



Facultad 10

**Sistema de Gestión de Auditoría de la Dirección de
Supervisión y Control de la Universidad de las Ciencias
Informáticas.**



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

Autores:

Arlee Enedina Paneque García

Noel González Fernández

Tutor:

Ing. Danayi Daniela Hernández Laguna

Ciudad de la Habana

2010, Año 52 de la Revolución.

Declaración de Autoría

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

Para que así conste firmo la presente a los ____ días del mes de _____ del año_____.

Arlee Enedina Paneque García
Autor

Noel González Fernández
Autor

Ing. Danayi Daniela Hernández Laguna
Tutor

Resumen

En este trabajo se presenta un sistema informático con el objetivo de gestionar la documentación resultante del proceso de auditoría que se realiza en la Dirección de Supervisión y Control de la Universidad de las Ciencias Informáticas.

Este software tiene funcionalidades como modelar el proceso de auditoría a través de la gestión de los diferentes papeles de trabajo, llevar el control de las empresas auditadas y un listado de los auditores que pertenecen a la Dirección y la realización de reportes estadísticos.

Este sistema está desarrollado bajo herramientas libres, y adaptado a las necesidades específicas de la organización para la cual fue hecho. En el trabajo se detallan aspectos concernientes a las auditorías y los sistemas que permiten su automatización, se utiliza como metodología de desarrollo XP, lenguaje de programación Python y Gestor de Base de Datos MySQL.

ÍNDICE DE CONTENIDOS

INTRODUCCIÓN.....	- 1 -
CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA	- 5 -
1.1 INTRODUCCIÓN.....	- 5 -
1.2 PROCESO DE AUDITORÍA	- 5 -
1.2.1 Auditoría y Auditoría Interna.....	- 5 -
1.2.2 Informe Final y Papeles de Trabajo.....	- 7 -
1.2.3 Proceso de Auditoría.....	- 8 -
1.2.4 SISTEMA DE GESTIÓN DE INFORMACIÓN.....	- 8 -
1.3 ESTADO DEL ARTE.....	- 9 -
1.3.1 Sistemas de Gestión de Auditorías en el mundo.....	- 9 -
1.3.2 Sistemas de Gestión de Auditorías en Cuba y la UCI.....	- 10 -
1.4 METODOLOGÍAS DE DESARROLLO	- 11 -
1.4.1 Proceso Unificado de Desarrollo (RUP).....	- 11 -
1.4.2 Programación Extrema (XP).....	- 12 -
1.4.3 Scrum.....	- 13 -
1.4.4 Selección de la Metodología de Desarrollo a utilizar.....	- 14 -
1.5 LENGUAJES DE PROGRAMACIÓN	- 15 -
1.5.1 Java.....	- 15 -
1.5.2 C#.....	- 15 -
1.5.3 Perl.....	- 16 -
1.5.4 C ++.....	- 17 -
1.5.5 Python.....	- 17 -
1.5.6 Selección del Lenguaje de Programación a utilizar.....	- 18 -
1.6 HERRAMIENTAS DE DESARROLLO.....	- 18 -
1.6.1 Umbrello.....	- 18 -
1.6.2 Visual Paradigm	- 19 -
1.6.3 Rational Rose.....	- 19 -
1.6.4 Selección de las Herramientas de Desarrollo a utilizar.....	- 20 -
1.7 SISTEMAS GESTORES DE BASES DE DATOS.....	- 21 -
1.7.1 MySql.....	- 21 -
1.7.2 Oracle.....	- 22 -
1.7.3 PostgreSQL.....	- 23 -
1.7.4 Selección del Sistema Gestor de Base de Datos a utilizar.....	- 24 -
1.8 CONCLUSIONES.....	- 24 -
CAPÍTULO 2 DESCRIPCIÓN DE LA SOLUCIÓN.....	- 25 -
2.1 INTRODUCCIÓN.....	- 25 -
2.2 DESCRIPCIÓN DEL FLUJO ACTUAL DEL PROCESO DE AUDITORÍA EN LA DIRECCIÓN DE SUPERVISIÓN Y CONTROL DE LA UCI.....	- 25 -
2.3 DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA	- 27 -

2.5 REQUISITOS NO FUNCIONALES DEL SISTEMA	- 28 -
2.5.1 <i>Requerimientos de apariencia o interfaz externa</i>	- 29 -
2.5.2 <i>Requerimientos de usabilidad</i>	- 29 -
2.5.4 <i>Requerimientos de rendimiento</i>	- 29 -
2.5.5 <i>Requerimientos de portabilidad</i>	- 29 -
2.5.6 <i>Requerimientos de seguridad</i>	- 29 -
2.5.7 <i>Requerimientos de software</i>	- 30 -
2.6 PLANIFICACIÓN	- 30 -
2.6.1 <i>Historias de usuarios</i>	- 30 -
2.6.2 <i>Plan de Iteraciones</i>	- 34 -
2.6.3 <i>Plan de entrega de versiones</i>	- 35 -
2.7 CONCLUSIONES	- 35 -
CAPÍTULO 3 DISEÑO E IMPLEMENTACIÓN	- 36 -
3.1 INTRODUCCIÓN	- 36 -
3.2 TARJETAS CRC “CLASE, RESPONSABILIDAD Y COLABORACIÓN”	- 36 -
3.3 DISEÑO DE LA BASE DE DATOS.....	- 38 -
3.4 TAREAS DE LA INGENIERÍA.	- 40 -
3.4.1 <i>Desarrollo de las tareas de la ingeniería</i>	- 41 -
3.4.2 <i>Iteración 1</i>	- 41 -
3.4.3 <i>Iteración 2</i>	- 44 -
3.4.4 <i>Iteración 3</i>	- 49 -
3.5 CONCLUSIONES	- 52 -
CAPÍTULO 4 PRUEBA	- 53 -
4.1 INTRODUCCIÓN	- 53 -
4.2 PRUEBAS DE ACEPTACIÓN.	- 53 -
4.2.1 <i>Iteración 1</i>	- 53 -
4.2.2 <i>Iteración 2</i>	- 58 -
4.2.3 <i>Iteración 3</i>	- 62 -
4.3 CONCLUSIONES	- 65 -
CONCLUSIONES GENERALES	- 66 -
RECOMENDACIONES	- 67 -
REFERENCIAS BIBLIOGRÁFICAS	- 68 -
BIBLIOGRAFÍA	- 69 -
GLOSARIO DE TÉRMINOS	- 71 -

ÍNDICE DE ILUSTRACIONES

ILUSTRACIÓN 1 CICLO DE VIDA RUP	- 12 -
ILUSTRACIÓN 2 CICLO DE VIDA XP	- 13 -
ILUSTRACIÓN 3 CICLO DE VIDA SCRUM.....	- 14 -
ILUSTRACIÓN 4 ARQUITECTURA DEL SISTEMA	- 28 -
ILUSTRACIÓN 5 MODELO DE DATOS DEL SISTEMA.....	- 40 -

ÍNDICE DE TABLAS

TABLA 1 TRABAJADORES Y RESPONSABILIDADES	- 26 -
TABLA 2 ROLES Y NIVELES DE ACCESO.....	- 28 -
TABLA 3 REPRESENTACIÓN DE HU 1.....	- 31 -
TABLA 4 REPRESENTACIÓN DE HU 2.....	- 31 -
TABLA 5 REPRESENTACIÓN DE HU 3.....	- 32 -
TABLA 6 REPRESENTACIÓN DE HU 4.....	- 32 -
TABLA 7 REPRESENTACIÓN DE HU 5.....	- 33 -
TABLA 8 REPRESENTACIÓN DE HU 6.....	- 33 -
TABLA 9 REPRESENTACIÓN DE HU 7.....	- 34 -
TABLA 10 REPRESENTACIÓN DE HU 8.....	- 34 -
TABLA 11 PLAN DE ITERACIONES	- 35 -
TABLA 12 PLAN DE ENTREGA DE VERSIONES	- 35 -
TABLA 13 TARJETA CRC 1	- 37 -
TABLA 14 TARJETA CRC 2.....	- 37 -
TABLA 15 TARJETA CRC 3.....	- 37 -
TABLA 16 TARJETA CRC 4.....	- 37 -
TABLA 17 TARJETA CRC 5.....	- 38 -
TABLA 18 TARJETA CRC 6.....	- 38 -
TABLA 19 TAREAS DE LA INGENIERÍA.....	- 41 -
TABLA 20 DESCRIPCIÓN DE LA TI 1 AUTENTICAR USUARIO	- 41 -
TABLA 21 DESCRIPCIÓN DE LA TI 1 ADMINISTRAR AUDITOR	- 42 -
TABLA 22 DESCRIPCIÓN DE LA TI 2 ADMINISTRAR AUDITOR.....	- 42 -
TABLA 23 DESCRIPCIÓN DE LA TI 3 ADMINISTRAR AUDITOR.....	- 43 -
TABLA 24 DESCRIPCIÓN DE LA TI 1 ADMINISTRAR EMPRESA	- 43 -
TABLA 25 DESCRIPCIÓN DE LA TI 2 ADMINISTRAR EMPRESA	- 44 -
TABLA 26 DESCRIPCIÓN DE LA TI 3 ADMINISTRAR EMPRESA	- 44 -
TABLA 27 DESCRIPCIÓN DE LA TI 1 ADMINISTRAR DOCUMENTO	- 45 -
TABLA 28 DESCRIPCIÓN DE LA TI 2 ADMINISTRAR DOCUMENTO	- 45 -
TABLA 29 DESCRIPCIÓN DE LA TI 3 ADMINISTRAR DOCUMENTO	- 46 -
TABLA 30 DESCRIPCIÓN DE LA TI 4 ADMINISTRAR DOCUMENTO	- 46 -
TABLA 31 DESCRIPCIÓN DE LA TI 5 ADMINISTRAR DOCUMENTO	- 47 -
TABLA 32 DESCRIPCIÓN DE LA TI 1 ADMINISTRAR EXPEDIENTE.....	- 47 -
TABLA 33 DESCRIPCIÓN DE LA TI 2 ADMINISTRAR EXPEDIENTE	- 47 -

TABLA 34 DESCRIPCIÓN DE LA TI 3 ADMINISTRAR EXPEDIENTE	- 48 -
TABLA 35 DESCRIPCIÓN DE LA TI 4 ADMINISTRAR EXPEDIENTE	- 48 -
TABLA 36 DESCRIPCIÓN DE LA TI 1 DECLARAR DOCUMENTO FINAL.....	- 49 -
TABLA 37 DESCRIPCIÓN DE LA TI 1 REALIZAR ESTADÍSTICAS	- 49 -
TABLA 38 DESCRIPCIÓN DE LA TI 2 REALIZAR ESTADÍSTICAS	- 50 -
TABLA 39 DESCRIPCIÓN DE LA TI 3 REALIZAR ESTADÍSTICAS	- 50 -
TABLA 40 DESCRIPCIÓN DE LA TI 4 REALIZAR ESTADÍSTICAS	- 51 -
TABLA 41 DESCRIPCIÓN DE LA TI 5 REALIZAR ESTADÍSTICAS	- 51 -
TABLA 42 DESCRIPCIÓN DE LA TI 1 EDITAR PROPIEDADES.....	- 52 -
TABLA 43 DESCRIPCIÓN DE LA PA 1 AUTENTICAR USUARIO	- 54 -
TABLA 44 DESCRIPCIÓN DE LA PA 2 AUTENTICAR USUARIO.....	- 54 -
TABLA 45 DESCRIPCIÓN DE LA PA 1 ADMINISTRAR AUDITOR.....	- 55 -
TABLA 46 DESCRIPCIÓN DE LA PA 2 ADMINISTRAR AUDITOR.....	- 55 -
TABLA 47 DESCRIPCIÓN DE LA PA 3 ADMINISTRAR AUDITOR.....	- 56 -
TABLA 48 DESCRIPCIÓN DE LA PA 4 ADMINISTRAR AUDITOR.....	- 56 -
TABLA 49 DESCRIPCIÓN DE LA PA 1 ADMINISTRAR EMPRESA	- 57 -
TABLA 50 DESCRIPCIÓN DE LA PA 2 ADMINISTRAR EMPRESA	- 57 -
TABLA 51 DESCRIPCIÓN DE LA PA 3 ADMINISTRAR EMPRESAS.....	- 58 -
TABLA 52 DESCRIPCIÓN DE LA PA 4 ADMINISTRAR EMPRESAS.....	- 58 -
TABLA 53 DESCRIPCIÓN DE LA PA 1 ADMINISTRAR DOCUMENTOS.....	- 59 -
TABLA 54 DESCRIPCIÓN DE LA PA 2 ADMINISTRAR DOCUMENTOS.....	- 59 -
TABLA 55 DESCRIPCIÓN DE LA PA 3 ADMINISTRAR DOCUMENTOS.....	- 60 -
TABLA 56 DESCRIPCIÓN DE LA PA 4 ADMINISTRAR DOCUMENTOS.....	- 60 -
TABLA 57 DESCRIPCIÓN DE LA PA 1 ADMINISTRAR EXPEDIENTES	- 61 -
TABLA 58 DESCRIPCIÓN DE LA PA 2 ADMINISTRAR EXPEDIENTES	- 61 -
TABLA 59 DESCRIPCIÓN DE LA PA 1 MODIFICAR EXPEDIENTES.....	- 62 -
TABLA 60 DESCRIPCIÓN DE LA PA 1 DECLARAR DOCUMENTO FINAL.....	- 62 -
TABLA 61 DESCRIPCIÓN DE LA PA 1 REALIZAR ESTADÍSTICAS	- 63 -
TABLA 62 DESCRIPCIÓN DE LA PA 2 REALIZAR ESTADÍSTICAS	- 63 -
TABLA 63 DESCRIPCIÓN DE LA PA 3 REALIZAR ESTADÍSTICAS	- 64 -
TABLA 64 DESCRIPCIÓN DE LA PA 4 REALIZAR ESTADÍSTICAS	- 64 -
TABLA 65 DESCRIPCIÓN DE LA PA 1 EDITAR PROPIEDADES	- 65 -
TABLA 66 DESCRIPCIÓN DE LA PA 2 EDITAR PROPIEDADES	- 65 -

INTRODUCCIÓN

En cada entidad pública debe existir alguna herramienta de control que permita constatar que los procesos que se desarrollan internamente están correctos y se corresponden con las disposiciones y leyes vigentes en la constitución cubana, para este objetivo surge la Auditoría Interna, que no es más que “un proceso sistemático, practicado por personal especializado de conformidad con normas y procedimientos técnicos establecidos, consistente en obtener y evaluar objetivamente las evidencias sobre las afirmaciones contenidas en los actos jurídicos o eventos de carácter técnico, económico, administrativo y otros, con el fin de determinar el grado de correspondencia entre esas afirmaciones, las disposiciones legales vigentes y los criterios establecidos.”¹, que se realiza por diferentes organismos y órganos del estado cubano, en la Universidad de las Ciencias Informáticas(UCI), la unidad encargada de la realización de la misma es la Dirección de Supervisión y Control, la cual tiene la tarea de valorar de forma independiente las actividades que se realizan en la UCI con el objetivo de contribuir a prevenir y detectar las indisciplinas, ilegalidades y manifestaciones de corrupción que pudieran afectar en el control de los recursos materiales, humanos y financieros de los que se dispone.

“La Auditoría Interna está llamada a convertirse, en una actividad planificada que contribuya al cumplimiento de las metas y objetivos planteados en esta, que agregue valor y mejore las operaciones de la entidad, enfocada sistemática y disciplinadamente a evaluar y mejorar la eficacia de los procesos de gestión de riesgos, control y dirección. Los servicios de auditoría comprenden la evaluación objetiva de las evidencias, efectuada por los auditores internos, para proporcionar una conclusión independiente que permita calificar el cumplimiento de las políticas, reglamentaciones, normas, disposiciones u otros requerimientos legales; respecto a un sistema, proceso, subproceso o actividad.”¹

El trabajo realizado por los supervisores debe reflejarse en un documento escrito como constancia de esta acción, que debe reflejar las insuficiencias o fallas técnicas detectadas y las recomendaciones que de ellas se deriven. Este documento forma parte de un expediente que incluye todos los papeles de trabajo que sean generados de la supervisión y control, estos documentos deben ser comprensibles y legibles, reflejando un diseño apropiado de acuerdo con el objetivo que se persigue y

¹ Ministerio de Auditoría y Control. Gaceta Oficial no 003 22 de enero del 2008. Resolución 350/07.

otras cuestiones de forma, con suficiente información, de manera que no requieran explicaciones complementarias, y con la constancia de todas las revisiones efectuadas por los funcionarios facultados para ello, es decir, toda la documentación que se obtiene esta realizada a papel y lápiz, por lo que la información sensible está expuesta a diferentes riesgos de seguridad como pérdida o daño en su traslado o manejo dentro del local donde se guarda, en el cual pueden sufrir accidentes tales como incendio o inundación, perdiéndose parcial o totalmente los datos que contiene, ya que, debido al formato en que se archivan se hace compleja la realización de una salva de la información guardada.

