladores técnicos responsables de implementarlos, garantizando la mejora de los mismos y finalmente a los usuarios del negocio quienes gestionarán y administrarán dichos procesos. Crea un puente estándar entre el diseño del proceso de negocio y su implementación.

La notación ha sido diseñada particularmente para coordinar la secuencia de procesos y los mensajes que fluyen entre los diferentes procesos participantes, es más expresivo que UML a la hora de especificar procesos debido a que es gráficamente más rico, con menos símbolos fundamentales pero con más variaciones de estos, lo que facilita la comprensión por parte de gente no experta.

Es necesario señalar que abarca únicamente los procesos de negocio, lo que significa que otro tipo de modelos relacionados (estructura de la organización, recursos, modelos de datos, estrategias, reglas de negocio) quedan fuera de la especificación.

BPMN es un facilitador de fondo para una nueva iniciativa en el mundo de la arquitectura empresarial, la Administración de Procesos de Negocio (BPM). Esta

permite gestionar el cambio para mejorar los procesos de negocio y unifica las disciplinas de Modelación de Procesos, Simulación, Flujo de trabajo, Integración de Aplicaciones Empresariales (EAI) y la integración Business-to-Business (B2B) en un único estándar.

1.6.3 Fundamentación del Lenguaje seleccionado

De los lenguajes de modelado antes expuestos se decide utilizar UML, fundamentalmente porque es un lenguaje estándar, fácil de aprender, que responde eficientemente ante el cambio constante de requisitos, además permite una comunicación fluida entre las partes involucradas en el desarrollo de software y ofrece una amplia variedad de diagramas para visualizar el sistema desde varias perspectivas por lo que facilita la comunicación entre clientes y desarrolladores.

1.7 Herramientas CASE (Computer-Aided Software Engineering)

CASE o Ingeniería de Sistemas Asistida por Ordenador es la aplicación de tecnología informática a las actividades, las técnicas y las metodologías propias de desarrollo, el objetivo es acelerar el proceso para automatizar o apoyar una o mas fases del ciclo de vida del desarrollo de sistemas. Estas herramientas persiguen a largo plazo proporcionar una generación automática de programas desde una especificación a nivel de diseño. (Landeros Gómez, y otros, 2006)

Esta tecnología supone la automatización del desarrollo del software, contribuyendo a mejorar la productividad y la calidad en el desarrollo de sistemas de información y aplicaciones de bases de datos. Permite el desarrollo y refinamiento visual de las aplicaciones, la aplicación práctica de metodologías estructuradas; facilita la reutilización de componentes de software, la realización de prototipos; simplifica el mantenimiento y mejora notablemente, además de que estandariza, la documentación de los procesos.

1.7.1 Visual Paradigm

Es una herramienta CASE que utiliza como lenguaje de modelado UML. Se caracteriza por su robustez, portabilidad y usabilidad principalmente, soporta el ciclo completo de desarrollo de software, desde análisis y diseño orientado a objetos hasta construcción, pruebas y despliegue. Permite proyectar todo tipo de diagramas de clases. Ofrece un diseño enfocado al negocio y centrado en los casos de usos, generando un software de mayor calidad, modelo y código sincronizados a lo largo del

ciclo de desarrollo, capacidades de ingeniería directa e inversa y un uso del lenguaje común a todo el equipo de desarrollo que facilita mucho la comunicación.

Partiendo de un modelo relacional en PostgreSQL, SQL Server o MySQL es capaz de desplegar todas las clases asociadas a las tablas, siguiendo el patrón de diseño Una Clase-Una Tabla. Soporta múltiples usuarios (ingenieros de software, analistas de sistemas, analistas de negocios, arquitectos y desarrolladores) trabajando sobre el mismo proyecto, permite control de versiones y genera la documentación automáticamente en formatos como Web o PDF. Es multiplataforma.

1.7.2 Rational Rose

Es una poderosa herramienta CASE que utiliza UML como lenguaje de modelado para el modelado visual del análisis, diseño y desarrollo de sistemas orientados a objetos. Permite especificar, analizar y diseñar el sistema antes de codificarlo y que se mantenga la consistencia de los modelos del sistema software, el chequeo de la sintaxis UML, la generación de documentación automática, generación de código a partir de los modelos, crear modelos a partir del código (ingeniería inversa) y trabajo en grupo. Cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases.

Propone la utilización de cuatro tipos de modelo para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico (vista de Casos de Uso, vista Lógica, vista de Componentes y vista de Despliegue). Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software. Es la herramienta líder en el mundo para el modelado de sistemas complejos y de tiempo real. Dispone de un navegador que muestra información de forma jerárquica de todos los elementos de los modelos de un proyecto. Ofrece varias opciones para el manejo de los diagramas en el momento de su visualización, exportación o impresión. Es privativo y no es multiplataforma.

1.7.3 Fundamentación de la Herramienta seleccionada

Se decide utilizar Visual Paradigm debido a que cuenta con un amigable entorno de desarrollo, soporta el ciclo de vida completo de desarrollo de un producto software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Este

software de modelado UML ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite elaborar todos los tipos de diagramas de clases, generar código desde diagramas y generar documentación.

La principal razón de esta selección es que es una herramienta multiplataforma, y en calidad de la migración que está llevando a cabo el país unánimemente se está adoptando como herramienta de modelado en la universidad, además de que existe un personal capacitado con la experiencia y los conocimientos necesarios para ser consultados en un momento puntual del modelado y experiencias positivas de su aplicación en los proyectos productivos de la universidad.

1.8 Lenguajes de programación Web

Específicamente los lenguajes de programación que se utilizan para aplicaciones Web se dividen en dos grupos fundamentales: lenguajes del lado del servidor y lenguajes del lado del cliente. Los lenguajes del lado del servidor se encargan de aspectos relacionados con la funcionalidad del sistema: la conexión entre la base de datos y el servidor, la lógica del negocio y la modificación del contenido que viene y va hacia el cliente. Los segundos tienen la característica de ejecutarse en el navegador del usuario y desempeñan funciones hasta cierto punto sencillas tales como los cambios de apariencia y la validación de los datos de entrada.

1.8.1 Lenguajes de programación del lado del servidor

Este tipo de lenguaje se encarga de atender las peticiones del cliente y generar páginas web dinámicas con los datos de respuesta de forma comprensible para este, son reconocidos, ejecutados e interpretados por el propio servidor. Entre los más usados en la actualidad se encuentran ASP, ASP.NET, JAVA y PHP entre otros.

PHP (Hypertext Pre-procesor)

Es un lenguaje interpretado de propósito general que está diseñado especialmente, y es ampliamente usado, para el desarrollo Web de forma dinámica. Publicado bajo la PHP License, la Free Software Foundation (FSF) considera esta licencia como software libre, por lo que se presenta como una alternativa de fácil acceso y amplias posibilidades para todos. Se utiliza embebido dentro de código HTML y para programar scripts¹¹ del lado del servidor.

¹¹ Es un conjunto de instrucciones que permiten la automatización de tareas, creando pequeñas utilidades.

Es gratuito e independiente de la plataforma, rápido, posee mucha documentación y una gran librería de funciones explicadas y ejemplificadas, permite las técnicas de programación orientada a objetos y no requiere de definición de tipo de variables. A partir de la versión 5 tiene manejo de excepciones y como principal objetivo solucionar las carencias de versiones anteriores mejorando los mecanismos de POO.

La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas. Se integra con mucha cohesión a la mayoría de los manejadores de base de datos que más se utilizan en la actualidad, destacando su conectividad con MySQL. Según datos del año 2006 se encuentra instalado en más de 20 millones de sitios Web y en un millón de servidores, dato que continúa aumentando constantemente.

ASP (Active Server Pages)

Es una tecnología creada por Microsoft para generar páginas Web dinámicamente, ha sido comercializada adjuntamente a Internet Information Server (IIS), por lo que posee un modelo tecnológico relacionado estrechamente a su fabricante. Programar en ASP es muy similar a programar en Visual Basic por lo que pretende ser una solución para un modelo de programación efectiva y rápida.

Las páginas que se ejecutan en el servidor pueden realizar diferentes conexiones de red, accesos a bases de datos, y otras tareas para crear la página final que mostrará al cliente. El cliente recibe únicamente una página con el código HTML que muestra el resultado de la ejecución de la página ASP. Como la página resultante contiene solamente código HTML, es compatible con todos los navegadores.

Es importante conocer que ASP es un sistema con nula portabilidad pues requiere necesariamente de un servidor Windows, con todas las implicaciones del alto costo derivado de las licencias restrictivas bajo las cuales es comercializada esta tecnología, poca flexibilidad y escasa seguridad que estos equipos conllevan.

JSP (Java Server Pages)

Es una tecnología orientada a crear páginas web potentes usando programación en el lenguaje Java. Permite crear aplicaciones web que se ejecuten en múltiples servidores, de diversas plataformas, debido a que en esencia es un lenguaje

multiplataforma. Las páginas JSP están compuestas de etiquetas especiales y código HTML/XML mezclado para programar scripts de servidor en sintaxis Java.

Por tanto, las JSP se pueden escribir en un editor HTML/XML común. El motor de las páginas JSP está basado en los servlets¹² de Java -programas en Java destinados a ejecutarse en el servidor-, aunque el número de desarrolladores que pueden afrontar la programación de JSP es mucho mayor, dado que resulta mucho más sencillo aprender que los servlets.

En JSP se crean páginas similarmente a como se crean en PHP o ASP. Se generan archivos con extensión .jsp que incluyen, dentro de la estructura de etiquetas HTML, las sentencias Java a ejecutar en el servidor. Antes de que sean funcionales los archivos, el motor JSP lleva a cabo una fase de traducción de esa página en un servlet. Esta fase de traducción se lleva a cabo habitualmente cuando se recibe la primera solicitud de la página .jsp, aunque existe la opción de precompilar el código para evitar ese tiempo de espera la primera vez que un cliente solicita la página.

1.8.2 Lenguajes de programación del lado del cliente

Los lenguajes del lado del cliente son aquellos interpretados directamente por un navegador y no necesitan un preprocesamiento ya que son totalmente independientes del servidor.

HTML (HyperText Markup Language)

El Lenguaje de Marcas de Hipertexto es el lenguaje de publicación estándar de la World Wide Web, un lenguaje de marcado para la construcción de páginas web que describe la estructura y el contenido en forma de texto empleando “etiquetas” (ej. <Body>...</Body> o <P>...</P>) para ello. Es un código descriptivo que tiene como objetivo dar formato al contenido que se pretende visualizar en el navegador, por lo que no constituye un lenguaje de programación.

Luego, HTML no es más que una serie de etiquetas que se utilizan para definir hasta cierto punto la forma o estilo en que se va a mostrar el documento puesto que está orientado a la estructura y no al aspecto. Debido a que es un estándar independiente de fabricantes y marcas puede ser interpretado por todos los navegadores, siendo sus

¹² Generan páginas web de forma dinámica a partir de los parámetros de la petición que envíe el navegador web.

aplicaciones rápidas, con mucho desarrollo y ocupan poco espacio. Permite combinar texto, sonido e imágenes con bastante éxito, hacer referencias a otros textos y mostrar los contenidos de una página web de forma organizada y sugestiva. Es fácil de aprender ya que posee un número reducido de etiquetas.

CSS (Cascading Style Sheets)

Las hojas de estilo en cascada son un lenguaje formal que define cómo se va a mostrar un documento estructurado, escrito en HTML o XML y persigue separar la estructura de la presentación. Al permitir especificar las hojas de estilo a los usuarios a través de sus navegadores consigue una gran accesibilidad y esto beneficia a personas con diferentes deficiencias visuales, que pueden adaptar los contenidos a sus capacidades, disminuye el tamaño de las páginas y por tanto optimiza el uso del ancho de banda.

