



Facultad 6

SISTEMA PARA LA GESTIÓN DE LA INFORMACIÓN DE LOS EXPEDIENTES TÉCNICOS EN EL CENTRO DE INFORMÁTICA INDUSTRIAL

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

Autor: Yadriel Cuesta Hechavarría

Tutora: MsC. Zobeida Rosa Pérez López Chávez

Co-tutores: Ing. Luis Andrés Valido Fajardo

Ing. Luanner Kerton Martínez

La Habana, marzo, 2015

Año 57 de la Revolución

Declaración de Autoría

Declaro ser el autor de la presente tesis: “**Sistema para la gestión de la información de los Expedientes Técnicos en el Centro de Informática Industrial**” 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 20 días del mes de marzo del 2015

Firma del Autor

Yadriel Cuesta Hechavarría

Firma de la Tutora

MSc. Zobeida Rosa Pérez López Chávez

Firma del Co-Tutor

Ing. Luis Andrés Valido Fajardo

Firma del Co-Tutor

Ing. Luanner Kerton Martínez

Datos del contacto

Autor: Yadriel Cuesta Hechavarría

Correo: yadriel@uci.cu

Tutora: Zobeida Rosa Pérez López Chávez

Especialidad de graduación: Licenciatura en educación- especialidad en Química

Categoría Docente: Principal Asistente

Categoría Científica: Máster

Años de experiencia en el tema: 11 años

Años de graduado: 21 años

Correo: zobeida@uci.cu

Co-tutor: Luanner Kerton Martínez

Especialidad de graduación: Ingeniero en Ciencias Informáticas

Años de experiencia en el tema: 4

Años de graduado: 3

Correo: luanner@uci.cu

Co-tutor: Luis Andrés Valido Fajardo

Especialidad de graduación: Ingeniero en Ciencias Informáticas

Categoría Docente: Adiestrado

Años de experiencia en el tema: 4

Años de graduado: 2

Correo: lvalido@uci.cu

Dedicatoria

*A todos los que me quieren y me
tienen en cuenta.*

Agradecimientos

Sería muy ingrato de mi parte si en estos momentos no demuestro mis profundos agradecimientos a todos aquellos que contribuyeron a que este día llegara. Quisiera comenzar agradeciendo:

- *A mi soberano e omnipotente Dios que con su ayuda permitió que yo naciera sano y con sabiduría a pesar de mi malformación congénita.*
- *A mi adorada y magistral madre que ha sido para mí un ejemplo de dedicación y empeño.*
- *A mi querida hermana que ha sido el motor impulsor de mi vida para que yo siga adelante.*
- *A mi abuelo por ser aquel hombre que no escatimó esfuerzos para que yo lograra ser lo que soy hoy. A toda mi familia de manera general.*
- *A Zobe que más que mi tutora, fue mi amiga y madre, gracias por tus enseñanzas y gracias por no molestarte cuando hacía lo que me daba la gana en el documento y más atrás venias tú a ayudarme a enmendar el desastre.*
- *A Valido por ser una persona tan especial que de forma desinteresada se brindó para ayudarme cuando más lo necesité.*
- *A Luanner por brindarme su ayuda en todo momento.*
- *A mis eternos compañeros de lucha, me refiero a mis compañeros de aula, gracias por soportarme en estos largos y sacrificados años de estudios.*
- *A mis compañeros de trabajo y ahora colegas por brindarme su ayuda, en especial a Yosvani, Ariannys, Yailín, Marisniulkis, Batista, Dayrien, etc.*
- *A mi exjefe de departamento, leo, que aunque en estos momentos está por los países me facilitó el tiempo para estudiar.*
- *A Maiquel y Bridón director y subdirector respectivamente del CEDIN, por cooperar conmigo siempre y aguantarme mis cuentos.*
- *Al grupo de calidad del Centro por ayudarme en todo momento, en especial a Lianet, Yarisnelys, Marelis y Yareivy.*

Agradecimientos

- *A mis amigos de la UCI Ayme, Baby, Porti, Yohana, Harold, Sayuri, Fifi, Yanelis Benítez, Rosa María, Dayní y Yornué, sepan que tienen un lugar en mi corazón.*
- *Al piquete de fiesta y parranda Gloria, Juan Carlos, Nenuco, Leydis por ser mis amigos y demostrármelo siempre.*
- *A Gina, Digna por siempre creer en mí, por quererme tanto y valorarme.*
- *Al tribunal por sus críticas constructivas para mejorar mi documento de tesis.*
- *A mis profesores de la Facultad 6 por dedicar su vida a la maravillosa tarea de enseñar y prepararnos como buenos profesionales.*
- *A mi oponente por su capacidad profesional y su humildad para conmigo.*

Y en especial a aquella persona que fue tan valiosa en mi vida y que diera lo que no tengo por tenerla en este minuto presente físicamente entre nosotros porque espiritualmente lo está, a ti mi abuela querida que siempre fuiste luz en medio de mi oscuridad y que parte de lo que ahora soy también te lo debo a tú. Que Dios te ilumine siempre.

A todos muchas gracias.

Resumen

El acelerado desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) ha permitido que los sistemas de información tomen un papel protagónico en las organizaciones, mediante la creciente utilización y ventajas ofrecidas. La Universidad de las Ciencias Informáticas (UCI) es uno de los frutos del desarrollo de las TIC en Cuba. La UCI cuenta con centros de desarrollo donde se encuentra el Centro de Informática Industrial (CEDIN), creado con el propósito de desarrollar productos informáticos para la automatización industrial; la constante asignación y cambio de los recursos tecnológicos a este centro, es necesario para lograr una alta calidad en sus productos. Para gestionar estos recursos tecnológicos se crea el expediente técnico con el objetivo de lograr un mejor control de estos. La presente investigación tiene como objetivo informatizar los procesos que se llevan a cabo en la gestión de los expedientes técnicos, para contribuir a la gestión de la información que se almacena en ellos. Se selecciona como metodología de desarrollo de software Xtreme Programming (XP) para guiar el proceso de desarrollo, como herramienta CASE el Visual Paradigm para describir y representar los artefactos del sistema; MySQL como sistema gestor de base de datos; PHP como lenguaje de programación y PHPStorm como entorno de desarrollo integrado.

Palabras Claves: Expediente técnico, Recursos tecnológicos.

The rapid development of Information Technology and Communications (ITC) has enabled information systems take a leading role in organizations, thanks to its constant use and the benefits they provide. The University of Informatics Sciences (UIS) is one of the fruits of development of ITC in Cuba. Within the hierarchical structure of the UIS is the Informatics of Industrial Center (IIC), created with the purpose of developing software products for industrial automation; allocation and constant technological change to this center resources, it is necessary to achieve high quality products. Therefore, the technical file to have better control of existing technology assets in the center is created. This research focuses on the automation of processes carried out with the use of records, for better management of information stored in them. Is selected as development methodology Xtreme Programming (XP) software to guide the development process; as modeling language UML CASE tool using the Visual Paradigm to describe and represent artifacts of the system; as a system manager MySQL database; PHP programming language and PhpStorm as integrated development environment.

Keywords: technical file, technological resources

Introducción.....	1
Capítulo I Fundamentación Teórica.....	5
1.1 Gestión de información.....	5
1.1.1 Componentes fundamentales del proceso de gestión.....	5
1.2 Sistemas de gestión de información.....	6
1.2.1 Sistemas de gestión de información para recursos tecnológicos.....	7
1.4 Herramientas y tecnologías de desarrollo.....	12
1.4.1 Herramientas CASE.....	13
1.4.2 Framework de desarrollo.....	14
1.4.3 Entorno de desarrollo.....	15
1.4.4 Lenguajes de programación.....	17
1.4.5 Servidor web.....	18
1.4.6 Sistema gestor de base de datos.....	19
Conclusiones parciales.....	21
Capítulo II Análisis y diseño del sistema.....	22
2.1 Consideraciones del negocio.....	22
2.2 Modelo conceptual.....	23
2.3 Propuesta del sistema.....	25
2.4 Fase de exploración y planificación.....	25
2.5 Patrones de diseño.....	36
2.6 Arquitectura del framework de desarrollo Yii.....	37
2.7 Diseño de Base de Datos.....	40
2.8 Tarjetas CRC.....	42
2.9 Diagrama de despliegue.....	44
Conclusiones parciales.....	46
Capítulo III Implementación y prueba.....	47
3.1 Tarea de ingeniería.....	47
3.2 Estándares de codificación.....	51
3.3 Pruebas.....	52
3.4 Resultados de las pruebas realizadas por el grupo de calidad del Centro de Informática Industrial.....	57
Conclusiones parciales.....	58

Conclusiones	59
Recomendaciones	60
Referencias bibliográficas	61
Anexos	64

Índice de figuras

TABLA 1 EJEMPLO DE UNA HISTORIA DE USUARIO.	28
TABLA 2 HISTORIA DE USUARIO. GESTIONAR EXPEDIENTE TÉCNICO	30
TABLA 3 HISTORIA DE USUARIO: GENERAR FICHA TÉCNICA.....	33
TABLA 4 ESTIMACIÓN DE ESFUERZO POR HISTORIA DE USUARIO	34
TABLA 5 PLAN DE ITERACIONES	35
TABLA 6 PLAN DE ENTREGA.....	36
TABLA 7 DESCRIPCIÓN DE LAS ENTIDADES DEL SISTEMA.....	41
TABLA 8 TARJETA CRC. EXPEDIENTE TÉCNICO.....	43
TABLA 9 TARJETA CRC. HISTORIAL DE MODIFICACIONES.....	43
TABLA 10 TARJETA CRC. LOCAL	44
TABLA 11 TARJETA CRC. NÚMERO DE INVENTARIO	44
TABLA 12 TARJETA CRC. TRASLADO.....	44
TABLA 13 TAREA DE INGENIERÍA. ADICIONAR EXPEDIENTE TÉCNICO	47
TABLA 14 RESUMEN DE LAS TAREAS DE INGENIERÍA POR CADA HISTORIA DE USUARIO	48
TABLA 15 TAREA DE INGENIERÍA. MODIFICAR EXPEDIENTE TÉCNICO	50
TABLA 16 TAREA DE INGENIERÍA. ELIMINAR EXPEDIENTE TÉCNICO	51
TABLA 17 TAREA DE INGENIERÍA. VISUALIZAR EXPEDIENTE TÉCNICO	51
TABLA 18 DESCRIPCIÓN DE LOS CASOS DE PRUEBA	53
TABLA 19 CASO DE PRUEBA DE ACEPTACIÓN. GESTIONAR EXPEDIENTE TÉCNICO	54
TABLA 20 CASO DE PRUEBA DE ACEPTACIÓN. VISUALIZAR EXPEDIENTE TÉCNICO	55
TABLA 21 CASO DE PRUEBA DE ACEPTACIÓN. GESTIONAR ROTURAS.....	56
TABLA 22 IMPORTANCIA DEL CONTROL DE DISPOSITIVOS.....	64
TABLA 23 NECESIDAD DE ACCESO AL ESTADO DE DISPOSITIVOS.....	65
TABLA 24 NECESIDAD DE SISTEMA INFORMÁTICO.....	66

Índice de figuras

FIGURA 1 CICLO DE VIDA DE LA METODOLOGÍA DE DESARROLLO DE SOFTWARE RUP	10
FIGURA 2 CICLO DE VIDA DE XP.....	11
FIGURA 3 MODELO CONCEPTUAL.....	24
FIGURA 4 ARQUITECTURA DEL MODELO - VISTA- CONTROLADOR	38
FIGURA 5 ARQUITECTURA DEL MODELO - VISTA- CONTROLADOR DEL SISTEMA SIGEXT	39
FIGURA 6 MODELO DE DATOS DEL SISTEMA SIGEXT.....	40
FIGURA 7 DIAGRAMA DE DESPLIEGUE DEL SISTEMA SIGEXT	45
FIGURA 8 RESULTADOS DE LAS NO CONFORMIDADES DETECTADAS.....	57

Introducción

El desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC), ha favorecido el cambio de los ambientes rutinarios de desarrollo de *software* por otros caracterizados por la innovación y la interacción permanente. Como nacimiento del Proyecto Futuro surge la Universidad de las Ciencias Informáticas (UCI) como casa de altos estudios, creada con el propósito de dar un impulso al desarrollo del *software* cubano y lograr productos de alta calidad en la industria de las nuevas tecnologías.

La UCI tiene como misión formar profesionales comprometidos con su Patria y altamente calificados en la rama de la Informática. Además de producir aplicaciones y servicios informáticos, a partir de la vinculación estudio-trabajo como modelo de formación y servir de soporte a la industria cubana de la informática. En la estructura organizativa de la UCI, se encuentra la Vicerrectoría de tecnología, área que se encarga de controlar los recursos tecnológicos existentes en la institución. En cursos anteriores al 2012-2013, esta vicerrectoría no contaba con un mecanismo que permitiera controlar con calidad las características de las estaciones de trabajo provocando posibles manifestaciones delictivas.

Para mitigar este riesgo, se crea en el año 2013, como instrumento de control el expediente técnico, que resume las principales características de los componentes internos y de los dispositivos de entrada y salida de la computadora. Actualmente en el Centro de Informática Industrial (CEDIN), el expediente técnico se gestiona de forma manual por el personal designado por los directivos de cada área, lo que provoca algunas deficiencias en la gestión de la información debido a que esta se encuentra dispersa en los diferentes departamentos que componen al centro, además no se cuenta con archivos para guardar y organizar estos expedientes provocando la pérdida de uno o varios de estos, por lo que existe la falta de evidencia provocada por el deterioro de los mismos.

Para la gestión de estos expedientes técnicos, el personal autorizado debe de desplazarse por cada una de las áreas que conforman al centro para recopilar a través de un software informático toda la información concerniente a las computadoras, como por ejemplo: la capacidad del disco duro, su modelo y número de serie, la cantidad de memoria RAM con la que cuenta la PC¹ entre otros datos de interés. En caso de que

¹ Computadora Personal o *Personal Computer* por sus siglas en inglés

se desee conocer alguna información específica, se debe revisar cada uno de los expedientes técnicos y tomar nota lo que provoca demora considerable en la información que se desea obtener.

Todo lo anterior provoca que el control de expedientes técnicos sea un proceso complejo, la búsqueda, el análisis y la actualización de la información sea lenta; el responsable de este proceso se sobrecarga al tener que desplazarse para mostrar la documentación además los directivos del centro no pueden realizar un estudio detallado de la situación de la tecnología existente en las estaciones de trabajo, lo que dificulta el control de los recursos tecnológicos y retarda el proceso de toma de decisiones en cuanto a la actualización de los componentes internos de las computadoras.

Teniendo en cuenta la problemática planteada anteriormente se define como **problema a resolver**: ¿Cómo contribuir a la gestión de la información de los expedientes técnicos en el Centro de Informática Industrial?

Para dar solución a la problemática planteada se asume como **objeto de estudio** los procesos de gestión de información.

En concordancia con lo anterior se propone como **objetivo general**: Desarrollar un sistema informático que contribuya a la gestión de la información de los expedientes técnicos en el Centro de Informática Industrial.

Enmarcado en el **campo de acción**: Sistema informático para la gestión de expedientes técnicos.

Tareas investigativas:

- Revisión bibliográfica de los sistemas de gestión de información para la elaboración del marco teórico de la investigación.
- Establecimiento de los fundamentos que deben sostener el sistema informático para la gestión de expedientes técnicos en el CEDIN.
- Definición de la metodología de desarrollo de software para guiar el proceso de desarrollo.
- Levantamiento de los requisitos de software que sustentan la solución.
- Definición de la arquitectura que tendrá la solución.
- Validación de la contribución dada a través de la introducción del sistema informático para la gestión de expedientes técnicos en el CEDIN.
- Valoración cualitativa de los resultados obtenidos en la validación del sistema informático para la gestión de expedientes técnicos en el CEDIN.

- Validación de los resultados mediante pruebas de liberación de calidad.

Durante el desarrollo de la investigación se combinaron varios métodos y técnicas de obtención de información. Como métodos teóricos se encuentran:

Métodos teóricos

Analítico - Sintético: Permitió el estudio y revisión de la bibliografía necesaria para tener un mejor entendimiento del problema a resolver con el objetivo de arribar a conclusiones que sustenten la necesidad de la investigación sobre los sistemas de gestión.

Histórico - Lógico: Se utilizó para analizar la evolución histórica de soluciones similares, las tendencias más recientes en la gestión de información, complementar las características necesarias y deseables para la solución que se propone.

Métodos empíricos

Observación: Se utilizó para realizar el estudio de las características y comportamientos de los usuarios de soluciones similares permitiendo la formulación global de la investigación.

Técnicas de obtención de información

Cuestionario: Se aplicó a especialistas para determinar requerimientos e impresiones que aporten valores y bases para la investigación.

La presente investigación está compuesta por tres capítulos, conclusiones, recomendaciones, bibliografía y anexo.

Capítulo 1: Fundamentación Teórica

En el presente capítulo se engloban los conceptos básicos relacionados con los sistemas de gestión de información, sus principales características, así como sus ventajas y desventajas. Además se analizan las principales herramientas, metodología y tecnologías para el desarrollo de aplicaciones informáticas y se fundamenta la selección de estas para el desarrollo de la solución propuesta.