En dependencia de la complejidad y profundidad del trabajo que se esté realizando en la entidad auditada, la masa de documentos generada puede ser grande y requiere ser archivada, además de que el acceso a un documento o dato determinado se hace más difícil y consume gran cantidad de tiempo y esfuerzo del personal, no permite un análisis estadístico profundo o que incluya una gran cantidad de elementos y no se tiene un control de la información organizada por aspectos como por ejemplo: por empresas, por fechas, por auditores y trae como consecuencia además que los reportes y estadísticas resultantes de los procesos de auditorías que se llevan a cabo presenten un alto nivel de dificultad en su realización. La información no está disponible las 24 horas del día y para estudiarla a fondo demanda gran cantidad de esfuerzo y tiempo y algunas veces resulta ineficaz la búsqueda.

Partiendo de lo antes expuesto se plantea el siguiente **Problema Científico**:

¿Cómo informatizar la gestión de información en el Proceso de Auditoría de la Dirección de Supervisión y Control de la UCI?

Siendo el **Objeto de Estudio** de la investigación que se presenta: el Proceso de Auditoría en la Dirección de Supervisión y Control de la UCI.

De ahí que el **Objetivo General** sea:

Desarrollar un sistema para la gestión de información en el Proceso de Auditoría de la Oficina de Supervisión y Control de la UCI.

Y derivándose de este los siguientes **Objetivos Específicos**:

- Definir la metodología, las herramientas y el lenguaje que intervendrán en el desarrollo del sistema.
- Analizar y diseñar el sistema.
- Realizar la implementación y las pruebas del sistema.

Basados en estos objetivos se actuará sobre el **Campo de Acción**: los Flujos de Información del Proceso de Auditoría en la Dirección de Supervisión y Control de la UCI.

Para lograr el cumplimiento de los objetivos se proponen las siguientes **Tareas Investigativas**:

- Resumen de la información sobre los Procesos de Auditorías.
- Evaluación de los Sistemas de Gestión de Procesos de Auditorías que existan a nivel mundial, nacional y de la UCI.
 - Evaluación de las metodologías, herramientas y lenguajes necesarios en el desarrollo de sistemas.
 - Selección de la metodología, las herramientas y el lenguaje que serán utilizados en el desarrollo del sistema.
 - Definición de las características y los requisitos del software.
 - Realización del análisis y diseño del sistema.
 - Implementación del sistema.
 - Realización de las pruebas al sistema.

Los **métodos teóricos** usados para dar cumplimiento a las tareas a desarrollar son:

Análisis Histórico-Lógico: Se usa este método para estudiar las formas de solución dadas de años anteriores a problemas similares al planteado, además de entender de forma teórica el Proceso de Auditoría. “A través de este método se explica la historia de su desarrollo, reproducen el objeto en su forma superior y permiten unir el estudio de la estructura del objeto de investigación con su concepción histórica”. ²

² Hernández León R.A. y Coello González S. *El Paradigma Cuantitativo de la investigación Científica*. Noviembre 2002.

Modelación: “La modelación es el método mediante el cual se crean abstracciones con el objetivo de explicar la realidad”.² Se utilizará para crear un modelo del Proceso de Auditoría para poderlo llevar a un sistema que cumpla los objetivos propuestos.

Los **métodos empíricos** utilizados para obtener información sobre el objeto de estudio son:

Entrevista: Se utilizará como medio para obtener el conocimiento cualitativo de los problemas que presenta el Proceso de Auditoría, sus basamentos legales y sociales y una descripción del proceso que incluya, etapas, acciones y flujos de información que se generan.

Observación No incluida: Se definirá qué parte del Proceso de Auditoría se va a observar de acuerdo con los objetivos planteados, utilizando un control adecuado y realizándose de forma reiterada para “conocer la realidad mediante la percepción directa de los objetos y el fenómeno.”²

La investigación científica se compondrá de cuatro capítulos:

Capítulo 1. Fundamentación Teórica: En este capítulo se valoran los sistemas de auditoría que existen relacionados con el problema planteado, se hace un análisis de metodologías y herramientas de desarrollo y lenguajes de programación necesarios para el desarrollo de la solución propuesta.

Capítulo 2. Descripción de la Solución: En este capítulo se expone el flujo básico de acción del proceso de auditoría, se definen roles y responsabilidades, se propone una arquitectura y se plantean los requisitos no funcionales del software. Se da inicio a la fase de Planificación, identificando las historias de usuarios y la cantidad de iteraciones.

Capítulo 3. Diseño e implementación: En este capítulo se plantean las tareas de la ingeniería, necesarias para dar cumplimiento a las historias de usuarios, se presenta el modelo de datos del software y las relaciones entre clases que se describieron en el sistema.

Capítulo 4. Prueba: Este capítulo aborda los elementos y resultados obtenidos en las pruebas de aceptación.

CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En este capítulo se expondrán los conceptos relacionados con la actividad de auditoría interna en la universidad, además de analizar los sistemas de auditorías existentes en el mundo. Se analizarán las metodologías de desarrollo, lenguajes, herramientas y gestores de bases de datos que podrán ser utilizadas en el desarrollo de la solución.

1.2 Proceso de Auditoría

1.2.1 Auditoría y Auditoría Interna

Tal como se establece en la legislación vigente, la Auditoría se define como: “Un proceso sistemático, practicado por personal especializado de conformidad con normas y procedimientos técnicos establecidos, consistente en obtener y evaluar objetivamente las evidencias sobre las afirmaciones contenidas en los actos jurídicos o eventos de carácter técnico, económico, administrativo y otros, con el fin de determinar el grado de correspondencia entre esas afirmaciones, las disposiciones legales vigentes y los criterios establecidos”.³ Entre los tipos de Auditoría que se realizan en nuestro país se encuentran la gubernamental, la externa y la interna.

La auditoría gubernamental es aquella que se realiza por contralores, miembros de la Contraloría General de la República y que está incluida dentro del plan anual de las acciones de auditoría, supervisión y control.

La auditoría externa es realizada por auditores que no pertenecen a la institución que se va a auditar, y “consiste en el examen o verificación de las transacciones, cuentas, informes, o estados financieros correspondientes a un período, evaluando el cumplimiento de las disposiciones legales vigentes y el sistema de control interno.”⁴

³ Ministerio de Auditoría y Control. *Gaceta Oficial no 003 22 de enero del 2008. Resolución 350/07.*

⁴ Asamblea Nacional Poder Popular. *Ley No. 107 De la Contraloría General de la República de Cuba. Agosto 2009.*

La auditoría interna es empleada por las organizaciones para la valoración independiente de sus actividades, con el objetivo de contribuir a prevenir y detectar las indisciplinas, ilegalidades y manifestaciones de corrupción que pudieran afectar en el control de los recursos materiales, humanos y financieros de que dispone, y que se rige por la Resolución 350 del Contraloría General de la República, dentro de esta se definen los conceptos y se regulan los procedimientos que tienen que ver con las actividades que realiza el auditor interno.

Entre sus principales funciones tiene la de contribuir al cumplimiento de los objetivos y metas a partir de un enfoque sistemático y disciplinado para evaluar y mejorar la eficacia de los procesos de gestión de riesgos, control y dirección, agregando así valor, y mejorar las operaciones de una organización. La Auditoría Interna se ajusta por diversas normas que incluyen las Normas Personales y Generales, las Normas de Ejecución y las Normas para la Presentación de Informes, ya que estas son las responsables de proveer pautas y lineamientos para la realización de la misma, de establecer las bases para medir su desempeño y fomentar la mejora en los procesos y operaciones de la entidad.

Según los objetivos fundamentales que se persiguen las auditorías se clasifican:

- **Auditoría de Gestión o Rendimiento:** consiste en el examen y evaluación de la gestión de un órgano, organismo, entidad, programa, proyecto, proceso o actividad, para establecer el grado de economía, eficiencia, eficacia, calidad e impacto de su desempeño en la planificación, control y uso de los recursos y en la conservación y preservación del medio ambiente, así como comprobar la observancia de las disposiciones que le son aplicables.
- **Auditoría Financiera o de Estados Financieros:** Consiste en el examen y evaluación de los documentos, operaciones, registros y estados financieros de la entidad, para determinar si éstos reflejan, razonablemente, su situación financiera y los resultados de sus operaciones, así como el cumplimiento de las disposiciones económico-financieras, con el objetivo de mejorar los procedimientos relativos a su gestión y el control interno.
- **Auditoría Forense:** Consiste en la investigación y verificación de información, operaciones, actividades y otras, para reunir y presentar el soporte técnico que sustente presuntos hechos delictivos y de corrupción.

- **Auditoría de Cumplimiento:** Es la comprobación, evaluación y examen que se realiza por especialistas del Derecho, con el objetivo de verificar el cumplimiento de las disposiciones jurídicas, reglamentarias, estatutarias y de procedimientos inherentes a la actividad sujeta a revisión, vinculando la eficacia de la norma en relación con los objetivos y metas de la entidad.
- **Auditoría Fiscal:** Consiste en el examen de las operaciones relacionadas con los tributos al fisco, a los que están obligadas las personas jurídicas o personas naturales con obligaciones tributarias y no tributarias, con el objetivo de determinar si se efectúan en la cuantía que corresponda, dentro de los plazos y formas establecidas y proceder conforme a derecho.
- **Auditoría de Tecnologías de la Información y las Comunicaciones:** Consiste en el examen de las políticas, procedimientos y utilización de los recursos informáticos; así como de la confiabilidad y validez de la información, la efectividad de los controles, aplicaciones, sistemas de redes y otros vinculados a la actividad informática.
- **Auditoría de Medio Ambiente:** Es el proceso para verificar el uso, administración, protección, preservación del medio ambiente y de los recursos naturales, con el objetivo de evaluar el cumplimiento de las normas y principios que rigen su control y cuando proceda, cuantificar el impacto por el deterioro ocasionado o que pueda producirse.
- **Auditoría Especial:** Consiste en la verificación y análisis de asuntos, temas y actividades, que responden a una necesidad específica.

1.2.2 Informe Final y Papeles de Trabajo

El trabajo realizado por los auditores debe reflejarse en un documento escrito como constancia de esta acción, que debe reflejar las insuficiencias o fallas técnicas detectadas en el organismo y las recomendaciones que de ellas se deriven. Este documento es el Informe Final en el cual se exige que sea fácil de leer y entender. Los informes deben redactarse en un lenguaje tan claro y sencillo como lo permitan los asuntos que traten. Tanto la organización lógica de la información como la exactitud y precisión para exponer los hechos y formular conclusiones son esenciales para la claridad y la comprensión. Cuando es conveniente, se utilizan elementos de apoyo visual (fotografías, diagramas, gráficos y mapas) para aclarar y resumir información compleja y de difícil comprensión.

Conjuntamente con este informe, en la auditoría se utiliza un gran número de documentos, algunos en formularios preestablecidos y otros preparados de acuerdo con proformas diseñadas al efecto, lo que facilita el trabajo del auditor, a estos documentos se les denomina Papeles de Trabajo. Cuando un

papel de trabajo ha sido revisado y se determina por parte del Jefe de Grupo que no se le va a realizar ninguna otra modificación este documento pasa a la condición de Papel de Trabajo Final.

1.2.3 Proceso de Auditoría

Una auditoría da inicio cuando se le ordena al Jefe de Grupo de un grupo de auditores que realice una determinada auditoría, estimando el alcance y extensión del trabajo, esto ocurre y se documenta a través de una Orden de Trabajo, en este papel de trabajo se debe incluir información que muestre los resultados de la auditoría, se registran datos de carácter estadístico para su uso y decisiones futuras.

El Proceso de Auditoría engloba todas las actividades que se realizan desde la indicación, planeamiento y diseño de una auditoría a una entidad u organización que se incluyen en la Orden de Trabajo, hasta la presentación del Informe Final que ha sido revisado por los supervisores. Una parte esencial de este proceso constituyen los Flujos de Información que incluye toda la información que se maneja durante su desarrollo, es decir, la gestión, documentación, recopilación y archivo de toda la información resultante del Proceso de Auditoría.

1.2.4 Sistema de Gestión de Información.

La información dentro de una institución constituye uno de los pilares más importantes a la hora del éxito y el desarrollo de la misma pues “los participantes de una acción humana colectiva toman sus decisiones tanto en base a sus propios puntos de referencia (morales, culturales, etc.) como en función de las informaciones que reciben”⁵, por esta razón es tan importante conocer qué cosa es un sistema de información y cómo influye en los mismos la Informática como ciencia.

La informática ha evolucionado a lo largo de su historia hacia el constante mejoramiento del hardware y la arquitectura de los computadores, del software de las aplicaciones que se desarrollan, los compiladores y las facilidades de los lenguajes de alto nivel, a las redes con la amplia cobertura de comunicación y colaboración que significa Internet y todo aquello que se relaciona con las tecnologías de la informática. En especial ha tenido impacto en la manera de trabajar con datos e información, realizando operaciones a velocidades impresionantes, por lo que ha transformado los sistemas de

⁵ Berthet Ch. *Manual de Informática*. Argentina Julio 1988

información que no son más que “la trama de las organizaciones y de las empresas, cuyos flujos de materiales, productos y servicios comandan, acompañan y controlan.”⁵

Se considera para los fines de la investigación que un Sistema de Gestión de Información es todo aquel programa informático que permita la administración, organización, salva y manipulación de datos e información perteneciente a una organización o entidad determinada.

En el caso de la Dirección de Supervisión y Control de la UCI como entidad, se genera un sistema de información que engloba todas las actividades que se realizan en la misma, los resultados de las auditorías, los listados de contactos y empresas que atienden y las estadísticas e informes resultantes de los procesos. Para gestionar este cúmulo de información se requiere de sistemas adaptados a las necesidades de este tipo de organización que permita la administración de los datos y un flujo de información amigable a los usuarios del sistema, por esta razón se realiza un estado del arte sobre los sistemas de gestión de auditorías a nivel internacional y nacional, con el objetivo de realizar un análisis acerca de las características de estos sistemas y sus posibilidades de uso.

1.3 Estado del arte.

1.3.1 Sistemas de Gestión de Auditorías en el mundo

En el mundo existen programas que se han creado con el objetivo de apoyar distintos tipos de auditorías y automatizar los procesos que se llevan a cabo, haciendo énfasis en el análisis y el almacenamiento de los datos que se obtienen en las mismas. Entre ellos se encuentra:

AUDITWorks TM: Es un software que ofrece diversas funcionalidades de apoyo a la realización de auditorías. Tiene entre sus funcionalidades un marco preparado para guardar los resultados de las auditorías, una guía para la realización de las mismas e incluye la gestión de los datos y protocolos para la evaluación de la información. Tiene listas de control que contienen preguntas para diversos programas de auditorías. Está adaptado para participar en todo tipo de auditorías, mostrándole al usuario un software fácil de aprender.

DOC61: Es un software programado en Java que está diseñado para los auditores de sistemas. Te permite mantener y planear registros e imprimir informes. Los registros son guardados en formato

XML, aprovechando este formato portable y que puede ser convertido a archivos de bases de datos con facilidad.

CaseWare IDEA: Es un software que utiliza como plataforma Windows, brinda herramientas de análisis de datos estadísticos y de auditorías. Está creado para que los auditores agilicen el tiempo de análisis de los datos y accedan a los archivos que estén en cualquier entorno, y puedan crear sus propias bases de datos flexibles. Entre sus ventajas se encuentra la eliminación de los riesgos estadísticos de las muestras y apoyar la creatividad de los auditores, permite encontrar problemas potenciales e identificar vulnerabilidades. Este software automatiza en un lenguaje IDEAScript los programas de Análisis de datos y Auditoría y la creación de bibliotecas de acuerdo con el trabajo personalizado de los auditores.

AutoAudit: Es un sistema de gestión que permite automatizar una auditoría desde la fase de planificación, el trabajo de campo y la realización del Informe Final. Digitaliza los documentos y permite seguir una metodología de evaluación de riesgos a nivel de entidad o proceso. Gestiona la planificación y los recursos de una auditoría, seguimiento de hallazgos, reportes de tiempo y gastos, contiene un módulo de reportes y control de la calidad. Permite que los auditores puedan trabajar en lugares distantes con sus réplicas locales de la auditoría en curso y su posterior sincronización a la base de datos centralizada y mantiene una Biblioteca de Estándares de programas de auditoría, papeles de trabajo, memos, listas de chequeo, hallazgos, informes y otros.