Posibilita un mantenimiento más efectivo ya que un cambio en la hoja de estilo modifica inmediatamente todas las páginas que estén relacionadas con esta sin necesidad de hacerlo directamente en cada una. Define, mediante una sintaxis especial la forma de presentación que se le aplicará ya sea a un sitio web completo, un documento o página html, una porción del documento o a una etiqueta en concreto. Facilita la comprensión de los documentos a través del empleo de técnicas visuales.

JavaScript

Es un lenguaje de programación interpretado que no requiere compilación previa, utilizado principalmente en páginas web a las cuales dota de gran dinamismo, es un lenguaje orientado a objetos puesto que implementa la Herencia y con una sintaxis muy similar a la del lenguaje C y el lenguaje Java. Todos los navegadores modernos interpretan el código JavaScript integrado dentro de las páginas web.

Entre las acciones típicas que se pueden realizar en JavaScript tenemos dos vertientes. Por un lado los **efectos especiales** sobre páginas web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo. Por el otro, JavaScript nos permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que podemos crear **páginas interactivas** con programas como calculadoras, agendas, o tablas de cálculo. JavaScript es un lenguaje con muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos,

con funciones, estructuras de datos complejas, etc. Además, JavaScript pone a disposición del programador todos los elementos que forman la página web, para que éste pueda acceder a ellos y modificarlos dinámicamente. (Alvarez, 2006)

Sus requerimientos son relativamente sencillos, es un lenguaje cuyos códigos se interpretan en el navegador trasladando algunas tareas poco complejas al lado del cliente, sin tener que estar yendo constantemente al servidor para actualizar la información, por lo que en la actualidad es el lenguaje más utilizado del lado del cliente.

1.8.3 Fundamentación de los Lenguajes seleccionados

Para desarrollar la aplicación se escoge como lenguaje de programación del lado del servidor, PHP en su versión 5 debido a que es libre y multiplataforma facilitando así el acceso desde Windows o Linux que son los dos sistemas operativos más usados dentro del país y la universidad. Está a disposición del que lo desee utilizar licenciado bajo la PHP License.

Existe una comunidad de desarrollo dentro de la universidad con varios años de experiencia por lo que sería muy fácil encontrar apoyo y documentación referente al tema, además cuenta con muchos desarrolladores con un alto nivel técnico dentro de la misma que podrían brindar asesoría en caso necesario. Como lenguajes de programación del lado del cliente se seleccionó HTML, CSS y JavaScript por la gran integración que poseen entre ellos y las buenas funcionalidades que se logran a través de su combinación, obteniéndose unas interfaces totalmente dinámicas e innovadoras.

Además, son interpretados por la mayoría de los navegadores que se utilizan en la universidad y en el país y son muy ligeros por lo que favorecen a los diversos anchos de banda de que disponen los usuarios a nivel nacional.

1.9 Entorno de Desarrollo Integrado (Integrated Development Environment o IDE)

Los IDEs son un conjunto de herramientas para el programador, que suelen incluir en una misma suite, un buen editor de código, administrador de proyectos y archivos, enlaces transparentes a compiladores y debuggers e integración con sistemas controladores de versiones o repositorios. (Yerotaro, 2008)

1.9.1 NetBeans IDE 6.8

NetBeans es un IDE de código abierto que cuenta con una inmensa comunidad de desarrolladores y colaboradores, creado y patrocinado por Sun Microsystem. El código fuente está disponible para su reutilización de acuerdo con la Common Development and Distribution License (CDDL) v1.0 y la GNU General Public License (GPL) v2.

Posee todas las herramientas necesarias para crear aplicaciones web, de escritorio y para teléfonos móviles con los lenguajes de programación Java, C, C++ e incluso lenguajes dinámicos como: PHP, JavaScript, Groovy y Ruby. Es fácil de usar, instalar y puede ser ejecutado en múltiples plataformas como Windows, Linux, Mac OS X y Solaris. Además posee un número importante de módulos para extender las funcionalidades del IDE.

Soporta todas las características estándar, como el autocompletado de código, los sucesos marca de resaltado de sintaxis, refactoring, plantillas de código, documentación de pop-up¹³, navegación de código, las advertencias de editor y lista de tareas. Tiene una excelente integración con MySQL, brindando acceso a la BD de forma nativa.

Entre sus principales características podemos mencionar:

- Brinda completamiento automático de código PHP, así como coloreado de código sintáctico y semántico.
- Permite depurar el código usando Xdebug¹⁴.
- Genera fragmentos de código para bases de datos MySQL.

Dadas las características, ventajas y soporte que brinda este potente entorno de desarrollo y sobre todo por ser su licencia “open source”, además de la experiencia con que cuentan muchos programadores de la Facultad 10 en el uso del mismo, no hubo duda alguna a la hora de seleccionarlo como IDE a utilizar para el desarrollo del sistema a implementar.

1.10 Sistemas Gestores de Bases de Datos

Los Sistemas Gestores de Base de Datos (SGBD) permiten diseñar, construir y gestionar una base de datos, tienen como objetivo principal proporcionar una forma

¹³ El término denomina a las ventanas que emergen automáticamente (generalmente sin que el usuario lo solicite) mientras se accede a ciertas páginas web.

¹⁴ Xdebug es una extensión PHP que proporciona capacidades de depuración y perfiles.

sencilla, práctica y eficiente de almacenar y recuperar la información, facilitando así el acceso a los datos y provocando que los tiempos de respuesta a las solicitudes de los usuarios sean más cortos, por lo que son muy eficaces para la gestión de grandes volúmenes de información.

De la correcta selección de un SGBD depende mucho la seguridad y consistencia de los datos del sistema que se va a construir, por lo que aquí se hace un análisis de algunos de los más usados a nivel mundial ya sean libres o privativos.

1.10.1 Oracle

Es un sistema de administración de base de datos relacional (RDBMS o SGBDR por sus siglas en inglés y español respectivamente) que ocupa un amplio sector del mercado internacional actualmente, sobre todo su dominio se ha extendido dentro del mercado de servidores empresariales debido a que sólo se puede adquirir a un alto costo, siendo empresas muy grandes y multinacionales las que puedan explotar todo su potencial, por lo que no es muy usado en el desarrollo de páginas web cediendo espacio a otros gestores como SQL Server, Access o MySQL; aunque está comenzando a experimentar la competencia de gestores como FireBird, PostgreSQL o MySQL que poseen licencias de SWL.

Tiene muchas características que garantizan la integridad y seguridad de los datos: las transacciones se ejecutan de forma correcta y sin inconsistencia; es muy robusto, posee gran estabilidad y escalabilidad, lo que facilita el almacenamiento y administración de grandes volúmenes de información y a partir de las últimas versiones está certificado para trabajar bajo Linux. Para su desarrollo y manejo de la BD se puede usar SQL o un lenguaje de 5ta generación bastante potente denominado PL/SQL.

1.10.2 MySQL

Es un SGBD relacional, multihilo aprovechando las posibilidades de sistemas multiprocesador, y multiusuario, gestionando gran cantidad de datos de forma eficiente. Está licenciado bajo la GNU/GPL como software libre y además se puede incorporar a productos privativos adquiriendo una licencia especial por lo que posee una variada portabilidad.

Posee una gran variedad de APIs¹⁵ que permiten acceder a la BD desde muchas aplicaciones escritas en otros lenguajes de programación. Soporta gran variedad de tipos de datos por columnas, alta conectividad al permitir el intercambio eficiente entre distintos sistemas operativos usando sockets Unix, TCP/IP o tuberías, gran escalabilidad al gestionar BD muy grandes, con millones de registros y miles de tablas.

Mantiene un alto nivel de seguridad en los datos de usuarios al cifrar el tráfico de contraseñas, se integra perfectamente con PHP, es relativamente fácil de usar y tiene un desempeño bastante rápido. Todo lo expuesto anteriormente ha propiciado que tenga una gran aceptación en la comunidad de SWL llegando a ser el SGBD más usado dentro de este movimiento.

1.10.3 PostgreSQL

Es un Sistema Gestor de Base de Datos Objeto-Relacional (ORDBMS) considerado como el SGBD de código abierto más avanzado actualmente. Posee una amplia gama de ventajas, entre las que se encuentran la extensibilidad, confiabilidad y estabilidad, está diseñado para desempeñarse en ambientes de altos volúmenes de demandas en las diversas plataformas en las que se puede instalar, contando con un alto grado de conformidad por parte de las compañías que lo usan debido a que en varios años de trabajo a alta demanda nunca ha presentado caídas la aplicación.

Usa una estrategia de almacenamiento de filas llamada MVCC para conseguir una mejor respuesta en ambientes de grandes volúmenes. MVCC, o Control de Concurrencia Multi-Versión (Multi-Versión Concurrency Control), es la tecnología que PostgreSQL usa para evitar bloqueos innecesarios, está considerada mejor que el bloqueo a nivel de tupla porque un lector nunca es bloqueado por un escritor. En su lugar, PostgreSQL mantiene una ruta a todas las transacciones realizadas por los usuarios de la base de datos.

La API de acceso al SGBD está disponible en C, C++, Java, Perl, PHP, Python y TCL fundamentalmente. Cuenta con un rico conjunto de tipos de datos, permitiendo además su extensión mediante tipos y operadores definidos y programados por el usuario. Su administración se basa en usuarios y privilegios. Sus opciones de

¹⁵ **Interfaz de Programación de Aplicaciones** es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

conectividad abarcan TCP/IP, sockets Unix y sockets NT, además de soportar completamente ODBC.

1.10.4 Fundamentación del SGBD seleccionado

Como SGBD se seleccionó MySQL porque está orientado para sistemas en los que la velocidad y el número de accesos concurrentes sean altos, brinda un alto nivel de seguridad, tiene una excelente integración con el CMS Drupal y con PHP además de que es libre y multiplataforma, es rápido y fácil de usar, por lo que goza de gran aceptación dentro de la comunidad de SWL siendo el gestor más usado.

1.11 Servidor web

El Servidor Web es el programa destinado a gestionar las peticiones de los usuarios que visitan una página Web, utilizando para esto el protocolo HTTP o la versión cifrada HTTPS. Cuando un usuario realiza una visita a un dominio este se encarga de mostrarle las páginas web que tiene alojadas. Pueden ser de dos tipos, un "servidor dedicado" o un "servidor compartido", el primero es para aplicaciones de alta demanda y se refiere a una PC servidor dedicada exclusivamente al sitio del cliente y el segundo se refiere a que varios clientes van a compartir los recursos de la PC servidor.

Un servidor Web básico cuenta con un esquema de funcionamiento muy simple, basado en ejecutar infinitamente el siguiente bucle:

1. Espera peticiones en el puerto TCP indicado (el estándar por defecto para HTTP es el 80).
2. Recibe una petición.
3. Busca el recurso.
4. Envía el recurso utilizando la misma conexión por la que recibió petición.
5. Vuelve al segundo punto. (Cibernetia, 2004)

Tomando como principio de funcionamiento el esquema anterior se han diseñado y desarrollado todos los servidores web existentes, variando sólo el tipo de peticiones (páginas estáticas, CGIs¹⁶, Servlets, etc.) que pueden atender, en función de que sean o no multiproceso o multihilo.

¹⁶ **Interfaz de Entrada Común** es una importante tecnología de la World Wide Web que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web.