Capítulo 2: Análisis y diseño del sistema

En el presente capítulo se presentan los fundamentos que sostienen al sistema propuesto, se describe sus funcionalidades y características esenciales. En el mismo se exponen los principales artefactos que rige la

metodología de desarrollo tales como Historias de Usuarios, Tareas de Ingeniería y las principales clases que componen el sistema representadas a través de las Tarjetas CRC, detallando la información del análisis y del diseño de la solución en cuestión. Se hace alusión a los conceptos que especifican al negocio y se utiliza como apoyo auxiliar un modelo conceptual que los refleja de forma sintetizada. Se elabora el diagrama de despliegue y se especifica la arquitectura que tendrá el sistema propuesto.

Capítulo 3: Implementación y pruebas

En este capítulo se aborda las tareas de implementación a través de las tareas de ingeniería. Se describe los estándares de codificación para el desarrollo de la solución propuesta a tener en cuenta, así como los resultados de las pruebas aplicadas para comprobar que las Historias de Usuario implementadas estén correctas a través de las pruebas de aceptación y el acta de liberación.

Capítulo I Fundamentación Teórica

Introducción

En el presente capítulo se abordan los principios fundamentales de la gestión de información. Se realiza un estudio de las herramientas que permiten obtener la información de los componentes internos y externos de las computadoras personales. Se define la metodología a utilizar y se seleccionan las diferentes tecnologías, herramientas y lenguajes de programación que favorecen la implementación de la solución propuesta.

1.1 Gestión de información

De acuerdo con Dans, la gestión de la información es un proceso que incluye operaciones como extracción, manipulación, tratamiento, depuración, conservación, acceso y/o colaboración de la información adquirida por una organización a través de diferentes fuentes y que gestiona el acceso y los derechos de los usuarios sobre la misma. (1)

La gestión de la información se vincula con la generación y la aplicación de estrategias, el establecimiento de políticas, así como con el desarrollo de una cultura organizacional y social dirigida al uso racional, efectivo y eficiente de la información en función de los objetivos y metas trazadas en materia de desempeño y de calidad.

1.1.1 Componentes fundamentales del proceso de gestión

Tomando los conceptos de Ana Mary, dentro de este proceso se mueven diferentes componentes conocidos como funciones de la gestión, estas son (2):

Planificar: es el proceso de establecer objetivos con el fin de alcanzar determinados resultados, así como identificar las acciones necesarias para alcanzarlos. Dentro de este concepto se contemplan un conjunto de decisiones o una selección de alternativas para el logro de tales resultados. Esta función se registra en diversos tipos de documentos: planes, programas, pronósticos y políticas.

Organizar: es el proceso de dividir el trabajo a realizar y de coordinar el logro de resultados que tienen un propósito común. Es la química de la organización donde se mezclan todos los elementos que interactúan entre sí a fin de obtener los resultados esperados. Es el acto de combinar habilidades, posibilidades técnicas, experiencias, recursos y todos los elementos que podrían convertirse en resultados.

Capítulo I Fundamentación Teórica

Dirigir: es el proceso de conducir y coordinar los esfuerzos laborales de las personas que integran una organización, ayudándolos a desarrollar tareas relevantes dentro de ella. La dirección es la función mediante la cual se ponen en marcha las actividades programadas. Comprende el compromiso de alcanzar un objetivo mediante el liderazgo de un grupo. La dirección ejerce una influencia notable en las personas para que trabajen voluntaria y entusiastamente para el logro de las metas colectivas de equipos y de la organización en su conjunto.

Controlar: es el proceso de supervisar las actividades y resultados, comparándolos con los objetivos y tomando las acciones correctivas, si son necesarias. Para ello se compara el desempeño con metas y planes, se muestran las desviaciones y al emprender medidas para corregir las desviaciones, se ayuda a asegurar el logro de los planes. Esta función comprende el establecimiento de normas de desempeño como base para la medida de los resultados, investigación, análisis, diseño, implantación y operación de los sistemas de información, registros contables y estadísticos, auditorías, inspecciones, controles y otros métodos de verificación directa.

1.2 Sistemas de gestión de información

Los principios de gestión existentes desde la antigüedad, han evolucionado y se han tornado indispensables para cualquier actividad humana. Cada día la sociedad enfrenta cambios más frecuentes y complejos que demandan una mayor eficiencia y optimización de los procesos. La filosofía de la gestión vincula los aspectos relativos al establecimiento de metas y objetivos con los aspectos referentes a la coordinación para alcanzar vías efectivas en la que la organización pueda manejar sus asuntos (2).

Los sistemas de gestión de información son un conjunto de elementos relacionados y ordenados según ciertas reglas que aporta al sistema objeto, es decir, a la organización a la que sirve y que marca sus directrices de funcionamiento, así como la información necesaria para el cumplimiento de sus fines; para ello debe recoger, procesar y almacenar datos procedentes tanto de la organización como de fuentes externas, con el propósito de facilitar su recuperación, elaboración y presentación.

Permiten la comprensión de cada organización desde un enfoque analítico, evaluador y creativo que permite develar las oportunidades que merezcan ser explotadas y contrarrestar las amenazas, además de establecer los factores que resulten críticos y buscar nuevas oportunidades (3) .

Un sistema de gestión de información realiza cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información. Este sistema ayuda a lograr los objetivos de una organización

Capítulo I Fundamentación Teórica

mediante una serie de estrategias que incluyen la optimización de procesos y el enfoque centrado en la gestión de la información (4). A su vez cumplen con tres objetivos básicos dentro de las organizaciones como la automatización de los procesos operativos, proporcionar información que sirva de apoyo al proceso de toma de decisiones y lograr ventajas competitivas a través de su uso e implantación.

La implementación de un sistema de gestión brinda ventajas como:

- Control efectivo de las actividades de la organización.
- Integración de nuevas tecnologías y herramientas.
- Disponibilidad de mayor y mejor información para los usuarios en tiempo real.
- Elimina la barrera de la distancia trabajando con un mismo sistema en puntos distantes.
- Disminuye errores, tiempo y recursos.

También presentan desventajas como:

- El tiempo que pueda tomar su implementación.
- La resistencia al cambio de los usuarios.
- Los problemas técnicos como fallas de *hardware* y/o *software*.
- La inadecuada implementación de las funcionalidades para el apoyo de actividades de la organización.

1.2.1 Sistemas de gestión de información para recursos tecnológicos

Los sistemas de gestión de información para recursos tecnológicos son herramientas que permiten almacenar información sobre las principales características de cada medio de cómputo. Dentro de estos sistemas se encuentran:

Aplicación web para el control de los dispositivos informáticos (GESPC)

Sistema informático creado en Camagüey, Cuba, para la administración de los dispositivos informáticos en la empresa de Servicios de Ingenieros. Programado en ASP.net y utiliza como sistema gestor de base de datos Microsoft SQL Server 2000.

Ventajas

- Registro de bajas de dispositivos y partes.
- Registro del estado técnico de cada dispositivo o parte.

Capítulo I Fundamentación Teórica

- Historial del estado técnico a través de la vida de cada dispositivo o parte.
- Resumen de dispositivos según estado técnico para cada área de la empresa.
- Historial de ubicación de cada dispositivo dentro de la empresa.

AIDA 64

Es una aplicación informática, de código cerrado, que sirve como herramienta de diagnóstico del sistema. Ofrece posibilidades de multilinguaje y está diseñado para ejecutarse sobre sistemas operativos, tanto de 32 bits como de 64 bits, basados en la familia de Microsoft Windows; siendo compatible con todas las versiones desde Windows 95 hasta Windows 7, incluyendo las versiones enfocadas hacia la línea de servidores. AIDA64 constituye uno de los programas más completos encargados de realizar un detallado análisis del sistema, el cual se muestra en una estructura de árbol. Incluye módulos de identificación del hardware de la computadora, módulos de diagnóstico y un conjunto completo de pruebas de rendimiento. Ofrece la posibilidad de acceder desde sus paneles a una amplia variedad de funciones específicas, proporcionando información sobre más de 120000 dispositivos.

EVEREST

Everest es una herramienta de diagnóstico que enumera todos los componentes de hardware del equipo separados en categorías. Solo funciona en Windows XP. La cantidad de información que Everest recopila es enorme. Para facilitar su asimilación, el sistema dispone de un apartado Resumen y un generador de informes exportables a HTML. La última versión de Everest, la 2.20, no es compatible con Windows Vista y Windows 7, puesto que se dejó de desarrollar en el año 2005. Sin embargo, Everest cumple su función en equipos que tengan instalado Windows XP.

Las principales causas por la que se desechan estas soluciones está dada porque las tecnologías utilizadas para su desarrollo son totalmente privativas. Solo son compatibles con el sistema operativo Windows en sus diferentes versiones. La última actualización de estas herramientas data del año 2005 por lo que para las tecnologías recientes no es funcional.

1.3 Metodologías de desarrollo de software

Las metodologías de desarrollo de software se definen como un conjunto de herramientas, técnicas, procedimientos y soporte documental para el diseño de sistemas de información. La metodología de desarrollo de software se divide en dos grandes grupos: las metodologías tradicionales o pesadas que se

Capítulo I Fundamentación Teórica

centran en el control del proceso mediante una rigurosa definición de roles, actividades, artefactos, herramientas y documentación detallada para el modelado donde el resultado final sería un proceso de desarrollo más complejo (5), entre las metodologías tradicionales se encuentran RUP y MSF siendo RUP (Proceso Unificado de Desarrollo) la metodología más utilizada.

Por otra parte las metodologías ágiles que se basan en el desarrollo iterativo e incremental, teniendo presente los cambios y respondiendo a estos mediante la colaboración de un grupo de desarrolladores auto-organizados y multidisciplinares (6), dentro de este grupo se encuentra la metodología Programación Extrema (XP).

Metodología de desarrollo RUP

Se caracteriza por ser dirigida por casos de uso, estar centrada en la arquitectura y ser iterativa e incremental. Entre sus principios se encuentran adaptar el proceso a la necesidad del cliente, equilibrar prioridades, demostrar valor iterativamente, fomentar la colaboración entre equipos, elevar el nivel de abstracción y enfocarse en la calidad. Sin embargo, RUP presenta algunos inconvenientes para su aplicación en algunos proyectos como son: límite de tiempo demasiado ajustado, la documentación al ser tan exhaustiva hace mucho énfasis en la planificación, no siempre se puede llevar a cabo la implementación del *software* de acuerdo con su planificación pues aparecen cambios los cuales solo pueden ser percibidos en el momento de la codificación.

RUP cuenta con cuatro fases de desarrollo (7) como se muestra en la figura 1

- Conceptualización (Concepción o Inicio): Se describe el negocio y se delimita el proyecto describiendo sus alcances con la identificación de los casos de uso del sistema.
- Elaboración: Se define la arquitectura del sistema y se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen. A pesar de que se desarrolla a profundidad una parte del sistema, las decisiones sobre la arquitectura se hacen sobre la base de la comprensión del sistema completo y los requerimientos (funcionales y no funcionales) identificados de acuerdo al alcance definido.
- Construcción: Se obtiene un producto listo para la utilización, que está documentado y tiene manual de usuario. Se obtienen tantas versiones de entrega como determinan las pruebas.
- Transición: Con la liberación, ya el producto está listo para su instalación en condiciones reales. Puede implicar la reparación de errores como parte del proceso de soporte.

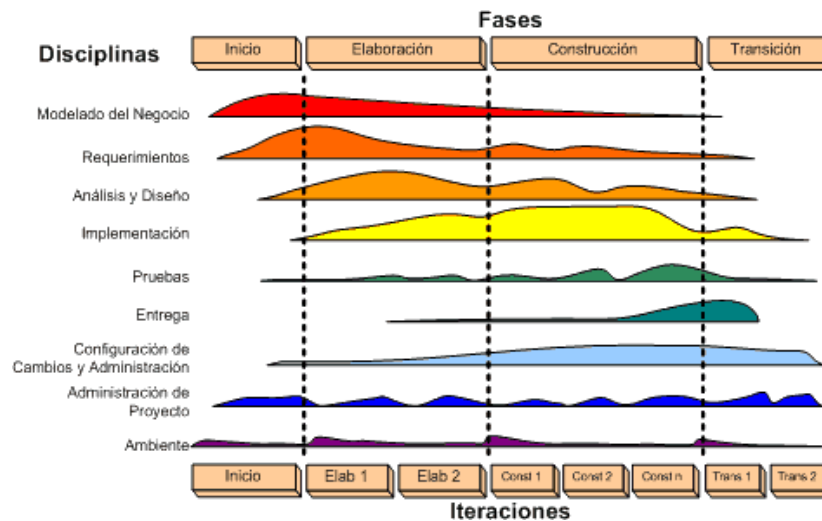


FIGURA 1 CICLO DE VIDA DE LA METODOLOGÍA DE DESARROLLO DE SOFTWARE RUP

Metodología de desarrollo XP

Metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de *software*, promueve el trabajo en equipo, se preocupa por el aprendizaje de los desarrolladores, y propicia un buen clima de trabajo. Se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios (8).

XP propone cuatro fases de desarrollo que son:

- *Planificación y estimación*: Se analiza el negocio, se identifican los requisitos y se establecen los planes de entrega.
- *Análisis y diseño*: Se diseña el sistema informático, se establece la arquitectura y se modelan las clases del sistema.
- *Implementación*: Se desarrolla el sistema a partir de las clases modeladas.
- *Pruebas*: Se prueba la solución y se verifica el correcto funcionamiento del sistema. A partir de las pruebas y cuando se valida la solución, comienza una nueva iteración con nuevas funcionalidades.

Valores de XP

- **Comunicación**: es lo más importante dentro de un equipo de desarrollo de software, cuando aparecen problemas durante el desarrollo es probable que alguien del equipo conozca la solución.

Capítulo I Fundamentación Teórica

- **Simplicidad:** hay que verla dentro del contexto del proyecto, al mejorar la comunicación se ayuda a ganar en simplicidad para eliminar requerimientos innecesarios.
- **Retroalimentación:** cuando se habla de desarrollo de software o de los requerimientos de un sistema, los cambios son inevitables. Es una parte crítica de la comunicación y contribuye a la simplicidad.
- **Coraje:** en conjunto con los otros valores es poderoso, el coraje para descartar soluciones fallidas y buscar nuevas, lleva a la simplicidad para buscar respuestas concretas y reales.

El proceso de desarrollo de XP sigue un ciclo, el cual consiste en los siguientes pasos (9) como se muestra en la figura 2

1. El cliente define el valor del negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor del negocio.
5. Vuelve al paso uno.



FIGURA 2 CICLO DE VIDA DE XP

Elección de la metodología de desarrollo de software

Se selecciona XP como metodología de desarrollo de *software* porque suprime la extensa documentación que proponen las metodologías tradicionales y asegura mediante sus principales valores, que el desarrollador sea versátil y conozca las necesidades elementales del cliente siendo este último parte del equipo de desarrollo, la documentación que se genera es suficiente y comprensible por desarrolladores y clientes, es ideal para proyectos donde el ciclo de vida es relativamente corto por lo que puede adaptarse a la solución propuesta ya que tiene un tiempo de duración de 6 meses, los cambios a los requerimientos son

bienvenidos, aún en fases tardías del desarrollo, se centra fundamentalmente en entregar un producto de *software* en el menor tiempo posible.

1.4 Herramientas y tecnologías de desarrollo

Las herramientas y tecnologías escogidas para el desarrollo de un proyecto deben ser seleccionadas cautelosamente, ya que pueden suponer el fracaso de éste o pueden aumentar su complejidad, para lo cual se deben conocer cuáles son las distintas alternativas y las necesidades del proyecto. A continuación se abordarán algunas de las tecnologías y herramientas más frecuentes, las cuales se utilizarán en el desarrollo del sistema propuesto.

Lenguaje de modelado

El Lenguaje Unificado de Modelado (UML), permite visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de *software*. Permite a los creadores de sistemas generar diseños que capturen sus ideas en una forma convencional y fácil de comprender para comunicarlas a otras personas. Este lenguaje dispone de reglas para combinar tales elementos y permite la modelación de sistemas con tecnología orientada a objetos. Los diagramas son entes importantes de UML, cuya finalidad es presentar diversas perspectivas de un sistema, a las cuales se les conoce como modelo. El modelo gráfico de UML tiene un vocabulario en el que se identifican: elementos, relaciones y diagramas (10).

Modelado de proceso de negocio

Notación de Modelado de Procesos de Negocio (BPMN) es utilizada como un estándar para la representación de procesos de negocios, independientemente de la metodología empleada para automatizar dichos procesos. Esta notación ha sido diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes en las actividades. Proporciona un lenguaje común para que las partes involucradas en el análisis de negocios (cliente y desarrollador), puedan intercambiar sobre un mismo modelo de forma clara, precisa y eficiente (11).