AUDIEASY: Es un software que automatiza también los papeles de trabajo, facilitando su revisión y análisis. Permite la realización de reportes que se basan en observaciones, sugerencias, ajustes, cálculos de nominas de trabajador utilizando las tablas y tarifas actualizadas del ejercicio a auditar que son generados en cualquier momento y tomando como base los propios papeles de trabajo. Permite tener todas las informaciones contables de la empresa, las balanzas de comprobación, diarios, auxiliares de cuentas, estados financieros y saldos de las cuentas contables, con un mayor control de forma electrónica.

1.3.2 Sistemas de Gestión de Auditorías en Cuba y la UCI.

En el ámbito nacional no se conoce ningún sistema que permita la automatización de procesos de auditoría dado que el trabajo se realiza en documentos físicos.

En particular en la UCI, el Director de la Dirección de Supervisión y Control Msc Pablo Pita, desarrolló un software de apoyo a la gestión de información que realizaba en su oficina. Fue presentado en diversos eventos científicos bajo el nombre de GAudit. Está creado en VisualFoxPro 9, contiene una base de datos donde se guardan las empresas a auditar, los principales documentos digitales de las auditorías y el registro de los auditores. Brinda opciones de impresión de documentos y conversión a formato portable. Tiene como desventaja que está creado para una sola máquina cliente, solo puede ser instalado en una estación de trabajo y no permite conexiones a la base de datos desde otras máquinas.

Realizado el análisis del estado de arte se llega a la conclusión de que no existe en estos momentos una alternativa de software libre que cumpla con los requisitos de las auditorías que se realizan en la Dirección de Supervisión y Control de la UCI, que esté adaptada a las necesidades específicas de esta organización y guiada por las normas que están establecidas en nuestro país.

1.4 Metodologías de Desarrollo

A continuación se brindan características de algunas metodologías de desarrollo que existen a fin de definir cuál será la que se usará para el desarrollo del sistema.

1.4.1 Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo “Es un proceso de desarrollo de software basado en casos de uso, centrado en la arquitectura e iterativo e incremental que utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema de software”.⁶ Este proceso está basado en casos de uso debido a que los casos de uso brindan una herramienta para especificar los requisitos que el sistema debe cumplir, son una guía para el diseño, implementación y prueba del software que se quiere lograr. Estos se desarrollan en paralelo con la arquitectura del sistema y la guían, aunque la arquitectura del sistema influye en la selección de los casos de uso.

Está centrado en la arquitectura pues en ella se definen los aspectos estáticos y dinámicos más significativos del sistema, surge de las necesidades de las empresas, de como la perciben los usuarios

⁶ Jacobson I, Booch G. y Rumbaugh J. *El Proceso Unificado de Desarrollo de Software*. España 2000.

y los inversores, y se refleja en los casos de uso. Con su relación con los casos de uso, la arquitectura permite el desarrollo de todos los casos de usos que se requieren en el sistema, en el momento de ser construido o en un futuro.

Es iterativo e incremental dado que los desarrolladores dividen los proyectos en pequeños fragmentos que se trabajaran en cada iteración, en cada una de ellas se comienza según hayan quedado los artefactos de desarrollo en la iteración anterior, incrementando el avance del proyecto en cada iteración, aunque este aumento en ocasiones puede no ser aditivo. “El Proceso Unificado se repite a lo largo de una serie de ciclos, que constituyen la vida de un sistema” ⁷ y estos a su vez constan de cuatro fases subdivididas en iteraciones. El resultado de cada ciclo de vida es una versión del producto para los clientes.

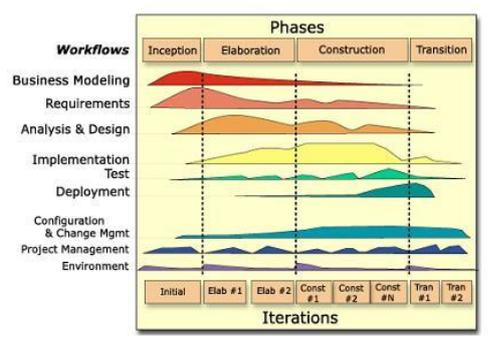


Ilustración 1 Ciclo de vida RUP

1.4.2 Programación Extrema (XP)

Otra de las metodologías de desarrollo es la Programación Extrema (XP), esta se basa principalmente en el trabajo en equipo y en la creación de una ambiente de trabajo propicio para el aprendizaje y las relaciones interpersonales de los desarrolladores como clave para el éxito del desarrollo del software. “La programación extrema se basa en una serie de reglas y principios que se han ido gestando a lo largo de toda la historia de la ingeniería del software”⁸.

⁷ Jacobson I, Booch G. y Rumbaugh J. *El Proceso Unificado de Desarrollo de Software*. España 2000

⁸ Beck K. *Extreme Programming Explained. Embrace Change*. 1999.

Promueve cuatro valores fundamentales para el desarrollo del software: la comunicación, evidenciándose en las relaciones entre clientes y desarrolladores, el coraje para expresar dudas, miedos o experiencias que puedan ayudar en la construcción del software, la simplicidad del software y retroalimentación que se toma del cliente y de los miembros del equipo, considerando que mientras más sencillo sea el software más fácil sería de probar.

Tiene cuatro principios que se apoyan en los valores y son la Retroalimentación Veloz, las Modificaciones Incrementales, el Trabajo de Calidad y la Asunción de Simplicidad.”Los principios suponen un puente entre los valores (algo intrínseco al equipo de desarrollo) y las practicas, que están más ligadas a las técnicas que hay que seguir.”⁹

Esta metodología se desarrolla en cuatro fases: la planificación, el diseño, desarrollo y pruebas.

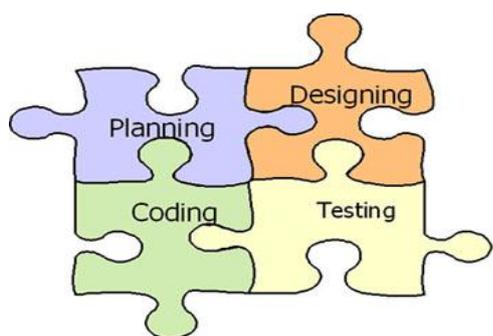


Ilustración 2 Ciclo de vida XP

1.4.3 Scrum

Scrum es una metodología ágil que se basa en el Patrón Scrum para el desarrollo de proyectos, de manera que toma los beneficios de la mejora continua y la institucionalización de los procedimientos como una estrategia de gestión que trabaja con la síntesis del conocimiento desarrollado por las teorías de procesos y las de agilidad. ”Una Scrum Management no se queda en el “cómo” de las prácticas, sino que trabaja desde el “porqué” para descartar, modificar o incorporar según las características de la empresa y los proyectos. “¹⁰

⁹ Escribano.G. F. eXtreme Programming / Programación Extrema. 2002.

¹⁰ Palacio J. *Flexibilidad con Scrum*. Noviembre 2007.

Esta metodología tiene en cuenta el beneficio para el trabajo en equipos, haciendo énfasis en la productividad y la comunicación directa, el Desarrollo incremental e iterativo con la producción frecuente de partes del producto que puede evaluar el cliente, integración y pruebas tempranas, el Diseño de procesos o rutinas en función de la principal necesidad del proyecto: previsibilidad o creatividad e innovación, el Grado de institucionalización de los procedimientos (procesos o rutinas) adecuado al tamaño y previsión de crecimiento de la organización y la Gestión sistémica de la organización.

“Scrum Management es un modelo de gestión de “conocimiento tácito”. El gestor no aporta “ejecución” para implantar un modelo, sino “talento” para desarrollar el propio. No consiste por tanto en aplicar un modelo, a modo de proceso, sino que son los gestores los que lo construyen con su conocimiento, sobre estas bases.”¹⁰

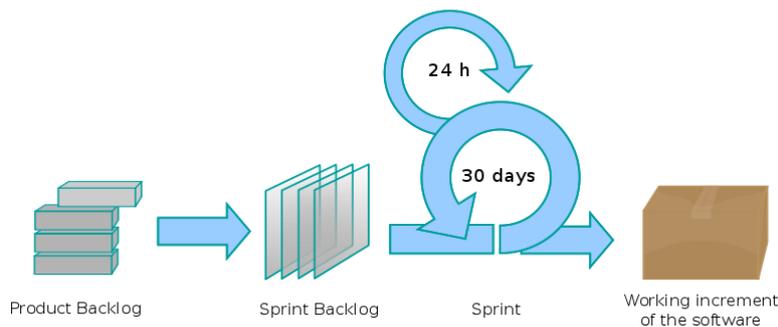


Ilustración 3 Ciclo de vida Scrum

1.4.4 Selección de la Metodología de Desarrollo a utilizar.

Realizado el análisis de las diferentes metodologías se llega a la conclusión que dado el tiempo de realización del proyecto y la cantidad de miembros del equipo de desarrollo es mejor usar XP, pues RUP es una metodología enfocada al desarrollo de proyectos grandes mientras que XP se enfoca más en proyectos pequeños, en el caso de Scrum al ser otra de las metodologías ágiles, no se elige para lograr organizar el desarrollo del proyecto con una sola metodología. En el caso de XP se destaca su facilidad para desarrollar el sistema y la constante comunicación y retroalimentación que tienen clientes y desarrolladores.

1.5 Lenguajes de Programación

1.5.1 Java

El lenguaje de programación Java fue desarrollado por Sun Microsystems a principios de los 90's con la idea original de usarlo para la creación de páginas WEB. Entre noviembre de 2006 y mayo de 2007, Sun Microsystems liberó la mayor parte de sus tecnologías Java bajo la licencia GNU GPL, de tal forma que prácticamente todo el Java de Sun es software libre. Es orientado a objetos, de una plataforma independiente y en sí mismo toma mucha de su sintaxis de C y C++ pero tiene un modelo de objetos más simple eliminando herramientas como la manipulación directa de punteros o memoria.

Entre sus características más importantes se destaca su robustez, por la forma en que está diseñado que no permite el manejo directo del hardware, ni de la memoria de modo que se puede decir que es virtualmente imposible colgar un programa Java. El intérprete siempre tiene el control. Además de que gestiona la memoria automáticamente, no permite el uso de técnicas de programación inadecuadas, posee mecanismos de seguridad incorporados y amplias herramientas de documentación que, con un poco de atención al poner los comentarios en las clases, se crea inclusive toda la documentación de las mismas en formato HTML.

1.5.2 C#

Es un lenguaje de programación orientado a objetos creado por Microsoft como parte de la iniciativa .NET Framework y escrito por Andres Hejlsberg en 1999. La última versión conocida es la 3.0 que salió a la luz el 17 de noviembre del 2007, propiedad de la compañía Microsoft. En su constitución, a diferencia de lenguajes como C / C++, que proporcionan un altísimo grado de control de los procesos permitiendo el uso de punteros y muchas otras funciones de bajo nivel, y otros incluidos dentro de este entorno .NET, como Microsoft Visual Basic, que posee un alto nivel y más facilidad a la hora de desarrollar una aplicación, C#, se encuentra en un término intermedio ya que ayuda a desarrollar aplicaciones rápidas pero que también permiten un gran control e integración con el desarrollo de aplicaciones Web, XML, y muchas otras de las tecnologías recientes.

Aunque C# forma parte de la plataforma .NET que es una interfaz de programación de Aplicaciones (API) este es un lenguaje independiente diseñado para generar programas sobre dicha plataforma. Como nacimiento reciente en la familia de lenguajes de programación C# tiene dos predecesores: C++

y Java dado que contiene muchas características de C++ y además añade la orientación a objetos de Java.

1.5.3 Perl

Es un lenguaje de programación diseñado por Larry Wall en 1987. La última versión conocida es la 5.10.1 que se publicó el 23 de agosto de 2009. Ha sido publicado bajo la licencia Artística y la licencia GNU/GPL. Estructuralmente, Perl está basado en un estilo de bloques, toma características de C y Shell, y fue ampliamente adoptado por su destreza en el procesado de texto y no tener ninguna de las limitaciones de los otros lenguajes script.

“Es un lenguaje de propósito general, originalmente desarrollado para la manipulación de texto y que ahora es utilizado para un amplio rango de tareas incluyendo administración de sistemas, desarrollo Web, programación en red, desarrollo de GUI y más. Se previó que fuera práctico (facilidad de uso, eficiente, completo) en lugar de hermoso (pequeño, elegante, mínimo).”¹¹

Sus principales características son:

- La estructura completa deriva ampliamente del lenguaje C.
- Soporta tanto la programación estructurada como la programación orientada a objetos y la programación funcional.
- Es un lenguaje imperativo, con variables, expresiones, asignaciones, bloques de código delimitados por llaves, estructuras de control y subrutinas.
- Todas las versiones de Perl hacen el tipificado automático de datos y la gestión de memoria.
- Las conversiones legales de tipo se hacen de forma automática en tiempo de ejecución; las conversiones ilegales son consideradas errores fatales.
- El intérprete conoce el tipo y requerimientos de almacenamiento de cada objeto en el programa reservando y liberando espacio para ellos según sea necesario.

Una característica principal introducida en Perl 5 la última versión alfa, fue la habilidad de empaquetar código reutilizable como módulos. Larry Wall, el creador, indicó más adelante que "la

¹¹ Jonathan Noel Tombs J.N. y Chávez Orzáez O. *Manual de Unix*, Revisión 2.4, 1995

intención del sistema de módulos de Perl 5 era apoyar el crecimiento de la cultura Perl en vez del núcleo de Perl".¹²

1.5.4 C ++

Es un lenguaje de programación creado por Bjarne Stroustrup en los años 80. Es un lenguaje imperativo orientado a objetos muy ligado al hardware subyacente, manteniendo una considerable potencia para programación a bajo nivel, pero con elementos que le permiten también un estilo de programación con alto nivel de abstracción. "No es un lenguaje orientado a objetos puro (en el sentido en que puede serlo Java por ejemplo), además no nació como un ejercicio académico de diseño. Se trata simplemente del sucesor de un lenguaje de programación hecho por programadores (de alto nivel) para programadores, lo que se traduce en un diseño pragmático al que se le han ido añadiendo todos los elementos que la práctica aconsejaba como necesarios, con independencia de su belleza o purismo conceptual ha experimentado un extraordinario éxito desde su creación. De hecho, muchos sistemas operativos, compiladores e intérpretes han sido escritos en C++"¹³

1.5.5 Python

Lenguaje de programación interpretado creado por Guido Van Rossum en 1991. La última versión estable del lenguaje es la 3.1.1. Entre sus principales características se encuentra que es de propósito general, se puede utilizar para realizar diferentes tipos de aplicaciones, es multiplataforma, puede correr sobre cualquier sistema operativo, siempre y cuando se tenga su intérprete.

Permite dividir el programa en módulos reutilizables desde otros programas. Viene con una gran colección de módulos estándar que se pueden utilizar como base que proporcionan E/S de ficheros, llamadas al sistema, sockets y hasta interfaces a GUI (interfaz gráfica con el usuario) como Tk, Gtk, Qt, entre otros. Es interpretado lo que ahorra un tiempo considerable en el desarrollo del programa, pues no es necesario compilar ni enlazar. El intérprete se puede utilizar de modo interactivo, lo que facilita experimentar con características del lenguaje, escribir programas desechables o probar

¹² Robert Kirrily, skud@cpan.org, 2007 [Revisada Enero 2010]

¹³ Zator Systems 2008 http://www.zator.com/Cpp/E1_2_1.htm [Revisada Enero 2010]

funciones durante el desarrollo del programa. El principal objetivo que persigue este lenguaje es la facilidad, tanto de lectura, como de diseño.

Es un lenguaje de programación multiparadigma. Esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación estructurada y programación funcional. Otros paradigmas están soportados mediante el uso de extensiones.

1.5.6 Selección del Lenguaje de Programación a utilizar.

Se utilizará en la programación del sistema el lenguaje Python porque presenta ventajas con relación a los demás lenguajes, entre ellas se encuentra que permite la programación multiparadigma, tanto orientada a objetos, como funcional y estructurada, que es dinámicamente tipado y que soporta herencia múltiple. Los programas escritos en python son cortos lo que elimina la desventaja de que por ser interpretado puede resultar más lento que los compilados. Es importante destacar la gran cantidad de librerías que tiene que logran una programación más flexible y ligera.