1.11.1 Apache

Apache es el servidor web más usado en internet alcanzando una cifra que oscila en el 70%, goza de gran reputación entre los servidores web gracias a su gran estabilidad y confiabilidad ya que alrededor del 90% de los servidores de más alta disponibilidad a nivel mundial lo usan, se trata de un servidor multiplataforma muy eficiente, robusto y modular. Se ha puesto mucho empeño en optimizar el rendimiento de su código por parte del grupo de soporte, conocido como "The Apache Group", un grupo de voluntarios muy comprometidos con su comunidad que planean y desarrollan todo lo relacionado a esta plataforma, comenzando por la documentación hasta el código en sí, resultando en una rápida aplicación que consume menos recursos de sistema que otros servidores.

Es muy personalizable, posee una arquitectura modular que permite construir un servidor a medida. Para configurar cada módulo es necesario configurar las directivas que están dentro del mismo. Dichos módulos se pueden clasificar en tres categorías: Módulos Base, Módulos Multiproceso y Módulos Adicionales.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código. El resto de las funcionalidades del servidor se consiguen por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades a este, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software. (Cuenca, 2003) Además es totalmente libre, licenciado bajo la Apache Software License que permite modificar el código y la implementación de los últimos y más nuevos protocolos.

Respecto a la administración, los archivos de configuración están en ASCII¹⁷, por lo que tiene un formato simple, y pueden ser editados empleando un editor de texto común. Estos son transferibles, lo que permite la clonación efectiva de un servidor. El servidor puede ser administrado vía línea de comandos, lo que hace la administración remota muy conveniente.

¹⁷ **Código Estadounidense Estándar** para el **Intercambio de Información** es un código de caracteres basado en el alfabeto latino tal como se usa en inglés moderno y en otras lenguas occidentales.

1.11.2 Internet Information Server (IIS)

Es un servidor web que provee una poderosa y flexible plataforma para desarrollar, desplegar y gestionar aplicaciones web utilizando los lenguajes más populares usados en la web, desde ASP.NET hasta PHP. Es completamente modular y extensible, por lo que se pueden añadir, quitar y sustituir fácilmente los componentes de IIS por componentes programados por el desarrollador, además de que se puede personalizar para satisfacer las necesidades de cada desarrollador. Hace simple de desplegar centralmente y manejar la configuración y el contenido de una granja de servidores web.

Proporciona soporte incorporado para delegar tareas de administración a dueños de sitios web, permitiendo realizar tareas de dirección y configuración básica sin la intervención administrativa, un veloz desempeño para el contenido estático y dinámico a través de una poderosa compresión HTTP y una más profunda integración con las peticiones de servicio del núcleo de Windows para sitios web SSL¹⁸ y autenticación de ventanas implícita que permite a los administradores autenticar a los usuarios de forma segura a través de servidores de seguridad y proxy.

Minimiza el tiempo de indisponibilidad y diagnostica rápidamente cuestiones del servidor y de las aplicaciones incorporando nuevas herramientas de diagnóstico. Maximiza la seguridad del servidor web por defecto con el aislamiento automático de las aplicaciones y acortando las huellas de memoria de procesos.

Incluye el protocolo de transferencia de noticias a través de la red (NNTP), el protocolo de transferencia de archivos (FTP) y el protocolo simple de transferencia de correo (SMTP). Es un producto de Microsoft por lo tanto totalmente privativo.

1.11.3 Fundamentación del Servidor Web seleccionado

La decisión del servidor web seleccionado no requirió mucho tiempo, debido a que Apache se ha convertido más que en una opción en una solución para los desarrolladores web, más que por su gratuidad que tanto influye, por la alta madurez e integridad en cuanto al código se refiere, constituyendo uno de los ejemplos más fehacientes de que el SWL puede alcanzar altos niveles de profesionalismo y convertirse en una vía de desarrollo para países con menos recursos.

¹⁸ **Protocolo de Capa de Conexión Segura** es un protocolo criptográfico que proporciona comunicación segura por una red.

CAPÍTULO

Exploración y Planificación

2.1 Fase de Exploración

Es la fase donde se define el alcance general del proyecto, el cliente especifica lo que necesita a través de la elaboración de sencillas Historias de Usuario apoyándose en los conocimientos y la experiencia de los desarrolladores. Es una fase relativamente corta pero muy importante ya que brinda una visión general del sistema.

Historias de Usuario identificadas:

Historia de Usuario: HU1-CU	
Número: 1	Nombre Historia de Usuario: <i>Crear Usuario</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Anónimo</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: 0.2
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: 0.2
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>El sistema debe brindar la facilidad de crear cuentas de usuario a todos los usuarios anónimos (visitantes).</i>	
Observaciones:	

Tabla 1: HU1 Crear Usuario

Historia de Usuario: HU2-AU	
Número: 2	Nombre Historia de Usuario: <i>Autenticar Usuario</i>

Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Autenticado, Usuario Administrador</i>	Iteración Asignada: <i>1</i>
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: <i>0.2</i>
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: <i>0.2</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios que poseen cuentas de usuario del sistema deben tener la posibilidad de autenticarse en el mismo.</i>	
Observaciones:	

Tabla 2: HU2 Autenticar Usuario

Historia de Usuario: HU3-MU	
Número: <i>3</i>	Nombre Historia de Usuario: <i>Modificar Usuario</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Autenticado, Usuario Administrador</i>	Iteración Asignada: <i>1</i>
Prioridad en Negocio: <i>Baja</i>	Puntos Estimados: <i>0.1</i>
Riesgo en Desarrollo: <i>Bajo</i>	Puntos Reales: <i>0.1</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>El sistema debe brindar la posibilidad a los usuarios con cuentas de usuario en el sistema que puedan modificar las mismas.</i>	
Observaciones:	

Tabla 3: HU3 Modificar Usuario

Historia de Usuario: HU4-EU

Número: 4	Nombre Historia de Usuario: <i>Eliminar Usuario</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Administrador</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Media</i>	Puntos Estimados: 0.1
Riesgo en Desarrollo: <i>Medio</i>	Puntos Reales: 0.1
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>El usuario Administrador debe contar con la capacidad de eliminar cuantas de usuario.</i>	
Observaciones:	

Tabla 4: HU4 Eliminar Usuario

Historia de Usuario: HU5-CR	
Número: 5	Nombre Historia de Usuario: <i>Crear Reporte</i>
Modificación de Historia de Usuario: 7	
Usuario: <i>Usuario Autenticado</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: 1
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: 1
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios que poseen cuentas de usuario del sistema deben tener la posibilidad de reportar sus experiencias con el Hardware y el SWL para colaborar con otros usuarios.</i>	
Observaciones: <i>Los reportes creados deben pasar por un proceso (Flujo de Trabajo) de revisión para comprobar que cumplen una serie de condiciones para su posterior publicación.</i>	

Tabla 5: HU5 Crear Reporte

Historia de Usuario: HU6-AR	
Número: 6	Nombre Historia de Usuario: <i>Actualizar Reporte</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Autenticado</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Media</i>	Puntos Estimados: 0.3
Riesgo en Desarrollo: <i>Bajo</i>	Puntos Reales: 0.2
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios que poseen cuentas de usuario del sistema que hayan realizado algún reporte deben tener la posibilidad de actualizar dicho reporte.</i>	
Observaciones:	

Tabla 6: HU6 Actualizar Reporte

Historia de Usuario: HU7-LR	
Número: 7	Nombre Historia de Usuario: <i>Listar Reporte</i>
Modificación de Historia de Usuario: 14	
Usuario: <i>Usuario Anónimo, Usuario Autenticado, Usuario Administrador</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: 1
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: 1.2
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>El sistema debe brindar la posibilidad de consultar los reportes y filtrarlos por varios criterios para optimizar la consulta.</i>	
Observaciones:	

Tabla 7: HU7 Listar Reporte

Historia de Usuario: HU8-ER	
Número: 8	Nombre Historia de Usuario: <i>Eliminar Reporte</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Autenticado, Usuario Administrador</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Baja</i>	Puntos Estimados: 0.2
Riesgo en Desarrollo: <i>Bajo</i>	Puntos Reales: 0.1
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>El sistema debe brindar la posibilidad de que los usuarios autenticados puedan eliminar sus propios reportes, así como a los administradores eliminar cualquier reporte.</i>	
Observaciones: <i>Los reportes que no cumplan las condiciones de publicación deben ser eliminados.</i>	

Tabla 8: HU8 Eliminar Reporte

Historia de Usuario: HU9-CA	
Número: 9	Nombre Historia de Usuario: <i>Crear Artículo</i>
Modificación de Historia de Usuario: 8	
Usuario: <i>Usuario Autenticado</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: 1
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: 1
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios que poseen cuentas de usuario del sistema deben tener la posibilidad de crear artículos relacionados con la compatibilidad del Hardware y el SWL para colaborar con otros usuarios.</i>	

Observaciones: Los artículos creados deben pasar por un proceso (Flujo de Trabajo) de revisión para comprobar que cumplen una serie de condiciones para su posterior publicación.

Tabla 9: HU9 Crear Artículo

Historia de Usuario: HU10-MA	
Número: 10	Nombre Historia de Usuario: <i>Modificar Artículo</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Autenticado</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Media</i>	Puntos Estimados: 0.2
Riesgo en Desarrollo: <i>Bajo</i>	Puntos Reales: 0.1
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios que poseen cuentas de usuario del sistema que han publicado algún artículo deben tener la posibilidad de modificarlo.</i>	
Observaciones:	

Tabla 10: HU10 Modificar Artículo

Historia de Usuario: HU11-LA	
Número: 11	Nombre Historia de Usuario: <i>Listar Artículos</i>
Modificación de Historia de Usuario: 10	
Usuario: <i>Usuario Anónimo, Usuario Autenticado, Usuario Administrador</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: 1
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: 1.1
Programador Responsable: <i>Jorge A. Avila Cruz</i>	

Descripción: *El sistema debe brindar la posibilidad de que los usuarios del sistema accedan a la lista de artículos publicados enviados por otros usuarios.*

Observaciones:

Tabla 11: HU11 Listar Artículos

Historia de Usuario: HU12-EA	
Número: 12	Nombre Historia de Usuario: <i>Eliminar Artículo</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Autenticado, Usuario Administrador</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Baja</i>	Puntos Estimados: 0.2
Riesgo en Desarrollo: <i>Bajo</i>	Puntos Reales: 0.1
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios autenticados que han colaborado con algún artículo deben tener la posibilidad de eliminar sus propios artículos, de la misma forma que los usuarios administradores pueden eliminar cualquier artículo.</i>	
Observaciones: <i>Los artículos que no cumplan las condiciones de publicación deben ser eliminados.</i>	

Tabla 12: HU12 Eliminar Artículo

Historia de Usuario: HU13-AH	
Número: 13	Nombre Historia de Usuario: <i>Adicionar Hardware</i>
Modificación de Historia de Usuario: 10	
Usuario: <i>Usuario Administrador</i>	Iteración Asignada: 1
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: 1

Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: <i>1</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>El sistema debe brindar la posibilidad a los usuarios administradores de que adicionen nueva información sobre Hardware.</i>	
Observaciones:	

Tabla 13: HU13 Adicionar Hardware

Historia de Usuario: HU14-UH	
Número: <i>14</i>	Nombre Historia de Usuario: <i>Actualizar Hardware</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Administrador</i>	Iteración Asignada: <i>1</i>
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: <i>0.2</i>
Riesgo en Desarrollo: <i>Medio</i>	Puntos Reales: <i>0.1</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios administradores deben tener la posibilidad de actualizar la información correspondiente a un Hardware específico.</i>	
Observaciones:	

Tabla 14: HU14 Actualizar Hardware

Historia de Usuario: HU12-LH	
Número: <i>15</i>	Nombre Historia de Usuario: <i>Listar Hardware</i>
Modificación de Historia de Usuario: <i>17</i>	
Usuario: <i>Usuario Administrador</i>	Iteración Asignada: <i>1</i>
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: <i>1</i>

Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: <i>1.2</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios administradores deben tener la posibilidad de listar el Hardware existente en la BD por diferentes criterios.</i>	
Observaciones:	

Tabla 15: HU 15 Listar Hardware

Historia de Usuario: HU16-EH	
Número: <i>16</i>	Nombre Historia de Usuario: <i>Eliminar Hardware</i>
Modificación de Historia de Usuario: <i>ninguna</i>	
Usuario: <i>Usuario Administrador</i>	Iteración Asignada: <i>1</i>
Prioridad en Negocio: <i>Media</i>	Puntos Estimados: <i>0.2</i>
Riesgo en Desarrollo: <i>Bajo</i>	Puntos Reales: <i>0.1</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: <i>Los usuarios administradores deben de tener la posibilidad de eliminar un Hardware de la BD.</i>	
Observaciones: <i>Esto sólo ocurrirá en un caso extremo en que se liberen todas las restricciones del Hardware, tanto comerciales como tecnológicas.</i>	

Tabla 16: HU16 Eliminar Hardware

2.2 Fase de Planificación

La Planificación es una fase relativamente corta, durante la cual se realiza una estimación del esfuerzo que tomará implementar cada historia de usuario. Como resultado de esta fase se obtiene un Plan de Entregas por el cuál se van a guiar las partes involucradas en el proyecto, dígase clientes, desarrolladores y gerentes.