Elección del lenguaje de modelado

Por lo descrito anteriormente se selecciona UML como lenguaje de modelado para especificar, visualizar, construir y documentar los artefactos del sistema, es un lenguaje estándar, fácil de aprender, y ofrece una amplia variedad de diagramas para mostrar el sistema desde varias perspectivas. Está especialmente diseñado para apoyar un estilo de desarrollo iterativo e incremental y presenta tecnología orientada a

objetos. Tiene una notación gráfica expresiva que permite representar en mayor o menor medida todas las fases de un proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos, hasta la implementación y configuración con los diagramas de despliegue.

1.4.1 Herramientas CASE

Las herramientas CASE (Ingeniería de Software Asistida por Computadoras) fueron diseñadas para aumentar la productividad en el desarrollo de *software*, reduciendo el costo de las mismas en términos de tiempo y dinero, reemplazan al lápiz y al papel por la computadora para transformar la actividad de creación de aplicaciones informáticas en un proceso automatizado. Esta tecnología contribuye a elevar la calidad en los sistemas (12). A continuación se realiza un análisis de las herramientas CASE más utilizadas:

Rational Rose Enterprise

IBM Rational Rose Enterprise proporciona un lenguaje de modelado común que permite crear con mayor velocidad aplicaciones informáticas de calidad, es compatible con UML, permite la ingeniería directa e inversa. Incluye un complemento de modelado web que proporciona el modelado para el desarrollo de aplicaciones web. Esta herramienta resulta de gran utilidad para los desarrolladores de proyectos por lograr cubrir todo el ciclo de vida del proceso desde su fase inicial hasta que termina (12).

Rational Rose posibilita establecer una trazabilidad entre los modelos. Cada rol tiene su propia vista de arquitectura, utilizando un lenguaje común para comprender y comunicar la estructura y funcionalidad del sistema en construcción (13). Sin embargo, cuando estas características la convierten en una herramienta eficaz, su particularidad de ser propietaria, la hace ser rechazada por muchas empresas y desarrolladores de *software*.

Visual Paradigm

Visual Paradigm en su versión 8.0 es una herramienta para diseñar los diagramas, integrar y desplegar las funcionalidades críticas de la aplicación a través de UML. Brinda la posibilidad de crear diagramas de clases del negocio, de datos y otros artefactos que tributan a la documentación del sistema. Posee amplias posibilidades de integración con IDE²de desarrollo, genera códigos a partir de diagramas y viceversa. Propicia el desarrollo de *software* de manera acelerada garantizando la calidad de los productos. Es utilizado para la representación *UML* de los modelos descriptivos del proceso de desarrollo del producto (10).

² IDE: Entorno de Desarrollo Integrado o *Integrated Development Environment* por sus siglas en inglés

Elección de la herramienta CASE

Se decide utilizar Visual Paradigm en su versión 8.0 por su robustez, usabilidad y portabilidad. Cuenta con licencia gratuita para su utilización y soporta el ciclo completo del proceso de desarrollo de *software*. Es multiplataforma. Utiliza UML como lenguaje de modelado que tiene como propósito visualizar, especificar, construir y documentar proyectos de *software*. Posee la capacidad de realizar ingeniería directa e inversa.

1.4.2 Framework de desarrollo

Un framework³ es una estructura de soporte definida a partir de la cual un proyecto de software puede ser organizado y desarrollado. Este incluye soporte de programas, bibliotecas y un lenguaje interpretado para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Entre los objetivos principales que persigue un framework se encuentran acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

Existen diferentes framework de desarrollo entre los cuales se encuentra Symfony y Yii⁴. A continuación se exponen las principales características de cada uno de ellos.

Symfony

Symfony en su versión 2.0 es un framework diseñado para optimizar el desarrollo de aplicaciones web. Se encuentra desarrollado completamente con PHP5. Ha sido utilizado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel; separa la lógica del negocio, la lógica del servidor y la presentación de la aplicación (13).

Reduce el tiempo de desarrollo de una aplicación web compleja. Automatiza las tareas más comunes y reutiliza el código cada vez que se crea una nueva aplicación web. Tiene compatibilidad con la mayoría de los gestores de base de datos. Es multiplataforma, sin embargo, su mapeo de objeto relacional para conectarse a las bases de datos es propenso a lentitud, pues conlleva a gastos innecesarios en la generación y configuración del código. Su curva de aprendizaje es elevada (13).

Yii

³ Framework: Marco de trabajo. Es una estructura de software de componentes personalizables e intercambiables para el desarrollo de una aplicación.

⁴ Yii: Sí, lo es o *Yes, It, Is* por sus siglas en inglés

Yii en su versión 1.1.4 es un framework PHP basado en componentes de alto rendimiento para desarrollar aplicaciones web de gran escala. El mismo permite la máxima reutilización en la programación web y puede acelerar el proceso de desarrollo.

Entre sus principales características se encuentran el uso del patrón de diseño Modelo Vista Controlador (MVC). La integración con bibliotecas de Javascript. Las entradas de formularios y las validaciones. Las extensiones de Ajax, como autocompletado de campos de texto. El soporte de autenticación incorporado y la autorización de control de acceso basado en roles jerárquicos. La personalización de aspectos y temas. Las medidas de seguridad incluyen la prevención de secuencias de comandos en sitios cruzados, prevención de falsificación de petición en sitios cruzados y prevención de la manipulación de cookies. La generación automática de código para el esqueleto de la aplicación como aplicaciones CRUD⁵. Contiene un editor WYSIWYG⁶ (14).

Elección del framework de desarrollo

Se selecciona como framework de desarrollo Yii, debido a que posee una extensa documentación. Es guiado sobre el patrón arquitectónico MVC. Utiliza un motor de plantilla para PHP y está diseñado para principiantes. Posee una alta popularidad a nivel internacional. La curva de aprendizaje de este framework es baja por lo que permite una mejor comprensión.

1.4.3 Entorno de desarrollo

Un entorno de desarrollo integrado, conocido también como IDE, es un programa informático compuesto por un conjunto de herramientas de programación que facilitan el desarrollo de aplicaciones web, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, Python, Java, PHP, C#, Delphi y Visual Basic (15).

Aptana Studio

Aptana Studio en su versión 3.0 es un entorno de desarrollo integrado gratuito basado en Eclipse y desarrollado por Aptana, Inc. Su uso es básicamente para la elaboración de aplicaciones web dinámicas que empleen PHP, es multiplataforma y de código abierto, es una distribución focalizada en el desarrollo

⁵ CRUD: acrónimo de Crear, Obtener, Actualizar y Borrar o *Create, Read, Update and Delete* por sus siglas en inglés.

⁶ WYSIWYG: Lo que ves es lo que obtiene o *What You See Is What You Get* en inglés.

Capítulo I Fundamentación Teórica

web, con soporte a HTML, CSS y JavaScript. Aptana Studio contiene también información de soporte para los principales navegadores web: Internet Explorer, Firefox, Opera, Netscape y Safari (16). Entre sus principales características se encuentran las ayudas visuales para la escritura de scripts⁷ en diversos lenguajes, como coloreado y auto escritura del código, ayudas contextuales de referencia y visualización de errores de sintaxis a medida que se escribe. El asistente de código para HTML y Javascript. La extensión de funcionalidad mediante macros y acciones.

Zend Studio

Zend Studio en su versión 9.0 es un completo entorno de desarrollo integrado para el lenguaje de programación PHP. Está escrito en Java y está disponible para las plataformas Microsoft Windows, Mac OS X y GNU/Linux.

Junto con su contraparte Zend Server (antes llamado *Zend Platform*), son la propuesta de *Zend Technologies* para el desarrollo de aplicaciones web utilizando PHP, actuando *Zend Studio* como la parte cliente y *Zend Server* como la parte servidora. Se trata en ambos casos de software comercial, lo cual contrasta con el hecho de que PHP es software libre (17). Entre sus principales características se encuentran que no requiere la instalación previa del entorno de ejecución de Java. Brinda soporte para PHP 4 y PHP 5 así como el resaltado de sintaxis, autocompletado de código, ayuda de código y lista de parámetros de funciones y métodos de clase. Ofrece soporte mediante PHPDocs y PHPDocumentor integrado. Posee soporte para gestión de grandes proyectos de desarrollo y para el control de versiones.

PHP Storm

PHP Storm en su versión 8.0 es un IDE de desarrollo que soporta la base de todas las nuevas tecnologías incluyendo Vagrant, Drupal, Zend 2, Web Components y TypeScript. Tiene soporte para la última versión de PHP en su versión 5.5 y mejoras en PHPDoc, así como una nueva forma de inspeccionar código. También provee colores en la sintaxis de PHP, la cual puede ser definida por el usuario, con lo que se mejora la lectura del código. Hay cuatro colores diferentes: para declaraciones de funciones y métodos, para clases e interfaces, campos e instancias de clases de campos (18).

Elección del entorno de desarrollo

⁷ scripts: es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano.

Se selecciona PHP Storm en su versión 8.0 por las características relevantes siguientes: Tiene un Step-Into, lo cual es una característica que permite elegir la función que se quiere analizar paso a paso mientras se depura. Tiene un plugin para Drupal 6 y 7. Introduce nuevas herramientas con soporte a tecnologías web, incluyendo tres alternativas de plantillas en JavaScript: EJS, Handlebars y Mustache (18). Su curva de aprendizaje es baja.

1.4.4 Lenguajes de programación

Un lenguaje de programación es un idioma diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Definido además como un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Es utilizado para controlar el comportamiento físico y lógico de una computadora (19).

Hipertext Pre-processor (PHP 5)

PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. PHP no necesita ser compilado para ejecutarse. Tiene como ventaja su facilidad de aprendizaje, es un lenguaje rápido, tiene licencia de código abierto, es multiplataforma, tiene una alta variedad de funciones. Tiene capacidad de conexión con la mayoría de los manejadores de base de datos como MySQL, PostgreSQL, Oracle y MS SQL Server. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas (20).

Hyper Text Markup Language 5 (HTML 5)

HTML, siglas de *Hyper Text Markup Language* (lenguaje de marcado de hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web, utilizado normalmente en la *www* (*World Wide Web*). Fue creado en 1986 por el físico nuclear Tim Berners-Lee. HTML se escribe en forma de etiquetas, rodeadas por corchetes angulares (<,>). El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos *Windows* (Bloc de notas), o el que ofrece Microsoft *Word* (21).

El conjunto de etiquetas que se creen, se deben guardar con la extensión .htm o.html. Estos documentos pueden ser mostrados por los visores de páginas web en internet, como Netscape *Browser*, Mozilla Firefox, Mosaic, Opera y Microsoft Internet Explorer. En el caso HTML5, es la quinta revisión importante del lenguaje básico, especifica dos variantes de sintaxis para HTML: un «clásico» HTML (text/html), la variante conocida como HTML5 y una variante XHTML conocida como sintaxis

XHTML5 que deberá ser servida como XML. Esta es la primera vez que HTML y XHTML se han desarrollado en paralelo (22).

JavaScript

JavaScript es un lenguaje de programación que permite a los desarrolladores crear acciones en sus páginas web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos. Entre sus principales características se puede mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas y cargas de páginas. El código JavaScript se puede encontrar dentro de las etiquetas `<body></body>` de las páginas web. JavaScript es soportado por la mayoría de los navegadores como Internet Explorer, Netscape, Opera y Mozilla Firefox (23).

Cascading Style Sheets (CSS)

CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML y por extensión en XHTML. Consiste en separar la estructura de un documento de su presentación. Una hoja de estilos CSS se fundamenta en una serie de reglas. Cada regla posee uno o más selectores y un bloque de estilos con los estilos a aplicar para los elementos del documento que cumplan con el selector que les precede. Cada bloque de estilos se define entre llaves, y está formado por una o varias declaraciones de estilo con el formato propiedad **valor** (22).

Algunas ventajas de utilizar CSS son:

- Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Separación del contenido de la presentación, lo que facilita al creador, diseñador, usuario o dispositivo electrónico que muestre la página, la modificación de la visualización del documento sin alterar el contenido del mismo.

1.4.5 Servidor web

Un servidor web es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de internet. El servidor web se encarga de contestar estas peticiones de forma adecuada, entregando como resultado una página web o información

Capítulo I Fundamentación Teórica

de todo tipo de acuerdo a los comandos solicitados. La principal función de un servidor web es almacenar los archivos de un sitio y emitirlos por internet para poder ser visitado por los usuarios.

Servidor web XAMPP

XAMPP en su versión 1.7.7 es un servidor independiente de plataforma, *software* libre, que consiste principalmente en la base de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script: PHP y Perl. El programa está liberado bajo la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux conocido como LAMP, Solaris y MacOS X (24).

1.4.6 Sistema gestor de base de datos

Un sistema gestor de base de datos (SGBD) es un conjunto de programas que permiten crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad. El propósito general de los SGBD es manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante para una organización. Estos sistemas permiten describir los elementos de datos con su estructura, sus interrelaciones y sus validaciones. Entre los más utilizados se encuentran Oracle, PostgreSQL y MySQL (25).

PostgresSQL

PostgresSQL en su versión 9.1 es un SGBD que comenzó como un proyecto en la Universidad Bekerley de California. Este proyecto sigue actualmente un activo proceso de desarrollo a nivel mundial por un equipo de desarrolladores y contribuidores de código abierto (26). Puede funcionar en múltiples plataformas y a partir de la versión 8.0, también en Windows de forma nativa. Para las versiones anteriores existen versiones binarias para este sistema operativo, pero no tienen respaldo oficial. Sus ventajas más notables son: Está publicado bajo licencia *BSD* (Berkeley Software Distribution por sus siglas en inglés). Es orientado a objetos. Está diseñado para ambientes de alto volumen. Aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas además de soportar la integridad referencial, que se utiliza para asegurar la validez de los datos de la base de datos.

Oracle

Capítulo I Fundamentación Teórica

Es un ORDBMS⁸ desarrollado por Oracle Corporation. Se considera como uno de los sistemas de bases de datos más completos, entre sus principales características se destaca el soporte de transacciones. La estabilidad, escalabilidad y el soporte multiplataforma. Su dominio en el mercado de servidores empresariales ha sido casi total hasta hace poco, recientemente sufre la competencia del Microsoft SQL Server de Microsoft y de la oferta de otros RDBMS⁹ con licencia libre como PostgreSQL, MySQL o Firebird.

MySQL

MySQL en su versión 5.5.16 es un sistema de gestión de bases de datos relacional multiplataforma. Su diseño multihilo le permite soportar una gran carga de información de forma muy eficiente. Está diseñado para entornos de producción críticos, con alta carga de trabajo así como para integrarse en aplicaciones para ser distribuido (27).

MySQL es una marca registrada de MySQL AB, aunque tiene una doble licencia. Los usuarios pueden elegir entre usar el producto MySQL como un sistema Open Source bajo los términos de la licencia GNU (del inglés General Public License) o pueden adquirir una licencia comercial estándar de MySQL AB. Se encuentra programado en C y en C++, probado con un amplio rango de compiladores diferentes, multiplataforma, proporciona sistemas de almacenamiento transaccionales y no transaccionales. El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor (27).

También es disponible como biblioteca y puede ser incrustado en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible. Los clientes se pueden conectar con el servidor MySQL usando sockets TCP/IP en cualquier plataforma (27).

Elección del sistema gestor de base de datos

Una vez analizado algunos de los gestores de base de datos más populares, se decide seleccionar MySQL en su versión 5.5.16 como gestor de base de datos debido a que es más ligero, es potente, viene integrado en los servidores libres. Posee mayor rendimiento, mejores utilidades de administración (backup y recuperación de errores), mejor control de acceso así como una mejor integración con PHP. No hay límites en el tamaño de los registros. Permite además la modificación o adición de campos a una tabla en caliente.

⁸ ORDBMS: Sistema de Gestión de Base de Datos Objeto-Relacional o *Object-Relational Data Base Management System* por sus siglas en inglés.

⁹ RDBMS: Sistema Gestor de Base de Datos Relacionales o *Relational Database Management System* por sus siglas en inglés.

Conclusiones parciales

El análisis bibliográfico permitió comprender que es imprescindible el uso de los sistemas de gestión, debido al avance de las tecnologías de la información y las comunicaciones en el mundo. Se analizaron las metodologías de desarrollo de *software* más utilizadas, se seleccionó a XP como metodología de desarrollo debido a la corta duración de la investigación para desarrollar un sistema informático. Se realizó un estudio de las principales tecnologías y herramientas llegándose a seleccionar las más adecuadas para el desarrollo del sistema.

Capítulo II Análisis y diseño del sistema

Introducción

En el presente capítulo se estudia las consideraciones del negocio. Se elabora un modelo conceptual, se realiza el levantamiento de los requisitos, con el fin de dar una solución factible al problema que se investiga. Se define los requisitos funcionales y no funcionales. Se desarrollan los artefactos principales de la metodología de desarrollo seleccionada. Se define la arquitectura que tendrá la solución propuesta y se muestra un modelo de despliegue.