1.6 Herramientas de Desarrollo

1.6.1 Umbrello

Es una herramienta de desarrollo desarrollada por Paul Hensgen que ayuda en el proceso de desarrollo de software usando el estándar 'Unified Modelling Language' (UML) que permite crear y editar diagramas para diseñar y documentar sistemas . Es una herramienta libre que maneja gran parte de los diagramas UML y está diseñado principalmente para KDE aunque funciona en otros entornos de escritorio, además de permitir la importación y exportación de los diagramas a partir de códigos en C++, Java, Pascal/Delphi, Python entre otros basados en el formato 'XML Metadata Interchange' (XMI). También permite la exportación de los modelos a DocBook y XHTML los cuales pueden ser publicados vía Web con la ventaja de que en los proyectos colaborativos se pueden distribuir sin necesidad de acceso directo al programa. La última versión estable de sistema es la 2.0.0 que salió el 12 de enero de 2007 bajo licencia GPL.

1.6.2 Visual Paradigm

Visual Paradigm es una herramienta que utiliza UML para realizar la modelación y construcción de aplicaciones. Apoya el ciclo de vida completo del desarrollo de software a través de todas sus fases haciendo énfasis en la calidad del mismo. Brinda un soporte integral para la realización de diagramas a través de tutoriales de UML, proyectos y demostraciones interactivas de UML.

Visual Paradigm tiene como características principales:

- Modelado Colaborativo con CVS y Subversión e interoperabilidad con modelos UML2 a través de XML, que no son más que metamodelos UML 2.x para plataforma Eclipse.
- Posibilidad de realizar ingeniería e ingeniería inversa ya sea de código a modelo, como de código a diagrama, en Java, C++, Esquemas XML, XML,.NET exe/dll, CORBA IDL, e ingeniería inversa a base de datos desde gestores a diagramas Entidad-Relación.
- Puede generar código objeto Java desde la base de datos por su soporte ORM, genera además código ya sea de Modelo a código como de diagrama a código, genera código y despliegue de EJBs (Beans) para el desarrollo y despliegue de aplicaciones, con visualización de sistemas EJB.
- Permite la transformación de diagramas de Entidad-Relación en tablas generando bases de datos.
- Puede generar documentación a partir de la realización de informes y contiene un editor de Detalles de casos de Uso con un entorno integral para la especificación de los detalles y las descripciones de los casos de uso y del modelo general.
- En su ambiente de trabajo se pueden distribuir automáticamente los diagramas, importar y exportar ficheros XML e imágenes, contar con un editor de figuras además de contener otras herramientas y plugging para Javaclipse, Netbeans, SunOne, Oracle JDeveloper, Jbuilder, IntelliJ Idea, WebLogic Workshop, Visual Estudio entre otras.

1.6.3 Rational Rose

IBM Rational Rose es un software que permite el desarrollo de aplicaciones usando UML, tiene entre sus principales características ayudar a los equipos a crear aplicaciones basadas en arquitecturas robustas, permitir a los usuarios crear arquitecturas independientes de las plataformas con soporte completo para UML 1.x, mantener una herramienta del modelado e idioma común para el

equipo de desarrollo, soportar el desarrollo de sistemas incluidos y en tiempo real y el modelado WEB, habilitar publicaciones y reportes web, integrarse con las herramientas de desarrollo de ciclos de vida de IBM Rational y permitir la realización de ingeniería ya sea para Ada, ANSI C++, el C, C++, CORBA, Java, Microsoft Visual Basic y los idiomas de C++ Visuales, así como diseño de base de datos.

“Arquitectos, analistas, diseñadores de software y base de datos, y diseñadores de sistemas incluidos y de tiempo real pueden usar la familia de productos de Rational Rose para producir para producir modelos visuales de arquitecturas de software, bases de datos, requisitos de la aplicación y recursos reusables, así como para formular comunicaciones por niveles de dirección. Usando el UML estándar, los usuarios pueden crear arquitecturas semánticamente ricas de aplicaciones que ayudan al equipo de desarrollo incluso stakeholders, a entender claramente el proyecto. Una herramienta e idioma común para modelar ayuda a simplificar el ambiente de trabajo y acelerar el desarrollo de aplicaciones de software de calidad.”¹⁴

Algunas de sus funciones son:

- Permitir Especificar, Analizar, Diseñar el sistema antes de Codificarlo.
- Mantener la consistencia de los modelos del sistema software.
- El chequeo de la sintaxis UML.
- La generación de la documentación automáticamente.
- La generación de código a partir de los Modelos.
- Y la realización de Ingeniería Inversa (crear modelo a partir código).¹⁵

1.6.4 Selección de las Herramientas de Desarrollo a utilizar

Analizadas las herramientas de desarrollo se elige a Visual Paradigm para el desarrollo de la aplicación por ser una herramienta libre que acompaña el desarrollo del software en todas las etapas del ciclo de vida del mismo, brindando potentes herramientas de modelado que permiten la integración con diversos sistemas y la generación de la documentación del proyecto, así como la base de datos y

¹⁴ IBM Corporation. *IBM Rational Rose. Rational Rose.pdf*.2006.

¹⁵ Gonzalez Blanco R. y Perez Tobalina S. *LESE-2 Introducción a Rational Rose*. Facultad de Informática de Barcelona.2002.

la realización de ingeniería inversa ya sea de código a diagramas como de bases de datos a diagramas.

1.7 Sistemas Gestores de Bases de Datos

Es un tipo de software dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que utilizan. Su objetivo es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante para una organización. Una base de datos, en su definición más sencilla, es una colección de archivos relacionados. La mayor parte de las bases de datos actuales son de tipo relacional. Se denominan así porque utilizan tablas de datos relacionadas por un campo en común.

Entre los objetivos que tienen los SGBD están: la abstracción de la información en la cual ocultan a los usuarios detalles del almacenamiento físico de los datos, la independencia de los datos que plantea que las aplicaciones que interactúan con el gestor no sean afectadas ante cambios producidos en la base de datos, la consistencia de los datos que estos sean lo menos redundante posible y en caso de existir que se puedan actualizar los datos repetidos de forma simultánea y la seguridad de los datos, garantizando a partir de niveles o categorías de permisos que la información está segura, el manejo de transacciones y el tiempo de respuesta.

1.7.1 MySql

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario que implementa diversas Interfaces de Programación de Aplicaciones (APIs) que permite que aplicaciones puedan acceder a las bases de datos, independientemente del lenguaje en que estén implementadas. La última versión conocida es la 5.1.42 que salió el 31 de diciembre del 2009 bajo licencia GPL y licencia comercial. El gestor incluye soporte para conexiones seguras entre los clientes MySQL y el servidor, utilizando el protocolo Secure Sockets Layer (SSL).

Contiene un interfaz Open Data Base Connectivity (ODBC), llamado MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL. Es muy utilizado en aplicaciones web como MediaWiki o Drupal, en plataformas Linux/Windows-Apache-MySQL-PHP/Perl/Python (WAMP), y por herramientas de seguimiento de errores como Bugzilla.

Entre sus ventajas se incluyen:

- “Coste: El coste de MySQL es gratuito para la mayor parte de sus usos y su servicio de asistencia resulta económico.
- Asistencia: MySQL AB ofrece contratos de asistencia a precios razonables y existe una nutrida y activa comunidad MySQL.
- Velocidad: MySQL es mucho más rápido que la mayor parte de sus rivales.
- Funcionalidad: MySQL dispone de muchas de las funciones que exigen sus desarrolladores profesionales, como compatibilidad completa con ACID,
- Compatibilidad para la mayor parte de SQL ANSI, volcados online, duplicación, funciones SSL e integración con la mayor parte de sus entornos de programación. Así mismo, se desarrolla y actualiza de forma mucho más rápida que muchos de sus rivales.
- Portabilidad: MySQL se ejecuta en la inmensa mayoría de sistemas operativos y, la mayor parte de sus casos, sus datos se pueden transferir de un sistema a otro sin dificultad.
- Facilidad de uso: MySQL resulta fácil de utilizar y de administrar. Gran parte de las viejas bases de datos presentan problemas por utilizar sistemas obsoletos, lo que complica innecesariamente las tareas de administración. Las herramientas de MySQL son potentes y flexibles, sin sacrificar su capacidad de uso.”¹⁶

1.7.2 Oracle

Es un sistema de gestión de base de datos que surge a finales de los 70 con el nombre de Relational Software propiedad de la compañía Oracle Corporation a partir de un estudio de George Koch. La última versión conocida es la 11.0 que salió en el 2007 y utiliza una licencia privativa. Se caracteriza por tener soporte de transacciones y contener registros de reconstrucción en línea los cuales están compuestos por todas las transacciones confirmadas o anilladas., su estabilidad, por ser multiplataforma y escalable.

¹⁶ Gilfillan I. *La Biblia de Mysql*.

“Toda la información que una base de datos Oracle necesita para gestionarse a si misma se almacena en una serie de tablas que se conocen comúnmente como el diccionario de datos. Imagine este diccionario de datos como una tabla de tablas.”¹⁷

“El método estándar utilizado para pasar solicitudes a la base de datos Oracle y obtener la información utiliza Oracle Net y TCP/IP. A Oracle Net se le conoce como conexión nativa, dado que es la conexión específica de la base de datos Oracle.”¹⁷

1.7.3 PostgreSQL.

Es un sistema de gestión base de datos relacional orientada a objetos, liberado bajo la licencia BSD y desarrollado por Michael Stonebraker que en 1989 publica la versión 1.0. La última versión estable es la 8.4 que salió el 1 de julio del 2009.

“Presenta una alta concurrencia ya que mediante un sistema denominado MVCC (Acceso concurrente multiversión) permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.”¹⁸

Algunas de sus principales características son:

- “Altamente Extensible: PostgreSQL soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario.
- Soporte SQL Comprensivo: PostgreSQL soporta la especificación SQL99 e incluye características avanzadas tales como las uniones SQL92.
- Integridad Referencial: PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.

¹⁷ Abbey M, Corey M. y Abramson I. *Oracle 9i Guía de aprendizaje.pdf*. Edición Oficial.

¹⁸ PostgreSQL 2007, PostgreSQL. Tutorial de PostgreSQL. *Tutorial de PostgreSQL* September 2007.

➤ API Flexible: La flexibilidad del API de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para el RDBMS PostgreSQL. Estas interfaces incluyen Object Pascal, Python, Perl, PHP, ODBC, Java/JDBC, Ruby, TCL, C/C++, y Pike.”¹⁹

1.7.4 Selección del Sistema Gestor de Base de Datos a utilizar

Analizados los SGBD se decide utilizar MySQL debido a que tiene una licencia libre y es un potente sistema que permite el acceso simultáneo de usuarios o aplicaciones a las bases de datos, usa un motor SQL permitiendo el ahorro de tiempo, aprovechando la estandarización de SQL para facilitar la portabilidad con otros sistemas y plataformas, y además asegura la integridad de los datos a través de diversos niveles de permisos y privilegios a usuarios, posibilitando compartir y proteger datos.

1.8 Conclusiones.

En este capítulo se definieron conceptos relacionados con el fundamento teórico de la investigación, se realizó el estado del arte sobre los sistemas de auditoría llegando a la conclusión de que no existe ninguna herramienta libre que cumpla con las necesidades específicas de la Dirección de Supervisión y Control de la Universidad.

Se estudiaron diferentes lenguajes, herramientas, metodologías de desarrollo, sistemas gestores de bases de datos para definir cuáles se utilizarán en la construcción de la solución, eligiéndose como metodología de desarrollo a XP, como lenguaje de programación a Python utilizando la librería visual GTK y como sistema gestor de base de datos a MySQL.

¹⁹ eAPRENDE.com. 2008. <http://www.eaprende.com/gestor-de-basededatos-mysql-postgresql-sqlite.html>.

CAPÍTULO 2 DESCRIPCIÓN DE LA SOLUCIÓN

2.1 Introducción

En este capítulo se describe el flujo del proceso de auditoría, definiendo la estructura organizativa de la oficina de Supervisión y Control conjuntamente con los diferentes niveles de seguridad, las fases que se divide una auditoría y se propone la arquitectura del sistema. Se definen requisitos no funcionales del sistema y se detallan las historias de usuarios, que permitirán planificar las iteraciones en que se desarrollará el sistema, además del plan de entrega de versiones, especificándose en qué fecha se entregan las versiones del sistema.

En esta etapa el equipo de desarrollo realiza un trabajo conjunto con el cliente en reuniones donde analizan aspectos relacionados con cada etapa, el equipo se adentra en la metodología y en las herramientas y lenguajes con que se trabajarán, y se definen los requisitos y reglas que compondrán el sistema.

2.2 Descripción del flujo actual del Proceso de Auditoría en la Dirección de Supervisión y Control de la UCI

La Dirección de Supervisión y Control de la UCI es la entidad encargada de la realización y control de la Auditoría Interna. En esta el director es el que emite la orden de trabajo, la cual luego de ser firmada por el rector de la universidad da inicio a una auditoría en un área determinada y designa el grupo de auditores encargados de su realización además de nombrar a un jefe de grupo que en lo adelante será el responsable de las informaciones que contendrá la orden de trabajo, y la conformación del expediente que contiene los informes que se obtienen por parte de los auditores. Una auditoría se considera concluida cuando se aprueba por parte del director y de los directivos de la entidad auditada el Informe Final, documento que se conforma de resúmenes de las principales cuestiones vistas en la auditoría y pasa a ser de carácter oficial.

La estructura organizativa de la Oficina está compuesta por el director y los auditores. Un auditor en una auditoría determinada puede ser Jefe de Grupo, por lo que un mismo trabajador se puede encontrar jugando roles diferentes en las auditorías. A continuación definimos responsabilidades de cada uno de los trabajadores.

TRABAJADOR	RESPONSABILIDAD
Director	Dirige todo el equipo de trabajo de la Oficina, planifica e indica la realización de auditorías, da por finalizada una auditoría con la aprobación del Informe Final y es el único autorizado a la eliminación de expedientes que cumplan con la una antigüedad de 5 años o más.
Jefe de Grupo	Rol temporal que juegan los auditores en una auditoría determinada. Dirige la realización de la auditoría, controla y planifica el trabajo y el cronograma de cumplimiento de la auditoría, es el único autorizado a declarar un documento como final, luego de ser revisado, redacta y presenta al Director el Informe Final.
Auditor	Trabajador de la oficina que se encarga de llevar a cabo procedimientos de control dentro de las auditorías, participa en las auditorías conformando documentos que registran toda la actividad de revisión que realiza, en correspondencias con leyes, procedimientos y proformas establecidas dentro de la Auditoría Interna.

Tabla 1 Trabajadores y responsabilidades

Se identificaron cuatro fases de acuerdo a la realización de una auditoría las cuales son: la Exploración, durante la cual se realiza una visita a la entidad a auditar, donde se observan los flujos de trabajo presentes en la misma, se definen los aspectos que requieren una verificación y se familiariza con el ambiente de trabajo de la entidad, la Planificación, en esta fase la auditoría se indica y aprueba según cronograma de realización, se le indica una fecha de inicio y fin, y se define e informa la entidad

o área que será auditada, el equipo de trabajo que llevará a cabo la auditoría con el Jefe de Grupo que la dirigirá y los objetivos de la auditoría.

La fase de Realización es en la cual se llevan a cabo procedimientos y controles que verifican las actividades de la entidad y controlan los parámetros definidos en los objetivos de la auditoría. Todas las actividades realizadas tienen un documento escrito que las ampara. A medida que avance la auditoría los documentos generados pueden ser declarados finales. La última fase es la Culminación y Presentación de Resultados, en esta fase se escribe y presenta el Informe Final al Director, se determina la fecha de fin real y se realiza una reunión de conclusiones con los directivos de la entidad donde se exponen las principales cuestiones encontradas durante la realización de la auditoría.

2.3 Descripción de la solución propuesta

Como propuesta de solución se encuentra el desarrollo de un software con entorno de trabajo gráfico que permita la modelación de las fases descritas. Se trata de una aplicación de escritorio con arquitectura cliente servidor, que cuenta con interfaces de trabajo de los diferentes datos que se van a manejar dentro del proceso tales como auditores que constituyen los usuarios de la aplicación, así como las áreas o empresas que sean necesarias.

Se incluirá tres niveles de seguridad de acuerdo al nivel de acceso y permisos definidos en el proceso de auditoría, evidenciados en la siguiente tabla.

ROL	NIVEL DE ACCESO
Director	<ul style="list-style-type: none"> ➤ Acceso a todos los datos y opciones del sistema. ➤ Permisos para finalizar una auditoría. ➤ Permisos para eliminar una auditoría. ➤ Permisos para administrar en el sistema usuarios y roles con sus niveles de acceso. ➤ Permisos para administrar en el sistema empresas y áreas.