2.2.1 Estimación de esfuerzo por HU

Para el desarrollo del sistema propuesto se realizó una estimación del esfuerzo, apoyado en la técnica denominada en XP como “spikes” que no es más que la realización de pequeños programas de pruebas como unidad de medida para reducir los riesgos de una planificación incorrecta, por cada HU identificada, obteniendo una medida bastante real de la duración del desarrollo del proyecto y ofreciendo un plan razonable al cual ajustarse, obteniendo así los resultados que se muestran en la siguiente tabla.

Módulos	Historias de usuario	Puntos estimados
Gestión de Usuario	Crear Usuario	0.2
	Autenticar Usuario	0.2
	Modificar Usuario	0.1
	Eliminar Usuario	0.1
Gestión de Reporte	Crear Reporte	1
	Actualizar Reporte	0.2
	Listar Reportes	1.2
	Eliminar Reporte	0.1
Gestión de Artículo	Crear Artículo	1
	Modificar Artículo	0.1
	Listar Artículos	1.1
	Eliminar Artículo	0.1
Gestión de Hardware	Adicionar Hardware	1
	Actualizar Hardware	0.1
	Listar Hardware	1.2
	Eliminar Hardware	0.1

Tabla 17: Organización por Módulos y Estimación de esfuerzo por HU

2.3 Plan de Iteraciones

Las HU identificadas que conformarán el sistema, serán desarrolladas y probadas en un ciclo de iteración, por cada módulo definido para cada entrega, de acuerdo al orden antes establecido.

Iteración 1:

Se implementará el primer módulo: Gestión de Usuario, estableciendo los niveles de acceso y las opciones de navegación.

Iteración 2:

Se realizará el segundo módulo: Gestión de Reporte, que tiene una prioridad alta en el negocio puesto que da respuesta a uno de los objetivos que se propone alcanzar el desarrollo de este sistema, de que los usuarios transmitan las experiencias adquiridas a través de la instalación de SWL en el Hardware que poseen.

Iteración 3:

Se desarrollará el tercer módulo: Gestión de Artículo, que también tiene una alta prioridad en el negocio y servirá para que los usuarios colaboren compartiendo información relacionada con el tema de compatibilidad y así otros usuarios puedan nutrirse de esta información.

Iteración 4:

Se llevará a cabo el cuarto módulo: Gestión de Hardware, considerado el más importante puesto que contiene las funcionalidades más importantes del sistema y de él se derivan una serie de servicios que va a ofrecer la aplicación.

2.4 Plan de duración de las iteraciones

A continuación se ilustra en una tabla el plan de duración estimado para cada Módulo de HU por iteraciones de implementación. Aplicando esta estimación al equipo de desarrollo que las implementará y el orden en que se van a desarrollar.

Iteración	Orden de la Historia de Usuario	Duración de las iteraciones
1	1.1 Crear Usuario 1.2 Autenticar Usuario 1.3 Modificar Usuario 1.4 Eliminar Usuario	1 Semana
2	2.1 Crear Reporte 2.2 Actualizar Reporte 2.3 Listar Reportes 2.4 Eliminar Reporte	2.5 Semanas
	3.1 Crear Artículo	

3	3.2 Modificar Artículo 3.3 Listar Artículos 3.4 Eliminar Artículo	2.3 Semanas
4	4.1 Adicionar Hardware 4.2 Actualizar Hardware 4.3 Listar Hardware 4.4 Eliminar Hardware	2.4 Semanas

Tabla 18: Plan de duración de las iteraciones

2.5 Plan de Entregas

El plan o cronograma de liberaciones (releases) tiene como objetivo definir el número de entregas que se cumplirán a lo largo del período de vida del proyecto y las iteraciones necesarias para desarrollar cada una. Para realizar el plan de entregas, el sistema propuesto se dividió en módulos que contienen las historias de usuario relacionadas lógicamente de acuerdo a su propósito. Cada una de las entregas se correspondió con una iteración de desarrollo.

Módulo	Final Iteración: 1ra	Final Iteración: 2da	Final Iteración: 3ra	Final Iteración: 4ta
	1ra semana de marzo.	4ta semana de marzo.	2da semana de abril.	1ra semana de mayo.

Tabla 19: Plan de Entregas

CAPÍTULO

Implementación y Pruebas

3.1 Fase de Implementación

Para dar solución a las tareas de implementación, según la metodología aplicada, las Historias de Usuario se agruparon por módulos y se desarrollaron a través de un proceso iterativo, que permitió la liberación continua de mini versiones del sistema y la integración inmediata de las mismas a las ya desarrolladas, en cada iteración se realizó el ciclo (análisis, diseño, implementación y pruebas) completo de desarrollo, proceso que se vio favorecido por las facilidades y potencialidades que brinda el CMS Drupal.

Iteración 1: Módulo Gestión de Usuario

Tarea:	
Número de Tarea: 1	Código de la HU: HU1-CU
Nombre de la Tarea: <i>Crear Usuario</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: 0.2
Fecha de Inicio: 1/03/2010	Fecha de Fin: 2/03/2010
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: Se diseña la interfaz que se va a mostrar al usuario con los campos previstos para que introduzca todos los datos correspondientes al perfil de usuario. Se capturan los datos y se guardan en la BD. Se muestra un mensaje de operación satisfactoria.	

Tabla 20: Tarea 1 Crear usuario

Tarea:	
Número de Tarea: 2	Código de la HU: HU2-AU
Nombre de la Tarea: <i>Autenticar Usuario</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: 0.2
Fecha de Inicio: 3/03/2010	Fecha de Fin: 4/03/2010
Programador Responsable: <i>Jorge A. Avila Cruz</i>	

Descripción: Se diseña la interfaz que se va a mostrar al usuario con los campos previstos para que introduzca los datos de su cuenta de usuario. Se capturan los datos y se comprueba su validez en la BD.

Tabla 21: Tarea 2 Autenticar Usuario

Tarea:	
Número de Tarea: 3	Código de la HU: HU3-MU
Nombre de la Tarea: <i>Modificar Usuario</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: 0.1
Fecha de Inicio: 5/03/2010	Fecha de Fin: 5/03/2010
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: Se diseñó la interfaz que se va a mostrar al usuario con los campos que contienen los datos de su cuenta de usuario. Se capturan los datos y se modifica la información en la BD. Se muestra un mensaje de cambios realizados satisfactoriamente.	

Tabla 22: Tarea 3 Modificar Usuario

Tarea:	
Número de Tarea: 4	Código de la HU: HU4-EU
Nombre de la Tarea: <i>Eliminar Usuario</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: 0.1
Fecha de Inicio: 6/03/2010	Fecha de Fin: 6/03/2010
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: Como el administrador es el único con permisos para eliminar cuentas de usuario, se diseña una interfaz donde se listan todos los usuarios del sistema a la cual sólo puede acceder el administrador. Este proceso ocurriría previa solicitud del usuario que desea eliminar su cuenta del sistema, contactando al administrador. Se eliminan los datos correspondientes de la BD.	

Tabla 23: Tarea 4 Eliminar Usuario

Iteración 2: Módulo Gestión de Reporte

Tarea:	
Número de Tarea: 5	Código de la HU: HU5-CR
Nombre de la Tarea: <i>Crear Reporte</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: 1

Fecha de Inicio: 8/03/2010	Fecha de Fin: 13/03/2010
Programador Responsable: Jorge A. Avila Cruz	
Descripción: Se diseñó la interfaz para crear reportes con todos los campos necesarios para capturar los datos y la información que indica qué datos recoge cada campo, para un mejor entendimiento por parte del usuario. Se almacenan los datos en la BD y se muestra un mensaje de operación realizada satisfactoriamente.	

Tabla 24: Tarea 5 Crear Reporte

Tarea:	
Número de Tarea: 6	Código de la HU: HU6-AR
Nombre de la Tarea: Actualizar Reporte	
Tipo de Tarea: Desarrollo	Tiempo Estimado: 0.2
Fecha de Inicio: 15/03/2010	Fecha de Fin: 16/03/2010
Programador Responsable: Jorge A. Avila Cruz	
Descripción: Se muestran los datos del reporte que desea actualizar el usuario a través de una interfaz diseñada específicamente para esto. Se capturan los datos y se guardan en la BD, luego se muestra un mensaje de actualización satisfactoria.	

Tabla 25: Tarea 6 Actualizar Reporte

Tarea:	
Número de Tarea: 7	Código de la HU: HU7-LR
Nombre de la Tarea: Listar Reportes	
Tipo de Tarea: Desarrollo	Tiempo Estimado: 1.1
Fecha de Inicio: 17/03/2010	Fecha de Fin: 24/03/2010
Programador Responsable: Jorge A. Avila Cruz	
Descripción: Se diseñó una interfaz específica para esta funcionalidad que muestra los títulos de la lista de reportes publicados en el sistema y varias opciones de filtrado de la información, con el objetivo de optimizar la búsqueda cuando el número de reportes sea muy grande, además cada título posee un link que redireccionará al usuario que lo desee a una interfaz que muestra toda la información del reporte. En caso de no encontrarse los reportes que le interesan consultar al usuario se muestra un mensaje que informa que no existen dichos reportes en la BD todavía.	

Tabla 26: Tarea 7 Listar Reportes

Tarea:	
Número de Tarea: 8	Código de la HU: HU8-ER

Nombre de la Tarea: <i>Eliminar Reporte</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: <i>0.2</i>
Fecha de Inicio: <i>25/03/2010</i>	Fecha de Fin: <i>26/03/2010</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: Para eliminar un reporte se accede a través de la interfaz para actualizar reporte, donde se brinda la opción de borrar dicho reporte. Luego se muestra un mensaje de confirmación de la eliminación.	

Tabla 27: Tarea 8 Eliminar Reporte

Iteración 3: Módulo Gestión de Artículo

Tarea:	
Número de Tarea: <i>9</i>	Código de la HU: <i>HU9-CA</i>
Nombre de la Tarea: <i>Crear Artículo</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: <i>1</i>
Fecha de Inicio: <i>27/03/2010</i>	Fecha de Fin: <i>2/04/2010</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: Se diseñó una interfaz específica para crear artículos con los campos necesarios para capturar la información y una opción de previsualización de cómo quedará el artículo al ser publicado. Se guarda dicha información en la BD del sistema y se muestra un mensaje de creación del artículo satisfactoria.	