2.1 Consideraciones del negocio

Un expediente técnico es un resumen de las principales características de los componentes internos y de los dispositivos de entrada y salida de los medios de cómputo. El expediente reúne los elementos que permiten caracterizar un medio computacional, el área al que pertenece, el local donde se encuentra ubicado, su responsable y los cambios que ha sufrido durante su tiempo de vida. El proceso de gestión de los expedientes técnicos lo realiza el personal designado por el responsable de cada área y registra los principales procesos que se lleva a cabo con estos como son: proceso de intercambio, rotura y traslado de la PC. En tal sentido, los expedientes técnicos interactúan con los responsables de área, de locales y de cada responsable de su puesto de trabajo.

El proceso de gestión tiene las siguientes fases:

Alta de medios: Cuando una PC es asignada a un área en forma de alta o traslado, se debe confeccionar su expediente técnico teniendo en cuenta sus propiedades internas y externas, el nuevo responsable y su ubicación dentro del área.

Modificaciones: Cada modificación que reciba la PC así como sus periféricos debe ser anotada en el expediente técnico. Las modificaciones deben estar aprobadas por el responsable del área y el cambio dentro del expediente debe ser autorizado por el personal responsable. Como contra partida, el encargado de servicios técnicos debe llevar un registro de control sobre los cambios realizados sobre cualquier medio de cómputo.

Capítulo II Análisis y diseño del sistema

Revisiones y auditorías: Los expedientes técnicos pueden ser revisados por el responsable del área o por el personal autorizado en el instante de tiempo en que se necesite consultar el estado de los medios, por ejemplo: en auditorías de seguridad informáticas, cierres de locales, verificación de submayores de activos y entrega de partes informativos a la dirección, la información de los expedientes técnicos debe ser consultada y se certifica como la fuente más verídica para conocer sobre procesos de traslados, intercambio, roturas, altas y modificación de los medios de cómputos.

Durante este proceso se han de tener en cuenta las siguientes reglas:

1. La modificación de los expedientes técnicos deben estar autorizados por los responsables de área y realizada solamente por el personal designado para esta tarea.
2. El personal de servicios técnicos no puede realizar ningún trámite sobre un medio de cómputo sin antes comprobar su estado a partir de la revisión de su expediente técnico.
3. El personal externo no puede tener acceso a los expedientes, solamente el personal designado debe tener acceso a este, y será el encargado de disponer de su accesibilidad durante los trámites con los medios.
4. La confección del expediente técnico estará hecha a través de un documento oficial emitido por la Vicerrectoría de tecnología.

Dada la cantidad de actores que involucra el negocio representado anteriormente, sus restricciones, la cantidad de medios por cada local y la importancia de la actualización pertinente de cada cambio sobre un medio, conllevan a que el control de los expedientes técnicos necesita ser automatizado mediante soluciones de *software* que agilicen este proceso en el CEDIN.

2.2 Modelo conceptual

Un modelo conceptual es una representación visual de las clases conceptuales u objetos del mundo real en un dominio de interés (28). La figura 3 muestra el modelo conceptual de la problemática planteada.

Capítulo II Análisis y diseño del sistema

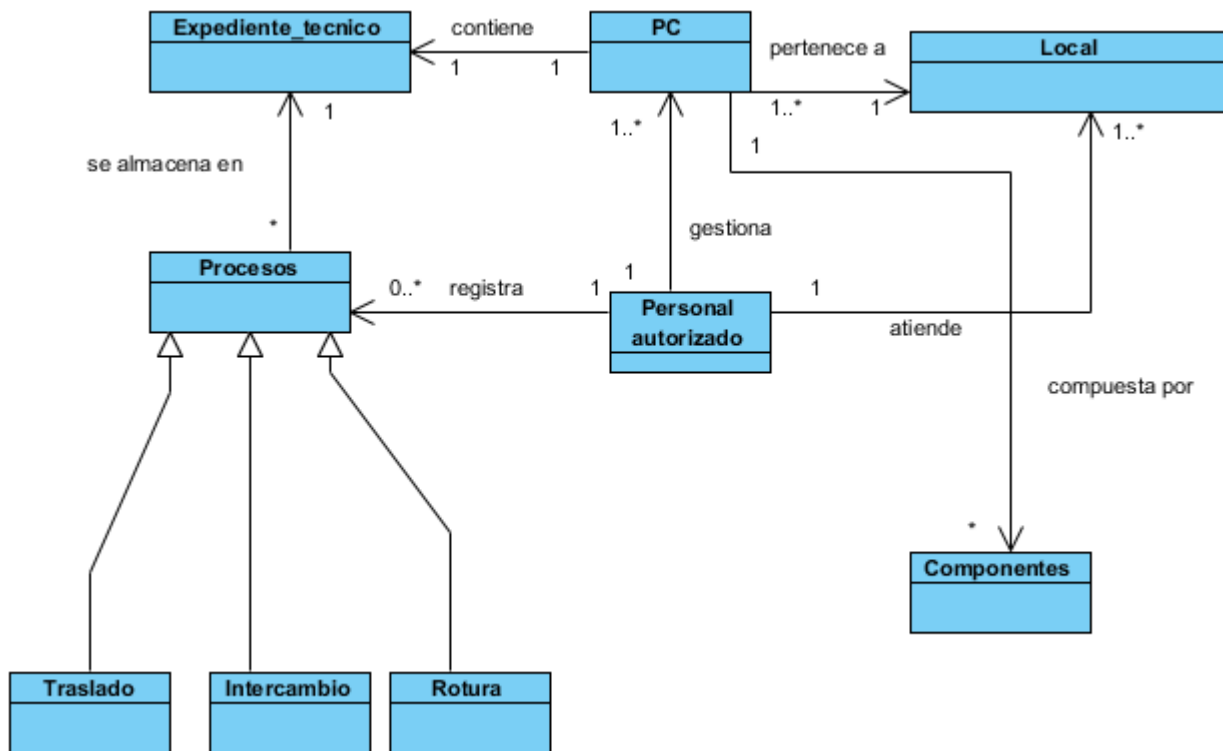


FIGURA 3 MODELO CONCEPTUAL

Descripción de las entidades del Modelo Conceptual:

Personal autorizado: Personal encargado de manipular los expedientes técnicos.

Expediente técnico: Documento que resume las principales características de cada estación de trabajo.

PC: Equipo de cómputo.

Local: Lugar donde se encuentra ubicada cada PC.

Componentes: Partes que integran la estación de trabajo.

Procesos: Principales procesos que se realizan en el expediente técnico.

- *Rotura:* Proceso que se registra en el expediente técnico cuando se identifica una rotura en una PC.
- *Intercambio:* Proceso que se registra en el expediente técnico cuando se realiza un intercambio.
- *Traslado:* Proceso que se registra en el expediente técnico cuando se realiza un traslado.

Capítulo II Análisis y diseño del sistema

2.3 Propuesta del sistema

Se propone un sistema llamado SIGEXT, utilizando para su desarrollo herramientas libres y que sean capaz de brindar todas las funcionalidades requeridas para la gestión de expedientes técnicos. Con SIGEXT se logrará informatizar los procesos complejos que se llevan a cabo en la gestión de expedientes técnicos. Cuenta con un entorno de fácil uso en cuanto a la manipulación de los principales módulos que intervienen en el sistema. Este sistema poseerá funcionalidades que le permitirá al usuario interactuar fácilmente con el sistema y obtener en un menor tiempo resultados que le permitan tomar pertinentes decisiones. Se logrará a través de éste gestionar toda la información de los expedientes técnicos. SIGEXT contará con una gestión de roles lo que permitirá tener una mayor seguridad en el control de la información, lo que garantizará que solo el personal autorizado tenga acceso a este.

Roles del sistema

Administrador: Encargado de gestionar toda la información del sistema. Establece los permisos pertinentes para los distintos tipos de usuarios.

Responsable del área: Encargo de gestionar toda la información referente a su departamento al cual es asignado por el sistema.

Responsable del local: Encargado de gestionar la información referente a su laboratorio al cual es asignado por el sistema.

Usuario: Visualiza la información de los expedientes técnicos asignado bajo su responsabilidad.

2.4 Fase de exploración y planificación

En la fase de exploración y planificación los clientes describen sus necesidades en las Historias de Usuario que son los requisitos funcionales del sistema y establecen las prioridades de cada una. Al mismo tiempo, se define el tiempo de desarrollo de cada Historia de Usuario y se familiariza con las tecnologías, herramientas y prácticas que se utilizarán en el desarrollo del sistema SIGEXT. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema. La fase de exploración toma poco tiempo, dependiendo de la capacidad del programador con la tecnología y el alcance del proyecto (29).

Requisitos funcionales

Los requisitos funcionales son aquellos que describen qué debe hacer el sistema, desde el punto de vista de las necesidades del usuario, son capacidades o condiciones que debe cumplir el sistema y que están

Capítulo II Análisis y diseño del sistema

fuertemente ligados a las opciones del programa. A continuación se muestran los principales requisitos funcionales del sistema.

RF -1: Autenticar usuario en el sistema SIGEXT.

RF-2: Gestionar roles de usuario en el sistema SIGEXT.

- 2.1 Modificar rol de usuario en el sistema SIGEXT.
- 2.2 Buscar usuario partiendo de un criterio especificado.
- 2.3 Eliminar rol de usuario del sistema SIGEXT.

RF-3: Gestionar expediente técnico

- 3.1 Registrar expediente técnico en el sistema SIGEXT.
- 3.2 Modificar expediente técnico registrado en el sistema SIGEXT.
- 3.3 Listar expediente técnico registrado en el sistema SIGEXT.
- 3.4 Eliminar expediente técnico registrado en el sistema SIGEXT.

RF-4: Visualizar expediente técnico registrado en el sistema SIGEXT.

RF-5: Gestionar Local

- 5.1 Registrar local en el sistema SIGEXT.
- 5.2 Modificar local registrado en el sistema SIGEXT.
- 5.3 Listar local registrado en el sistema SIGEXT.
- 5.4 Eliminar local registrado en el sistema SIGEXT.

Requisitos no funcionales

Los requerimientos no funcionales son requisitos que imponen restricciones en el diseño o la implementación. Son propiedades o cualidades que el producto debe cumplir (10). Sin embargo estos requisitos no definen el éxito del producto, pero influyen considerablemente en la evaluación del mismo. Teniendo en cuenta las características del sistema se definieron los siguientes requerimientos no funcionales:

RFN-1: Usabilidad

Capítulo II *Análisis y diseño del sistema*

El sistema debe contar con una adecuada organización de los módulos del sistema permitiendo la fácil interacción con el usuario. Contará con secciones tabulares lo que permitirá tener una mejor organización de la información al registrar un expediente técnico.

RFN-2: *Seguridad*

El sistema debe responder a la política de acceso basada en roles, permisos de los usuarios para las operaciones y permisos funcionales sobre las operaciones. El acceso al sistema se realizará a través de protocolo seguro (HTTPS).

➤ *Confidencialidad*

Existencia de distintos roles que establezcan que la información sólo sea vista por aquellos usuarios que posean los privilegios suficientes; restringir la ejecución de acciones a usuarios sin credenciales que intenten acceder a las mismas y la verificación de que el usuario esté autenticado antes de realizar cualquier operación.

RFN-3 *Copia de Respaldo*

Se debe permitir la realización de copias de respaldo hacia otros dispositivos de almacenamiento de forma automática a partir de tareas programadas periódicas para prevenir pérdidas de información. Las copias de seguridad se realizarán cada siete días.

RFN-4: *Interfaces externas*

La interfaz debe ser simple, con una navegación por pestañas donde se puedan identificar de forma rápida las acciones a llevar a cabo por los botones presentes. De esta manera se logra una navegación efectiva en la aplicación. Los mensajes deben ser claros, sin ambigüedades y con buena ortografía.

RFN-4.1: *Software:*

- Servidor:
 - Sistema Operativo Linux, Windows
 - Servidor web XAMPP 1.7.7 o superior.
 - Servidor de Base de Datos MySQL 5.5.16.
 - PHP 5.4
- Cliente:

Capítulo II Análisis y diseño del sistema

- Navegador web:
 - Mozilla Firefox 16.0 o superior.
 - Google Chrome 16.0 o superior.

RFN-4.2: Hardware:

- Servidor:
 - Procesador Core i3 a 3.4 GHz.
 - Memoria RAM: 4 GB.
 - Disco duro: 160 Gb.
 - Tarjeta de Red (100 Mb/s).
- Cliente
 - Procesador Pentium IV a 1.8 GHz.
 - Memoria RAM: 512 MB.
 - Tarjeta de Red (100 Mb/s).

Historias de Usuario

Las Historias de Usuario se escriben en el lenguaje del cliente, representa un requisito que se debe satisfacer con la implementación del sistema y debe ser lo suficientemente sencilla para que el programador pueda saber que va a implementar. Si la Historia de Usuario cuando el cliente la escribe el programador entiende que no es lo completamente sencilla como para implementarla como una funcionalidad, entonces se divide en dos o más Historias de Usuario (30). En la tabla 1 se muestra un ejemplo de una de las Historias de Usuarios del sistema.

TABLA 1 EJEMPLO DE UNA HISTORIA DE USUARIO.

HISTORIA DE USUARIO			
Orden	HU_09	Nombre	Visualizar expediente técnico
Riesgo	Medio	Prioridad	Alta
Iteración	1	Puntos estimados	2
Descripción	Para visualizar el expediente técnico, el usuario sin privilegio puede visualizar los expedientes técnicos que tenga bajo su responsabilidad con solo entrar al sistema, se le mostrará la vista principal un icono en PDF con los números de inventarios de las PC que tiene bajo su custodia. Dará clic sobre ellos y se le mostrará su expediente técnico en formato PDF para su posterior impresión y/o descarga.		

Capítulo II Análisis y diseño del sistema

	Por otro lado, los usuarios con algún tipo de privilegio pueden visualizar los expedientes técnicos al dar clic en el ícono "Aceptados", se le mostrará una vista con todos los expedientes registrados con esa condición y se dará clic en el ícono llamado Mostrar.
--	---

Descripción de los campos que componen las Historias de Usuario:

Orden: Está constituido por dos partes. La primera está referido al nomenclador HU (Historia de Usuario y la segunda corresponde el número de la funcionalidad que representa.

Nombre: Nombre que identificará a la Historia de Usuario

Riesgo: Es el grado de incertidumbre en el desarrollo que se asocia a la Historia de Usuario. Determina la posibilidad real de implementarse o no con las condiciones previstas por el equipo de desarrollo (tiempo, recursos, personal). Puede ser Bajo, Medio o Alto.

Prioridad: La prioridad la define el cliente, y es el grado de importancia que le concede a la funcionalidad.

Iteración: Es el número de la fase en la cual se define la Historia de Usuario.

Puntos estimados: Es un número entero que representa la cantidad de semanas que se dispone para el desarrollo de la Historia de Usuario. Las Historias de Usuario con altos puntos estimados deben ser separadas en varias tareas. Un punto es una semana efectiva de desarrollo.

Descripción: Se escribe una fundamentación de lo que hace la funcionalidad.

A partir de la solución propuesta se identificaron 55 requisitos funcionales agrupados en las siguientes Historias de Usuario:

1. Autenticar usuario en sistema SIGEXT.	10. Exportar expediente técnico.
2. Administrar usuario en el sistema SIGEXT.	11. Gestionar rotura.
3. Gestionar responsable del local.	12. Gestionar traslado.
4. Gestionar área de responsabilidad.	13. Registrar intercambio.
5. Gestionar locales.	14. Generar reportes.
6. Gestionar responsable del área.	15. Graficar estadísticas.
7. Gestionar componentes de hardware.	16. Generar ficha técnica.
8. Gestionar expediente técnico.	17. Generar alertas.

Capítulo II Análisis y diseño del sistema

9. Visualizar expediente técnico.	18. Imprimir gráficas de estadísticas.
-----------------------------------	--

En las tablas 2 y 3 se muestran la descripción de dos Historias de Usuario.