<p>Jefe de Grupo</p>	<ul style="list-style-type: none"> ➤ Permisos para modificar un expediente de auditoría. ➤ Permisos para declarar un Documento Final. ➤ Permisos como Auditor.
<p>Auditor</p>	<ul style="list-style-type: none"> ➤ Permisos para ver auditorías en las que no esté trabajando. ➤ Permisos para eliminar un documento en una auditoría en que forme parte del equipo de trabajo. ➤ Permisos para personalizar su ambiente de trabajo cambiando las propiedades del sistema.

Tabla 2 Roles y niveles de acceso

2.4 Arquitectura del sistema propuesto

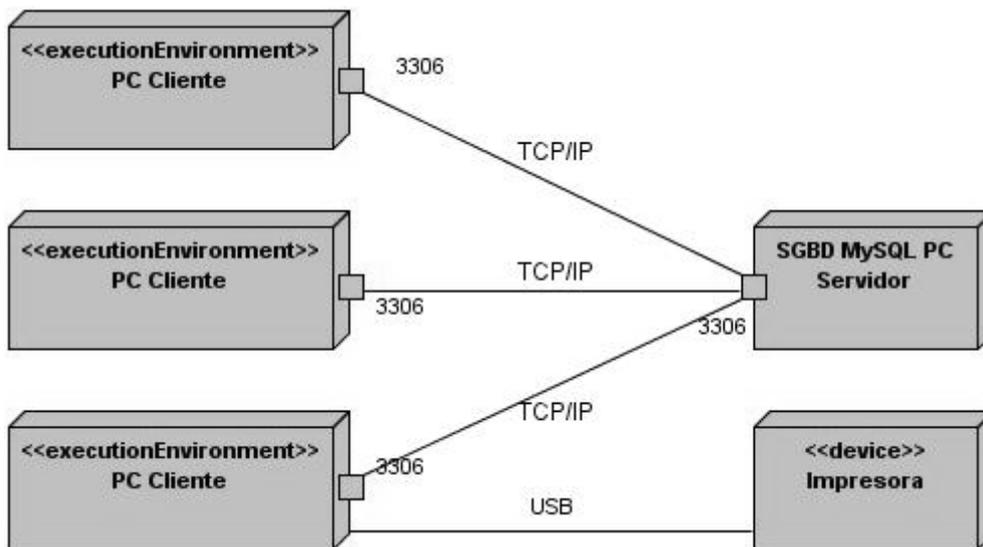


Ilustración 4 Arquitectura del sistema

2.5 Requisitos no funcionales del sistema

Se identificaron requisitos que se tendrán en cuenta en la realización del sistema, que permiten una mejor integración con otros sistemas, facilidad de uso y adaptación por parte del cliente y mejor rendimiento del sistema.

2.5.1 Requerimientos de apariencia o interfaz externa

La aplicación debe tener una interfaz amigable al usuario permitiéndole realizar operaciones de la forma más sencilla posible. Debe poseer un interfaz configurable que permita personalizar el entorno a gusto del cliente.

2.5.2 Requerimientos de usabilidad

El sistema debe contar con un manual de usuario que contenga informaciones acerca de las funcionalidades ofrecida además de los principales cortes de camino o atajos de teclado ofrecidos. Debe contar con la opción de ayuda y la oferta de sugerencias en las diferentes vistas en la cuales se encuentre el cliente.

2.5.4 Requerimientos de rendimiento

El sistema debe administrar eficiente y eficazmente los recursos dado que debe poder ejecutarse bajo diversas plataformas o sistemas operativos que administran de forma diferente los recursos además de poseer diferentes arquitecturas de máquina con distintas cantidades de recursos.

2.5.5 Requerimientos de portabilidad

El sistema debe poder instalarse y luego ejecutarse en las plataformas más conocidas y usadas adaptando tanto el motor visual como el sistema de gestión de datos a la arquitectura en la cual se encuentre funcionando.

2.5.6 Requerimientos de seguridad

La información que maneja el sistema debe poder ser accedida solo por personas autorizadas previamente identificada, del modo autorizado de acuerdo a su rol dentro de la aplicación, y en el momento que lo soliciten. Las operaciones que se realicen sobre los documentos deben estar registradas en el momento que sean realizadas, el auditor que la realizó y la ubicación desde la cual

fue realizada. Los datos sensibles dentro de la aplicación como contraseñas deben ser tratadas mediante procedimientos de encriptado.

2.5.7 Requerimientos de software

El sistema debe ser capaz de funcionar sobre Windows XP o superior y Linux en sus versiones más conocidas entornos Gnome y KDE. Para la gestión de datos se requiere MySQL 4.0 o superior. Para la lógica del negocio se requiere Python 2.6 o superior además de la biblioteca visual Gtk+ 2.10.

2.6 Planificación

En esta etapa se describen las historias de usuario, que son las funcionalidades que se compondrá el sistema, son escritas por el cliente el cual es parte indispensable del equipo de trabajo, y se usan para estimar el tiempo de desarrollo de dicha funcionalidad, es decir, cuánto se demorará en implementarse.

Se crea además un plan de iteraciones que se compone de todas las versiones con fecha de entrega al cliente, en cada una de las versiones se definen las historias de usuarios que se realizarán, se establecen los tiempos de implementación ideales de las historias de usuario y la prioridad con la que serán implementadas.

Al finalizar esta etapa la colaboración entre cliente y desarrolladores dará como resultados fundamentales la agrupación de historias de usuarios por versiones estableciéndose objetivos por etapas, el tiempo que se demorará en implementarse cada versión y el número de personas que trabajaran en el desarrollo del sistema.

2.6.1 Historias de usuarios

“Las historias de usuario tienen el mismo propósito que los casos de uso, pero no son lo mismo. Las escriben los propios clientes, tal y como ven ellos las necesidades del sistema. Por tanto, serán descripciones cortas y escritas en el lenguaje del usuario, sin terminología técnica.”²⁰ Su principal

²⁰ Rodríguez M. y Ordoñez M. *La metodología XP aplicable al desarrollo del software educativo en Cuba*. UCI. Julio 2007.

diferencia está en el nivel de detalle pues las historias de usuarios solo brindan la información necesaria para que los programadores estimen un tiempo ideal de desarrollo y a la hora de implementar la historia de usuario, deberán reunirse con el cliente y ampliar los detalles de la misma.

En la realización del sistema propuesto se identificaron 8 historias de usuarios las cuales se detallan a continuación:

Historia de usuario 1 Autenticar usuario

HISTORIA DE USUARIO	
NÚMERO: 1	USUARIO: Auditor, Director
NOMBRE HISTORIA: Autenticar usuario	
PRIORIDAD EN NEGOCIO: Media	RIESGO EN DESARROLLO: Medio
PUNTOS ESTIMADOS: 1	ITERACIÓN ASIGNADA: 1
PROGRAMADOR RESPONSABLE: Arlee E. Paneque y Noel González	
DESCRIPCIÓN: El usuario accede al sistema escribiendo su usuario y contraseña válidos.	
OBSERVACIONES:	

Tabla 3 Representación de HU 1

Historia de usuario 2 Administrar auditor

HISTORIA DE USUARIO	
NÚMERO: 2	USUARIO: Director
NOMBRE HISTORIA: Administrar auditor	
PRIORIDAD EN NEGOCIO: Alta	RIESGO EN DESARROLLO: Medio
PUNTOS ESTIMADOS: 1	ITERACIÓN ASIGNADA: 1
PROGRAMADOR RESPONSABLE: Arlee E. Paneque y Noel González	
DESCRIPCIÓN: El usuario podrá realizar operaciones sobre usuarios como son insertar un nuevo usuario, modificar datos y eliminar usuarios del listado. Debe poder ver en un listado con todos los usuarios y cuando lo requiera poder ver los datos de los mismos.	
OBSERVACIONES:	

Tabla 4 Representación de HU 2

Historia de usuario 3 Administrar empresa

HISTORIA DE USUARIO	
NÚMERO: 3	USUARIO: Director
NOMBRE HISTORIA: Administrar empresa	
PRIORIDAD EN NEGOCIO: Media	RIESGO EN DESARROLLO: Medio
PUNTOS ESTIMADOS: 1	ITERACIÓN ASIGNADA: 1
PROGRAMADOR RESPONSABLE: Arlee E. Paneque y Noel González	
DESCRIPCIÓN: El usuario podrá realizar operaciones sobre empresas como son insertar una nueva empresa, modificar datos y eliminar empresas del listado. Debe poder ver en un listado con todas las empresas y cuando lo requiera poder ver los datos de las mismas.	
OBSERVACIONES:	

Tabla 5 Representación de HU 3

Historia de usuario 4 Administrar documento

HISTORIA DE USUARIO	
NÚMERO: 4	USUARIO: Director, Jefe de Grupo, Auditor
NOMBRE HISTORIA: Administrar documentos	
PRIORIDAD EN NEGOCIO: Alta	RIESGO EN DESARROLLO: Medio
PUNTOS ESTIMADOS: 1.2	ITERACIÓN ASIGNADA: 1
PROGRAMADOR RESPONSABLE: Arlee E. Paneque y Noel González	
DESCRIPCIÓN: El usuario podrá realizar operaciones sobre documentos como son insertar un nuevo documento, modificar datos y eliminar documentos del listado. Debe poder ver en un listado con todos los documentos organizados en expedientes y cuando lo requiera poder ver los datos de los mismos.	
OBSERVACIONES:	

Tabla 6 Representación de HU 4

Historia de usuario 5 Administrar Expedientes

HISTORIA DE USUARIO	
NÚMERO: 5	USUARIO: Jefe de Oficina, Auditor, Jefe de Grupo
NOMBRE HISTORIA: Administrar expedientes	
PRIORIDAD EN NEGOCIO: Alta	RIESGO EN DESARROLLO: Medio
PUNTOS ESTIMADOS: 1	ITERACIÓN ASIGNADA: 2
PROGRAMADOR RESPONSABLE: Arlee E. Paneque y Noel González	
DESCRIPCIÓN: El usuario podrá realizar operaciones sobre expedientes como son insertar un nuevo expediente, modificar datos y eliminar expedientes del listado. Debe poder ver en un listado con todos los expedientes y cuando lo requiera poder ver los datos de los mismos.	
OBSERVACIONES:	

Tabla 7 Representación de HU 5

Historia de usuario 6 Declarar Documento Final

HISTORIA DE USUARIO	
NÚMERO: 6	USUARIO: Jefe de Oficina, Jefe de Grupo
NOMBRE HISTORIA: Declarar Documento Final	
PRIORIDAD EN NEGOCIO: Media	RIESGO EN DESARROLLO: Medio
PUNTOS ESTIMADOS: 0.6	ITERACIÓN ASIGNADA: 2
PROGRAMADOR RESPONSABLE: Arlee E. Paneque y Noel González	
DESCRIPCIÓN: El usuario podrá declarar un documento como final, es decir que luego de esto no podrá ser modificado este documento.	
OBSERVACIONES:	

Tabla 8 Representación de HU 6

Historia de usuario 7 Realizar estadísticas

HISTORIA DE USUARIO	
NÚMERO: 7	USUARIO: Jefe de Oficina, Jefe de Grupo, Auditor

NOMBRE HISTORIA: Realizar estadísticas	
PRIORIDAD EN NEGOCIO: Media	RIESGO EN DESARROLLO: Medio
PUNTOS ESTIMADOS: 2	ITERACIÓN ASIGNADA: 2
PROGRAMADOR RESPONSABLE: Arlee E. Paneque y Noel González	
DESCRIPCIÓN: El usuario podrá realizar consultar estadísticas sobre cumplimiento del planes, evaluaciones y acceder reportes sobre estas estadísticas. Además de realizar búsquedas de documentos, personas o empresas y ver gráficas de las estadísticas para un mejor entendimiento de los resultados que se muestran.	
OBSERVACIONES:	

Tabla 9 Representación de HU 7

Historia de usuario 8 Editar propiedades

HISTORIA DE USUARIO	
NÚMERO: 8	USUARIO: Jefe de Oficina, Jefe de Grupo, Auditor
NOMBRE HISTORIA: Editar propiedades	
PRIORIDAD EN NEGOCIO: Media	RIESGO EN DESARROLLO: Medio
PUNTOS ESTIMADOS: 2	ITERACIÓN ASIGNADA: 3
PROGRAMADOR RESPONSABLE: Arlee E. Paneque y Noel González	
DESCRIPCIÓN: El usuario podrá cambiar aspectos del software, así como establecer configuraciones que personalicen el sistema.	
OBSERVACIONES:	

Tabla 10 Representación de HU 8

2.6.2 Plan de Iteraciones

HISTORIAS DE USUARIOS	DURACIÓN ESTIMADA	ITERACIÓN	DURACIÓN ITERACIÓN
Autenticar usuario	1	1	3 Semanas
Administrar auditor	1		
Administrar empresa	1		
Administrar documento	1.2	2	3 semanas

Administrar Expediente	1.4		
Declarar Documento Final	0.6		
Realizar Estadísticas	2	3	4 semanas
Editar propiedades	2		
Duración Total Planificada			3 meses

Tabla 11 Plan de iteraciones

2.6.3 Plan de entrega de versiones

ITERACIÓN	Iteración 1	Iteración 2	Iteración 3
CANTIDAD DE HISTORIAS DE USUARIOS	3	3	2
FECHA DE ENTREGA	17 de marzo 2010	18 de mayo 2010	18 de junio 2010

Tabla 12 Plan de entrega de versiones

2.7 Conclusiones

En este capítulo se describió el flujo actual del Proceso de Auditoría en la Oficina de Supervisión y Control de la Universidad, donde se definieron trabajadores que actúan en el sistema, fases de desarrollo de una auditoría y permitió sentar las bases para el desarrollo del sistema propuesto. Se describió la solución, señalando requisitos no funcionales del sistema, los niveles de seguridad y la arquitectura propuesta.

Se comenzó con la etapa de Planificación detallando las historias de usuarios por parte del cliente, estimando el tiempo de desarrollo de las mismas, se plantea el Plan de iteraciones, que serán tres iteraciones que agrupan las 8 historias de usuarios identificadas y se crea el Plan de entrega de versiones, donde se publicarán versiones del software de acuerdo con la iteración que se esté realizando.

CAPÍTULO 3 DISEÑO E IMPLEMENTACIÓN

3.1 Introducción

En este capítulo se detallan el diseño e implementación del software a partir de la realización de las tareas de la ingeniería que se obtuvieron de las historias de usuarios, se definen relaciones y responsabilidades entre clases y se muestra el diseño correspondiente a la base de datos del sistema.

3.2 Tarjetas CRC “Clase, responsabilidad y colaboración”

Las tarjetas CRC significan las relaciones que existen entre Clases, Responsabilidades y Colaboraciones, permitiendo al equipo de desarrollo entender la posición de cada objeto dentro del espectro del software, con quien interactúan y a cuales clases afectan directa o indirectamente. Esto permite que exista un esclarecimiento continuo de la realización de cada clase a medida que esta se implementa ayudando al programador a centrarse en su trabajo.

“Las tarjetas CRC permiten desprenderse del método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos. Las tarjetas CRC permiten que el equipo completo contribuya en la tarea del diseño.”²¹

Tarjeta CRC 1 Auditor

CLASE: Auditor	
RESPONSABILIDADES	CLASES RELACIONADAS
Valida la entrada al sistema.	Expediente
Responde a diferentes roles dentro del sistema.	Documento
	Empresa
	Estadística

²¹ Rodríguez M. y Ordoñez M. *La metodología XP aplicable al desarrollo del software educativo en Cuba*. UCI. Julio 2007.

Tabla 13 Tarjeta CRC 1

Tarjeta CRC 2 Expediente

CLASE: Expediente	
RESPONSABILIDADES	CLASES RELACIONADAS
Contener listado de documentos.	Documento
Permitir la modificación de los datos de un expediente.	Empresa
Transitar por las fases de Planificación, Realización y Culminación del proceso de auditoría.	Estadística
	Auditor

Tabla 14 Tarjeta CRC 2

Tarjeta CRC 3 Documento

CLASE: Documento	
RESPONSABILIDADES	CLASES RELACIONADAS
Visualizar datos de documento.	Expediente
	Auditor

Tabla 15 Tarjeta CRC 3

Tarjeta CRC 4 Empresa

CLASE: Empresa	
RESPONSABILIDADES	CLASES RELACIONADAS
Visualizar datos de empresa.	Expediente
	Auditor

Tabla 16 Tarjeta CRC 4

Tarjeta CRC 5 Estadística

CLASE: Estadística	
RESPONSABILIDADES	CLASES RELACIONADAS
Visualizar las estadísticas.	Expediente
Realiza los reportes del sistema.	Auditor

Tabla 17 Tarjeta CRC 5

Tarjeta CRC 6 Propiedades

CLASE: Propiedades	
RESPONSABILIDADES	CLASES RELACIONADAS
Administra los privilegios de los auditores.	Expediente
Administrar la interfaz del sistema.	Auditor
	Empresas
	Estadística
	Documento

Tabla 18 Tarjeta CRC 6

3.3 Diseño de la Base de Datos.

A continuación se presenta el diseño de la base de datos.