Tabla 28: Tarea 9 Crear Artículo

Tarea:	
Número de Tarea: <i>10</i>	Código de la HU: <i>HU10-MA</i>
Nombre de la Tarea: <i>Modificar Artículo</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: <i>0.1</i>
Fecha de Inicio: <i>3/04/2010</i>	Fecha de Fin: <i>3/04/2010</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: Se diseñó una interfaz donde se muestra la información que conforma un artículo, la cual puede ser modificada. Se captura la nueva información y se actualiza la BD, luego se muestra un mensaje de modificación de la información satisfactoria.	

Tabla 29: Tarea 10 Modificar Artículo

Tarea:	
Número de Tarea: <i>11</i>	Código de la HU: <i>HU11-LA</i>

Nombre de la Tarea: <i>Listar Artículos</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: <i>1.1</i>
Fecha de Inicio: <i>5/04/2010</i>	Fecha de Fin: <i>12/04/2010</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: Se diseñó una interfaz específica para mostrar los artículos publicados en el sistema, donde cada artículo muestra el título en forma de link que redireccionará al usuario al artículo completo y también muestra el primer párrafo del artículo. De no existir ningún artículo publicado se muestra un mensaje que informa al usuario que no hay artículos publicados todavía.	

Tabla 30: Tarea 11 Listar Artículos

Tarea:	
Número de Tarea: <i>12</i>	Código de la HU: <i>HU12-EA</i>
Nombre de la Tarea: <i>Eliminar Artículo</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: <i>0.1</i>
Fecha de Inicio: <i>13/04/2010</i>	Fecha de Fin: <i>13/04/2010</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: A través de la interfaz para modificar un artículo se brinda una opción para borrar un artículo creado previamente. Se elimina la información correspondiente al artículo en la BD y se muestra un mensaje de operación eliminar satisfactoriamente ejecutada.	

Tabla 31: Tarea 12 Eliminar Artículo

Iteración 4: Módulo Gestión de Hardware

Tarea:	
Número de Tarea: <i>13</i>	Código de la HU: <i>HU13-AH</i>
Nombre de la Tarea: <i>Adicionar Hardware</i>	
Tipo de Tarea: <i>Desarrollo</i>	Tiempo Estimado: <i>1</i>
Fecha de Inicio: <i>14/04/2010</i>	Fecha de Fin: <i>20/04/2010</i>
Programador Responsable: <i>Jorge A. Avila Cruz</i>	
Descripción: Para realizar esta labor se diseñó una interfaz específica que contiene los campos necesarios para capturar la información referente a un Hardware en particular. La información introducida en los campos se guarda en la BD y se muestra un mensaje de adición satisfactoria.	

Tabla 32: Tarea 13 Adicionar Hardware

Tarea:	
Número de Tarea: 14	Código de la HU: HU14-UH
Nombre de la Tarea: Actualizar Hardware	
Tipo de Tarea: Desarrollo	Tiempo Estimado: 0.1
Fecha de Inicio: 21/04/2010	Fecha de Fin: 21/04/2010
Programador Responsable: Jorge A. Avila Cruz	
Descripción: Para actualizar la información relativa a un Hardware en específico se diseñó una interfaz que muestra la información correspondiente al Hardware y permite editar la misma. Dicha información es capturada nuevamente y actualizada en la BD, luego se muestra un mensaje de actualización satisfactoria.	

Tabla 33: Tarea 14 Actualizar Hardware

Tarea:	
Número de Tarea: 15	Código de la HU: HU15-LH
Nombre de la Tarea: Listar Hardware	
Tipo de Tarea: Desarrollo	Tiempo Estimado: 1.2
Fecha de Inicio: 22/04/2010	Fecha de Fin: 30/05/2010
Programador Responsable: Jorge A. Avila Cruz	
Descripción: Se diseñó una interfaz específica para realizar esta tarea, la cual muestra la lista de Hardware almacenada en la BD, mostrando algunos de los datos que posee un Hardware, para consultar todos estos datos se consigue a través del dato Tipo de Hardware que posee un link a una interfaz donde se muestran todos los datos antes mencionados, además se facilitan una serie de filtros para optimizar el proceso de búsqueda. De no encontrarse ningún resultado en la BD se muestra un mensaje que informa que la búsqueda no obtuvo ningún resultado.	

Tabla 34: Tarea 15 Listar Hardware

Tarea:	
Número de Tarea: 16	Código de la HU: HU16-EH
Nombre de la Tarea: Eliminar Hardware	
Tipo de Tarea: Desarrollo	Tiempo Estimado: 0.1
Fecha de Inicio: 3/05/2010	Fecha de Fin: 3/05/2010
Programador Responsable: Jorge A. Avila Cruz	

Descripción: Esta acción le corresponde solamente al Administrador y la realiza en caso que se liberen todas las restricciones del Hardware, tanto comerciales como tecnológicas. Para esto se ofrece una opción Borrar a través de la interfaz de Actualizar Hardware. Luego se eliminan los datos de la BD y se muestra un mensaje de operación realizada satisfactoriamente.

Tabla 35: Tarea 16 Eliminar Hardware

3.2 Fase de Pruebas

El proceso de pruebas en XP constituye uno de los pilares fundamentales de la metodología, en este proceso se convoca a los desarrolladores a probar constantemente, tanto como sea posible. Mediante esta filosofía se reduce el número de errores no detectados así como el tiempo entre la introducción en el sistema y su detección. Proceso que contribuye a garantizar la calidad de los productos desarrollados y a que los programadores se sientan seguros cuando necesitan introducir cambios o modificaciones en el desarrollo.

3.2.1 Pruebas de Aceptación

Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Las pruebas de aceptación son consideradas como “pruebas de caja negra”. Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. (Joskowicz, 2008)

Para realizar este proceso de pruebas de aceptación y con el objetivo de no repetir tantas tablas de prueba, en los campos **Descripción** y **Entradas/Pasos de ejecución** se especificarán los diferentes escenarios de pruebas para una misma HU, siempre y cuando no se estime necesario realizarlo por separado.

Iteración 1: Módulo Gestión de Usuario

Caso de Prueba de Aceptación	
Código CP: <i>HU1-CP1</i>	Nombre HU: <i>Crear Usuario</i>
Nombre de la persona que realiza la prueba: <i>Jorge A. Avila Cruz</i>	
Descripción de la Prueba: <i>Se verifica en la BD que el usuario creado</i>	

efectivamente se adicionó a la tabla de usuarios.

Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.

Entrada / Pasos de ejecución:

1. Se introducen los datos correspondientes a un usuario ficticio y se presiona en el botón Crear.
2. Se intenta crear un usuario ficticio dejando uno o todos los campos vacíos.

Resultado Esperado:

1. El usuario es creado correctamente apareciendo el registro en la BD del sistema y se muestra un mensaje de operación satisfactoria.
2. Se muestra un mensaje de error informando que no se pueden crear usuario con campos de datos vacíos.

Evaluación de la Prueba: Satisfactoria.

Tabla 36: CP Crear Usuario

Caso de Prueba de Aceptación	
Código CP: HU2-CP2	Nombre HU: Autenticar Usuario
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que un usuario previamente creado se autentique correctamente en el sistema.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución:	
<ol style="list-style-type: none">1. Se introducen los datos correspondientes a un usuario existente en la BD y se presiona en el botón Entrar.2. Se intenta la autenticación de un usuario con datos falsos.3. Se intenta la autenticación dejando uno o todos los campos vacíos.	
Resultado Esperado:	
<ol style="list-style-type: none">1. El usuario cambia de rol y pasa a ser un usuario autenticado por lo que obtiene nuevos privilegios en el sistema.2. Se muestra un mensaje de error informando al usuario que no se reconocen los datos provistos y se brindan opciones para recuperar la contraseña o recordarle su nombre de usuario en caso de que esta sea su situación.3. Se muestra un mensaje de error informando que no se puede acceder al sistema dejando campos de datos vacíos en cualquiera de sus variantes.	
Evaluación de la Prueba: Satisfactoria.	

Tabla 37: CP Autenticar Usuario

Caso de Prueba de Aceptación	
Código CP: HU3-CP3	Nombre HU: Modificar Usuario
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que un usuario previamente creado tenga la posibilidad de modificar los datos de su cuenta de usuario.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución:	

1. Se introducen los nuevos datos que el usuario desea cambiar en su cuenta de usuario y se presiona el botón Guardar.
2. Se intenta la modificación de un dato de la cuenta de usuario dejándolo vacío.
3. Se intenta la modificación de los datos de una cuenta de usuario con datos que coinciden con los datos de una cuenta de usuario que ya existe en la BD.

Resultado Esperado:

1. Se muestra un mensaje de operación satisfactoria.
2. Se muestra un mensaje de error informando al usuario que la operación no es válida y se mantiene en la interfaz de modificación.
3. Se muestra un mensaje de error informando que el usuario introducido ya está asignado a otra cuenta y que ese correo ya está registrado en el sistema. Se mantiene al usuario en la interfaz de modificación.

Evaluación de la Prueba: Satisfactoria.

Tabla 38: CP Modificar Usuario

Caso de Prueba de Aceptación	
Código CP: HU4-CP4	Nombre HU: Eliminar Usuario
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que un usuario previamente creado en el sistema sea eliminado correctamente por el Administrador.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución:	
<ol style="list-style-type: none"> 1. El Administrador selecciona el usuario que desea eliminar y pincha en la opción borrar. 	
Resultado Esperado:	
<ol style="list-style-type: none"> 1. Se muestra un mensaje de operación satisfactoria y los datos del usuario son eliminados de la BD. 	
Evaluación de la Prueba: Satisfactoria.	

Tabla 39: CP Eliminar Usuario

Iteración 2: Módulo Gestión de Reporte

Caso de Prueba de Aceptación	
Código CP: HU5-CP5	Nombre HU: Crear Reporte
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que un usuario autenticado tenga la opción de crear un reporte y que esto ocurra de manera correcta y eficaz.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Se llenan todos los campos requeridos para realizar un reporte y se pincha en el botón Guardar. 2. Se intenta crear un reporte dejando algunos de los campos obligatorios sin llenar. 	
Resultado Esperado:	

1. Se muestra un mensaje de operación satisfactoria y los datos del reporte son almacenados en la BD.
2. Se muestra un mensaje de error y se indican los campos obligatorios que fueron dejados en blanco, la interfaz de Crear Reporte se mantiene activa debajo del mensaje.

Evaluación de la Prueba: Satisfactoria.

Tabla 40: CP Crear Reporte

Caso de Prueba de Aceptación	
Código CP: HU6-CP6	Nombre HU: Actualizar Reporte
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que un usuario autenticado tenga la opción de actualizar un reporte que haya realizado previamente y que esto ocurra de manera correcta y eficaz.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución: <ol style="list-style-type: none"> 1. Se modifican los datos del reporte y se presiona en el botón Guardar. 2. Se intenta actualizar un reporte dejando algunos o todos los campos obligatorios sin llenar. 	
Resultado Esperado: <ol style="list-style-type: none"> 3. Se muestra un mensaje de operación satisfactoria y los datos del reporte son actualizados en la BD. 4. Se muestra un mensaje de error y se indican los campos obligatorios que fueron dejados en blanco, la interfaz de Crear Reporte se mantiene activa debajo del mensaje. 	
Evaluación de la Prueba: Satisfactoria.	

Tabla 41: CP Actualizar Reporte

Caso de Prueba de Aceptación	
Código CP: HU7-CP7	Nombre HU: Listar Reportes
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que los usuarios tengan la opción de listar reportes publicados en el sistema y que esto ocurra de manera correcta y eficaz.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución: <ol style="list-style-type: none"> 1. Se accede a la interfaz donde se puede consultar los reportes realizados por otros usuarios en la que se listan los títulos de los reportes en forma de link que redirecciona al usuario a todo su contenido. 2. Se listan los reportes publicados por orden del más reciente al más antiguo. 3. Se brindan una serie de opciones para hacer la búsqueda más específica. 4. Se comprueba que el paginado de la interfaz funcione correctamente. 	
Resultado Esperado: <ol style="list-style-type: none"> 1. Se muestra la interfaz Listar Reportes con todos los reportes que contiene y los títulos de los reportes funcionan como un link que redirecciona a todo el contenido del reporte seleccionado. 2. Los reportes están ordenados por orden de publicación, el último en ser 	

- publicado es el primero que se muestra.
3. Los filtros disponibles hacen que la búsqueda sea más precisa.
 4. El paginado funciona correctamente mostrando las páginas siguientes y las anteriores.
 5. En todos los casos anteriores si la acción no encontró reportes publicados se muestra un mensaje de error.

Evaluación de la Prueba: Satisfactoria.

Tabla 42: CP Listar Reportes

Caso de Prueba de Aceptación	
Código CP: HU8-CP8	Nombre HU: Eliminar Reporte
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que los usuarios que tengan reportes publicados en el sistema tengan la posibilidad de eliminarlos y que esto ocurra de manera correcta y eficaz.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución: <ol style="list-style-type: none"> 1. Se selecciona un reporte ficticio creado previamente y se presiona el botón Borrar. 	
Resultado Esperado: <ol style="list-style-type: none"> 1. Se muestra un mensaje de advertencia y las opciones para confirmar o cancelar la operación. 2. Se eliminan los datos de la BD y se muestra un mensaje de operación satisfactoria. 	
Evaluación de la Prueba: Satisfactoria.	

Tabla 43: CP Eliminar Reporte

Iteración 3: Módulo Gestión de Artículo

Caso de Prueba de Aceptación	
Código CP: HU9-CP9	Nombre HU: Crear Artículo
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que los usuarios autenticados puedan crear sus propios artículos de forma correcta y eficiente.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución: <ol style="list-style-type: none"> 1. Se accede a la interfaz para crear un artículo, se introduce la información ficticia del artículo y se presiona el botón Guardar. 2. Se intenta crear un artículo dejando alguno o todos los campos vacíos. 	
Resultado Esperado: <ol style="list-style-type: none"> 1. Se guarda la información en la BD y se muestra un mensaje de operación satisfactoria. 2. Se muestra un mensaje de error informando los campos obligatorios que se dejaron vacíos y se mantiene activa la interfaz Crear Artículo. 	
Evaluación de la Prueba: Satisfactoria.	

Tabla 44: CP Crear Artículo

Caso de Prueba de Aceptación	
Código CP: HU10-CP10	Nombre HU: <i>Modificar Artículo</i>
Nombre de la persona que realiza la prueba: <i>Jorge A. Avila Cruz</i>	
Descripción de la Prueba: <i>Se verifica que los usuarios autenticados puedan modificar sus propios artículos de forma correcta y eficiente.</i>	
Condiciones de Ejecución: <i>La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.</i>	
Entrada / Pasos de ejecución: <ol style="list-style-type: none"> <i>Se accede a la interfaz para modificar un artículo, se introduce la nueva información del artículo y se presiona el botón Guardar.</i> <i>Se intenta crear un artículo dejando alguno o todos los campos vacíos.</i> 	
Resultado Esperado: <ol style="list-style-type: none"> <i>Se guarda la información en la BD y se muestra un mensaje de operación satisfactoria.</i> <i>Se muestra un mensaje de error informando los campos obligatorios que se dejaron vacíos y se mantiene activa la interfaz Crear Artículo.</i> 	
Evaluación de la Prueba: <i>Satisfactoria.</i>	

Tabla 45: CP Modificar Artículo

Caso de Prueba de Aceptación	
Código CP: HU11-CP11	Nombre HU: <i>Listar Artículos</i>
Nombre de la persona que realiza la prueba: <i>Jorge A. Avila Cruz</i>	
Descripción de la Prueba: <i>Se verifica que todos los usuarios puedan consultar los artículos publicados de forma correcta y eficiente.</i>	
Condiciones de Ejecución: <i>La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.</i>	
Entrada / Pasos de ejecución: <ol style="list-style-type: none"> <i>Se accede a la interfaz donde se listan todos los artículos publicados.</i> <i>Se pincha en el título de un artículo que es un link que redirecciona al artículo completo.</i> 	
Resultado Esperado: <ol style="list-style-type: none"> <i>Se muestran todos los artículos publicados en el sistema ordenados en forma tal que el último en publicarse es el primero en mostrarse y también se muestra el primer párrafo del artículo.</i> <i>Se muestra el artículo íntegramente.</i> <i>De no existir ningún artículo publicado se muestra un mensaje de error.</i> 	
Evaluación de la Prueba: <i>Satisfactoria.</i>	

Tabla 46: CP Listar Artículos

Caso de Prueba de Aceptación	
Código CP: HU12-CP12	Nombre HU: <i>Eliminar Artículo</i>
Nombre de la persona que realiza la prueba: <i>Jorge A. Avila Cruz</i>	
Descripción de la Prueba: <i>Se verifica que los usuarios autenticados que han escrito algún artículo puedan eliminarlo de forma correcta y eficiente.</i>	
Condiciones de Ejecución: <i>La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el</i>	

servidor de BD MySQL deben estar ejecutándose normalmente.

Entrada / Pasos de ejecución:

1. Se accede a la interfaz donde se brinda la opción Borrar el artículo y se pincha en esta.

Resultado Esperado:

1. Se muestra un mensaje de advertencia y las opciones para confirmar o cancelar la operación.
2. Se elimina la información de la BD y se muestra un mensaje de operación satisfactoria.

Evaluación de la Prueba: Satisfactoria.

Tabla 47: CP Eliminar Artículo

Iteración 4: Módulo Gestión de Hardware

Caso de Prueba de Aceptación	
Código CP: HU13-CP13	Nombre HU: Adicionar Hardware
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que los usuarios administradores puedan adicionar hardware de forma correcta y eficiente a la BD.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución: <ol style="list-style-type: none">1. Se accede a la interfaz de adicionar hardware, se introduce toda la información correspondiente a un hardware y se presiona el botón Guardar.2. Se intenta adicionar un hardware dejando alguno o todos los campos obligatorios de la interfaz vacíos.	
Resultado Esperado: <ol style="list-style-type: none">1. Se adiciona la información a la BD y se muestra un mensaje de operación satisfactoria.2. Se muestra un mensaje de error informando los campos que se dejaron vacíos y se mantiene activa la interfaz Adicionar Hardware.	
Evaluación de la Prueba: Satisfactoria.	

Tabla 48: CP Adicionar Hardware

Caso de Prueba de Aceptación	
Código CP: HU14-CP14	Nombre HU: Actualizar Hardware
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que los usuarios administradores puedan actualizar los datos de un hardware de forma correcta y eficiente en la BD.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución: <ol style="list-style-type: none">1. Se accede a la interfaz de actualizar hardware, se introduce la nueva información correspondiente a un hardware y se presiona el botón Guardar.2. Se intenta actualizar un hardware dejando alguno o todos los campos obligatorios de la interfaz vacíos.	
Resultado Esperado:	

1. Se actualiza la información en la BD y se muestra un mensaje de operación satisfactoria.
2. Se muestra un mensaje de error informando los campos que se dejaron vacíos y se mantiene activa la interfaz Adicionar Hardware.

Evaluación de la Prueba: Satisfactoria.

Tabla 49: CP Actualizar Hardware

Caso de Prueba de Aceptación	
Código CP: HU15-CP15	Nombre HU: Listar Hardware
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que los usuarios con permisos adecuados puedan consultar los datos de un hardware de forma correcta y eficiente.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Se accede a la interfaz donde se lista el hardware existente en la BD, se muestran varios campos de los que contiene cada hardware y un link en el campo Tipo de Hardware que redirecciona a la información completa del hardware. 2. Se brindan varias opciones de filtrado de búsqueda para hacerla más específica. 3. Se comprueba que el paginado funcione correctamente. 	
Resultado Esperado:	
<ol style="list-style-type: none"> 1. Se muestran algunos campos de información del hardware y el link redirecciona a toda la información de un hardware en específico. 2. Los filtros acotan la búsqueda y la hacen más específica. 3. El paginado permite ir a la página siguiente y a la anterior de forma eficiente. 4. De no producirse ningún resultado con la búsqueda se muestra un mensaje de error. 	
Evaluación de la Prueba: Satisfactoria.	

Tabla 50: CP Listar Hardware

Caso de Prueba de Aceptación	
Código CP: HU16-CP16	Nombre HU: Eliminar Hardware
Nombre de la persona que realiza la prueba: Jorge A. Avila Cruz	
Descripción de la Prueba: Se verifica que los usuarios con permisos adecuados puedan eliminar los datos de un hardware de forma correcta y eficiente de la BD.	
Condiciones de Ejecución: La aplicación debe ejecutarse en condiciones normales simulando el desempeño real de la misma. El servidor de aplicaciones Apache y el servidor de BD MySQL deben estar ejecutándose normalmente.	
Entrada / Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Se accede a la interfaz de eliminar hardware y se pincha en el botón Borrar. 	
Resultado Esperado:	
<ol style="list-style-type: none"> 1. Se muestra un mensaje de advertencia y las opciones para confirmar o cancelar la operación. 2. Se elimina la información de la BD y se muestra un mensaje de operación satisfactoria. 	
Evaluación de la Prueba: Satisfactoria.	

Tabla 51: CP Eliminar Hardware

CONCLUSIONES

La investigación desarrollada permitió constatar la importancia que adquiere la correcta selección de hardware para el buen desempeño del mismo ejecutando una distribución de SWL, acción que toma mayor importancia si se tiene en cuenta el proceso de migración que se está llevando a cabo en el país y los altos costos de adquisición de estos equipos.

Como se pudo comprobar existen varios sistemas similares al que se desea desarrollar que podrían servir como punto de partida, aunque las necesidades específicas y las características del país en torno a la informática y en particular al SWL hacen necesario el desarrollo de un sistema hecho a la medida, que satisfaga las demandas de las empresas y los usuarios cubanos.

El estudio realizado arrojó como resultado la existencia de una serie de técnicas, tecnologías y herramientas libres que responden perfectamente a las necesidades tecnológicas que requiere el proyecto software que se va a realizar.

Permitió comprobar que el Software Libre está alcanzando niveles de popularidad y excelencia que lo convierten en una solución factible para muchos problemas que hace unos años eran imposibles enfrentar debido al costo que acarreaban, constituyendo una nueva forma de plantearse el desarrollo y la cooperación en cuanto a las TIC se refiere.

Se desarrolló un sistema que gestiona de forma precisa y eficiente la información relacionada con la compatibilidad entre el Hardware y el SWL, además con una arquitectura sencilla para que se faciliten la inclusión de nuevas funcionalidades y que los usuarios finales puedan utilizar la aplicación de forma intuitiva y correcta, por lo que se considera que se le dio solución al problema planteado al inicio del trabajo.

RECOMENDACIONES

Por la importancia que tiene el desarrollo de este trabajo de diploma y con el objetivo de contribuir a mejorar un proceso tan importante para el país como la migración a SWL, en medio de una crisis económica de magnitud global, se recomienda:

- Continuar profundizando la investigación con respecto a la compatibilidad entre el Hardware y el Software Libre con el objetivo de incluir nuevas funcionalidades al sistema desarrollado.
- La creación del Centro de Certificación y Homologación de Hardware para SWL, para que su desempeño contribuya a mantener la BD de Hardware del sistema desarrollo continuamente actualizada.
- La integración del sistema desarrollado a la Plataforma Integral para la Migración a SWL del proyecto SIMAYS.
- La Regulación del Hardware importado, bajo estrictas condiciones de compatibilidad certificada para SWL.
- El despliegue del sistema dentro de la UCI durante un corto período de prueba para su posterior despliegue a todo el país.
- Que se utilice como fuente de apoyo para las operaciones comerciales que se lleven a cabo por parte del Comité Coordinador de Compras e Importaciones de equipos de cómputo del Ministerio de la Informática y las Comunicaciones.

REFERENCIAS BIBLIOGRÁFICAS

Stallman, Richard. 2007. The GNU Operating System. [En línea] 2007. [Citado el: 25 de 03 de 2010.] <http://www.gnu.org/philosophy/can-you-trust.es.html>.

CMS Hispano. 2007. CMS Hispano. [En línea] 2007. [Citado el: 22 de 02 de 2010.] <http://cms-hispano.org/index.php?s=content&p=necesidad>.

Developers4Web. 2005. Foro de Plone. [En línea] 2005. [Citado el: 22 de 02 de 2010.] <http://www.plonehispano.com/>.

Drupal Hispano. 2005. Drupal Hispano. [En línea] 2005. [Citado el: 22 de 02 de 2010.] <http://drupal.org.es/drupal>.

Tecnologías Racionales. 2007. Tecnologías Racionales. [En línea] 2007. [Citado el: 22 de 02 de 2010.] <http://www.tecnologiasracionales.com/servicios/consulting/rup.php>.

Joskowicz, José. 2008. Reglas y Prácticas en eXtrame Programing. [En línea] 2008. [Citado el: 08 de 03 de 2010.] <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>.

Proyectos Ágiles. 2007. Proyectos Ágiles. [En línea] 2007. [Citado el: 13 de 03 de 2010.] <http://www.proyectosagiles.org/que-es-scrum>.

Landeros Gómez, Ruth Priscila y Del Juncal Huerta, Jorge Luis. 2006. Monografías. [En línea] 2006. [Citado el: 23 de 02 de 2010.] <http://www.monografias.com/trabajos14/herramicase/herramicase.shtml>.

Alvarez, Miguel Angel. 2006. Desarrolloweb. [En línea] 2006. [Citado el: 26 de 02 de 2010.] <http://www.desarrolloweb.com/articulos/25.php>.

Yerotaro. 2008. Yerotaro. [En línea] 2008. [Citado el: 26 de 02 de 2010.] <http://yerotaro.wordpress.com/2008/09/>.

Cibernetia. 2004. Cibernetia. [En línea] 2004. [Citado el: 01 de 03 de 2010.] http://www.cibernetia.com/manuales/instalacion_servidor_web/1_conceptos_basicos.php.

Cuenca, Carlos Luis. 2003. Desarrolloweb. [En línea] 2003. [Citado el: 02 de 03 de 2010.] <http://www.desarrolloweb.com/articulos/1112.php>.

Joskowicz, José. 2008. Reglas y Prácticas en eXtrame Programing. [En línea] 2008. [Citado el: 28 de 04 de 2010.] <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>.

BIBLIOGRAFÍA

Alvarez, Miguel Angel. 2006. Desarrolloweb. [En línea] 2006. [Citado el: 26 de 02 de 2010.] <http://www.desarrolloweb.com/articulos/25.php>.

AppLabs. 2009. Linux Tested. [En línea] 2009. [Citado el: 18 de 02 de 2010.] <http://www.linuxtested.com>.

Astroza, Felipe. 2008. Open-Free. [En línea] 2008. [Citado el: 25 de 03 de 2010.] <http://freeopen.wordpress.com/2008/09/29/%C2%BFque-es-tivoization/>.

Bloggea2. 2010. Bloggea2. [En línea] 2010. [Citado el: 27 de 03 de 2010.] <http://www.bloggea2.com/netbeans-descargar-gratis-programacion-ide-java/>.

Bracci Roa, Luigino. 2009. Somos Libres. [En línea] 2009. [Citado el: 26 de 03 de 2010.] <http://www.somoslibres.org/modules.php?name=News&file=article&sid=2494>.

Cibernetia. 2004. Cibernetia. [En línea] 2004. [Citado el: 01 de 03 de 2010.] http://www.cibernetia.com/manuales/instalacion_servidor_web/1_conceptos_basicos.php.

CMS Hispano. 2007. CMS Hispano. [En línea] 2007. [Citado el: 22 de 02 de 2010.] <http://cms-hispano.org/index.php?s=content&p=necesidad>.

Comunidad Joomla! 2008. Comunidad Joomla! [En línea] 2008. [Citado el: 22 de 02 de 2010.] <http://comunidadjoomla.org/>.

Corai, Ricardo. 2010. CMS en Español. [En línea] 2010. [Citado el: 22 de 02 de 2010.] <http://www.cmsenespanol.com/>.

Cuenca, Carlos Luis. 2003. Desarrolloweb. [En línea] 2003. [Citado el: 02 de 03 de 2010.] <http://www.desarrolloweb.com/articulos/1112.php>.

de la Torre, Aníbal. 2006. Bitácora de Aníbal de la Torre. [En línea] 2006. [Citado el: 26 de 02 de 2010.] http://www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html.

DedaSys LLC. 2008. Linux Incompatibility List. [En línea] 2008. [Citado el: 18 de 02 de 2010.] <http://leenooks.com/>.

DefectiveByDesign. 2009. DefectiveBydesign. [En línea] 2009. [Citado el: 25 de 03 de 2010.] <http://www.defectivebydesign.org/>.

Delgado Expósito, Eryly. Monografías. [En línea] [Citado el: 22 de 02 de 2010.] <http://www.monografias.com/trabajos60/metodologias-desarrollo-software/metodologias-desarrollo-software.shtml>.

- Developers4Web. 2005.** Foro de Plone. [En línea] 2005. [Citado el: 22 de 02 de 2010.] <http://www.plonehispano.com/>.
- Drupal Hispano. 2005.** Drupal Hispano. [En línea] 2005. [Citado el: 22 de 02 de 2010.] <http://drupal.org.es/drupal>.
- Drupal Web.** Drupal Web. [En línea] [Citado el: 22 de 02 de 2010.] <http://www.drupalweb.com/>.
- Drupal-es.** Drupal-es. [En línea] [Citado el: 22 de 02 de 2010.] <http://www.drupal-es.org/>.
- Eguíluz Pérez, Javier. 2009.** LibrosWeb. [En línea] 2009. [Citado el: 26 de 02 de 2010.] <http://librosweb.es/javascript>.
- Enciclopedia Libre Universal. 2008.** Enciclopedia Libre Universal. [En línea] 2008. [Citado el: 22 de 02 de 2010.] http://enciclopedia.us.es/index.php/Lenguaje_de_programaci%C3%B3n.
- Factoría de Internet S.L. 2008.** WebTaller. [En línea] 2008. [Citado el: 24 de 02 de 2010.] <http://www.webtaller.com/maletin/articulos/lenguajes-programacion-web.php>.
- Free Download Manager. 2007.** Free Download Manager. [En línea] 2007. [Citado el: 24 de 02 de 2010.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.
- Free Software Foundation. 2010.** Free Software Foundation. [En línea] 2010. [Citado el: 18 de 02 de 2010.] <http://www.fsf.org/resources/hw>.
- G.S.I. 2007.** GSInnova. [En línea] 2007. [Citado el: 24 de 02 de 2010.] <http://www.rational.com.ar/herramientas/roseenterprise.html>.
- García, Joaquin. 2006.** Ingeniero Software. [En línea] 2006. [Citado el: 13 de 03 de 2010.] <http://www.ingenierosoftware.com/equipos/scrum.php>.
- Joomla! Hispano.** Joomla! Hispano. [En línea] [Citado el: 22 de 02 de 2010.] <http://www.joomlahispano.org/>.
- Joskowicz, José. 2008.** Reglas y Prácticas en eXtreme Programming. [En línea] 2008. [Citado el: 08 de 03 de 2010.] <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>.
- La Revista Informática. 2006.** La Revista Informática. [En línea] 2006. [Citado el: 25 de 02 de 2010.] <http://www.larevistainformatica.com/lenguajes-programacion-web.htm>.
- Landeros Gómez, Ruth Priscila y Del Juncal Huerta, Jorge Luis. 2006.** Monografías. [En línea] 2006. [Citado el: 23 de 02 de 2010.] <http://www.monografias.com/trabajos14/herramicase/herramicase.shtml>.
- Lenguajes de Programación. 2009.** Lenguajes de Programación. [En línea] 2009. [Citado el: 22 de 02 de 2010.] <http://www.lenguajes-de-programacion.com/lenguajes-de-programacion.shtml>.

Lenguajes de Programación. 2009. Lenguajes de Programación. [En línea] 2009. [Citado el: 25 de 02 de 2010.] <http://www.lenguajes-de-programacion.com/programacion-web.shtml>.

Lepied, Frederic. 2009. Hardware4Linux. [En línea] 2009. [Citado el: 18 de 02 de 2010.] <http://hardware4linux.info/>.

Linalco. 2009. Linalco. [En línea] 2009. [Citado el: 26 de 03 de 2010.] <http://www.linalco.com/noticias-linalco/contra-la-crisis-linux.html>.

LinLap. 2006. LinLap. [En línea] 2006. [Citado el: 17 de 02 de 2010.] <http://www.linlap.com/>.

Linux Compatible. 2009. Linux Compatible. [En línea] 2009. [Citado el: 17 de 02 de 2010.] <http://www.linuxcompatible.org>.

Linux on Laptops. 2008. Linux on Laptops. [En línea] 2008. [Citado el: 17 de 02 de 2010.] <http://www.linux-on-laptops.com/>.

LinuxCertified. 2009. LinuxCertified. [En línea] 2009. [Citado el: 17 de 02 de 2010.] <http://www.linuxcertified.com/index.html>.

LinuxQuestions. LinuxQuestions. [En línea] [Citado el: 17 de 02 de 2010.] <http://www.linuxquestions.org/hcl>.

LinWik. 2006. LinWik. [En línea] 2006. [Citado el: 17 de 02 de 2010.] <http://www.linwik.com/>.

Mandriva. Mandriva HCL. [En línea] [Citado el: 18 de 02 de 2010.] <http://hcl.mandriva.com/>.

Masadelante. 2010. Masadelante. [En línea] 2010. [Citado el: 26 de 02 de 2010.] <http://www.masadelante.com/faqs/javascript>.

Mastermagazine. 2008. Mastermagazine. [En línea] 2008. [Citado el: 17 de 02 de 2010.] <http://www.mastermagazine.info/articulo/13092.php>.

Menéndez, Rafael y Asensio, Barzanallana. 2010. Ingeniería de Software. [En línea] 2010. [Citado el: 23 de 02 de 2010.] http://www.um.es/docencia/barzana/IAGP/Enlaces/CASE_principales.html.

Microsoft Corporation. 2010. IIS. [En línea] 2010. [Citado el: 02 de 03 de 2010.] <http://www.iis.net/>.

Mis Respuestas. 2010. Mis Respuestas. [En línea] 2010. [Citado el: 01 de 03 de 2010.] <http://www.misrespuestas.com/que-es-un-servidor-web.html>.