TABLA 2 HISTORIA DE USUARIO. GESTIONAR EXPEDIENTE TÉCNICO

HISTORIA DE USUARIO			
Orden	HU_08	Nombre	Gestionar expediente técnico
Riesgo	Alto	Prioridad	Alta
Iteración	1	Puntos estimados	3
	<p>Para gestionar un expediente técnico el responsable de área o del local, una vez registrado en el sistema y con los permisos necesarios para realizar esta acción, se desplazará hacia el módulo <i>Expediente técnico</i> y dará clic sobre él. Se mostrará una vista con las funcionalidades siguientes: Registrar expediente técnico, Administrar expediente técnico, Listar expediente técnico, Visualizar expediente técnico nuevo, en revisión y aceptados y Generar un reporte de los expedientes técnicos. Para agregar un expediente nuevo dará clic en el icono “<i>Registrar expediente</i>” y se visualizará una vista dividida en cuatro secciones en forma tabular: La primera sección Datos generales almacenará las informaciones siguientes: Nombre del área de responsabilidad: donde el personal encargado podrá escoger el área al que pertenece el expediente técnico. El área de responsabilidad se cargará a partir del área asignada al personal encargado por el administrador del sistema en la vista <i>administración de área de responsabilidad</i>; el nombre del local donde aparecerá en una lista desplegable los locales que pertenecen al área seleccionada; nombre del responsable: este nombre aparecerá automáticamente a partir de la autenticación del usuario en el sistema; puesto, número de inventario de la PC, éste aparecerá en una lista desplegable con todos los números de inventario registrados en el sistema, número de serie de la PC y Número de sello de copextel. Una vez llenados los datos de esta sección se procederá a la segunda sección llamada Datos técnicos la que almacenará las características técnicas de la PC como son: motherboard, de ella se conoce el fabricante, el modelo y el número de serie; microprocesador, de él se conoce el modelo y su número de serie; memoria RAM, se conoce de esta la capacidad, tipo y cantidad de slot; disco duro, del mismo se conoce el modelo y su número de serie; unidad óptica, de ella se conoce el fabricante, el modelo y su número de</p>		

Capítulo II Análisis y diseño del sistema

Descripción	<p>serie; fuente, se conoce de esta su modelo y su número de serie; tarjeta de video, se conoce de esta su modelo, el fabricante y su número de serie; tarjeta de red, se conoce su modelo, el fabricante y su número de serie. Se registrarán todos los fabricantes y modelos de los dispositivos con los que cuenta el centro y se mostrarán en una lista desplegable. Luego de haber registrado todos los datos en esta sección se pasará para la tercera sección llamada Datos periféricos donde se registrarán los periféricos de la PC como son: monitor, del mismo se conoce el número de inventario, el modelo y su número de serie; teclado, del cual se conoce el modelo; mouse, se conoce del mismo el fabricante y modelo; speaker, se conoce del mismo el fabricante y modelo; UPS¹⁰, de él se conoce el modelo y su número de serie; la impresora, se le conoce el modelo, el número de inventario y el número de serie; scanner, se conoce del mismo el modelo, el número de inventario y el número de serie. Todos los datos sobre el modelo y el fabricante de los dispositivos periféricos se mostrarán en forma de desplegable, escogiendo también los números de inventario de una lista desplegable. Luego de haber llenado los datos de esa sección se pasará a la cuarta sección llamada Observaciones la cual registrará algunos datos de interés como son el estado de revisión del cual se conoce si es Nuevo, En revisión o Aceptado. Los datos del estado de revisión deben mostrarse en una lista desplegable para permitir al usuario su elección. El sistema tomará la fecha actual. Una vez llenado todos los campos de cada sección se dará clic en el botón REGISTRAR, si todos los datos introducidos son correctos, el sistema registrará el expediente técnico. En caso de que no exista correspondencia con los datos solicitados el sistema emitirá un mensaje de error, lo que indica que debe corregir los errores señalados.</p> <p>Si el usuario una vez llenado todos los datos escoge como estado de revisión la opción NUEVO y da clic en el botón REGISTRAR entonces el expediente técnico se registrará en la carpeta "Nuevo". Para visualizar los expedientes técnicos nuevos se debe dar clic en el módulo <i>Expediente técnico</i>, se le mostrará una vista con todas las funcionalidades referentes a este módulo y se dará clic en icono "Expedientes nuevos". Se le mostrará un listado con todos los expedientes técnicos que cumplan esa condición. Para modificar algún expediente se procederá a dar</p>
--------------------	---

¹⁰ UPS: Sistema de Alimentación Ininterrumpida o *Uninterrupted Power System* por sus siglas en inglés

Capítulo II Análisis y diseño del sistema

clic en el icono representado por un lápiz llamado Modificar. En caso de eliminar algún expediente técnico se dará clic en el icono representado por un cesto de basura, llamado Borrar. El sistema emitirá un mensaje de confirmación de eliminación, brindándole la posibilidad al usuario de decidir si ejecuta o no esta acción. El sistema permitirá filtrar los expedientes técnicos según los siguientes criterios: número de inventario, área de responsabilidad, local y responsable, para su mejor localización.

Si el usuario una vez llenado todos los datos escoge como estado de revisión la opción **EN REVISIÓN** y da clic en el botón **REGISTRAR** entonces el expediente técnico se registrará en la carpeta "Revisión". Para visualizar los expedientes técnicos en revisión se debe dar clic en el módulo *Expediente técnico*, se le mostrará una vista con todas las funcionalidades referentes a este módulo y se dará clic en icono "Expedientes en revisión". Se le mostrará un listado con todos los expedientes técnicos que cumplan esa condición. Para modificar algún expediente se procederá a dar clic en el icono representado por un lápiz llamado Modificar. En caso de eliminar algún expediente técnico se dará clic en el icono representado por un cesto de basura llamado Borrar. El sistema emitirá un mensaje de confirmación de eliminación, brindándole la posibilidad al usuario de decidir si ejecuta o no esta acción. El sistema permitirá filtrar los expedientes técnicos según los siguientes criterios: número de inventario, área, local y responsable, para su mejor localización.

Si el usuario una vez llenado todos los datos escoge como estado de revisión la opción **ACEPTADO** y da clic en el botón **REGISTRAR** entonces el expediente técnico se registrará en la carpeta "Aceptado". Para visualizar los expedientes técnicos aceptados se debe dar clic en el módulo *Expediente técnico*, se le mostrará una vista con todas las funcionalidades referentes a este módulo y se dará clic en icono "Expedientes aceptados". Se le mostrará un listado con todos los expedientes técnicos que cumplan esa condición. Para modificar algún expediente técnico se procederá a dar clic en el icono representado por un signo **+ de color verde** Se le mostrará el expediente técnico para la modificación de sus datos en cada sección según sea el caso. Una vez modificado los valores se procederá a ir a la sección modificación; registrará el número de sello de copextel y el número de orden del GAT y se dará clic en el botón **REGISTRAR**.

Capítulo II Análisis y diseño del sistema

	Automáticamente se modifica el valor y se adiciona una modificación en el expediente técnico donde se explica los valores que fueron modificados. En caso de eliminar algún expediente técnico se siguen los pasos anteriores para su modificación y se dará clic en el icono representado por un cesto de basura llamado Borrar. El sistema emitirá un mensaje de confirmación de eliminación, brindándole la posibilidad al usuario de decidir si ejecuta o no esta acción. El sistema permitirá filtrar los expedientes técnicos según los siguientes criterios: número de inventario, área, local y responsable para su mejor localización.
--	---

TABLA 3 HISTORIA DE USUARIO: GENERAR FICHA TÉCNICA

HISTORIA DE USUARIO			
Orden	HU_16	Nombre	Generar ficha técnica
Riesgo	Bajo	Prioridad	Baja
Iteración	2	Puntos estimados	1
Descripción	Para generar fichas técnicas el administrador del sistema, responsable de área o responsable del local una vez registrados en el sistema, se desplazará hacia el módulo Locales , se mostrará una vista con todas las funcionalidades siguientes: Registrar local, administrar locales, generar estadísticas del local, registrar responsable, administrar responsable, generar reporte del local. Una vez ahí se desplazará hacia el icono "Administración de locales", se mostrará todos los locales registrados en el sistema, luego se dará clic en el icono que representa la ficha técnica. El sistema generará un documento en formato PDF con un resumen de las principales características de cada PC donde se visualizará en un pequeño cuadro las características siguientes: Local, Número de inventario, Número de serie de la PC, Tipo de motherboard, Cantidad de RAM y Capacidad del disco duro. Se podrá visualizar una o varias fichas en formato PDF.		

Estimación de esfuerzo por Historias de Usuario

Las estimaciones de esfuerzo asociado a la implementación de las Historias de Usuario se realizan con el objetivo de lograr una planificación real en el desarrollo del sistema SIGEXT y llevar un registro de la velocidad de desarrollo, basándose principalmente en la suma de puntos correspondientes a las Historias de Usuario.

Capítulo II Análisis y diseño del sistema

La planificación se puede realizar basándose en el tiempo. La velocidad de desarrollo es utilizada para establecer cuántas Historias de Usuario se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de éstas. Se establece como medida el punto estimado. Un punto estimado, equivale a una semana ideal de programación. Las Historias de Usuario generalmente valen de 1 a 3 puntos (10). Teniendo en cuenta lo expuesto anteriormente la estimación de esfuerzo de las Historias de Usuario queda como se muestra en la tabla 4.

TABLA 4 ESTIMACIÓN DE ESFUERZO POR HISTORIA DE USUARIO

Historia de Usuario	Estimación de esfuerzo
Autenticar usuario en el sistema SIGEXT	1 semana
Administrar usuario en el sistema SIGEXT	1 semana
Gestionar área de responsabilidad	1 semana
Gestionar Locales	1 semana
Gestionar responsable del local	1 semana
Gestionar responsable del área	1 semana
Gestionar expediente técnico	3 semanas
Gestionar componentes de hardware	2 semanas
Visualizar expediente técnico	1 semana
Exportar expediente técnico	1 semana
Gestionar roturas	1 semana
Gestionar traslado	1 semana
Registrar intercambio	1 semana
Generar reportes	1 semana
Graficar estadísticas	2 semanas
Generar ficha técnica	1 semana
Generar alertas	1 semana
Imprimir gráficas de estadísticas	1 semana
Total de Historias de Usuario: 18	Total de esfuerzo: 22 semanas

A partir de la suma de los puntos de estimación de esfuerzo por cada Historia de Usuario, se calcula que el desarrollo del sistema tendrá una duración de 22 semanas.

Plan de iteraciones

Capítulo II Análisis y diseño del sistema

Para lograr una mejor organización del trabajo y proporcionar un desarrollo iterativo e incremental, se crea el plan de iteraciones donde se planifica el orden de desarrollo de las Historias de Usuario. Se definió realizar tres iteraciones, su clasificación está determinada según las prioridades de las Historias de Usuario. La duración total de cada iteración dependerá de los puntos estimados de las Historias de Usuario que en él se desarrollan. A continuación en la tabla 5 se muestra el plan de iteraciones.

TABLA 5 PLAN DE ITERACIONES

Iteración	Historias de Usuario
1	Autenticar usuario en el sistema SIGEXT
	Administrar usuario en el sistema SIGEXT
	Gestionar área de responsabilidad
	Gestionar local
	Gestionar expediente técnico
	Gestionar componentes de hardware
2	Gestionar traslado
	Gestionar responsable de área
	Gestionar responsable del local
	Registrar intercambio
	Visualizar expediente técnico
	Exportar a PDF
3	Generar ficha técnica
	Gestionar roturas
	Graficar estadísticas
	Generar alertas
	Generar reportes
	Imprimir gráficas de estadísticas

Plan de entrega

El plan de entrega es un documento que especifica con exactitud qué Historias de Usuario serán implementadas en cada entrega del sistema y sus prioridades, de modo que también permita conocer con claridad qué Historias de Usuario serán implementadas en la próxima iteración. Debe ser negociado y elaborado en forma conjunta entre el cliente y el equipo de desarrollado durante las reuniones de

Capítulo II Análisis y diseño del sistema

planificación de entregas, la idea es hacer entregas frecuentes para obtener una mayor retroalimentación (31).

A continuación se muestra en la tabla 6 el plan de entrega definido para el ciclo de desarrollo.

TABLA 6 PLAN DE ENTREGA

Iteración	Primera iteración	Segunda iteración	Tercera iteración
Historias de Usuario	6	6	6
Fecha de entrega	3 de noviembre de 2014	15 de diciembre de 2014	16 de febrero de 2015

2.5 Patrones de diseño

El uso de patrones es considerado como una buena práctica de desarrollo en la construcción de *software*. Un patrón de diseño es una buena práctica documentada o solución, que se ha aplicado con éxito en múltiples ambientes para erradicar problemas comunes de diseño de *software*, con una probada efectividad y con características de reutilización (32).

Patrones GRASP¹¹

Experto: Se encarga de asignar una responsabilidad al experto en información, o sea, aquella clase que cuenta con la información necesaria para cumplir la responsabilidad.

Creador: Guía la asignación de responsabilidades relacionadas con la creación de objetos. El propósito fundamental de este patrón es encontrar un creador que conecte con el objeto producido en cualquier evento. Le asigna a una clase determinada la responsabilidad de crear una instancia de otro tipo de clase.

Alta cohesión: Es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme.

Bajo acoplamiento: Es una medida de la fuerza con que una clase está conectada a otras clases, con que las conoce y con que recurre a ellas. Acoplamiento bajo significa que una clase no depende de muchas clases.

¹¹ GRASP: Patrón de Asignación de Responsabilidad o *General Responsibility Assignment Software Patterns* por sus siglas en inglés

Capítulo II Análisis y diseño del sistema

Controlador: Asigna la responsabilidad del manejo de mensajes de los eventos del sistema.

Patrones GoF¹²

Observador: Es un patrón de diseño que define una dependencia del tipo *uno-a-muchos* entre objetos, de manera que cuando uno de los objetos cambia su estado, notifica este cambio a todos los dependientes.

Acción/Comando: Encapsula una operación en un objeto, permitiendo ejecutar dicha operación sin necesidad de conocer el contenido de la misma.

Cadena de responsabilidades: Permite establecer la línea que deben llevar los mensajes para que los objetos realicen la tarea indicada.

2.6 Arquitectura del framework de desarrollo Yii

La arquitectura Modelo–Vista–Controlador surgió como patrón arquitectónico para el desarrollo de interfaces gráficas de usuario en entornos web. Su concepto se basa en separar el modelo de datos de la aplicación, de su representación de cara al usuario y de la interacción de éste con la aplicación, mediante la división de la aplicación en tres partes fundamentales (33). Yii utiliza este patrón y proporciona una estructura de carpetas explícita para cada pieza del modelo, la vista y código del controlador (33), como se muestra en la figura 4.

Modelo: Es la representación de la información en el sistema. Trabaja junto a la vista para mostrar la información al usuario y es accedido por el controlador para añadir, eliminar, consultar o actualizar datos. Un modelo en Yii es cualquier instancia del marco de la clase CModel o su clase hija (33).

Vista: Es responsable de hacer la interfaz de usuario, basada en los datos del modelo. Una vista en Yii es un script PHP que contiene elementos de interfaz de usuario relacionada, a menudo construida con HTML (33).

Controlador: El controlador es el principal director de una solicitud encaminada y es responsable de tomar la entrada del usuario, interactuando con el modelo, e instruir a la vista para actualizar y mostrarse adecuadamente. Un controlador en Yii es una instancia de CController o su hijo. Cuando un controlador se ejecuta, realiza la acción solicitada, la cual interactúa con los modelos necesarios y hace una conclusión acertada (33).

¹²GoF: Banda de los Cuatro o *Gang of four por sus siglas en inglés*.

Capítulo II Análisis y diseño del sistema

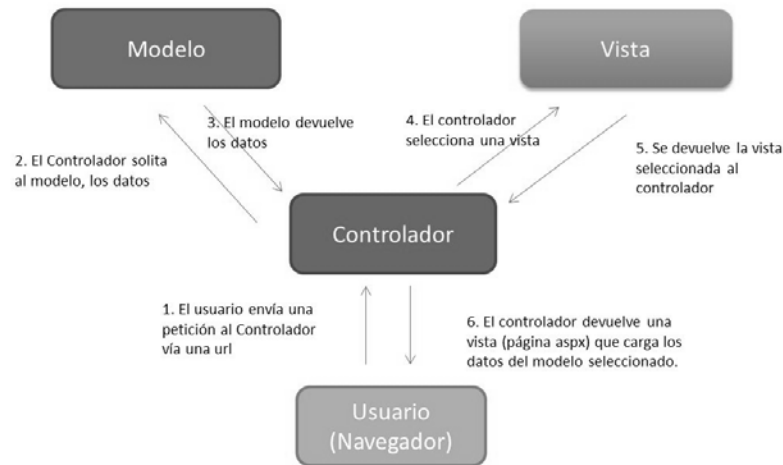


FIGURA 4 ARQUITECTURA DEL MODELO - VISTA- CONTROLADOR

Arquitectura del sistema SIGEXT

Para la definición de la arquitectura del sistema SIGEXT, se tuvo en cuenta la arquitectura definida por el framework de desarrollo Yii. Por lo que se seleccionó para el desarrollo del sistema SIGEXT, el patrón arquitectónico MVC. En la figura 5 se muestra un ejemplo de la arquitectura del sistema SIGEXT.