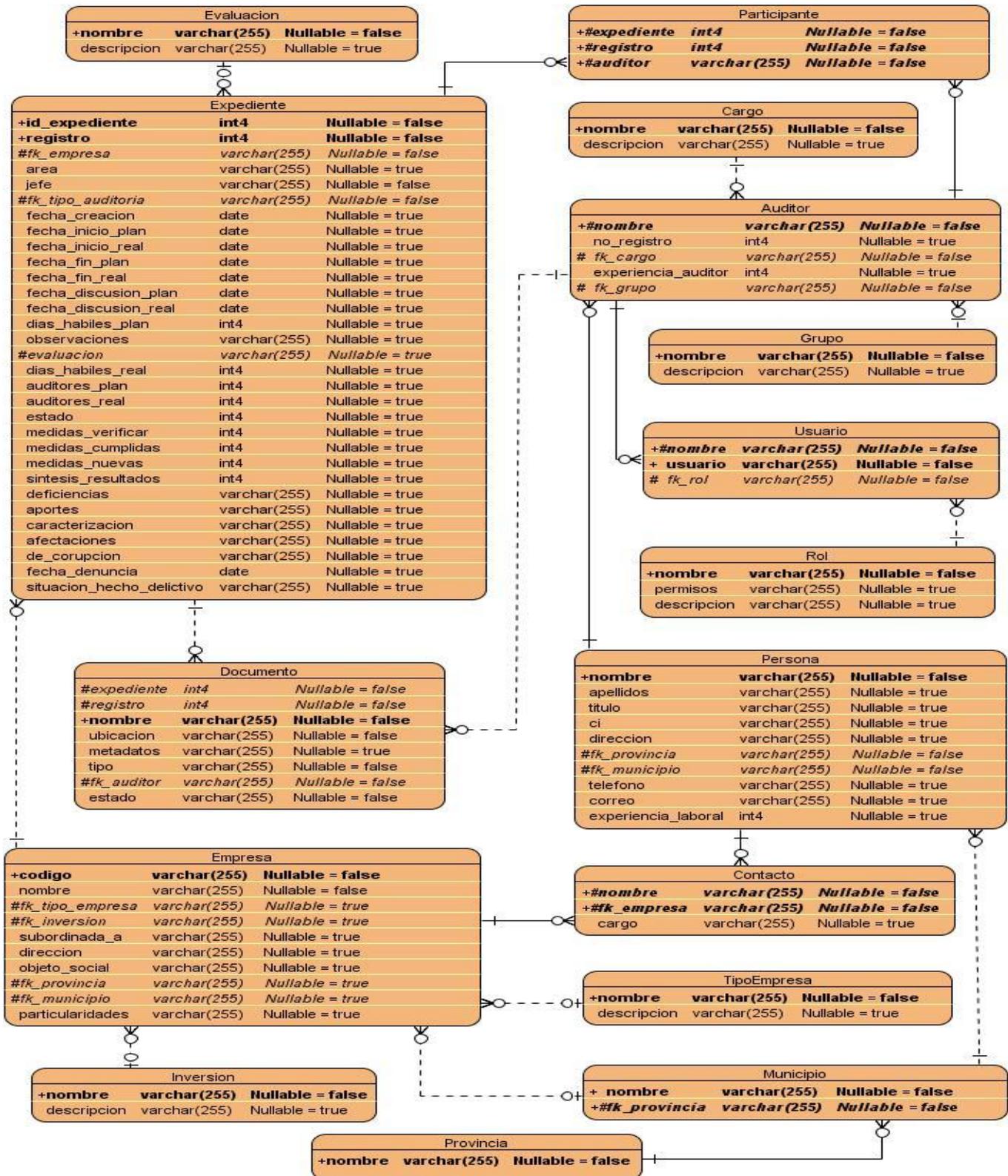


Ilustración 5 Modelo de datos del sistema

3.4 Tareas de la ingeniería.

En la implementación del software es de vital importancia la presencia del cliente, pues es él, el que divide el sistema en funcionalidades realizando las historias de usuarios. A partir de aquí, el cliente conjuntamente con el programador detalla que se va a hacer con cada una de las historias de usuarios, dividiéndolas en tareas de la ingeniería.

Estas no son más que las distintas funcionalidades operativas que conforman una historia de usuario, y que permiten testear si se está trabajando bien. Este trabajo conjunto constituye un paso decisivo para comenzar la implementación del software, pues permite organizar el trabajo en pasos lógicos, de acuerdo a la planificación correspondiente a esa historia de usuario.

En el sistema propuesto se identificaron 8 historias de usuarios, las cuales se dividieron en las siguientes Tareas de la Ingeniería.

No. HU	Nombre Historia de Usuario	No. TI	Tarea de la Ingeniería
1	Autenticar usuario	1	Autenticar usuario
2	Administrar auditor	1	Insertar usuario
		2	Modificar usuario
		3	Eliminar usuario
3	Administrar empresa	1	Insertar empresa
		2	Modificar empresa
		3	Eliminar empresa
4	Administrar documento	1	Insertar documento
		2	Abrir documento
		3	Modificar documento
		4	Eliminar documento
		5	Imprimir documento
		1	Insertar Orden de Trabajo
		2	Eliminar expediente

5	Administrar Expediente	3	Abrir expediente
		4	Modificar Expediente
6	Declarar documento final	1	Declarar documento final
7	Realizar Estadísticas	1	Cumplimiento de plan
		2	Evaluaciones
		3	Reportes totales
		4	Realizar búsqueda
		5	Graficar estadísticas
8	Editar propiedades	1	Editar propiedades

Tabla 19 Tareas de la Ingeniería

3.4.1 Desarrollo de las tareas de la ingeniería.

A continuación se detallan el desarrollo de las tareas de la ingeniería identificadas divididas en la iteración correspondiente a cada historia de usuario. Esta se realizó en conjunto con el cliente y el equipo de desarrollo.

3.4.2 Iteración 1

Tarea de la ingeniería 1 Autenticar usuario

TAREA DE LA INGENIERÍA	
Número Tarea: 1	Historia de usuario (No.1): Autenticar Usuario
Nombre Tarea: Autenticar usuario	
Tipo Tarea: Desarrollo	Puntos Estimados: 1
Fecha Inicio: 2 de febrero 2010	Fecha Fin: 10 de febrero 2010
Programador Responsable: Arlee E. Paneque y Noel González	
<p>Descripción: Durante el desarrollo de esta tarea se muestra una interfaz de acceso al sistema, donde el usuario debe introducir un usuario y una contraseña, el sistema verifica que los datos estén correctos y le da acceso al sistema según los permisos definidos para ese usuario.</p>	

Tabla 20 Descripción de la TI 1 Autenticar usuario

Tarea de la ingeniería 1 Insertar auditor

TAREA DE LA INGENIERÍA	
Número Tarea: 1	Historia de usuario (No.2): Administrar auditor
Nombre Tarea: Insertar usuario	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.4
Fecha Inicio: 11 de febrero 2010	Fecha Fin: 17 de febrero 2010
Programador Responsable: Arlee E. Paneque y Noel González	
<p>Descripción: Durante el desarrollo de esta tarea se crean las tablas Auditor, Persona, Usuario y Rol en la base de datos, además de la interfaz de inserción de auditor, donde se le brinda la posibilidad al usuario de introducir datos del nuevo auditor.</p>	

Tabla 21 Descripción de la TI 1 Administrar auditor

Tarea de la ingeniería 2 Modificar auditor

TAREA DE LA INGENIERÍA	
Número Tarea: 2	Historia de usuario (No.2): Administrar auditor
Nombre Tarea: Modificar usuario	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.4
Fecha Inicio: 18 de febrero 2010	Fecha Fin: 24 de febrero 2010
Programador Responsable: Arlee E. Paneque y Noel González	
<p>Descripción: Durante el desarrollo de esta tarea se muestra un listado con los auditores y se brinda la posibilidad de elegir que auditor modificar, se crea la interfaz de cambio de datos la cual verifica que los datos estén correctos y realiza los cambios, y por último se actualiza el listado con los cambios realizados.</p>	

Tabla 22 Descripción de la TI 2 Administrar auditor

Tarea de la ingeniería 3 Eliminar auditor

TAREA DE LA INGENIERÍA

Número Tarea: 3	Historia de usuario (No.2): Administrar auditor		
Nombre Tarea: Eliminar usuario			
Tipo Tarea: Desarrollo		Puntos Estimados: 0.2	
Fecha Inicio: 25 de febrero 2010		Fecha Fin: 2 de febrero	
Programador Responsable: Arlee E. Paneque y Noel González			
Descripción: Durante el desarrollo de esta tarea se muestra un listado con los auditores y se brinda la posibilidad de elegir que auditor eliminar, se elimina el auditor seleccionado luego de verificar que esta correcta esa eliminación, y por último se actualiza el listado con los cambios realizados.			

Tabla 23 Descripción de la TI 3 Administrar auditor

Tarea de la ingeniería 4 Insertar empresa

TAREA DE LA INGENIERÍA			
Número Tarea: 1	Historia de usuario (No.3): Administrar empresa		
Nombre Tarea: Insertar empresa			
Tipo Tarea: Desarrollo		Puntos Estimados: 0.4	
Fecha Inicio: 3 de marzo 2010		Fecha Fin: 8 de marzo 2010	
Programador Responsable: Arlee E. Paneque y Noel González			
Descripción: Durante el desarrollo de esta tarea se crean las tablas Empresa en la base de datos, además de la interfaz de inserción de empresa, donde se le brinda la posibilidad al usuario de introducir datos de la nueva empresa.			

Tabla 24 Descripción de la TI 1 Administrar empresa

Tarea de la ingeniería 5 Modificar empresa

TAREA DE LA INGENIERÍA	
Número Tarea: 2	Historia de usuario (No.3): Administrar empresa
Nombre Tarea: Modificar empresa	

Tipo Tarea: Desarrollo	Puntos Estimados: 0.4
Fecha Inicio: 9 de marzo 2010	Fecha Fin: 13 de marzo 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se muestra un listado con las empresas y se brinda la posibilidad de elegir qué empresa modificar, se crea la interfaz de cambio de datos la cual verifica que los datos estén correctos y realiza los cambios, y por último se actualiza el listado con los cambios realizados.	

Tabla 25 Descripción de la TI 2 Administrar empresa

Tarea de la ingeniería 6 Eliminar empresa

TAREA DE LA INGENIERÍA	
Número Tarea: 3	Historia de usuario (No.3): Administrar empresa
Nombre Tarea: Eliminar empresa	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.2
Fecha Inicio: 13 de marzo 2010	Fecha Fin: 17 de marzo 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se muestra un listado con las empresas y se brinda la posibilidad de elegir qué empresa eliminar, se elimina la empresa seleccionada luego de verificar que esté correcta esa eliminación, y por último se actualiza el listado con los cambios realizados.	

Tabla 26 Descripción de la TI 3 Administrar empresa

4.4.3 Iteración 2

Tarea de la ingeniería 7 Insertar documento

TAREA DE LA INGENIERÍA	
Número Tarea: 1	Historia de usuario (No.4): Administrar documento
Nombre Tarea: Insertar documento	

Tipo Tarea: Desarrollo	Puntos Estimados: 0.2
Fecha Inicio: 22 de marzo 2010	Fecha Fin: 25 de marzo 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se crea la tabla Documento en la base de datos, además de la interfaz de inserción de documento, donde se le brinda la posibilidad al usuario de introducir datos del nuevo documento.	

Tabla 27 Descripción de la TI 1 Administrar documento

Tarea de la ingeniería 2 Abrir documento

TAREA DE LA INGENIERÍA	
Número Tarea: 2	Historia de usuario (No.4): Administrar documento
Nombre Tarea: Abrir documento	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.2
Fecha Inicio: 26 de marzo 2010	Fecha Fin: 30 de marzo 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se muestra un listado de documentos y se le brinda la posibilidad al usuario de abrir el documento seleccionado.	

Tabla 28 Descripción de la TI 2 Administrar documento

Tarea de la ingeniería 3 Modificar Documento

TAREA DE LA INGENIERÍA	
Número Tarea: 3	Historia de usuario (No.4): Administrar documento
Nombre Tarea: Modificar empresa	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.4
Fecha Inicio: 31 de marzo 2010	Fecha Fin: 5 de abril 2010
Programador Responsable: Arlee E. Paneque y Noel González	

Descripción: Durante el desarrollo de esta tarea se muestra un listado con los documentos, se brinda la posibilidad de elegir qué documento modificar, se crea la interfaz de cambio de datos la cual verifica que los datos estén correctos, se realizan los cambios, y por último se actualiza el listado con los cambios realizados.

Tabla 29 Descripción de la TI 3 Administrar documento

Tarea de la ingeniería 4 Eliminar documento

TAREA DE LA INGENIERÍA	
Número Tarea: 4	Historia de usuario (No.4): Administrar documento
Nombre Tarea: Eliminar empresa	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.2
Fecha Inicio: 6 de abril 2010	Fecha Fin: 10 de abril 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se muestra un listado con los documentos, se brinda la posibilidad de elegir qué documento eliminar, se elimina el documento seleccionado luego de verificar que esta correcta esa eliminación, y por último se actualiza el listado con los cambios realizados.	

Tabla 30 Descripción de la TI 4 Administrar documento

Tarea de la ingeniería 5 Imprimir documento

TAREA DE LA INGENIERÍA	
Número Tarea: 5	Historia de usuario (No.4): Administrar documento
Nombre Tarea: Imprimir documento	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.2
Fecha Inicio: 12 de abril 2010	Fecha Fin: 15 de abril 2010
Programador Responsable: Arlee E. Paneque y Noel González	

Descripción: Durante el desarrollo de esta tarea se muestra un listado de documentos y se brinda la posibilidad al usuario de que seleccione un documento y éste se imprima.

Tabla 31 Descripción de la TI 5 Administrar documento

Tarea de la ingeniería 8 Insertar Orden de Trabajo

TAREA DE LA INGENIERÍA	
Número Tarea: 1	Historia de usuario (No.5): Administrar expediente
Nombre Tarea: Insertar Orden de Trabajo	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.6
Fecha Inicio: 16 de abril 2010	Fecha Fin: 22 de abril 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se crea la tabla Expediente en la base de datos, además de la interfaz de inserción de expediente, donde se le brinda la posibilidad al usuario de introducir datos del nuevo expediente.	

Tabla 32 Descripción de la TI 1 Administrar expediente

Tarea de la ingeniería 9 Abrir expediente

TAREA DE LA INGENIERÍA	
Número Tarea: 2	Historia de usuario (No.5): Administrar expediente
Nombre Tarea : Abrir expediente	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.2
Fecha Inicio: 23 de abril 2010	Fecha Fin: 28 de abril 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se muestra un listado de expediente y se le brinda la posibilidad al usuario de abrir el expediente seleccionado.	

Tabla 33 Descripción de la TI 2 Administrar expediente

Tarea de la ingeniería 10 Eliminar expediente

TAREA DE LA INGENIERÍA	
Número Tarea: 3	Historia de usuario (No.5): Administrar expediente
Nombre Tarea: Eliminar expediente	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.2
Fecha Inicio: 29 de abril 2010	Fecha Fin: 5 de mayo 2010
Programador Responsable: Arlee E. Paneque y Noel González	
<p>Descripción: Durante el desarrollo de esta tarea se muestra un listado con los expedientes y se brinda la posibilidad de elegir que expediente eliminar, se elimina el expediente seleccionado luego de verificar que esté correcta esa eliminación, y por último se actualiza el listado con los cambios realizados.</p>	

Tabla 34 Descripción de la TI 3 Administrar expediente

Tarea de la ingeniería 4 Modificar expediente

TAREA DE LA INGENIERÍA	
Número Tarea: 4	Historia de usuario (No.5): Modificar expediente
Nombre Tarea: Modificar expediente	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.4
Fecha Inicio: 6 de mayo 2010	Fecha Fin: 11 de mayo 2010
Programador Responsable: Arlee E. Paneque y Noel González	
<p>Descripción: Durante el desarrollo de esta tarea se muestra un listado con los expedientes y se brinda la posibilidad de elegir qué expediente modificar, se crea la interfaz de cambio de datos la cual verifica que los datos estén correctos y realiza los cambios, y por último se actualiza el listado con los cambios realizados.</p>	

Tabla 35 Descripción de la TI 4 Administrar expediente

Tarea de la ingeniería 1 Declarar Documento Final

TAREA DE LA INGENIERÍA

Número Tarea: 1	Historia de usuario (No.6): Declarar Documento Final		
Nombre Tarea: Declarar documento final			
Tipo Tarea: Desarrollo		Puntos Estimados: 0.6	
Fecha Inicio: 12 de mayo 2010		Fecha Fin: 18 de mayo 2010	
Programador Responsable: Arlee E. Paneque y Noel González			
Descripción: Durante el desarrollo de esta tarea se muestra un listado de documentos, y luego de seleccionar el documento que se quiere declarar como final, se le brinda la posibilidad al usuario de realizar esa operación.			