Montejo, Juan Carlos. 2006. ADR Formación. [En línea] 2006. [Citado el: 01 de 03 de 2010.] <http://www.adrformacion.com/articulo.php?ref=261&cont=Comparativa%20MySQL%20vs%20Oracle>.

Mozilla Developer Center. 2009. Mozilla Developer Center. [En línea] 2009. [Citado el: 26 de 02 de 2010.] https://developer.mozilla.org/en/About_JavaScript.

Muto, Kenshi. 2010. Debian GNU/Linux device driver check & report. [En línea] 2010. [Citado el: 18 de 02 de 2010.] <http://kmuto.jp/debian/hcl/>.

Netcraft. 2010. Netcraft. [En línea] 2010. [Citado el: 02 de 03 de 2010.]
http://news.netcraft.com/archives/2008/01/28/january_2008_web_server_survey.html.

Noción Digital. 2008. Noción Digital. [En línea] 2008. [Citado el: 26 de 02 de 2010.]
<http://www.nociondigital.com/webmasters/php-tutorial-lenguajes-del-lado-del-servidor-clientehtmlque-es-eso-detalle-177.html>.

Nouveau. 2009. Nouveau Wiki. [En línea] 2009. [Citado el: 17 de 02 de 2010.]
<http://nouveau.freedesktop.org/wiki/>.

Novell. 2009. OpenSUSE. [En línea] 2009. [Citado el: 18 de 02 de 2010.]
<http://es.opensuse.org/Hardware>.

Object Management Group. 2010. BPMN Information Home. [En línea] 2010. [Citado el: 23 de 02 de 2010.] <http://www.bpmn.org/>.

—. **2010.** Unified Modeling Language. [En línea] 2010. [Citado el: 22 de 02 de 2010.]
<http://www.uml.org/>.

Pérez Valdés, Damian. Maestros del Web. [En línea] [Citado el: 24 de 02 de 2010.]
<http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>.

Plataforma Anti DRM. Plataforma Anti DRM. [En línea] [Citado el: 25 de 03 de 2010.]
<http://www.anti-drm.org/articulos/introduccion-al-drm>.

Plone Foundation. 2010. Plone CMS. [En línea] 2010. [Citado el: 22 de 02 de 2010.]
<http://plone.org/>.

PostgreSQL Global Development Group. 2010. PostgreSQL. [En línea] 2010. [Citado el: 01 de 03 de 2010.] <http://www.postgresql.org>.

Pritchard, Steven. 2007. Linux Hardware Compatibility HOWTO. [En línea] 2007. [Citado el: 18 de 02 de 2010.] <http://www.tldp.org/HOWTO/Hardware-HOWTO/index.html>.

Proyectos Ágiles. 2007. Proyectos Ágiles. [En línea] 2007. [Citado el: 13 de 03 de 2010.]
<http://www.proyectosagiles.org/que-es-scrum>.

Ramos, Baruch. 2009. Blogultura. [En línea] 2009. [Citado el: 27 de 03 de 2010.]
<http://blogultura.com/java/netbeans-6-8-liberado/>.

Red Hat. 2007. Red Hat. [En línea] 2007. [Citado el: 18 de 02 de 2010.]
<https://hardware.redhat.com/>.

Scribd. 2007. Scribd. [En línea] 2007. [Citado el: 22 de 02 de 2010.]
<http://www.scribd.com/doc/297224/RUP>.

Softonic. 2010. Softonic. [En línea] 2010. [Citado el: 01 de 03 de 2010.]
<http://mysql.softonic.com/>.

- Solotuweb. 2009.** Solotuweb. [En línea] 2009. [Citado el: 26 de 02 de 2010.]
<http://www.solotuweb.com/lenguajes-lado-servidor-o-cliente-14038.html>.
- Stallman, Richard. 2009.** GNU Operating System. [En línea] 2009. [Citado el: 25 de 03 de 2010.]
<http://www.gnu.org/licenses/rms-why-gplv3.html>.
- . **2007.** The GNU Operating System. [En línea] 2007. [Citado el: 25 de 03 de 2010.]
<http://www.gnu.org/philosophy/can-you-trust.es.html>.
- Tecnologías Racionales. 2007.** Tecnologías Racionales. [En línea] 2007. [Citado el: 22 de 02 de 2010.] <http://www.tecnologiasracionales.com/servicios/consulting/rup.php>.
- The Linux Foundation. 2010.** The Linux Foundation. [En línea] 2010. [Citado el: 18 de 02 de 2010.] <http://www.linuxfoundation.org/collaborate/workgroups/openprinting>.
- The Linux Information Project. 2007.** [En línea] 2007. [Citado el: 25 de 03 de 2010.]
<http://www.linfo.org/tivoization.html>.
- The Power of Mind. 2006.** The Power of Mind. [En línea] 2006. [Citado el: 26 de 03 de 2010.]
<http://draxus.org/weblog/2006/04/01/hardware-con-certificacion-linux/>.
- Todo Expertos. 2010.** Todo Expertos. [En línea] 2010. [Citado el: 26 de 02 de 2010.]
<http://www.todoexpertos.com/categorias/tecnologia-e-internet/bases-de-datos/respuestas/163504/porque-utilizar-oracle>.
- TuFunción. 2006.** TuFunción. [En línea] 2006. [Citado el: 25 de 02 de 2010.]
<http://www.tufuncion.com/diferentes-lenguajes-programacion>.
- Ubuntu. UbuntuHCL. [En línea] [Citado el: 18 de 02 de 2010.]** <http://www.ubuntuhcl.org/>.
- W3C. 2007.** W3C. [En línea] 2007. [Citado el: 26 de 02 de 2010.] <http://www.w3.org/MarkUp/>.
- . **2008.** W3C. [En línea] 2008. [Citado el: 26 de 02 de 2010.]
<http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>.
- Wells, Don. 2009.** Extreme Programming. [En línea] 2009. [Citado el: 22 de 02 de 2010.]
<http://www.extremeprogramming.org/>.
- Yerotaro. 2008.** Yerotaro. [En línea] 2008. [Citado el: 26 de 02 de 2010.]
<http://yerotaro.wordpress.com/2008/09/>.

GLOSARIO DE TÉRMINOS Y DEFINICIONES

Hardware: corresponde a todas las partes físicas y tangibles de una computadora: sus componentes eléctricos, electrónicos, electromecánicos y mecánicos; sus cables, gabinetes o cajas, periféricos de todo tipo y cualquier otro elemento físico involucrado.

Tivoización: fenómeno derivado de la utilización de SWL en Hardware restrictivo, denominado así porque fue la compañía TiVo quien lo protagonizó, explotando una brecha legal en la licencia GPLv2.0 y que se corrigió en la v3.0 del 2007.

Multiplataforma: es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.

Plataforma: es una combinación de hardware y software usada para ejecutar aplicaciones; en su forma más simple consiste únicamente de un sistema operativo, una arquitectura, o una combinación de ambos.

Compatibilidad: es la condición que hace que un programa y un sistema, arquitectura o aplicación logren comprenderse correctamente tanto directamente o indirectamente (mediante un algoritmo).

Driver: un controlador de dispositivo (llamado normalmente controlador, o, en inglés, driver) es un programa informático que permite al sistema operativo interactuar con un periférico, haciendo una abstracción del hardware y proporcionando una interfaz posiblemente estandarizada para usarlo.

Ubuntu: es una distribución de GNU/Linux basada en Debian que proporciona un sistema operativo actualizado y estable para el usuario promedio, con un fuerte enfoque en la facilidad de uso y de instalación del sistema.

Gentoo: es una distribución de GNU/Linux orientada a usuarios con cierta experiencia en estos sistemas operativos.

Debian: es una comunidad conformada por desarrolladores y usuarios, que mantiene un sistema operativo GNU basado en software libre precompilado y empaquetado, en un formato sencillo en múltiples arquitecturas de computador y en varios núcleos.

Red Hat: es la compañía responsable de la creación y mantenimiento de una distribución del sistema operativo GNU/Linux que lleva el mismo nombre: Red Hat Enterprise Linux.

OpenSUSE: es el nombre de la distribución y proyecto libre auspiciado por Novell y AMD para el desarrollo y mantenimiento de un sistema operativo basado en GNU/Linux.

GNU/Linux: es el término empleado para referirse al sistema operativo similar a Unix que utiliza como base las herramientas de sistema de GNU y el núcleo Linux.

Internet: Gran red descentralizada de ordenadores, de ámbito global y públicamente accesible, que proporciona una ingente cantidad de servicios de comunicación de varios tipos, incluyendo la World Wide Web, el correo electrónico y muchos otros.

ANEXOS

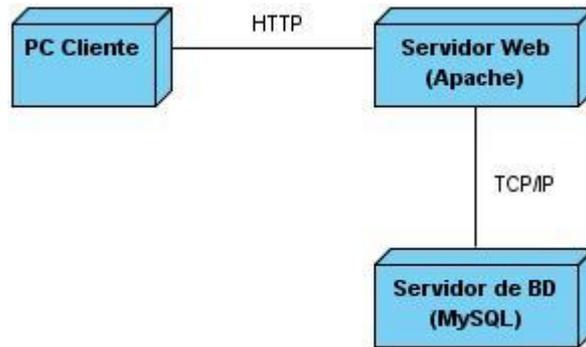


Figura 4: Diagrama de Despliegue

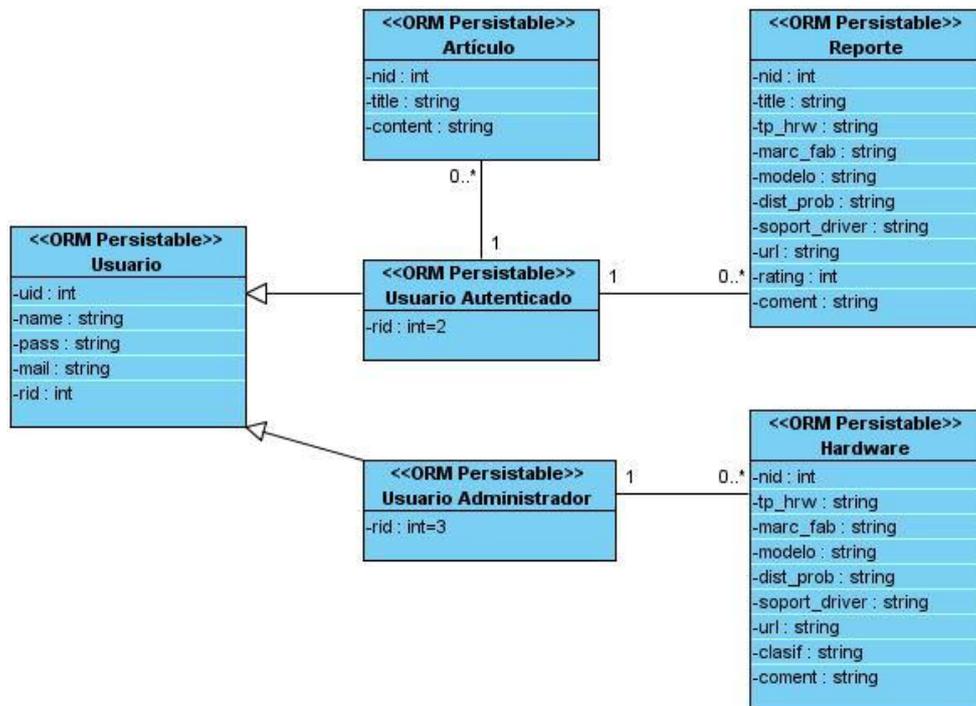


Figura 5: Diagrama de Clases Persistentes