Modelo: Yii implementa dos tipos de modelos: el modelo de formulario (clase CFormModel) y el modelo de registro activo (clase CActiveRecord). Ambos se extienden desde la misma clase base CModel. CFormModel representa un modelo de datos que recoge las entradas en forma de HTML y encapsula toda la lógica para la validación de campo de formulario y cualquier otra lógica de negocio que pueden necesitar ser aplicado a los datos de campo de formulario. Active Record (AR) es un patrón de diseño utilizado para el acceso de base de datos abstracta de una manera orientada a objetos. Los valores de los campos de datos de cada columna de la fila de la tabla se representan como propiedades del objeto AR. Esta clase se encuentra ubicada dentro del directorio protected y contiene las principales clases siguientes:

- ExpedienteTecnico.php
- AreaResponsabilidad.php
- HistorialModificaciones.php

Capítulo II Análisis y diseño del sistema

Vista: Presenta la información obtenida con el modelo de manera que pueda ser visible al usuario. Esta clase está ubicada dentro del directorio protected y contiene las principales clases siguientes:

- exptecpdf.php
- areaResponsabilidad.php
- expedienteTecnico.php
- historialModificaciones.php

Controlador: Es la capa intermedia e implementa toda la capa de negocio del sistema, en la misma se encuentra el núcleo del sistema que implementa toda la lógica de la aplicación. Esta clase está ubicada dentro del directorio protected y contiene las principales clases siguientes:

- ExpedienteTecnicoController.php
- AreaResponsabilidadController.php
- HistorialModificacionesController.php

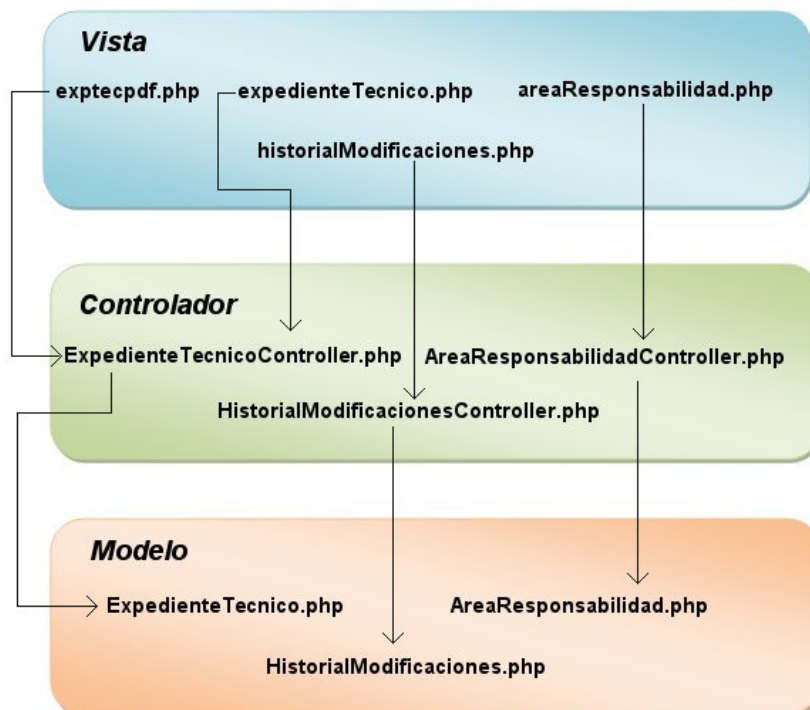


FIGURA 5 ARQUITECTURA DEL MODELO - VISTA- CONTROLADOR DEL SISTEMA SIGEXT

Capítulo II Análisis y diseño del sistema

2.7 Diseño de Base de Datos

Para el desarrollo del sistema se modeló el siguiente esquema de base de datos, que contiene toda la información necesaria para gestionar los expedientes técnicos. A continuación en la figura 6 se presenta el modelo de datos obtenido para el sistema SIGEXT.

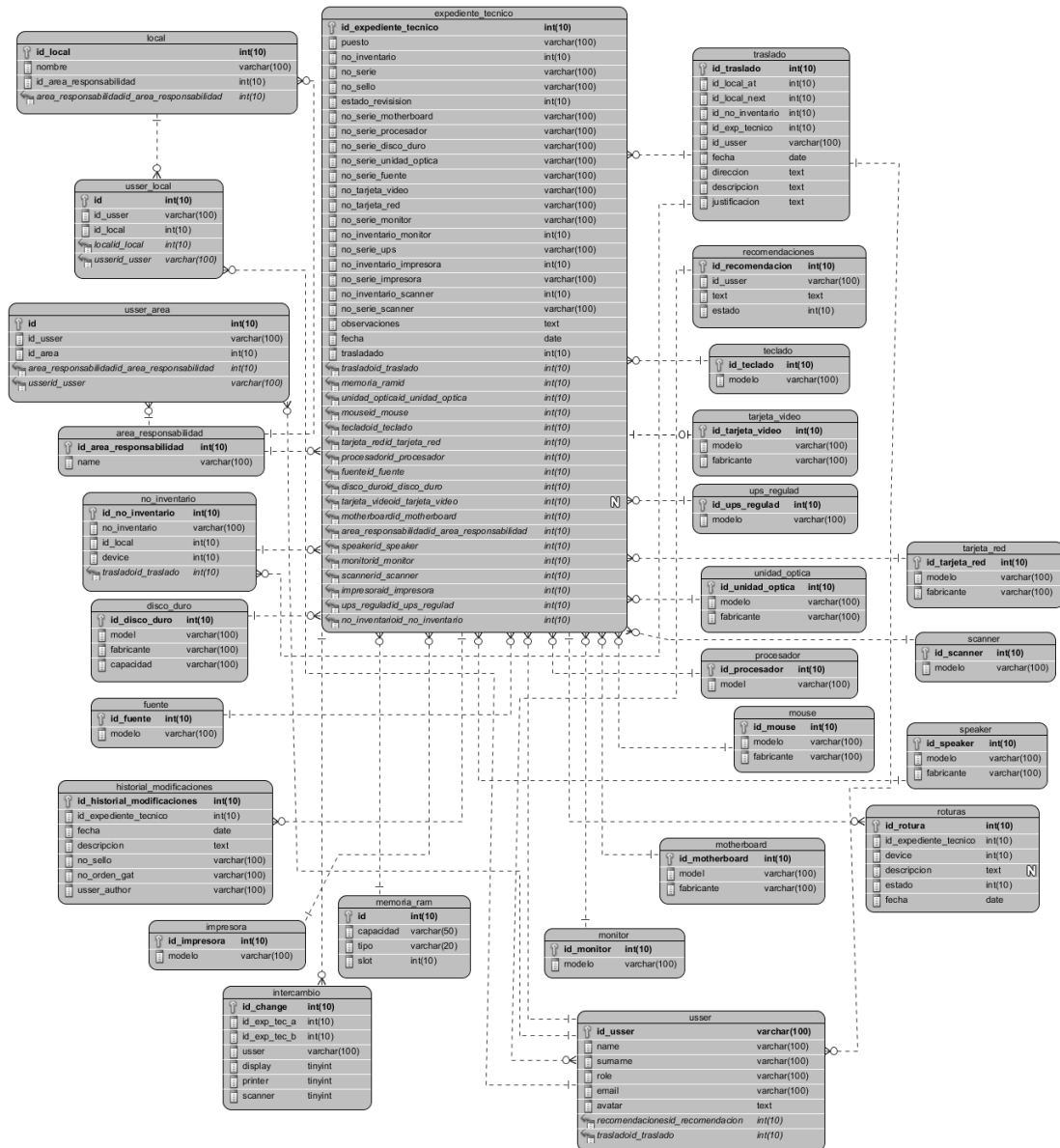


FIGURA 6 MODELO DE DATOS DEL SISTEMA SIGEXT

Capítulo II Análisis y diseño del sistema

La tabla 7 que a continuación se presenta, describe cada una de las entidades que conforma el modelo de datos:

TABLA 7 DESCRIPCIÓN DE LAS ENTIDADES DEL SISTEMA

Nombre	Descripción
expediente_tecnico	Almacena la información en la base de datos las características de los componentes internos y de los dispositivos de entrada y salida de una estación de trabajo.
traslado	Almacena la información en la base de datos la fecha en que se realiza el traslado, la dirección para donde se traslada el medio de cómputo, una breve justificación del porqué del traslado y la descripción del medio a trasladar, todos estos datos registrados en el expediente técnico.
tarjeta_video	Almacena la información en la base de datos el modelo y el fabricante de la tarjeta de vídeo.
historial_modificaciones	Almacena la información en la base de datos la fecha, descripción, número de sello de Copextel, número de la orden de servicio del GAT y el usuario autorizado a realizar una modificación al expediente técnico.
disco_duro	Almacena la información en la base de datos el modelo, el fabricante y la capacidad del disco duro.
memoria_ram	Almacena la información en la base de datos la capacidad, la cantidad de slot y el tipo de memoria RAM.
motherboard	Almacena la información en la base de datos el modelo y el fabricante de la motherboard.
usser	Almacena la información en la base de datos el nombre, los apellidos, el rol que desempeña, el correo electrónico y la foto que identifica al usuario registrado.
usser_area	Almacena la información en la base de datos, el nombre del usuario responsable del área.
recomendaciones	Almacena la información en la base de datos el usuario, el texto y el estado de la recomendación realizada al sistema.
area_responsabilidad	Almacena la información en la base de datos el nombre del área de responsabilidad.
unidad_optica	Almacena la información en la base de datos el modelo y el fabricante de la unidad óptica.
speaker	Almacena la información en la base de datos el nombre y el fabricante del speaker.

Capítulo II *Análisis y diseño del sistema*

usser_local	Almacena la información en la base de datos el nombre del local, el usuario responsable del área y el área al que pertenece.
monitor	Almacena la información en la base de datos el modelo del monitor.
local	Almacena la información en la base de datos el nombre del local y el área al que pertenece.
scanner	Almacena la información en la base de datos el modelo del scanner.
mouse	Almacena la información en la base de datos el modelo y fabricante del mouse.
teclado	Almacena la información en la base de datos el modelo del teclado.
impresora	Almacena la información en la base de datos el modelo de la impresora.
ups_regulad	Almacena la información en la base de datos el modelo del UPS o regulador de voltaje.
no_inventario	Almacena la información en la base de datos el número de inventario del medio, el local al que pertenece y la descripción del medio.
roturas	Almacena la información en la base de datos el número de inventario del medio, el dispositivo o componente defectuoso, una breve descripción de la rotura, el estado en que se encuentra y la fecha que se reporta la rotura.
tarjeta_red	Almacena la información en la base de datos el modelo y el fabricante de la tarjeta de red.
procesador	Almacena la información en la base de datos el modelo del procesador.
fuelle	Almacena la información en la base de datos el modelo de la fuente.
intercambio	Almacena la información en la base de datos los números de inventario de las PC, el usuario que realiza el intercambio y el medio a intercambiar, estos pueden ser: monitor, impresora y scanner.

2.8 Tarjetas CRC

Las características más sobresalientes de las tarjetas CRC son su simpleza y adaptabilidad. Se define a una tarjeta CRC como una ficha de papel o cartón que representa a una entidad del sistema, las cuales permiten también que el equipo completo contribuya en la tarea del diseño. Estas tarjetas se utilizan para estructurar las clases y a su vez definir las responsabilidades sobre las mismas, así como la simulación de escenarios en el sistema (30).

Capítulo II Análisis y diseño del sistema

El nombre de la clase se coloca a modo de título en la tarjeta, las responsabilidades se colocan en la parte izquierda, y las clases que se implican en cada responsabilidad en la parte derecha, en la misma línea que su requerimiento correspondiente.

Clase: Es cualquier persona, evento, concepto, pantalla o reporte.

Responsabilidades: Las responsabilidades de una clase son las entidades que conoce y las que realizan sus atributos y métodos.

Colaboradoras: Los colaboradoras de una clase son las demás clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades. A continuación se muestran algunas tarjetas CRC consideradas las más importantes.

En las tablas 8, 9, 10, 11, y 12 se presentan las tarjetas CRC más significativas obtenidas del sistema SIGEXT.

TABLA 8 TARJETA CRC. EXPEDIENTE TÉCNICO

Nombre de la clase: ExpedienteTecnicoController	
Responsabilidades	Colaboradoras
Adicionar un expediente técnico	Traslado
Modificar un expediente técnico	HistorialModificaciones
Eliminar un expediente técnico	Roturas
Generar reporte de expediente técnico	Intercambio

TABLA 9 TARJETA CRC. HISTORIAL DE MODIFICACIONES

Nombre de la clase: HistorialModificacionesController	
Responsabilidades	Colaboradoras
Adicionar una modificación	ExpedienteTecnico
Modificar una modificación	NoInventario
Eliminar una modificación	AreaResponsabilidad
Mostrar modificación	Local

Capítulo II Análisis y diseño del sistema

TABLA 10 TARJETA CRC. LOCAL

Nombre de la clase: LocalController	
Responsabilidades	Colaboradoras
Adicionar un local	UsserLocal
Modificar un local	NoInventario
Eliminar un local	AreaResponsabilidad
Generar reporte de local	Expedientetecnico

TABLA 11 TARJETA CRC. NÚMERO DE INVENTARIO

Nombre de la clase: NoInventarioController	
Responsabilidades	Colaboradoras
Adicionar un número de inventario	
Modificar un número de inventario	ExpedienteTecnico
Eliminar un número de inventario	
Generar reporte de número de inventario	

TABLA 12 TARJETA CRC. TRASLADO

Nombre de la clase: TrasladoController	
Responsabilidades	Colaboradoras
Adicionar un traslado	Local
Modificar un traslado	AreaResponsabilidad
Eliminar un traslado	ExpedienteTecnico
Generar reporte de traslado	HistorialModificaciones

2.9 Diagrama de despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema. En la figura 7, se muestra la distribución de los nodos del sistema SIGEXT.

Capítulo II Análisis y diseño del sistema



FIGURA 7 DIAGRAMA DE DESPLIEGUE DEL SISTEMA SIGEXT

Descripción de los nodos

PC Cliente: El nodo representa una PC cliente desde la cual se podrá acceder al sistema por medio de un navegador Web e interactuar con todas las funcionalidades que este brinda.

Servidor Web: El nodo representa el servidor Web donde estará alojado el sistema así como los componentes almacenados en el mismo.

Servidor de Base de Datos: El nodo representa el servidor de Base de Datos MySQL en el que estará alojada la base de datos del sistema.

Impresora: El nodo representa la impresora a través de la cual se imprimen los reportes generados por el sistema.

Descripción de los protocolos utilizados:

- ✓ **HTTPS:** Protocolo para Transferencia de Hipertexto seguro. Extensión del HTTP para la autenticación y encriptación de datos entre un servidor Web y un navegador Web.
- ✓ **TCP/IP:** Proviene de dos protocolos, Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP). Forma de comunicación básica que usa el Internet.
- ✓ **LPT:** Línea terminal de impresión/línea de la impresora (*Line Print Terminal / Line PrinTer*). Es un conector que permite la transmisión de datos desde un dispositivo externo hacia la computadora.

Capítulo II *Análisis y diseño del sistema*

- ✓ **USB:** Bus Universal en Serie (*Universal Serial Bus*) Interfaz entre la PC y ciertos dispositivos.

Conclusiones parciales

El estudio de los aspectos fundamentales que rodean al negocio, permitió establecer las reglas principales que se deben cumplir para el desarrollo del sistema. Se realizó un estudio de los principales artefactos que guían la metodología de desarrollo XP lo que permitió definir los aspectos necesarios para el desarrollo del sistema. Se definió la arquitectura que tendrá el sistema guiado por el patrón arquitectónico Modelo-Vista-Controlador y se mostró a través del diagrama de despliegue la manera en que se implantará el sistema propuesto en un ambiente real.

Capítulo III Implementación y prueba

Introducción

En este capítulo se realiza la implementación del sistema propuesto a través de las tareas de ingeniería, se define los estándares de codificación que tendrá el sistema y se diseñarán los casos de prueba realizadas para la validación de la solución propuesta, además de mostrarse los resultados de estas.

3.1 Tarea de ingeniería

Las tareas de ingeniería son actividades que los programadores conocen que el sistema debe hacer. Deben ser estimables, su tiempo de implementación debe ser corto, aproximadamente entre uno y tres días, su objetivo es resolver las Historias de Usuario. Una Historia de Usuario puede tener una o varias tareas de ingeniería en dependencia de la funcionalidad a desarrollar. Pueden existir también tareas de ingeniería técnicas, que son aquellas que aunque no derivan directamente de una Historia de Usuario, es necesaria su consideración para que el sistema funcione (34).

En la tabla 13 se muestra el formato utilizado para la confección de las tareas de ingeniería:

TABLA 13 TAREA DE INGENIERÍA. ADICIONAR EXPEDIENTE TÉCNICO

Tarea de ingeniería	
Número tarea: 16	Número de Historia de Usuario: HU_08
Nombre tarea: Adicionar expediente técnico	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 8/09/2014	Fecha fin: 9/09/2014
Programador responsable: Yadriel Cuesta Hechavarría	
Descripción: Adiciona un expediente técnico a la base de datos.	

Los campos de la tarjeta de las tareas de ingeniería reflejan lo siguiente:

Número tarea: Representa el número por el que se identifica a la tarea. Cada tarea tiene un único número que la identifica.

Número Historia de Usuario: Es el número de la Historia de Usuario a la que responde la tarea.

Capítulo III Implementación y prueba

Nombre de tarea: Define el nombre o funcionalidad concreta a la que se dedica la tarea, debe estar expresado en forma infinitiva.

Tipo de tarea: Información del tipo de tarea a realizar, la misma puede ser:

- *Desarrollo:* Tarea que se realizará por primera vez.
- *Corrección:* Tarea que se realiza a partir de una anterior que no se realizó correctamente, es decir, que no pasó los casos de prueba satisfactoriamente.
- *Mejora:* Tarea que se realiza a partir de una anterior incorporándole nuevos requerimientos.
- *Otra:* Tarea que no corresponde con una de las anteriores, en este caso es necesario especificar el tipo de tarea o realizar una descripción más profunda de esta.

Puntos estimados: Tiempo de duración de la tarea. El tiempo estimado es reflejado en días. La suma de los puntos estimados de las tareas de ingeniería de una Historia de Usuario no puede superar la cantidad de puntos estimados definidos para la Historia de Usuario.

Fecha inicial: Fecha en la que se inicia el desarrollo de la tarea de ingeniería.

Fecha final: Fecha en la que se concluye el desarrollo de la tarea de ingeniería.

Programador responsable: Nombre del responsable de la realización de la tarea.

Descripción: Es una breve descripción sobre lo que la tarea debe hacer o resolver.

Durante la etapa de implementación se realizaron un total de 55 tareas de ingeniería, las cuales responden al número de Historias de Usuario implementadas. En la tabla 14 se muestra un resumen de las tareas de ingenierías implementadas por cada Historia de Usuario.

TABLA 14 RESUMEN DE LAS TAREAS DE INGENIERÍA POR CADA HISTORIA DE USUARIO

Historia de Usuario	No. Tarea	Tarea de Ingeniería
Autenticar usuario en el sistema SIGEXT	01	Autenticar usuario
Administrar usuario en el sistema SIGEXT	02	Modificar usuario
	03	Eliminar usuario
Gestionar área de responsabilidad	04	Adicionar área de responsabilidad
	05	Modificar área de responsabilidad
	06	Eliminar área de responsabilidad

Capítulo III Implementación y prueba

Gestionar locales	07	Adicionar local
	08	Modificar local
	09	Eliminar local
Gestionar responsable del local	10	Adicionar responsable del local
	11	Modificar responsable del local
	12	Eliminar responsable del local
Gestionar responsable del área	13	Adicionar responsable del área
	14	Modificar responsable del área
	15	Eliminar responsable del área
Gestionar expediente técnico	16	Adicionar expediente técnico
	17	Modificar expediente técnico
	18	Eliminar expediente técnico
Gestionar componentes de hardware	19	Gestionar componentes de hardware
Visualizar expediente técnico	20	Visualizar expediente técnico
Exportar expediente técnico	21	Exportar expediente técnico
Gestionar roturas	22	Adicionar rotura
	23	Modificar rotura
	24	Eliminar rotura
Gestionar traslado	25	Adicionar traslado
	26	Modificar traslado
	27	Eliminar traslado
Registrar intercambio	28	Registrar intercambio
	29	Generar reportes de usuario
	30	Generar reportes de área de responsabilidad
	31	Generar reportes de locales
	32	Generar reportes de expedientes técnicos
	33	Generar reportes de número de inventario
	34	Generar reportes de roturas
	35	Generar reportes de traslado
	36	Generar reportes de intercambio
	37	Generar reportes de motherboard
	38	Generar reportes de microprocesadores

Capítulo III Implementación y prueba

Generar reportes	39	Generar reportes de disco duro
	40	Generar reportes de fuente
	41	Generar reportes de mouse
	42	Generar reportes de teclado
	43	Generar reportes de monitor
	44	Generar reportes de speaker
	45	Generar reportes de impresora
	46	Generar reportes de scanner
	47	Generar reportes de tarjeta de red
	48	Generar reportes de tarjeta de video
	49	Generar reportes de lector de DVD
	50	Generar reportes de UPS
	51	Generar reportes de memoria RAM
Graficar estadísticas	52	Graficar estadísticas
Generar ficha técnica	53	Generar ficha técnica
Generar alertas	54	Generar alertas
Imprimir gráfica de estadísticas	55	Imprimir gráficas de estadísticas

Las siguientes tablas 15, 16 y 17 ejemplifican el trabajo realizado.

TABLA 15 TAREA DE INGENIERÍA. MODIFICAR EXPEDIENTE TÉCNICO

Tarea de ingeniería	
Número tarea: 17	Número de Historia de Usuario: HU_08
Nombre tarea: Modificar expediente técnico	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 10/09/2014	Fecha fin: 11/09/2014
Programador responsable: Yadriel Cuesta Hechavarría	
Descripción: Modifica uno o varios campos del expediente técnico registrado en la base de datos haciendo una actualización en la misma.	

Capítulo III Implementación y prueba

TABLA 16 TAREA DE INGENIERÍA. ELIMINAR EXPEDIENTE TÉCNICO

Tarea de ingeniería	
Número tarea: 18	Número de Historia de Usuario: HU_08
Nombre tarea: Eliminar expediente técnico	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 12/09/2014	Fecha fin: 13/09/2014
Programador responsable: Yadriel Cuesta Hechavarría	
Descripción: Elimina de la base de datos el expediente técnico seleccionado tomando en consideración la eliminación de los valores contenidos en las clases: Traslado, HistorialModificaciones, Roturas e Intercambio asociado al expediente técnico.	

TABLA 17 TAREA DE INGENIERÍA. VISUALIZAR EXPEDIENTE TÉCNICO

Tarea de ingeniería	
Número tarea: 20	Número de Historia de Usuario: HU_09
Nombre tarea: Visualizar expediente técnico	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 15/09/2014	Fecha fin: 16/09/2014
Programador responsable: Yadriel Cuesta Hechavarría	
Descripción: Visualiza en formato PDF el expediente técnico tomando los valores almacenados en la base de datos.	

3.2 Estándares de codificación

Las convenciones o estándares de codificación son pautas de programación que no están enfocadas a la lógica del programa, sino a su estructura y apariencia física para facilitar la lectura, comprensión y mantenimiento del código (35). La metodología XP enfatiza la comunicación de los programadores a través del código, con lo cual es necesario que se sigan ciertos estándares de programación. Los estándares de programación mantienen el código legible para los miembros del equipo. Para la implementación del sistema SIGEXT se siguieron normas y estándares desarrollados, que se relacionan a continuación:

Indentación: La indentación es la separación de un bloque de código hacia la derecha, ésta debe ser a cuatro espacios.

Capítulo III Implementación y prueba

Estructuras de control: Las estructuras de control deben tener un espacio entre las palabras clave de la estructura, el signo de apertura de paréntesis para distinguir entre las llamadas de las funciones y el signo de llaves deben estar sobre la línea de la estructura.

Llamadas de funciones: Las funciones deben ser llamadas sin espacios entre el nombre de la función, el signo de paréntesis y el primer parámetro, espacios entre cada coma por parámetro y sin espacios entre el último paréntesis, el signo de paréntesis cerrado y el signo de punto y coma (;).

Estilos de comentarios: El estilo de los comentarios debe ser como el estilo de comentarios para C (`/*, */` o `//`), no debe utilizarse el estilo de comentarios de Perl (`#`).

Bloques de código: Siempre se debe utilizar las etiquetas `<?php>` para abrir un bloque de código.

Nombres: Los nombres de las clases deben de iniciar con letra mayúscula. Los nombres de las variables y de las funciones pueden iniciar con letra minúscula, pero si estas tienen más de una palabra, cada nueva palabra debe iniciar con letra mayúscula; el nombre puede escribirse separado por signos de guión bajo. Si una función, en una clase, es privada; deberá comenzar con el signo de guión bajo para una fácil identificación. Las constantes deben de escribirse siempre en mayúsculas y tanto estas como las variables globales deben de tener como prefijo el nombre de la clase a la que pertenecen.

Los estándares de código resultan importantes en cualquier proyecto de desarrollo, ayudan a asegurar que el código tenga calidad, menos errores y pueda ser mantenido fácilmente.

3.3 Pruebas

La metodología XP enfatiza en la realización de pruebas a lo largo de todo el desarrollo del *software*, con el fin de lograr un producto con calidad, reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su corrección. En este proceso no solo participa el desarrollador, también es importante la colaboración del cliente, sobre todo en las pruebas de aceptación. En XP las pruebas se dividen en dos grupos: pruebas unitarias encargadas de verificar el código y pruebas de aceptación orientadas a probar las funcionalidades del sistema (36).

Pruebas de aceptación

Las pruebas de aceptación son las especificaciones para el comportamiento deseado y la funcionalidad de un sistema. Muestra por una Historia de Usuario dada, cómo el sistema se encarga de ciertas condiciones y con qué tipo de resultados. Los clientes junto a un miembro del equipo de desarrollo son los responsables

Capítulo III Implementación y prueba

de verificar que los resultados de estas pruebas sean los correctos para así tomar decisiones acerca de las mismas. Una Historia de Usuario no se puede considerar terminada hasta que no pase las pruebas de aceptación. Es recomendable publicar los resultados de las pruebas de aceptación, para que todo el equipo de desarrollo esté al tanto de esta información (37). Al sistema SIGEXT a parte de las pruebas de funcionalidad, se le realizaron Pruebas de Seguridad para comprobar que solo los usuarios autorizados tenían acceso a la información, Pruebas de Regresión para comprobar que las no conformidades detectadas habían sido resueltas y que no se había afectado otras funcionalidades, Pruebas de Usabilidad para comprobar el cumplimiento de este requisito no funcional. El proceso de realización de estas pruebas culminó con la firma de conformidad del Acta de Aceptación por parte del cliente. (Anexo no. 3).

Casos de prueba

Los casos de prueba son evidencias de pruebas funcionales o unitarias que se realizan al sistema para comprobar su funcionamiento. Las pruebas funcionales son validaciones escritas desde la perspectiva del cliente, y las pruebas unitarias son validaciones desde la perspectiva del programador. El objetivo general es tener una forma para decirle al cliente que la Historia de Usuario está lista. Las pruebas funcionales o pruebas de aceptación, son las más importantes, ya que representan la medida de satisfacción del cliente para una funcionalidad que el sistema debe tener (9). Los casos de pruebas fueron definidos para cada Historia de Usuario establecida. Se comprueba el funcionamiento de cada una de las funcionalidades implementadas que responden a la misma. El formato utilizado para la confección de casos de pruebas se muestra a continuación en la tabla 18:

TABLA 18 DESCRIPCIÓN DE LOS CASOS DE PRUEBA

Caso de prueba de aceptación	
Código:	Historia de Usuario:
Nombre:	
Descripción:	
Condiciones de ejecución:	
Pasos de ejecución:	
Resultados esperados:	
Evaluación de la prueba:	

Campos del caso de prueba

Capítulo III Implementación y prueba

Código: Identificador del caso de prueba. Dividido en dos partes. La primera representa la inicial del artefacto y la segunda representa el número con que se identifica la prueba.

Historia de Usuario: Es el número de la Historia de Usuario a la que responde el caso de prueba.

Descripción: Es una breve descripción del propósito de la prueba.

Condiciones de ejecución: Condiciones especiales que deben tenerse en cuenta para ejecutar el caso de prueba.

Entradas / pasos de ejecución: Entradas o funciones que deben ejecutarse para realizar el caso de prueba.

Resultado esperado: Salida u objetivo que debe cumplir la funcionalidad a la que se le realiza el caso de prueba.

Evaluación: Evaluación de éxito del caso de prueba. Prueba satisfactoria en caso de éxito o prueba insatisfactoria en caso de fallo.

Los casos de prueba son agregados a los artefactos de entrega que se realiza al cliente al terminar cada fase o iteración del proyecto. Las Historias de Usuario con evaluación insatisfactoria, serán corregidas en la próxima iteración a partir de nuevas tareas de ingeniería. A continuación en las tablas 19, 20 y 21 se argumentan los casos de prueba del sistema implementado clasificados según la iteración a la que pertenecen.

Primera iteración

TABLA 19 CASO DE PRUEBA DE ACEPTACIÓN. GESTIONAR EXPEDIENTE TÉCNICO

Caso de prueba de aceptación	
Código: HU08-P8	Historia de Usuario: 8
Nombre: Gestionar expediente técnico	
Descripción: Prueba diseñada para comprobar la Historia de Usuario: Gestionar expediente técnico	
Condiciones de ejecución: Que el usuario tenga privilegios de administración, de responsable de área o responsable del local para gestionar los expedientes técnicos.	
Pasos de ejecución: <ul style="list-style-type: none">• El usuario accede al sistema a través de la web.• Se le muestra una ventana de autenticación.• El usuario debe de introducir su usuario y contraseña.	

Capítulo III Implementación y prueba

<ul style="list-style-type: none"> • El sistema verifica si el usuario está registrado en el sistema y tiene el rol correspondiente. • El administrador del sistema, responsable del área o responsable del local se desplazará hacia el módulo Expediente técnico y dará clic en el icono “<i>Registrar expediente técnico</i>”. <p>Registrar</p> <ul style="list-style-type: none"> • Podrá registrar un nuevo expediente dando clic en el icono “<i>Registrar expediente</i>”. • Registrará los datos que son solicitados y dará clic en el botón <u>Registrar</u>. <p>Modificar</p> <ul style="list-style-type: none"> • El administrador, responsable de área o de local dará clic en el módulo Expediente técnico. • Se mostrará una vista. • Dará clic en la carpeta Expedientes técnicos aceptados. • Visualizará todos los expedientes técnicos en la condición de aceptados. • Dará clic en el icono representado por un signo + de color verde. • Se le mostrará el expediente técnico para la modificación de sus datos en cada sección según sea el caso. Una vez modificado los valores se procederá a ir a la sección Modificación, registrará el número de sello de copextel y el número de orden del GAT y se dará clic en el botón <u>Registrar</u>. <p>Eliminar</p> <ul style="list-style-type: none"> • El administrador, responsable de área o de local dará clic en el módulo Expediente técnico. • Se mostrará una vista • Dará clic en el icono “<i>Administración de expedientes técnicos</i>”. • Visualizará todos los expedientes técnicos en la condición de aceptados. • Podrá eliminar el expediente técnico dando clic en el icono representado por un cesto de basura llamado <u>Borrar</u>.
Resultados esperados: Se registrará, modificará y eliminará el expediente técnico.
Evaluación de la prueba: Satisfactoria

Segunda iteración

TABLA 20 CASO DE PRUEBA DE ACEPTACIÓN. VISUALIZAR EXPEDIENTE TÉCNICO

Caso de Prueba de aceptación	
Código: HU09-P9	Historia de Usuario: 9
Nombre: Visualizar expediente técnico	
Descripción: Prueba diseñada para comprobar la Historia de Usuario: Visualizar expediente técnico	
Condiciones de ejecución: Que el usuario esté registrado en el sistema.	
Pasos de ejecución:	
<ul style="list-style-type: none"> • El usuario accede al sistema a través de la web. • Se le muestra una ventana de autenticación. • El usuario debe de introducir su usuario y contraseña. • El sistema verifica si el usuario está registrado en el sistema. • Al entrar al sistema se le mostrará en la vista principal los expedientes técnicos que tenga asignado el usuario registrado. • El administrador del sistema, el responsable de área dará clic en el módulo Expediente técnico. 	

Capítulo III Implementación y prueba

<ul style="list-style-type: none">• Se desplazará hacia el icono “Administrar expediente técnico”.• Visualizara todos los expedientes técnicos registrados.• Dará clic en el icono Mostrar.• Se le mostrará el expediente.
Resultados esperados: Se visualizará el expediente técnico.
Evaluación de la prueba: Satisfactoria

Tercera iteración

TABLA 21 CASO DE PRUEBA DE ACEPTACIÓN. GESTIONAR ROTURAS

Caso de Prueba de Aceptación	
Código: HU11-P11	Historia de Usuario: 11
Nombre: Gestionar roturas	
Descripción: Prueba diseñada para comprobar la Historia de Usuario: Gestionar rotura	
Condiciones de ejecución: Que el usuario esté registrado en el sistema y tenga el rol de administrador del sistema, responsable de área o responsable del local.	
Pasos de ejecución: <ul style="list-style-type: none">• El usuario accede al sistema a través de la web.• Se le muestra una ventana de autenticación.• El usuario debe de introducir su usuario y contraseña.• El sistema verifica si el usuario tiene los privilegios requeridos.• El administrador del sistema, responsable de área o responsable del local se desplazarán al módulo Expediente técnico y dará clic en el icono “Expedientes técnicos aceptados”.	
Registrar <ul style="list-style-type: none">• Buscará el número de inventario de la PC que desea registrarle la rotura.• Dará clic en el icono “Rotura”.• Introducirá los datos requeridos y presionará el botón llamado <u>Registrar</u>.	
Modificar <ul style="list-style-type: none">• El administrador, responsable de área o de local dará clic en el módulo Roturas.• Se mostrará una vista.• Seleccionará el número de inventario de la PC a la que quiere modificarle la rotura.• Dará clic en el icono representado por un lápiz llamado <u>Modificar</u>.• Modificará el dispositivo roto, dará una breve explicación si conoce el tipo de rotura y seleccionará de ser necesario la fecha de la misma. Y dará clic en el botón <u>Registrar</u>.	
Eliminar <ul style="list-style-type: none">• El administrador, responsable de área o de local dará clic en el módulo Roturas.• Se mostrará una vista.• Seleccionará el número de inventario de la PC a la que quiere eliminar la rotura.• Dará clic en el icono representado por un cesto de basura llamado <u>Borrar</u>.	

Capítulo III Implementación y prueba

Resultados esperados: Se registrará, modificará y se eliminará una rotura en el expediente técnico.

Evaluación de la prueba: Satisfactoria

3.4 Resultados de las pruebas realizadas por el grupo de calidad del Centro de Informática Industrial

En el proceso de liberación se realizaron pruebas al sistema SIGEXT. Para la realización de estas pruebas se tuvo en cuenta el método de Caja Negra y como técnica la Partición de Equivalencia. Con el objetivo de lograr una mejor organización de la ejecución de las pruebas se diseñaron los casos de prueba. Se detectaron un total de 29 no conformidades. A medida que se fue avanzando en las iteraciones disminuyeron el número de no conformidades hasta no quedar ninguna, demostrándose de esta manera que el sistema estaba listo para ser utilizado. En la figura 8 se resume las no conformidades detectadas en cada una de las iteraciones.

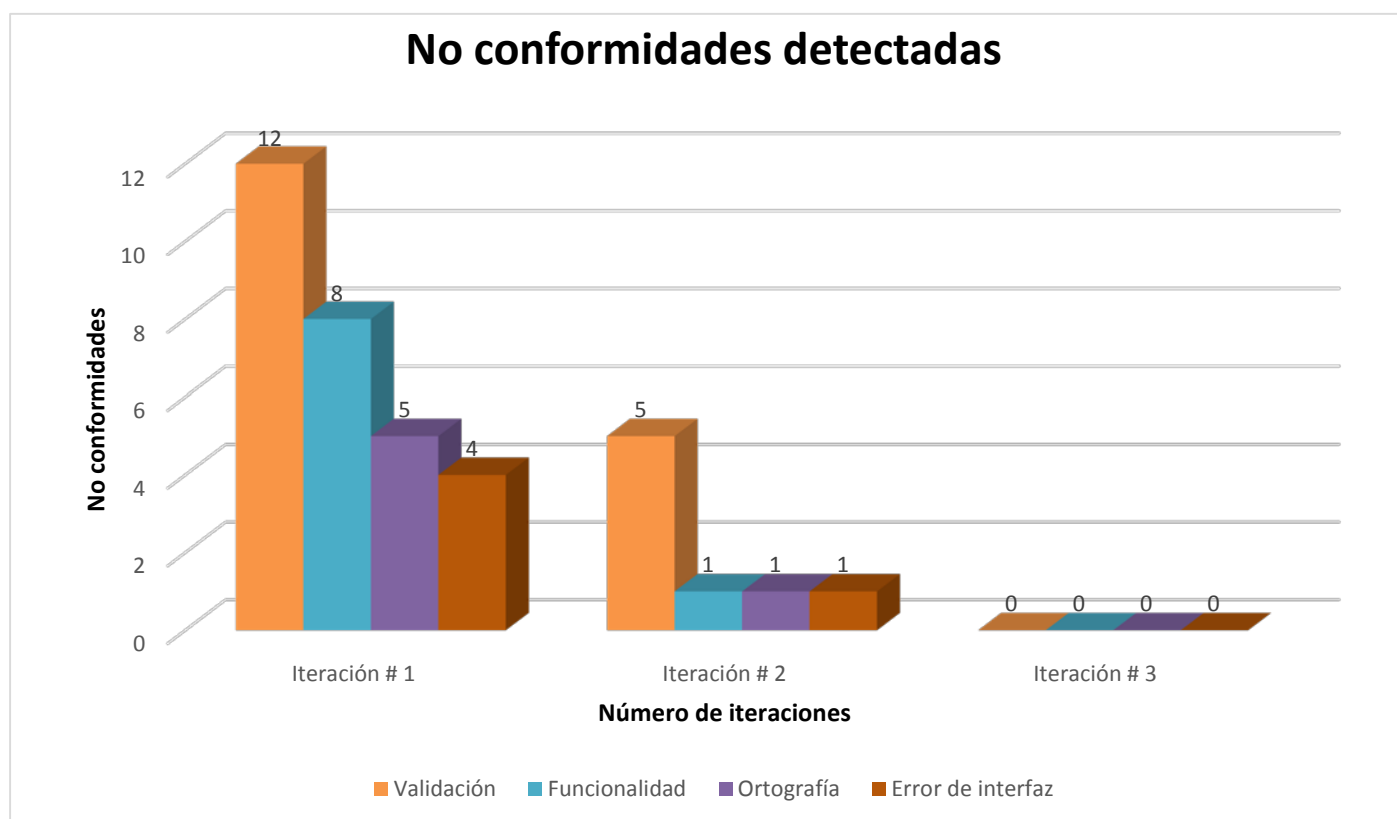


FIGURA 8 RESULTADOS DE LAS NO CONFORMIDADES DETECTADAS

Capítulo III Implementación y prueba

Conclusiones parciales

En este capítulo se definió el estándar de codificación a tener en cuenta para la implementación del sistema SIGEXT. Se desarrollaron las principales tareas de ingeniería para dar solución a las Historias de Usuario. Se ejecutaron las pruebas de aceptación diseñadas para la validación, obteniendo resultados positivos. A partir del resultado arrojado por las pruebas realizadas se llegó a la conclusión de que SIGEXT se encuentra listo para su puesta en funcionamiento.

Conclusiones

- El estudio de diferentes fuentes bibliográficas, permitió obtener una visión de las principales ventajas y carencias que poseen los sistemas de gestión para los recursos tecnológicos.
- La implantación del sistema SIGEXT contribuyó a la gestión de la información en el Centro de Informática Industrial dando cumplimiento al objetivo de la investigación.
- La realización de las pruebas de caja negra contribuyeron a garantizar que el sistema cumple con la calidad requerida para ser desplegado en el Centro de Informática Industrial.
- Con la validación del sistema SIGEXT a través del acta de liberación interna del grupo de calidad del Centro de Informática Industrial y la firma del acta de aceptación por parte del cliente garantizó la calidad del sistema para ser utilizado en el centro.

Recomendaciones

- Implementar funcionalidades que permitan la integración con el cliente GRHS para garantizar una mejor integridad y confiabilidad en la información que se procesa.
- Desplegar la solución propuesta en el Centro de Informática Industrial para facilitar el proceso de toma de decisiones de sus directivos.

Referencias bibliográficas

1. **DANS, ENRIQUE.** Information Management. *Reflexiones sobre las tecnologías de la información.* [En línea] 28 de noviembre de 2006. [Citado el: 7 de septiembre de 2014.] Disponible en <http://informationmanagement.wordpress.com/category/gestion/gestion-de-la-informacion/>.
2. **SERAFÍN LINARES, A.M.** *SISTEMA AUTOMATIZADO PARA EL CONTROL Y GESTIÓN DEL TRANSPORTE EN EL GRUPO DE LA ELECTRÓNICA.* Universidad de las Ciencias Informáticas, La Habana 2007. 2007.0841_07.
3. **DÍAZ PÉREZ, MAYDELÍN.** *Características de los sistemas de información que permiten la gestión oportuna de la información y el conocimiento institucional.* [En línea] 13 de noviembre de 2009. [Citado el: 10 de septiembre de 2014.] Disponible en <http://scielo.sld.cu/scielo.php?script=sc>. ISSN 1561-2880.
4. **INSTITUTE, THE BRITISH STANDARDS.** *The British Standards Institute.* [En línea]. [Citado el: 10 de septiembre de 2014.] Disponible en <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion>.
5. **SÁNCHEZ MUGUERCIA, NILMAR.** *Aplicación de escritorio del Sistema de Entrega Digital.* Universidad de las Ciencias Informáticas, La Habana, 2013.
6. **BENT, KENT Y OTROS.** *Planning Extreme Programming.* s.l: Addison Wesley, 2001. ISBN: 0-201-71091-9.
7. **GÓMEZ, A Y WILSON, J.** *Metodología de desarrollo de software un enfoque práctico y global versión 1.0.11.* [En línea] 23 de marzo de 2011. Disponible en <http://www.otcolombia.com/documentos/mds360-1.0.11-beta.pdf>.
8. **BENT, KENT Y FOWLER, MARTIN.** *Planning Extreme Programming.* s.l: Addison Wesley, 2001. ISBN: 0-201-71091-9.
9. **JEFFRIES, RON Y OTROS.** *Extreme Programming Installed.* s.l: Addison Wesley, 2000. ISBN: 0-201-70842-6.
10. **JACOBSON Y OTROS.** *El proceso unificado de desarrollo de software.* Madrid: Pearson Education, 2000. ISBN:84-7829-036-2.
11. **BAISLEY, B.** *Unified Modeling Language Infrastructure.* s.l: Pearson Editorial, 2006.
12. **FRIAS, C.** *Desarrollo de una librería de componentes de interfaz gráfica de usuario basada en Swing Application Framework y AWT Framework para el desarrollo de la capa de presentación en aplicaciones de software de escritorio.* Universidad de las Ciencias Informáticas . La Habana, s.n, 2012.
13. **POTENCIER, F.** *Symfony 1.2, la guía definitiva.* 2010. ISBN-13: 978-1590597866.
14. **POYO GARRIDO, ALEJANDRO.** *Comparativa Framework.* [En línea] 2012. [Citado el: 26 de octubre de 2014.] Disponible en <http://es.scribd.com/doc/189203494/Comparativa-Framework>.

Referencias bibliográficas

15. **MENA, D.** Los IDE de desarrollo. [En línea] 2011. Disponible en <http://www.cursosporinternet.info/index.php/the-news/43-programacion/167-los-ides-de-programacion.html> .
16. **ÁLVAREZ, M.** Un IDE para el desarrollo de aplicaciones web, enfocado en Ajax y la Web 2.0. [En línea] 2013. Disponible en <http://www.desarrolloweb.com/articulos/aptana-studio.html>.
17. **CRUSAT Y VAQUER, MARC.** *Zend Studio: el entorno de desarrollo para PHP*. 2004.
18. **LÓPEZ MICHELONE, MANUEL.** PhpStorm. [En línea] 2013. Disponible en <http://www.unocero.com/2013/10/25/se-libera-phpstorm-8/>.
19. **LOUDEN, KENNETH.** *Lenguajes de programación: Principios y práctica*. s.l: Cengage Learning Editores, 2004.
20. **SÆTHER BAKKEN, STIG.** *Manual de PHP*. s.l: Free Software Foundation, 2001.
21. **PÉREZ, MAURI Y OTROS.** *Desarrollo del sistema de gestión de contenidos de la colección El Navegante en su versión multiplataforma*. La Habana: s.n, 2012.
22. **AJONA PRADO, GUILLERMO.** Página Web sobre HTML y CSS. [En línea] 2012. [Citado el: 26 de octubre de 2014.] Disponible en <https://belenus.unirioja.es/~guprado/pagweb/caraccss.html>.
23. **EGUILUZ PÉREZ, JAVIER.** *Introducción a JavaScript*. 2009.
24. **XAMPP.** [En línea] 2015. [Citado el: 16 de enero de 2015.] Disponible en <https://www.apachefriends.org/es/index.html>.
25. **TELLEZ, I Y OTROS.** *Sistema para la gestión de la información de los medicamentos, turnos médicos y vacunas en el hospital Ernesto Ché Guevara*. Universidad de las Ciencias informáticas. La Habana: s.n, 2012.
26. **GUERRERO, Z Y OTROS.** *Sistema de Captura y Transcripción de Audio*. Universidad de las Ciencias Informáticas. La Habana, 2009.
27. **SUEHRING, S.** *MySQL Bible*. New York: Wiley Publishing Inc, 2001. ISBN: 0-7645-4932-4.
28. **LARMAN, CRAIG.** *UML y Patrones*. s.l: Prentice Hall, 2003.
29. **LETELIER, P. Y OTROS.** *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. Universidad de Valencia. Valencia: Valencia: Universidad de Valencia: s.n, 2008. Vol. 05, 26, 2008. Vols. 05,26. ISSN 1666-1680.
30. **LETELIER.** *Metodologías ágiles en el desarrollo de software*. Valencia: s.n, 2007.
31. **RIZO CONSTANTEN, ROSA MARÍA.** *Sistema de Comunicación interna. Tesis de grado para optar por el título de Ingeniera en Ciencias Informáticas*. Universidad de las Ciencias Informáticas, La Habana, 2013.
32. **FOWLER, MARTÍN.** *Pattern Enterprise Application Architecture*. s.l: Addison Wesley, 2002. ISBN: 0-321-12742-0.

Referencias bibliográficas

33. **SÁNCHEZ GONZÁLEZ, CARLOS.** *ONess: un proyecto open source para el negocio textil mayorista desarrollado con tecnologías open source innovadoras.* Caruña: s.n, 2004.
34. **WINESETT, JEFFERY.** *Agile Web Application Development with Yii 1.1 and PHP 5.* s.l: Birmingham: Packt Publishing Ltd, 2010. ISBN 978-1-847199-58-4.
35. **WALLACE.** *Extreme Programming for Web Projects.* s.l: Addison Wesley, 2002. ISBN: 0-201-79427-6.
36. **CASAS, S. Y OTROS.** *Identificación y Modelado de Aspectos Tempranos dirigido por Tarjetas de Responsabilidades y Colaboraciones.* 2008. <http://www.oocities.org/espanol/profeprog2/INVPAPER25.pdf>.
37. **RODRIGUEZ CABRERA, M. Y OTROS.** *La Metodología XP Aplicable al Desarrollo del Software educativo en Cuba.* [En línea] 2007. Disponibles en http://repositorio_institucional.uci.cu//jspui/handle/ident/.TD_0837_07.
38. **KOSKELA, LASSE.** *TEST DRIVEN, Practical TDD and Acceptance TDD for Java Developers.* s.l: Manning Publication Co, 2008. ISBN 1-932394-85-0.
39. **ITSON, BIBLIOTECA.** [En línea] 2014. [Citado el: 3 de diciembre de 2014.] Disponible en http://biblioteca.itson.mx/oa/dip_ago/introduccion_sistemas/p3.htm.
40. **ESTRADA, SALVADOR Y OTROS.** *Gestión de recursos tecnológicos. Doctorado Interuniversitario en Economía y Gestión de la Innovación y Política Tecnológica.* Madrid, España: s.n, 2001.
41. **REAL ACADEMIA DE LA LENGUA ESPAÑOLA.** Real Academia Española. [En línea] 2013. Disponible en <http://lema.rae.es/drae/?val=informaci%C3%B3n>.
42. **PRESSMAN, ROGER.** *Ingeniería de Software. Un enfoque práctico.* 2008. ISBN 978-0-07-337597-7.
43. **DAVIS, G. Y OLSÓN.** *Management Information Systems: Conceptual foundations, Structure and Development.* Medellín: Universidad de Antioquia, 1985.
44. **GAMMA, ERICH Y OTROS.** *Design Patterns: Elements of Reusable Object-Oriented Software.* s.l: KevinZhang, 1997.

Anexos

Anexo no. 1. Cuestionario para conocer la necesidad de la investigación

Estimado especialista con el siguiente cuestionario nos proponemos conocer la importancia que tendría el desarrollo de un sistema informático para mejorar la gestión de la información de los expedientes técnicos en el Centro de Informática Industrial. Los resultados de esta investigación arrojarán elementos importantes que servirán de base para la investigación.

1. ¿Cree usted importante llevar el control de dispositivos de entrada y salida de cada estación de trabajo en el Centro de Informática Industrial? Si__ No__ Tal vez__.
2. ¿Considera usted necesario que cada trabajador del Centro de Informática Industrial tenga acceso a visualizar la información contenida en su expediente técnico? Si__ No__ Tal vez__.
3. ¿Cree usted necesario la utilización de un sistema informático para la gestión de los expedientes técnicos? Si__ No__ Tal vez__.

Resultados de la encuesta

TABLA 22 IMPORTANCIA DEL CONTROL DE DISPOSITIVOS

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos	No	6	21,4	21,4
	Tal vez	2	7,1	28,6
	Si	20	71,4	100,0
Total	28	100,0	100,0	

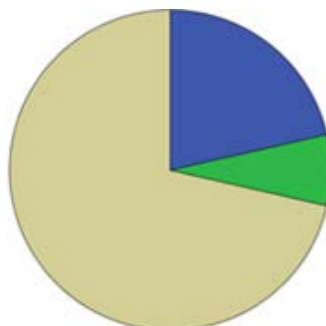


TABLA 23 NECESIDAD DE ACCESO AL ESTADO DE DISPOSITIVOS

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
No	2	7,1	7,1	7,1
Tal vez	1	3,6	3,6	10,7
Válidos Si	25	89,3	89,3	100,0
Total	28	100,0	100,0	

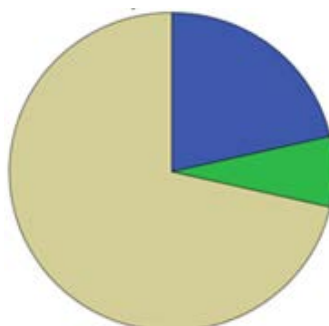


TABLA 24 NECESIDAD DE SISTEMA INFORMÁTICO

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Tal vez	1	3,6	3,6	3,6
Válidos Si	27	96,4	96,4	100,0
Total	28	100,0	100,0	