Tabla 36 Descripción de la TI 1 Declarar Documento Final

4.4.4 Iteración 3

Tarea de la ingeniería 1 Estadística Cumplimiento del Plan

TAREA DE LA INGENIERÍA			
Número Tarea: 1	Historia de usuario (No.7): Realizar estadísticas		
Nombre Tarea: Realizar estadística Cumplimiento del Plan			
Tipo Tarea: Desarrollo		Puntos Estimados: 0.2	
Fecha Inicio: 19 de mayo 2010		Fecha Fin: 22 de mayo 2010	
Programador Responsable: Arlee E. Paneque y Noel González			
Descripción: Durante el desarrollo de esta tarea se muestra una interfaz donde el usuario puede acceder a estadística de cumplimiento del plan realizada con los datos que estén en el sistema.			

Tabla 37 Descripción de la TI 1 Realizar estadísticas

Tarea de la ingeniería 2 Estadística Cumplimiento del Plan

TAREA DE LA INGENIERÍA	
Número Tarea: 2	Historia de usuario (No.7): Realizar estadísticas

Nombre Tarea: Realizar estadística Evaluaciones	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.2
Fecha Inicio: 24 de mayo 2010	Fecha Fin: 26 de mayo 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se muestra una interfaz donde el usuario puede acceder a estadísticas de evaluaciones realizadas con los datos que estén en el sistema.	

Tabla 38 Descripción de la TI 2 Realizar estadísticas

Tarea de la ingeniería 3 Reportes totales

TAREA DE LA INGENIERÍA	
Número Tarea: 3	Historia de usuario (No.7): Realizar estadísticas
Nombre Tarea: Realizar reportes totales	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.6
Fecha Inicio: 27 de mayo 2010	Fecha Fin: 31 de mayo 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se muestra una interfaz donde el usuario puede acceder a los resultados de resúmenes de datos que estén en el sistema y que se reúnen en reportes totales.	

Tabla 39 Descripción de la TI 3 Realizar estadísticas

Tarea de la ingeniería 4 Realizar búsqueda

TAREA DE LA INGENIERÍA	
Número Tarea: 4	Historia de usuario (No.7): Realizar estadísticas
Nombre Tarea: Realizar búsqueda	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.4
Fecha Inicio: 1 de junio 2010	Fecha Fin: 5 de junio 2010

Programador Responsable: Arlee E. Paneque y Noel González
Descripción: Durante el desarrollo de esta tarea se muestra una opción al usuario de buscar documentos o auditores eligiendo varios criterios de búsquedas. Los resultados saldrán reflejados en un explorador y el usuario tendrá acceso a las opciones propias de los objetos que busque.

Tabla 40 Descripción de la TI 4 Realizar estadísticas

Tarea de la ingeniería 5 Graficar estadísticas

TAREA DE LA INGENIERÍA	
Número Tarea: 5	Historia de usuario (No.7): Realizar estadísticas
Nombre Tarea: Graficar estadísticas	
Tipo Tarea: Desarrollo	Puntos Estimados: 0.6
Fecha Inicio: 7 de junio 2010	Fecha Fin: 12 de junio 2010
Programador Responsable: Arlee E. Paneque y Noel González	
Descripción: Durante el desarrollo de esta tarea se muestra la opción al usuario de visualizar las estadísticas a través de graficas que muestren en forma de dibujo los resultados.	

Tabla 41 Descripción de la TI 5 Realizar estadísticas

Tarea de la ingeniería 1 Editar propiedades

TAREA DE LA INGENIERÍA	
Número Tarea: 1	Historia de usuario (No.8): Editar propiedades
Nombre Tarea: Editar propiedades	
Tipo Tarea: Desarrollo	Puntos Estimados: 2
Fecha Inicio: 12 de junio 2010	Fecha Fin: 18 de junio 2010
Programador Responsable: Arlee E. Paneque y Noel González	

Descripción: Durante el desarrollo de esta tarea se muestran opciones al usuario que le permitan personalizar el sistema, como es cambiar fechas, cambiar imágenes de iconos que representen objetos dentro del sistema y adicionar roles y permisos dentro del mismo, esta última solo podrá disponer el administrador del sistema.

Tabla 42 Descripción de la TI 1 Editar propiedades

3.5 Conclusiones

En este capítulo se presentaron aspectos relacionados con las fases de Diseño e Implementación de la solución propuesta. Se mostraron las tarjetas CRC que se crearon para establecer las relaciones entre clases, se detallaron las tareas de la ingeniería identificadas por cada historia de usuario, divididas en las iteraciones que le correspondían y se presentó además el diseño de la base de datos utilizada.

Esta etapa se caracterizó por el trabajo conjunto entre clientes y desarrolladores, donde el cliente representa un papel fundamental, ya que es él quien define que funcionalidades contendrá el software, el grado de importancia que tienen, negocia con el programador el tiempo que demorará en implementarse y la iteración donde se desarrollará. Brinda además todos los detalles que contiene cada una de las tareas de la ingeniería.

CAPÍTULO 4 PRUEBA

4.1 Introducción

Unos de los pilares de XP es el uso de las pruebas para medir el desarrollo exitoso de cada funcionalidad del software. “Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección.”²² Durante la realización de este capítulo se detallarán las pruebas utilizadas para probar el software.

4. 2 Pruebas de aceptación.

Las pruebas de aceptación son diseñadas por los clientes pues estas pruebas miden el grado de satisfacción que tiene una funcionalidad dada para el cliente. Las pruebas de aceptación surgen de las historias de usuarios, el cliente especifica los aspectos a probar de la historia de usuario y a partir de ahí se comienza a diseñar la prueba. “Una historia de usuario puede tener más de una prueba de aceptación, tantas como sean necesarias para garantizar su correcto funcionamiento y no se considera completa hasta que no supera sus pruebas de aceptación.”²²

A continuación se exponen las pruebas de aceptación utilizadas para probar el software divididas en las iteraciones correspondientes a cada historia de usuario.

4.2.1 Iteración 1

²² Rodríguez M. y Ordoñez M. *La metodología XP aplicable al desarrollo del software educativo en Cuba*.UCI.Julio 2007.

Prueba de Aceptación 1 HU Autenticar usuario

Caso de Prueba de Aceptación	
Código: HU1_p1	Historia de Usuario (No.1): Autenticar usuario
Nombre: Entrada al sistema con datos erróneos	
Descripción: Se desea probar que el sistema no de acceso a un usuario con datos erróneos.	
Condiciones de ejecución: Juego de datos incorrectos o incompletos.	
Entrada/ Pasos de ejecución: Datos incorrectos. El sistema comprueba que los datos estén correctos para dar acceso al usuario.	
Resultado esperado: Mostrar un mensaje de error.	
Evaluación de la prueba: Satisfactoria	

Tabla 43 Descripción de la PA 1 Autenticar usuario

Prueba de Aceptación 2 HU Autenticar usuario

Caso de Prueba de Aceptación	
Código: HU1_p2	Historia de Usuario (No.1): Autenticar usuario
Nombre: Entrada al sistema con datos válidos	
Descripción: Se desea probar que el sistema de acceso a un usuario.	
Condiciones de ejecución: Juego de datos válidos de autenticación de usuario.	
Entrada/ Pasos de ejecución: El sistema comprueba que los datos estén correctos para dar acceso al usuario.	
Resultado esperado: Entrada a sistema del usuario esperado con el rol que le corresponde.	
Evaluación de la prueba: Satisfactoria	

Tabla 44 Descripción de la PA 2 Autenticar usuario

Prueba de Aceptación 1 HU Administrar auditor

Caso de Prueba de Aceptación	
Código: HU2_p1	Historia de Usuario (No.2): Administrar auditor
Nombre: Insertar un usuario con juego de datos incompletos.	
Descripción: Probar que no se puede insertar un usuario con datos incompletos o incorrectos	
Condiciones de ejecución: Deben existir errores en el juego de datos del nuevo usuario.	
Entrada/ Pasos de ejecución: Juego de datos con casillas en blanco y datos con errores, se intenta que el sistema acepte este usuario.	
Resultado esperado: El sistema debe mostrar una ventana de error.	
Evaluación de la prueba: Satisfactoria	

Tabla 45 Descripción de la PA 1 Administrar auditor

Prueba de Aceptación 2 HU Administrar auditor

Caso de Prueba de Aceptación	
Código: HU2_p2	Historia de Usuario (No.2): Administrar auditor
Nombre: Insertar un usuario repetido.	
Descripción: Se desea probar que no se puede insertar un usuario repetido.	
Condiciones de ejecución: Insertar juego de datos de un usuario que ya existe y se intenta que el sistema cree un nuevo usuario con este juego de datos.	
Entrada/ Pasos de ejecución: Juego de datos de un usuario ya existente y que este correcto.	
Resultado esperado: El sistema debe mostrar una ventana de error.	
Evaluación de la prueba: Satisfactoria	

Tabla 46 Descripción de la PA 2 Administrar auditor

Prueba de Aceptación 3 HU Administrar auditor

Caso de Prueba de Aceptación	
Código: HU2_p3	Historia de Usuario (No.2): Administrar auditor
Nombre: Modificar datos de un usuario con juego de datos correctos.	
Descripción: Se desea probar que el sistema cambia los datos de usuarios correctamente.	
Condiciones de ejecución: Se cambian datos de usuarios en el sistema.	
Entrada/ Pasos de ejecución: Juego de datos correctos que modifiquen un usuario existente.	
Resultado esperado: Se actualice el listado con los cambios realizados.	
Evaluación de la prueba: Satisfactoria	

Tabla 47 Descripción de la PA 3 Administrar auditor

Prueba de Aceptación 4 HU Administrar auditor

Caso de Prueba de Aceptación	
Código: HU2_p4	Historia de Usuario (No.2): Administrar auditor
Nombre: Eliminar usuario	
Descripción: Se desea probar que el sistema elimina satisfactoriamente un usuario.	
Condiciones de ejecución: Se accede a la opción de eliminación de un usuario	
Entrada/ Pasos de ejecución: Seleccionar el usuario a eliminar, verificar, y confirmar la eliminación.	
Resultado esperado: Se actualice el listado de usuarios.	
Evaluación de la prueba: Satisfactoria	

Tabla 48 Descripción de la PA 4 Administrar auditor

Prueba de Aceptación 2 HU Administrar empresa

Caso de Prueba de Aceptación	
Código: HU3_p1	Historia de Usuario (No.3): Administrar empresa
Nombre: Insertar una empresa con juego de datos incompletos.	
Descripción: Probar que no se pueda insertar una empresa con datos incompletos o incorrectos.	
Condiciones de ejecución: Deben existir errores en el juego de datos de la nueva empresa.	
Entrada/ Pasos de ejecución: Juego de datos con casillas en blanco y datos con errores, se intenta que el sistema acepte esta empresa.	
Resultado esperado: Se muestra una ventana de error donde se expone un mensaje con el error cometido.	
Evaluación de la prueba: Satisfactoria	

Tabla 49 Descripción de la PA 1 Administrar empresa

Prueba de Aceptación 2 HU Administrar empresas

Caso de Prueba de Aceptación	
Código: HU3_p2	Historia de Usuario (No.3): Administrar empresas
Nombre: Insertar un empresa repetida.	
Descripción: Se desea probar que el sistema no permite la duplicación de datos correspondientes a empresas.	
Condiciones de ejecución: Los datos de la empresa nueva deben existir en el sistema.	
Entrada/ Pasos de ejecución: Juego de datos de una empresa repetida.	
Resultado esperado: Se muestra una ventana de error notificando la existencia de esa empresa.	
Evaluación de la prueba: Satisfactoria	

Tabla 50 Descripción de la PA 2 Administrar empresa

Prueba de Aceptación 3 HU Administrar empresas

Caso de Prueba de Aceptación	
Código: HU3_p3	Historia de Usuario (No.3): Administrar empresas
Nombre: Modificar datos de una empresa.	
Descripción: Se desea probar que el sistema permite cambiar los datos de una empresa.	
Condiciones de ejecución: Juegos de datos correctos.	
Entrada/ Pasos de ejecución: Se selecciona una empresa y se modifican sus datos.	
Resultado esperado: El sistema debe actualizar la empresa mostrando los cambios realizados.	
Evaluación de la prueba: Satisfactoria	

Tabla 51 Descripción de la PA 3 Administrar empresas

Prueba de Aceptación 4 HU Administrar empresas

Caso de Prueba de Aceptación	
Código: HU4_p4	Historia de Usuario (No.4): Administrar empresas
Nombre: Eliminar empresa	
Descripción: Se desea probar que el sistema elimina satisfactoriamente una empresa.	
Condiciones de ejecución: Seleccionar de un listado la empresa que se desea eliminar-	
Entrada/ Pasos de ejecución: Selección de una empresa, verificación y confirmación de eliminación por parte del usuario.	
Resultado esperado: Se actualice el listado de empresas.	
Evaluación de la prueba: Satisfactoria	

Tabla 52 Descripción de la PA 4 Administrar empresas

4.2.2 Iteración 2

Prueba de Aceptación 3 HU Administrar documentos

Caso de Prueba de Aceptación	
Código: HU4_p1	Historia de Usuario (No.4): Administrar documentos
Nombre: Insertar un documento con juego de datos incompletos.	
Descripción: Probar que no se puede insertar un documento con datos incompletos o incorrectos.	
Condiciones de ejecución: Deben existir errores en el juego de datos del nuevo documento.	
Entrada/ Pasos de ejecución: Juego de datos con casillas en blanco y datos con errores, se intenta que el sistema acepte este documento.	
Resultado esperado: Se muestra una ventana de error donde se expone un mensaje con el error cometido.	
Evaluación de la prueba: Satisfactoria	

Tabla 53 Descripción de la PA 1 Administrar documentos

Prueba de Aceptación 2 HU Administrar documentos

Caso de Prueba de Aceptación	
Código: HU4_p2	Historia de Usuario (No.4): Administrar documentos
Nombre: Insertar un documento repetido.	
Descripción: Se desea probar que no se pueden duplicar documentos en el sistema.	
Condiciones de ejecución: Se inserta un documento nuevo con datos de un documento existente ya en el sistema.	
Entrada/ Pasos de ejecución: Juego de datos de un documento ya existente, se intenta que el sistema acepte la creación de este nuevo documento.	
Resultado esperado: Se muestre una ventana de error, no permitiendo la acción.	
Evaluación de la prueba: Satisfactoria	

Tabla 54 Descripción de la PA 2 Administrar documentos

Prueba de Aceptación 3 HU Administrar documentos

Caso de Prueba de Aceptación	
Código: HU4_p3	Historia de Usuario (No.4): Administrar documentos
Nombre: Modificar datos de un documento con juego de datos correctos.	
Descripción: Se desea probar que el sistema permite la modificación de datos de los documentos.	
Condiciones de ejecución: Cambio de datos de un documento que esté en el sistema.	
Entrada/ Pasos de ejecución: Se accede a un documento y se cambian sus datos.	
Resultado esperado: Se actualiza el documento con los nuevos datos.	
Evaluación de la prueba: Satisfactoria	

Tabla 55 Descripción de la PA 3 Administrar documentos

Prueba de Aceptación 4 HU Administrar documentos.

Caso de Prueba de Aceptación	
Código: HU4_p4	Historia de Usuario (No.4): Administrar documentos.
Nombre: Imprimir documento.	
Descripción: Se desea probar la opción de impresión de documento.	
Condiciones de ejecución: Seleccionar un documento a imprimir y se tenga la impresora en línea.	
Entrada/ Pasos de ejecución: Se selecciona un documento a imprimir y se acepta la opción de impresión.	
Resultado esperado: Se obtiene el documento en formato físico.	
Evaluación de la prueba: Satisfactoria	

Tabla 56 Descripción de la PA 4 Administrar documentos

Prueba de Aceptación 4 HU Administrar Expediente

Caso de Prueba de Aceptación	
Código: HU5_p1	Historia de Usuario (No.5): Administrar Expedientes
Nombre: Insertar una orden de trabajo con juegos de datos incorrectos.	
Descripción: Probar que no se puede insertar una orden de trabajo con errores en los datos.	
Condiciones de ejecución: Se intenta que el sistema acepte una orden de trabajo con juego con errores.	
Entrada/ Pasos de ejecución: Juego de datos con casillas en blanco o datos con errores.	
Resultado esperado: Se muestra una ventana de error no permitiendo realizar la acción.	
Evaluación de la prueba: Satisfactoria	

Tabla 57 Descripción de la PA 1 Administrar expedientes

Prueba de Aceptación 2 HU Administrar Expediente

Caso de Prueba de Aceptación	
Código: HU5_p2	Historia de Usuario (No.5): Administrar Expediente
Nombre: Eliminar expedientes por antigüedad.	
Descripción: Se desea probar que el sistema elimine expedientes.	
Condiciones de ejecución: Comprobar que el expediente tenga la antigüedad requerida para ser eliminado.	
Entrada/ Pasos de ejecución: Se intenta que el sistema elimine un expediente que cumple las condiciones para ser eliminado.	
Resultado esperado: Se actualice el listado de expedientes eliminando el seleccionado.	
Evaluación de la prueba: Satisfactoria	

Tabla 58 Descripción de la PA 2 Administrar expedientes

Prueba de Aceptación 1 HU Modificar Expediente

Caso de Prueba de Aceptación	
Código: HU5_p1	Historia de Usuario (No.5): Modificar Expediente
Nombre: Modificar datos de un expediente con juego de datos correctos.	
Descripción: Probar que el sistema es capaz de cambiar datos de un expediente.	
Condiciones de ejecución: Los datos a cambiar deben estar correctos.	
Entrada/ Pasos de ejecución: Se accede a un expediente y se cambian datos, se intenta que el sistema acepte los nuevos datos.	
Resultado esperado: El sistema actualice el expediente con los nuevos datos.	
Evaluación de la prueba: Satisfactoria	

Tabla 59 Descripción de la PA 1 Modificar expedientes

Prueba de Aceptación 1 HU Declarar documento final

Caso de Prueba de Aceptación	
Código: HU6_p4	Historia de Usuario (No.6): Declarar documento final
Nombre: Modificación de un documento final.	
Descripción: Se desea probar que el sistema no permite la modificación de un documento final.	
Condiciones de ejecución: El documento debe haber sido declarado final.	
Entrada/ Pasos de ejecución: Se selecciona un documento y se intenta modificar, y se intenta que el sistema acepte los cambios realizados en el documento.	
Resultado esperado: Se muestra un mensaje de error, no permitiendo el cambio del documento.	
Evaluación de la prueba: Satisfactoria	

Tabla 60 Descripción de la PA 1 Declarar Documento Final

4.2.3 Iteración 3

Prueba de Aceptación 1 HU Realizar Estadísticas.

Caso de Prueba de Aceptación	
Código: HU7_p1	Historia de Usuario (No.7): Realizar estadísticas.
Nombre: Estadística nula.	
Descripción: Se desea probar que el sistema realiza las estadísticas correctamente.	
Condiciones de ejecución: La base de datos debe estar vacía.	
Entrada/ Pasos de ejecución: Se selecciona la opción de estadísticas con la base de datos vacía.	
Resultado esperado: Se muestra un mensaje de error por falta de datos para la realización de la estadística.	
Evaluación de la prueba: Satisfactoria	

Tabla 61 Descripción de la PA 1 Realizar estadísticas

Prueba de Aceptación 2 HU Realizar Estadísticas.

Caso de Prueba de Aceptación	
Código: HU7_p2	Historia de Usuario (No.7): Realizar estadísticas.
Nombre: Ver estadísticas.	
Descripción: Se desea probar que el sistema realiza las estadísticas correctamente.	
Condiciones de ejecución: Deben existir datos en el sistema.	
Entrada/ Pasos de ejecución: Se selecciona la opción de estadísticas.	
Resultado esperado: Se muestra una ventana con el resultado consultado.	
Evaluación de la prueba: Satisfactoria	

Tabla 62 Descripción de la PA 2 Realizar estadísticas

Prueba de Aceptación 3 HU Realizar estadísticas.

Caso de Prueba de Aceptación	
Código: HU7_p3	Historia de Usuario (No.7): Realizar estadísticas.
Nombre: Realizar búsquedas.	
Descripción: Se desea probar la opción de búsqueda del sistema.	
Condiciones de ejecución: Deben existir datos en el sistema.	
Entrada/ Pasos de ejecución: Se selecciona la opción de búsqueda, se seleccionan diferentes opciones de búsquedas y se espera el resultado.	
Resultado esperado: Se muestra una ventana con el resultado consultado.	
Evaluación de la prueba: Satisfactoria	

Tabla 63 Descripción de la PA 3 Realizar estadísticas

Prueba de Aceptación 4 HU Realizar estadísticas.

Caso de Prueba de Aceptación	
Código: HU7_p4	Historia de Usuario (No.7): Realizar estadísticas.
Nombre: Ver gráficas de estadísticas.	
Descripción: Se desea visualizar una estadística a través de gráficas.	
Condiciones de ejecución: Deben existir datos en el sistema.	
Entrada/ Pasos de ejecución: Se realiza una estadística y se selecciona la opción de ver los resultados en forma de gráfica.	
Resultado esperado: Se muestra una gráfica con los datos de la estadística.	
Evaluación de la prueba: Satisfactoria	

Tabla 64 Descripción de la PA 4 Realizar estadísticas

Prueba de Aceptación 1 HU Editar propiedades.

Caso de Prueba de Aceptación	
Código: HU8_p1	Historia de Usuario (No.8): Editar propiedades.
Nombre: Personalizar iconos.	
Descripción: Se desea que el sistema permita personalizar los iconos del sistema.	
Condiciones de ejecución: Los iconos deben ser imágenes de tipo .jpg o .png.	
Entrada/ Pasos de ejecución: Se selecciona la ventana de propiedades, opción cambio de tema, seleccionar nuevo icono por elemento, y se intenta que el sistema acepte el cambio.	
Resultado esperado: Se actualizan los listados del elemento seleccionado con la nueva imagen.	
Evaluación de la prueba: Satisfactoria	

Tabla 65 Descripción de la PA 1 Editar propiedades

Prueba de Aceptación 2 HU Editar propiedades.

Caso de Prueba de Aceptación	
Código: HU8_p2	Historia de Usuario (No.8): Editar propiedades.
Nombre: Cambiar permisos a un rol determinado.	
Descripción: Se desea cambiar los permisos de un rol.	
Condiciones de ejecución: Debe ser un usuario con cuenta administrativa.	
Entrada/ Pasos de ejecución: Se selecciona la ventana de propiedades, opción cambio de rol, seleccionar rol a modificar y los nuevos permisos a los que tendrá acceso, y se intenta que el sistema acepte el cambio.	
Resultado esperado: Que el nuevo rol pueda realizar operaciones que se le determinaron.	
Evaluación de la prueba: Satisfactoria	

Tabla 66 Descripción de la PA 2 Editar propiedades

4.3 Conclusiones

Durante la realización de este capítulo se realizaron las pruebas al sistema para comprobar el nivel de satisfacción del software con los requerimientos que planteó el cliente en las historias de usuarios.

CONCLUSIONES GENERALES

Durante la realización de esta investigación se realizó un software para la automatización del Proceso de Auditoría de la Oficina de Supervisión y Control de la Universidad de las Ciencias Informáticas. Al culminar la investigación se concluyó:

- Se informatizó el proceso de auditoría en todas sus fases, brindando una mayor facilidad y ahorro de tiempo en realizar las diversas actividades que lo componen.
- Se determinó para desarrollar el sistema como metodología de desarrollo XP, lenguaje de modelado UML con la herramienta Visual Paradigm y lenguaje de programación Python, resultando un producto orientado a objetos con arquitectura cliente servidor.
- La realización del análisis y diseño del sistema permitió obtener un software con opciones como la administración de documentos, auditores, empresas, trabajo con proformas definidas y declaraciones de documentos finales. Se obtiene un listado de informes finales, facilita la realización de estadísticas con gráficas incluidas y de búsquedas con varias opciones. Permite además la personalización de la aplicación brindando una mayor facilidad y mejor acondicionamiento del personal de la oficina al trabajo con el software.
- Se sometió el software a las pruebas permitiendo medir el grado de satisfacción y el cumplimiento de los requerimientos del cliente.

La versión obtenida y documentada por el equipo de desarrollo a través de XP, permitirá la adición de nuevos módulos y la integración con otros sistemas que se utilizan en la oficina de Supervisión y Control de la Universidad, además de servir como base para la realización de nuevas versiones.

RECOMENDACIONES

Durante el desarrollo de la investigación y el desarrollo del software surgen aspectos que se deben tener en cuenta para el futuro perfeccionamiento del mismo, los cuales se sugieren a continuación:

- Continuar la implementación de la aplicación, adicionándole las interfaces requeridas para su integración con otros programas que se requieren en la Dirección de Supervisión y Control de la Universidad.
- Optimizar las funcionalidades del sistema, realizando mejoras en el rendimiento de la aplicación.
- Reforzar los requerimientos de seguridad establecidos en el sistema.

REFERENCIAS BIBLIOGRÁFICAS

- Ministerio de Auditoría y Control. *Gaceta Oficial no 003 22 de enero del 2008.Resolución 350/07.*
- Hernández León R.A. y Coello González S. *El Paradigma Cuantitativo de la investigación Científica.* Noviembre 2002.
- Asamblea Nacional Poder Popular. *Ley No. 107 De la Contraloría General de la República de Cuba.* Agosto 2009.
- Berthet Ch. *Manual de Informática.* Argentina, Julio 1988.
- Jacobson I, Booch G. y Rumbaugh J. *El Proceso Unificado de Desarrollo de Software.*España 2000
- BECK K. *Extreme Programming Explained. Embrace Change.* 1999.
- Escribano.G. F. *eXtreme Programming / Programación Extrema.* 2002.
- Palacio J. *Flexibilidad con Scrum.* Noviembre 2007.
- Tombs J.N. y Chávez Orzáez O. *Manual de Unix, Revisión 2.4,* 1995
- Robert Kirrily, *skud@cpan.org,* 2007[Revisada Enero 2010]
- Zator Systems 2008 http://www.zator.com/Cpp/E1_2_1.htm [Revisada Enero 2010]
- Gilfillan I. *La Biblia de Mysql.*
- IBM Corporation. *IBM Rational Rose. Rational Rose.pdf.*2006.
- González Blanco R. y Perez Tobalina S. *LESE-2 Introducción a Rational Rose.* Facultad de Informática de Barcelona.2002.
- Abbey M, Corey M. y Abramson I. *Oracle 9i Guía de aprendizaje.* Edición Oficial.
- PostgreSQL 2007, PostgreSQL. Tutorial de PostgreSQL. *Tutorial de PostgreSQL* September 2007.
- eAPRENDE.com. 2008. <http://www.eaprende.com/gestor-de-basededatos-mysql-postgresql-sqlite.html>.
- Rodríguez M. y Ordoñez M. La metodología XP aplicable al desarrollo del software educativo en Cuba. UCI. Julio 2007.

BIBLIOGRAFÍA

- <http://www.ciberconta.unizar.es/enlaces/mejor/contaudit/011.HTM> . Revisada Enero 2010.
- <http://www.eniac.com/productos/product9.htm> . Revisada Enero 2010.
- <http://www.aplicaeasy.com/Audieasy.aspx> Revisada Enero 2010.
- Msc. Pozo Ceballos. *La importancia relativa de la auditoría. Una propuesta metodológica*. Universidad de la Habana.
- Ministerio de Auditoría y Control. *Resolución 100/04*. Abril 2001.
- Ministerio de Auditoría y Control. *Resolución 023/06*. Abril 2001.
- Ministerio de Auditoría y Control. *Resolución 276/03*. Abril 2001.
- Dr. C. Ponjuan Dante R. y otros. *Procesos Estratégicos de la Gestión del Conocimiento*. Disponible en <http://www.imbiomed.com> . Enero 2010.
- Ballesteros Padilla A. *Importancia de la gestión de la información y el conocimiento en el proceso de cambio organizacional*. Disponible en <http://www.monografias.com> . Enero 2010.
- Dra. Ponuan Dante R. *Gestión del Conocimiento y el Profesional de la Información: Una visión desde los países de Iberoamérica*. Brasil 2006.
- Gonzalo G, Fuentes J. M. y Llorens J. *Evaluación de herramientas CASE para UML*. Málaga, Noviembre 2004.
- Marzal A. y García I. *Introducción a la programación con Python*. 2003
- Butow E. y Ryan T. *C# Hungry Minds*. Nueva York, 2002.
- Brueck D. y Tanner S. *Python Bible Hungry Minds*. Nueva York, 2001.
- Fergunsson J. y otros. *La Biblia de C#*. España 2003.
- Javier Ceballos F. *Visual C++. Programación Avanzada en Win32*. México 1999.
- Lopez A. *Java, la programación del futuro*. Uruguay 1997.
- Flanagan D. *Java en pocas palabras*. 1998.
- Dr. Cs. Alvarez de Zayaz C. *Metodología de la Investigación Científica*. Universidad de Oriente. Santiago de Cuba. 1995.
- Castellanos Simona B. *La Planificación de la Investigación educativa*. Instituto Superior Pedagógico Enrique José Varona. 1998.

-
- Colectivo de Autores. *Metodología de la Investigación Educativa*. Editorial Felix Varela, 2003.
 - Colectivo de autores. *Conferencias curso 2009-2010. Metodología Investigación Científica*. UCI, 2010. Disponibles en [http:// www.teleformacion.uci.cu](http://www.teleformacion.uci.cu) .
 - PostgreSQL Global Development Group. *PostgreSQL 8.1.0 Documentation*. 2005.
 - Gabriel Poratti G. *Redes, la guía de referencia actual y definitiva*.
 - Pressman R. *Ingeniería de Software. Un enfoque practico*. Quinta Edición.

GLOSARIO DE TÉRMINOS

- UCI: Universidad de las Ciencias Informáticas.
- XML: El Extensible Markup Language, es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium.
- GUI: La interfaz gráfica de usuario, conocida también como GUI es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz.
- Tk, Gtk, Qt: Bibliotecas de elementos básicos para construir una interfaz gráfica de usuario (GUI).
- UML: El Lenguaje Unificado de Modelado es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

- XMI: XMI o XML Metadata Interchange (XML de Intercambio de Metadatos) es una especificación para el Intercambio de Diagramas. La especificación para el intercambio de diagramas fue escrita para proveer una manera de compartir modelos UML entre diferentes herramientas de modelado.

- GPL: es una licencia creada por la Free Software Foundation en 1989 (la primera versión), y está orientada principalmente a proteger la libre distribución, modificación y uso de software.
- CVS: El Concurrent Versions System es una aplicación informática que implementa un sistema de control de versiones
- CORBA IDL: Es un lenguaje de informática utilizado para describir la interfaz de componentes software.
- ORM: El mapeo objeto-relacional es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional.
- EJBs: Los Enterprise JavaBeans son una de las API que forman parte del estándar de construcción de aplicaciones empresariales. Los EJB proporcionan un modelo de componentes distribuido estándar del lado del servidor.
- WEB: World Wide Web (también conocida como "la Web"), el sistema de documentos (o páginas web) interconectados por enlaces de hipertexto, disponibles en Internet.

- IBM: International Business Machines es una empresa multinacional que fabrica y comercializa herramientas, programas y servicios relacionados con la informática.
- Ada: es un lenguaje de programación orientado a objetos y fuertemente tipado de forma estática que fue diseñado por Jean Ichbiah de CII Honeywell Bull.
- CORBA: Common Object Request Broker Architecture — arquitectura común de intermediarios en peticiones a objetos, es un estándar que establece una plataforma de desarrollo de sistemas distribuidos facilitando la invocación de métodos remotos bajo un paradigma orientado a objetos.
- SGBD: Los sistemas de gestión de bases de datos son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.
- API: Una interfaz de programación de aplicaciones o API es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
- ACID: Es un acrónimo de Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad, y no es más que un conjunto de características necesarias para que una serie de instrucciones puedan ser consideradas como una transacción.
- TCP/IP: Es un conjunto de protocolos de red en los que se basa Internet y que permiten la transmisión de datos entre redes de computadoras.
- Net: Es un framework de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones.
- BSD: Es la licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution).