



Universidad de las Ciencias Informáticas

Facultad Regional “Mártires de Artemisa”

Título: Módulo de la Dirección de Inspección de la Administración Provincial de Artemisa.

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

Autor: Daymi Duque Carrodegua.

Tutor: Ing. Leosmel Zayas Castillo.

Co-Tutor: Mario Díaz Román.

Artemisa, Junio 2012
“Año 54 de la Revolución”

DECLARACION DE AUTORIA

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Daymi Duque Carrodegua

Ing. Leosmel Zayas Castillo

Firma del Autor

Firma del Tutor

*“Está bien celebrar el éxito, pero es más importante prestar atención a
las lecciones del fracaso”*

Bill Gates

AGRADECIMIENTOS

Agradezco a mis padres Sol Angel y Enrique por el apoyo que siempre me han dado y por saber guiarme siempre por el camino correcto.

A mi hermana Dayamí por siempre estar ahí para mí y por siempre ayudarme en todo.

A mi abuela Nidia por su apoyo y por ayudarme siempre.

A mi tía Marisol y mis primas Yaira y Yamila por siempre darme su apoyo y cariño.

A la Gallega por apoyarme y darme aliento para terminar la carrera.

A Michael por apoyarme y ayudarme siempre que lo necesite.

A mis tíos por su apoyo y ayuda para terminar mi carrera.

A Yani por ayudarme siempre que lo necesité y por ser mi amiga.

A Eleydis que sin su ayuda y la de sus compañeras no hubiese sido posible la realización del módulo para la Dirección de Inspección.

A todos mis compañeros de estudios que de una forma u otra hicieron posible que me gradúe y por siempre ayudarme en los momentos difíciles de mi carrera.

DEDICATORIA

A mis padres Sol Angel y Enrique en especial.

A mi hermana Dayamí.

A toda mi familia en general.

RESUMEN

Resumen

La revolución informática de estos tiempos se desarrolla vertiginosamente; muchas veces tiende a sorprender su rápida y constante evolución. Para el desarrollo de las aplicaciones es necesario estudiar como es el procesamiento de información. Para el desarrollo de la investigación se tuvo como objeto físico la Administración Provincial de Artemisa, específicamente la Dirección de Inspección, donde fueron estudiados los procesos que allí se desarrollan.

Se estudiaron varios conceptos relacionados con la investigación, se definió la metodología utilizada, se describieron varias herramientas y tecnologías. Además, se realiza el análisis y el diseño de la propuesta de solución y se validan mediante pruebas el resultado, en busca de errores.

Palabras claves: metodología, tecnologías, procesamiento, pruebas.

INDICE

Índice

Introducción.....	1
Capítulo 1. Fundamentación Teórica.....	8
Introducción.....	8
1.1 Conceptos Fundamentales.....	8
1.2 Estado del Arte.....	12
1.3 Metodologías de desarrollo de software.....	15
1.5 Conclusiones Parciales.....	32
Capítulo 2. Características, Análisis y Diseño del Sistema.....	33
Introducción.....	33
2.1 Características y Principales Funcionalidades del Sistema.....	33
2.2 Planificación del Proyecto por Roles.....	34
2.3 Modelo de Dominio.....	35
2.4 Lista de Reserva del Producto (LRP).....	36
2.6 Plan de Releases.....	47
2.7 Diseño con Metáforas.....	48
Capítulo 3. Adquisición y Validación de los Resultados del Sistema.....	51
Introducción.....	51
3.1 Pruebas unitarias realizadas.....	51
3.2 Resultados Obtenidos.....	57
3.3 Funcionalidades Obtenidas.....	58
3.4 Aporte social y económico.....	59
3.5 Conclusiones Parciales.....	59
Conclusiones Generales.....	60
Recomendaciones.....	61
Referencias Bibliográficas.....	62
Bibliografía.....	66
Anexos.....	67

ÍNDICE DE FIGURAS

Índice de Figuras

Figura 1: Fases de RUP.....	16
Figura 2: Modelo de dominio	36
Figura 3. Diagrama de paquetes	49
Figura 4. Diagrama de componente	50

ÍNDICE DE TABLAS

Índice de Tablas

Tabla 1. Planificación del proyecto por roles	35
Tabla 2. Lista de reserva del producto	44
Tabla 3. Historia de usuario 1	45
Tabla 4. Tarea de ingeniería 1.1	46
Tabla 5. Tarea de ingeniería 1.2	46
Tabla 6. Tarea de ingeniería 1.3	46
Tabla 7. Tarea de ingeniería 1.4	47
Tabla 8. Tarea de ingeniería 1.5	47
Tabla 9. Plan de Releases	48

INTRODUCCION

Introducción

Desde la edad media se conceptualizó a la información como una organización activa que dirige los modelos del pensamiento humano. Pero lo que marca la diferencia en los seres humanos, ya que la información se encuentra en la naturaleza misma, es la posibilidad de crear y perfeccionar sistemas de símbolos con significados. Este es el origen del lenguaje, útil para la convivencia social. Puede pensarse a la información como la manera de exteriorizar la producción del pensamiento humano.

La información es un mensaje significativo que se transmite de la fuente a los usuarios, es la expresión material del conocimiento con fines de uso, está destinada a resolver determinados problemas. Debe estar disponible públicamente y servir para el desarrollo individual y corporativo pues se encuentra presente en todos los niveles de actividad y ramas de la economía, la política y la sociedad, constituye el cimiento para cumplir la satisfacción de sus objetivos y metas.

La información y el conocimiento constituyen los dos pilares vitales que sustentan el trabajo de cualquier organización y que hacen realidad su producción, servicios o razón social. Su gestión demanda un riguroso trabajo de diseño, organización, control y comunicación a fin de que sus diferentes miembros, proveedores y usuarios puedan obtener el máximo de beneficios con el mínimo de costos. Sin una adecuada gestión de la información no se puede alcanzar la eficacia y la eficiencia que reclaman los tiempos actuales, además ocupa, cada vez más, un espacio mayor en la economía de los países a escala mundial.

Las Tecnologías de la Información y las Comunicaciones (TICs), son también un conjunto de tecnologías desarrolladas para gestionar información y enviarla de un lugar a otro. Abarcan un abanico de soluciones muy amplio. Incluyen las tecnologías para almacenar información y recuperarla después, enviar y recibir

INTRODUCCION

información de un sitio a otro, o procesar información para poder calcular resultados y elaborar informes.

En Cuba se ha emprendido el reto de la informatización de la sociedad, con el objetivo de elevar la calidad de vida del pueblo y lograr más eficiencia en la economía con un proyecto de desarrollo que tiene como objetivo la justicia social, la participación popular, la equidad y la solidaridad. Se han diseñado e iniciado la aplicación de estrategias que permiten convertir los conocimientos y las tecnologías de la información y las comunicaciones en instrumentos a disposición del avance y las profundas transformaciones revolucionarias.

Como resultado de los grandes esfuerzos realizados por el gobierno cubano, muchos son los frutos que se han alcanzado hasta el punto de que se puede decir que las TIC se han insertado en casi todas las ramas de la sociedad por lo que el trabajo continúa perfeccionándose, además se amplía el radio de acción de las nuevas tecnologías en beneficio de todas las personas. Se organizan y planifican metas ambiciosas que están a la altura de los países del primer mundo y que ya hoy no están muy lejos de poderse alcanzar. Ejemplo de ello es la Universidad de las Ciencias Informáticas (UCI) que cuenta con un gran avance tecnológico, y en la que se desarrollan muchos proyectos productivos que son de beneficio en lo nacional y también en lo internacional.

Esta universidad cuenta con varias facultades entre ellas las regionales las cuales surgen por una idea del Comandante en Jefe Fidel Castro Ruz, entre estas está la Facultad Regional "Mártires de Artemisa", la cual cumple la misión de formar profesionales y de llevar un proceso de desarrollo de software para aportar como la UCI a la economía del país.

En el año 2011 en Cuba se realizó una nueva distribución política administrativa debido a la gran cantidad de municipios que contenía la provincia de La Habana.

INTRODUCCION

Con esta nueva distribución surgen las provincias de Mayabeque y Artemisa. En esta última, en el municipio cabecera se encuentra ubicado el órgano del Gobierno Provincial, el cual está dividido en 32 órganos de direcciones y una de ellas es la de Inspección.

Este órgano de dirección tiene la misión de planificar, organizar y realizar las inspecciones (reinspecciones) para comprobar y evaluar en el territorio el cumplimiento y efectividad de la legislación vigente correspondiente y coordinar en un plan único las acciones de control que se ejecutarán en la Administración Provincial.

El proceso de gestión de información es defectuoso debido a que se realiza de forma manual, por vía telefónica, correo electrónico o mediante mensajeros de las entidades que se encuentran en los municipios de la provincia que entregan la información en formato duro o digital. Se dificulta el análisis de los reportes ya que los datos se presentan en formato "Excel", la información no está íntegra y es inconsistente. Además, es un riesgo la posibilidad real de sufrir un daño que puede contribuir negativamente en el desarrollo del funcionamiento del gobierno.

Debido a los problemas de organización y seguridad del órgano de dirección, la información puede ser afectada de origen accidental o intencionado como es el caso de desastres naturales o domésticos (incendios, inundaciones, cortocircuitos, terremotos), averías en los equipos informáticos o accesorios (plantas generadoras, instalación eléctrica), errores en la introducción de datos, como borrado indebido de archivos o modificaciones no autorizadas y difusión mal intencionada de información.

La información con la que se toman las decisiones muchas veces no se encuentra centralizada y se tiene que buscar de forma manual y dentro de grandes volúmenes de información, esto ocasiona problemas de eficiencia y confiabilidad en la

INTRODUCCION

dirección.

Esta mala gestión de la información trae consigo baja calidad en los reportes, poca fiabilidad en los mismos y retrasos en la entrega de información al presidente del gobierno, causando grandes consecuencias para la toma de decisiones en la provincia.

Las razones expuestas anteriormente dan lugar a que el **problema de investigación** a solucionar sea:

¿Cómo contribuir al proceso de gestión de la información de la Dirección de Inspección, de forma que garantice la eficiencia y confiabilidad en dicho proceso?

Definiendo como **objeto de estudio**: La gestión de información y el **campo de acción**: los procesos de gestión de la información en el sector de inspección para el Sistema Informativo de la Administración Provincial de Artemisa.

Por lo que se toma como **idea a defender**: El desarrollo de un módulo para la Dirección de Inspección que garantizará la eficiencia y confiabilidad en el proceso de gestión de información en la Administración Provincial de Artemisa.

Para dar respuesta al problema planteado se define como **objetivo general**:

Desarrollar un módulo para la Dirección de Inspección que garantice la eficiencia y confiabilidad en el proceso de gestión de información en la Administración Provincial de Artemisa.

INTRODUCCION

Los **objetivos específicos** trazados para dar cumplimiento al objetivo general son:

1. Elaborar la Fundamentación Teórica de la investigación.
2. Realizar el análisis y el diseño de la solución de software propuesta para la dirección de Inspección de la AP de Artemisa.
3. Implementar el soporte para dar solución a los requerimientos de las aplicaciones clientes para la gestión de la información de la dirección de Inspección de la AP de Artemisa.
4. Validar mediante pruebas funcionales los resultados obtenidos con la solución.

Las **tareas de la investigación** trazadas para dar cumplimiento a estos objetivos son:

1. Establecer los fundamentos teórico-metodológicos para el desarrollo de los procesos de gestión de información.
2. Caracterizar el proceso de gestión de la información en la dirección de Inspección.
3. Establecer los fundamentos que deben sostener los procesos de gestión de información en la dirección de Inspección.
4. Desarrollar un módulo para la dirección de Inspección.
5. Validar la contribución lograda a fin de que cumpla con los requisitos propuestos mediante el uso de técnicas para este fin.

Esto lleva a los **aportes prácticos**:

1. Módulo para la Dirección de Inspección de la Administración Provincial de Artemisa que garantice la correcta gestión de la información.

Métodos de investigación empleados:

Métodos Teóricos:

INTRODUCCION

- **Analítico-Sintético:** Posibilitó la realización del estudio teórico de la investigación y el análisis previo sobre el funcionamiento de la dirección de inspección permitiendo extraer sus elementos más importantes y establecer relaciones entre ellos.
- **Histórico-Lógico:** Permitió analizar la trayectoria de la dirección de inspección desde su origen hasta la actualidad y revelar sus etapas principales.
- **Modelación:** Con el objetivo de realizar el análisis y diseño del software y posteriormente la implementación.

Métodos empíricos:

- **Entrevista:** Se realizó entrevistas con el director de la dirección de inspección para tener un informe más detallado de cómo se lleva a cabo la realización del trabajo y de cómo se realiza la recopilación de información en esta dirección. (Anexo 1)

Variable Independiente: Módulo de la Dirección de Inspección de la Administración Provincial de Artemisa.

Variables Dependientes: eficiencia y confiabilidad.

La investigación está estructurada en tres capítulos de la siguiente forma:

Capítulo 1: Fundamentación Teórica: Comprende los conceptos generales y básicos que permiten el entendimiento de temas relacionados con la investigación. Además, se precisa el estado del arte en el ámbito nacional e internacional, así como se fundamentan la metodología, tecnologías y herramientas utilizadas para el desarrollo del sistema de gestión.

INTRODUCCION

Capítulo 2: Características, análisis y diseño del sistema: Se realiza una propuesta del sistema, se describe cómo debe funcionar y se destaca sus características distintivas; además, se especifican sus Requisitos Funcionales y No Funcionales. Se realiza el análisis y diseño del sistema donde se especifican los principales artefactos generados en las primeras fases.

Capítulo 3: Adquisición y validación de los resultados del sistema: Incluye toda la información relacionada con el desarrollo del sistema. Además, se muestra la información adquirida a raíz de la realización de las pruebas al sistema mediante las pruebas de caja blanca.

Finalmente se presentan las Conclusiones, Recomendaciones, Referencias Bibliográficas, Bibliografía, Anexos y Glosario de Términos.

Capítulo 1. Fundamentación Teórica

Introducción

En el presente capítulo se presentan los elementos teóricos que sirven de base a la investigación del problema planteado, se hace un estudio acerca de los sistemas de gestión de la información, detallando conceptos importantes utilizados durante la investigación referente con el objeto de estudio. Además de, un estudio de las tecnologías y herramientas a utilizar en el desarrollo del módulo de la Dirección de Inspección.

1.1 Conceptos Fundamentales.

Concepto de Gestión

La Gestión es un proceso emprendido por una o más personas para coordinar las actividades laborales de otros individuos, se define como la acción o trámite que hay que llevar a cabo para conseguir o resolver una determinada cosa. Cuando se gestiona se realizan diligencias adecuadas al logro de un negocio. Por lo tanto la gestión implica un conjunto de trámites que se llevan a cabo para resolver un asunto o concretar un proyecto. (Rebolledo Saavedra Gustavo, 1993)

El concepto de gestión hace referencia a la acción y a los efectos de gestionar o administrar. Gestionar es realizar diligencias conducentes al logro de un negocio o de un deseo cualquiera. Administrar, por otra parte, consiste en gobernar, dirigir, ordenar, disponer u organizar. Es un conjunto de trámites que se llevan a cabo para resolver un asunto o concretar un proyecto. La gestión es también la dirección o administración de una empresa o de un negocio.

(Concepto de gestión, 2012)

Después de analizar los conceptos de gestión anteriores se puede llegar a la conclusión de que la gestión es como una guía para orientar la acción, previsión, visualización y empleo de los recursos y esfuerzos a los fines que se desean de actividades que habrán de realizarse para lograr objetivos y el tiempo requerido alcanzar, la secuencia para efectuar cada una de sus partes y todos aquellos eventos involucrados en su consecución.

CAPITULO 1: FUNDAMENTACION TEORICA

Concepto de Información

Como punto de partida, es importante definir y caracterizar el elemento principal con que trabaja todo sistema de gestión de la información, es decir: la información.

La información es un conjunto organizado de datos, que constituye un mensaje sobre un cierto fenómeno o ente, permite resolver problemas y tomar decisiones, ya que su uso racional es la base del conocimiento, además son datos modificados o combinados que poseen significado de utilidad para quien los recibe. (Concepto Información, 2008, 2012)

Emery (1977) considera a la información como “los conocimientos necesarios para los distintos procesos de decisión que determinan el comportamiento de la organización” (J. G EMERY, 1977); por lo que se puede establecer que la información será la base para la toma de decisiones de expertos y/o directivos.

Castilla Plaza (1999) establece que “en el ámbito de una organización, una decisión supone la elección entre diversos cursos de acción, pudiéndose considerar a la decisión como un complejo proceso de transformación de información en acción. El sujeto decisor necesita disponer de información sobre las alternativas disponibles y sus consecuencias y sobre la situación del sistema y sus objetivos” (CASTILLA PLAZA, 1999).

Analizados estos conceptos de información se puede definir que la información es un conjunto de datos obtenidos, que se pueden modificar, o utilizar en beneficio propio. Se ha considerado que la información ha sido siempre un recurso muy valioso.

Concepto de Sistemas de Gestión

Un sistema de gestión es una estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de la organización. Las mejores empresas funcionan como unidades completas con una visión compartida. Ello engloba la información compartida, evaluaciones comparativas, trabajo en equipo y un funcionamiento acorde con los más rigurosos principios de calidad y del medio ambiente. Un sistema de gestión ayuda a lograr los objetivos de la organización

CAPITULO 1: FUNDAMENTACION TEORICA

mediante una serie de estrategias, que incluyen la optimización de procesos, el enfoque centrado en la gestión y el pensamiento disciplinado. (Sistema de gestión, 2012) El uso de sistemas de gestión, permite a las organizaciones automatizar un conjunto de funciones importantes dentro del proceso empresarial, dentro de ellas se encuentra la contabilidad, distribución de productos, recursos humanos e información. La implantación de un sistema de gestión trae consigo mejoras a la empresa, ya que permite facilitar y agilizar procesos que generalmente se realizan de forma manual. Además, permite tener un mayor control de los recursos, trabajadores e información que en ella se manipula. (Sistema de gestión, 2012)

Se puede decir que mediante un sistema de gestión se gestionan tipos específicos de contenidos de información, o sea, se buscan los aspectos comunes y fundamentales que puedan orientar a los usuarios en cuanto al tema a tratar para resolver algún problema en particular, para ellos se procesa, almacena y publica la información necesaria y con la calidad requerida al alcance de todo el que la solicite, para ayudar en el desarrollo y buen funcionamiento de la empresa o del negocio.

Concepto de Gestión de la Información

La gestión de la información es un amplio campo de estudio, pero esencialmente es el proceso de gestión de datos de una organización o empresa, es el proceso de desarrollo de prácticas y procedimientos relativos a los datos. La correcta gestión de la información conoce, incorpora y vincula todos los tipos de datos, de todas las áreas de la organización y se relaciona con todos los procesos, desde la generación de datos internos, la selección y adquisición de documentos hasta la organización de su uso. Por lo tanto la gestión de la información implica:

- Determinar la información que se precisa.
- Recoger y analizar la información.
- Registrar y recuperar la información cuando sea necesaria.
- Utilizar la información.

CAPITULO 1: FUNDAMENTACION TEORICA

- Divulgar la información. (Phil Bartle, 2009)

La Gestión de la Información integra en su organización el conjunto de instancias que la constituyen en función de hacer cumplir: cómo la información se adquiere, registra, almacena, distribuye y usa, cómo el personal designado maneja y hace llegar la información a los usuarios directos, cómo las personas usan la información, desarrollan habilidades e informativas y se convierten en divulgadores de la misma, cómo las tecnologías de la información se incorporan y perfeccionan los diferentes procesos de la gestión, cómo la capacitación y uso de la información incide en el crecimiento humano y organizacional con mejores resultados, lo que incide en los costos y beneficios de la organización.

(AUGIER ESCALONA, ALEJANDRO, 2012)

Gestión de los proyectos utilizando SXP

La Gestión de Proyectos implica la planificación, supervisión y control no solo del proceso de software, sino también del personal y los eventos que ocurren mientras evoluciona el software desde la fase preliminar a la implementación operacional. El proyecto debe planificarse estimando el esfuerzo y el tiempo para cumplir las tareas, definiendo los productos del trabajo, estableciendo puntos de control de calidad y estableciendo mecanismos para controlar y supervisar el trabajo definido en la planificación. Lo que se obtiene del mismo es un plan de proyecto que se realiza al comienzo de la actividad de gestión, el plan define el proceso y las tareas a realizar el personal que realizará el trabajo y los mecanismos para evaluar los riesgos, controlar el cambio y evaluar la calidad. Además, de la gestión del alcance del proyecto. En SXP esta gestión le viene directamente complementada de SCRUM. (ROMERO PEÑALVER, 2008)

Eficiencia

Relación entre los recursos utilizados en un proyecto y los logros conseguidos con el mismo. Se entiende que la eficiencia se da cuando se utilizan menos recursos para lograr un mismo objetivo. O al contrario, cuando se logran más objetivos con

los mismos o menos recursos. (MARTINEZ LOPEZ, 2012).

Confiabilidad

Se puede definir como la capacidad de un producto de realizar su función de la manera prevista. (GARCIA SANTILLAN, 2000)

1.2 Estado del Arte

Sistemas de gestión de la información en el ámbito internacional

Sistema de gestión de inspecciones.

El sistema de Gestión de Inspecciones resuelve el proceso completo de inspecciones previas a las que se somete a los automotores; desde que se decide que un vehículo debe recibir una inspección hasta que se realiza y los datos se encuentran cargados en los sistemas de la compañía.

Cada usuario participante en algún punto de los procesos cuenta con un escritorio personalizado de trabajo totalmente basado en la Web, desde donde realiza o participa en todas las tareas que le competen de acuerdo a su rol (peritos, legales, administrador, inspector).

En cuanto el sistema recibe el requerimiento de inspección, se dispara el flujo de trabajo que comienza con la tarea de coordinación de la inspección con el cliente. Esta tarea la llevan a cabo los responsables de inspección; que luego de coordinada, es asignada a un inspector. El sistema administra zonas geográficas y los inspectores están asignados a una o más zonas donde realizan su trabajo.

Las inspecciones se asignan en forma automática a los inspectores más adecuados una vez que están coordinadas y se conoce la dirección donde se realizará la inspección. A partir de la asignación, el inspector cuenta con un plazo definido para llevar a cabo la inspección, esto se controla mediante un "evento asincrónico" del motor del flujo de trabajo.

Si al inspector se le vence el plazo, la inspección vuelve a los responsables de inspección para que éstos tomen una acción al respecto.

CAPITULO 1: FUNDAMENTACION TEORICA

Cuando el inspector termina su día, entra al sistema desde su casa y carga las inspecciones que realizó, incluyendo las fotos digitales que forman parte del Formulario de Inspección Digital. Luego de un proceso de aprobación, se transfiere un subconjunto de la información a las aplicaciones internas de la compañía.

El sistema también permite monitorear el estado de las inspecciones, y realizar reportes de control y soporte de decisiones por el personal autorizado de la compañía. (Seguros, 2010)

Sistema de Información para la Gestión y Manejo de los Procesos de Inspección de Redes de Distribución Eléctrica de Baja, Media y Alta Tensión de la Empresa del Pacífico EPSA.

En este trabajo se presenta el diseño e implementación de un sistema de información y gestión que permite la integración de la información de inspección de todas las zonas y que a su vez normaliza todos los procedimientos al momento de programar y gestionar la información de las mismas. Así, la empresa en especial, los supervisores e inspectores de redes en particular, pueden contar con los mecanismos suficientes para el levantamiento y gestión de información de una manera integral.

Las inspecciones a las redes de distribución del mercado eléctrico, constituyen un mecanismo técnico e informativo para la observación y estudio preciso del estado actual de las redes que distribuyen energía y a su vez proveen un mecanismo para detectar las necesidades de mantenimiento para los elementos u objetos eléctricos. Las inspecciones de las redes buscan reflejar el éxito en la calidad de las redes de distribución de las empresas energéticas.

Sistema de Gestión Informativa de Vehículos (VIMS)

El Sistema de Gestión Informativa de Vehículos (VIMS) es una herramienta eficaz para administrar flotas de vehículos de la manera más eficiente dentro de un sistema descentralizado de Inspección Técnica de Vehículos (ITV), o a modo de extensión de la Inspección de Seguridad de Vehículos o Emisión de Gases.

CAPITULO 1: FUNDAMENTACION TEORICA

- Los servicios indicados a continuación han sido diseñados expresamente para su uso por las Autoridades gubernamentales.
- Registro informatizado y mantenimiento de datos.
- Control de registro de vehículos y documentos de propiedad.
- Recaudación y registro precisos y de tasas e impuestos de vehículos.
- Base de datos para fiscalización policial.
- Gestión y fiscalización de los requisitos del seguro obligatorio.
- Herramienta de gestión para programas centralizados de ITV reglamentaria.
- Análisis de estándares de rendimiento de flotas de vehículos.
- Diversas bases de datos técnicas.
- Sistema de gestión/ fiscalización para programas descentralizados.

SGS tiene la capacidad para diseñar y operar una variedad de sistemas de gestión de la información para programas de inspección de vehículos. Esto incluye disponer de tecnología punta en el área de hardware, software y telecomunicaciones, que soporten el requisito de reportes y análisis de los usuarios, así como la formación de los gestores Gubernamentales y el personal operativo. (VIMS, 2011, 1997)

Sistemas de gestión de la información en el ámbito nacional

Sistema de inspección de la planta de reactivos clínicos

El Sistema de Inspección de la calidad tiene como fundamento garantizar la obtención de productos confiables. Éste debe abarcar todas las etapas del proceso productivo, por lo que va encaminado a minimizar los desperdicios, reprocesos, devoluciones y quejas; haciendo las cosas bien, previendo los defectos y cumpliendo con los requisitos de los clientes dentro de un marco de productividad, costo y tiempo que garantice las actividades de la Empresa para permitir su crecimiento a largo plazo.

En la Empresa desde 1992 se está trabajando en la implantación de un Sistema de Calidad en la Planta de Reactivos Clínicos, por lo que es conveniente el diseño de

CAPITULO 1: FUNDAMENTACION TEORICA

un sistema a partir del cual se definan todas las variables que brinden la información necesaria para poder tomar las acciones correctivas y aplicar los proyectos de mejoras. El objetivo de este trabajo fue realizar el diseño e implantación del Sistema de Inspección en la Planta de Reactivos Clínicos. (Ing. Nancy Oña Aldama, Idania Hernández Oramas, 1999)

1.3 Metodologías de desarrollo de software

Desarrollar un buen software depende de una serie de actividades y etapas, donde elegir la mejor metodología para un equipo de desarrollo en un determinado proyecto, es trascendental para el éxito del producto. A continuación se describen algunas de las metodologías.

Proceso Unificado de Desarrollo

El Proceso Unificado de Racional (RUP) es un proceso para el desarrollo de un proyecto de un software que define claramente quien, cómo, cuándo y qué debe hacerse en el proyecto. Con 3 características esenciales está dirigido por los Casos de Uso: que orientan el proyecto a la importancia para el usuario y lo que este quiere, está centrado en la arquitectura: que Relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden, y es iterativo e incremental: donde divide el proyecto en mini proyectos donde los casos de uso y la arquitectura cumplen sus objetivos de manera más depurada.

El ciclo de vida del RUP:

RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en los distintas actividades.

CAPITULO 1: FUNDAMENTACION TEORICA

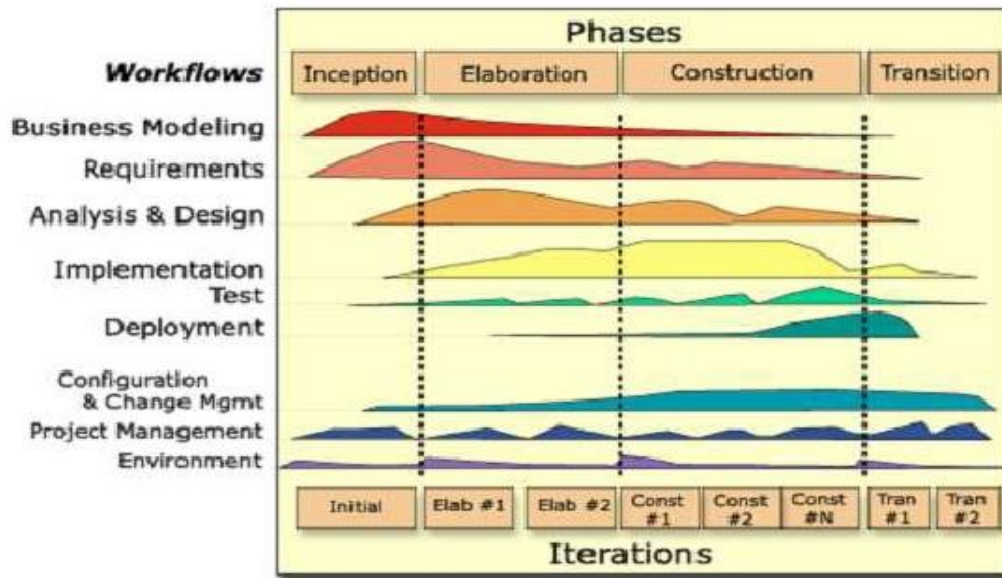


Figura 1: Fases de RUP.

En las iteraciones de cada fase se hacen diferentes esfuerzos en diferentes actividades.

- Inicio: Se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos. Se define el alcance del proyecto.
- Elaboración: se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos.
- Construcción: se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario.
- Transición: se instala el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados. (Juan Pablo Gómez Gallego, 2007)

Principales características:

- Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
- Pretende implementar las mejores prácticas en Ingeniería de Software.
- Desarrollo iterativo.
- Administración de requisitos.
- Uso de arquitectura basada en componentes.
- Control de cambios.

CAPITULO 1: FUNDAMENTACION TEORICA

- Modelado visual del software.
- Verificación de la calidad del software.

La metodología RUP es más apropiada para proyectos grandes (Aunque también pequeños), dado que requiere un equipo de trabajo capaz de administrar un proceso complejo en varias etapas. En proyectos pequeños, es posible que no se puedan cubrir los costos de dedicación del equipo de profesionales necesarios. (Ingeniería software dos, 2010)

SCRUM

Scrum, más que una metodología de desarrollo software, es una forma de auto-gestión de los equipos de programadores. Un grupo de programadores deciden cómo hacer sus tareas y cuánto van a tardar en ello. Scrum ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro. Permite además seguir de forma clara el avance de las tareas a realizar, de forma que los jefes puedan ver día a día cómo progresa el trabajo. (SCRUM, 2007) Es una de las más conocidas metodologías ágiles, y se basa en un enfoque iterativo, donde cada iteración se denomina Sprint.

Permite tener un control continuo sobre el estado actual de nuestro software, el cliente establece las prioridades y el equipo se auto-organiza para determinar la mejor forma de entregar los requerimientos con más prioridad.

Cada Semana acaba con software entregable que se puede publicar o bien seguir mejorando. Scrum implica una filosofía de trabajo que no solo implica al desarrollador, sino también al cliente dando prioridad a los individuos y las interacciones sobre los procesos y las tareas, prefiriendo el software funcional sobre la excesiva documentación, promocionando la colaboración con el cliente en lugar de la negociación de contratos y sobre todo teniendo capacidad de respuesta sobre los cambios en lugar de seguir estrictamente una planificación. (Omitsis Consulting, 2010)

Entre las cosas que nos ofrece esta metodología tenemos que:

- Permite a las organizaciones eliminar los impedimentos clásicos en el

CAPITULO 1: FUNDAMENTACION TEORICA

desarrollo de los proyectos, aumentando la satisfacción de los clientes mediante la realización de entregas frecuentes de resultados tangibles e integrándolos activamente en el ciclo de desarrollo, lo cual proporciona además una mayor adaptación y adecuación a sus necesidades.

- Potencia la formación de equipos de trabajo autosuficiente y multidisciplinarios, reduciendo la carga de gestión y proporcionando a los miembros del equipo un entorno amigable y productivo para desarrollar sus habilidades al máximo. Este entorno proporciona además, mayor calidad de vida a los trabajadores y mejora drásticamente la moral en las organizaciones.
- Se centra en el producto y las personas, y hace especial hincapié en la eliminación proactiva de todas las trabas e impedimentos que surjan durante el desarrollo. Así pues, permite a muchas organizaciones alcanzar el llamado “Efecto Toyota”: cuatro veces la productividad media del sector, con doce veces la calidad.
- Es sencillo combinar SCRUM con otras metodologías y marcos de gestión de proyectos.
- SCRUM es iterativo e incremental. Eso quiere decir que se liberan código (o producto) frecuentemente, y cada liberación representa un incremento sobre la anterior.
- El producto que funciona es la única medida del avance del proyecto.
- Todo en SCRUM tiene un límite de tiempo.
- Las entregas son productos potencialmente utilizables y/o comercializables. Eso significa que nada de presentación y nada de planos o documentos describiendo lo que hará la aplicación: se quiere tocar, ver y usar algo, aunque sea un prototipo de la función conexión.

El proceso de desarrollo Scrum se compone de 5 actividades importantes:

- Planes de lanzamientos.
- Distribución, revisión y ajuste de los estándares de producto.
- Sprint.

CAPITULO 1: FUNDAMENTACION TEORICA

- Revisión del Sprint.
- Cierre. (Omitsis Consulting, 2010)

La fase de Sprint es donde el desarrollo de software se lleva a cabo. Un Sprint consta de las siguientes sub-actividades: Elaborar; Integrar; Revisar y Ajustar. Esta fase no tiene una secuencia. A veces un elemento del backlog se tiene que desarrollar, integrar, y revisar cuando otras solo debe ser revisado o ajustado.

Revisión de Sprint

Cada Sprint es seguido por una revisión de Sprint. Durante esta revisión, el software desarrollado en el Sprint anterior se revisa y si es necesario se le añaden nuevos ítems del backlog. El grupo de revisores pueden ser: las partes interesadas del proyecto, gestores, desarrolladores y, en ocasiones los clientes, ventas y marketing.

Las actividades, y la revisión de Sprint Sprint se repiten hasta que el producto se considere listo para su distribución por los participantes en el proyecto. Luego, el proyecto pasa a la fase de cierre en que el producto se prepara para el lanzamiento y la distribución.

Cierre

En esta fase tienen lugar las actividades de debugging, marketing y promoción. Al acabar esta fase el proyecto quedará cerrado. (Omitsis Consulting, 2010)

Extreme Programming (XP)

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizada para proyectos de corto plazo y equipo de desarrollo pequeño. Esta es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo

CAPITULO 1: FUNDAMENTACION TEORICA

de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define especialmente para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. (Metodología Extreme Programming, 2009)

Principalmente esta metodología se caracteriza por:

1. Pruebas Unitarias: Se basa en las pruebas realizadas a los principales procesos, de tal manera que se adelanta en algo hacia el futuro, haciendo pruebas de las fallas que pudieran ocurrir. Es como adelantarse a obtener los posibles errores.
2. Refabricación: Se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
3. Programación en pares: Una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen de un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa.

Lo fundamental en este tipo de metodología es:

1. La comunicación, entre los usuarios y los desarrolladores.
2. La simplicidad, al desarrollar y codificar los módulos del sistema.

La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales para así impedir la aparición de riesgos críticos que perjudiquen la realización con éxito del proyecto y simultáneamente, ayuda en gran medida a disminuir el tiempo de desarrollo y sus costos. (Diego Gómez, 2009)

SXP

SXP es una metodología compuesta por las metodologías SCRUM y XP que ofrece una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad

CAPITULO 1: FUNDAMENTACION TEORICA

productiva fomentando el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo.

Consta de 4 fases principales:

- Planificación-Definición donde se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.
- Desarrollo, es donde se realiza la implementación del sistema hasta que esté listo para ser entregado.
- Entrega, puesta en marcha.
- Mantenimiento, donde se realiza el soporte para el cliente.

De cada una de estas fases se realizan numerosas actividades tales como el levantamiento de requisitos, la priorización de la Lista de Reserva del Producto, definición de las Historias de Usuario, diseño, implementación, pruebas, entre otras; de donde se generan artefactos para documentar todo el proceso. Las entregas son frecuentes, y existe una refactorización continua, lo que nos permite mejorar el diseño cada vez que se le añada una nueva funcionalidad.

Metodología de Desarrollo seleccionada: SXP

Después de analizadas las metodologías se decidió usar SXP ya que lo que se va a realizar no es de gran envergadura, constituye un proyecto pequeño y el cliente forma parte del equipo de desarrollo.

1.4 Técnicas y herramientas utilizadas para el desarrollo

Lenguaje de Modelado

“UML (Unified Modeling Language) es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos”. (Herramientas Case, 2006) UML sirve para el modelado completo de sistemas complejos, tanto en el diseño de los sistemas software como para la arquitectura hardware donde se ejecuten, además es independiente del lenguaje de implementación, de tal forma que los diseños realizados usando UML se puedan

CAPITULO 1: FUNDAMENTACION TEORICA

implementar en cualquier lenguaje que soporte las posibilidades de UML (principalmente lenguajes orientados a objetos).

UML es además un método formal de modelado. Esto aporta las siguientes ventajas:

- Mayor rigor en la especificación.
- Permite realizar una verificación y validación del modelo realizado.
- Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos). Esto permite que el modelo y el código estén actualizados, con lo que siempre se puede mantener la visión en el diseño, de más alto nivel, de la estructura de un proyecto.

Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones:

- Visualizar: UML permite expresar de una forma gráfica un sistema, de modo que otro lo puede entender.
- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión. (Ferré Grau, Xavier & Sánchez Segura, María Isabel, 2009)

Visual Parading 6.4

Visual Paradigm es una herramienta profesional considerada en el mercado mundial como muy completa y fácil de usar, es multiplataforma y proporciona excelentes facilidades de interoperabilidad con otras aplicaciones, la misma soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad y a un menor coste.

CAPITULO 1: FUNDAMENTACION TEORICA

Permite graficar todos los tipos de diagramas de clases, además de ello permite invertir código fuente de programas, archivos ejecutables y binarios en modelos UML al instante, creando de manera simple toda la documentación. La herramienta CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas y proyectos UML.

Tiene disponibilidad de múltiples versiones para cada necesidad y disponibilidad de integrarse en los principales IDEs. Apoya los estándares más recientes de las notaciones de Java y de UML. Incorpora el soporte para trabajo en equipo, que permite que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios hechos por sus compañeros. (Herramientas Case, 2006)

Lenguajes de Programación

Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente.

Un lenguaje de programación permite a un programador especificar de manera precisa: sobre qué datos una computadora debe operar, cómo deben ser estos almacenados y transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar relativamente próximo al lenguaje humano o natural, tal como sucede con el lenguaje léxico. (R Martín, D Enríquez, H Viltres, 2006)

Java 7

Supone un significativo avance en el mundo de los entornos software, y esto viene avalado por tres elementos claves que diferencian a este lenguaje desde un punto de vista tecnológico:

- Es un lenguaje de programación que ofrece la potencia del diseño orientado a objetos con una sintaxis fácilmente accesible y un entorno robusto y

CAPITULO 1: FUNDAMENTACION TEORICA

agradable.

- Proporciona un conjunto de clases potente y flexible.
- Pone al alcance de cualquiera la utilización de aplicaciones que se pueden incluir directamente en páginas Web (aplicaciones denominadas applets).

Java aporta a la Web una interactividad que se había buscado durante mucho tiempo entre usuario y aplicación. (Características de Java, 2007)

Las principales características de Java son:

- Potente.
- Simple.
- Interactivo y orientado a red.

A continuación se explica cada una de estas características:

Potente:

Orientado a Objetos: En este aspecto Java fue diseñado partiendo de cero, no siendo derivado de otro lenguaje anterior y no tiene compatibilidad con ninguno de ellos.

En Java el concepto de objeto resulta sencillo y fácil de ampliar. Además se conservan elementos "no objetos", como números, caracteres y otros tipos de datos simples.

Riqueza Semántica: Pese a su simpleza se ha conseguido un considerable potencial, y aunque cada tarea se puede realizar de un número reducido de formas, se ha conseguido un gran potencial de expresión e innovación desde el punto de vista del programador.

Robusto: Java verifica su código al mismo tiempo que lo escribe, y una vez más antes de ejecutarse, de manera que se consigue un alto margen de codificación sin errores. Se realiza un descubrimiento de la mayor parte de los errores durante el tiempo de compilación, ya que Java es estricto en cuanto a tipos y declaraciones, y así lo que es rigidez y falta de flexibilidad se convierte en eficacia. Respecto a la gestión de memoria, Java libera al programador del compromiso de tener que controlar especialmente la asignación que de ésta hace a sus necesidades específicas. Este lenguaje posee una gestión avanzada de memoria llamada

CAPITULO 1: FUNDAMENTACION TEORICA

gestión de basura, y un manejo de excepciones orientado a objetos integrados. Estos elementos realizarán muchas tareas antes tediosas a la vez que obligadas para el programador.

Modelo de objeto rico: Existen varias clases que contienen las abstracciones básicas para facilitar a los programas una gran capacidad de representación. Para ello se contará con un conjunto de clases comunes que pueden crecer para admitir todas las necesidades del programador.

Además la biblioteca de clases de Java proporciona un conjunto único de protocolos de Internet.

El conjunto de clases más complicado de Java son sus paquetes gráficos AWT (Abstract Window Toolkit) y Swing. Estos paquetes implementan componentes de una interfaz de usuario gráfica básica común a todos los ordenadores personales modernos. (Características de Java, 2007)

Simple:

Fácil Aprendizaje: El único requerimiento para aprender Java es tener una comprensión de los conceptos básicos de la programación orientada a objetos. Así se ha creado un lenguaje simple (aunque eficaz y expresivo) pudiendo mostrarse cualquier planteamiento por parte del programador sin que las interioridades del sistema subyacente sean desveladas.

Java es más complejo que un lenguaje simple, pero más sencillo que cualquier otro entorno de programación. El único obstáculo que se puede presentar es conseguir comprender la programación orientada a objetos, aspecto que, al ser independiente del lenguaje, se presenta como insalvable.

Completado con utilidades: El paquete de utilidades de Java viene con un conjunto completo de estructuras de datos complejas y sus métodos asociados, que serán de inestimable ayuda para implementar applets y otras aplicaciones más complejas. Se dispone también de estructuras de datos habituales, como pilas y tablas hash, como clases ya implementadas. (Características de Java, 2007)

Interactivo y orientado a red:

Interactivo y animado: Uno de los requisitos de Java desde sus inicios fue la

CAPITULO 1: FUNDAMENTACION TEORICA

posibilidad de crear programas en red interactivos, por lo que es capaz de hacer varias cosas a la vez sin perder rastro de lo que debería suceder y cuándo. Se da soporte a la utilización de múltiples hilos de programación (multithread).

Las aplicaciones de Java permiten situar figuras animadas en las páginas Web, y éstas pueden concebirse con logotipos animados o con texto que se desplace por la pantalla. También pueden tratarse gráficos generados por algún proceso. Estas animaciones pueden ser interactivas, permitiendo al usuario un control sobre su apariencia.

Arquitectura neutral: Java está diseñado para que un programa escrito en este lenguaje sea ejecutado correctamente independientemente de la plataforma en la que se esté actuando (Macintosh, PC, UNIX). Para conseguir esto utiliza una compilación en una representación intermedia que recibe el nombre de códigos de byte, que pueden interpretarse en cualquier sistema operativo con un intérprete de Java. La desventaja de un sistema de este tipo es el rendimiento; sin embargo, el hecho de que Java fuese diseñado para funcionar razonablemente bien en microprocesadores de escasa potencia, unido a la sencillez de traducción a código máquina hacen que Java supere esa desventaja sin problemas.

Applets: Una applet (miniaplicación) es un pequeño programa en Java transferido dinámicamente a través de Internet. Presentan un comportamiento inteligente, pudiendo reaccionar a la entrada de un usuario y cambiar de forma dinámica. Sin embargo, la verdadera novedad es el gran potencial que Java proporciona en este aspecto, haciendo posible que los programadores ejerzan un control sobre los programas ejecutables de Java que no es posible encontrar en otros lenguajes. (Características de Java, 2007)

Entorno de Desarrollo Integrado

Un entorno de desarrollo integrado, llamado también IDE (integrated development environment), es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un solo lenguaje de programación o bien poder utilizarse para varios. Un IDE es un entorno de

CAPITULO 1: FUNDAMENTACION TEORICA

programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

NetBeans7.0.1

El proyecto NetBeans consiste en un IDE de código abierto con todas las características escritas en el lenguaje de programación Java y una plataforma de aplicaciones cliente rico, que puede ser utilizado como un marco genérico para construir cualquier tipo de aplicación. Proporciona a los programadores un paquete con todas las funciones de las aplicaciones que necesitan para crear profesionales de la cruz-plataforma de escritorio, empresarial, web y aplicaciones móviles. (NetBeans IDE 7.1, 2012)

Frameworks

Un framework puede ser traducido como marco de trabajo pero en realidad es más que eso, simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes, proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Además facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas. (Francois Zaninotto Potencier, 2009)

Un framework simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes ya que en un momento determinado puede que el programador necesite utilizar cierta funcionalidad para su aplicación y oportunamente la tendrá a su disposición con solo llamarla. Proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener, facilitan la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas. Pueden incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un

CAPITULO 1: FUNDAMENTACION TEORICA

proyecto.

Son capaces de representar una arquitectura de software que modele las relaciones generales de las entidades del dominio, y llevan consigo una estructura y una metodología de trabajo la cual extienden o utilizan las aplicaciones del dominio. (Loismarx Peña, Achang González, José Rafael Viera, 2008)

JWebSocket

Este framework será utilizado para facilitar el desarrollo y funcionamiento de la aplicación. Contando con una gran variedad de funcionalidades tanto para la conectividad como para las propias funcionalidades y sistemas de seguridad por el lado del servidor.

Sistema Gestor de base de datos

Un Sistema Gestor de base de datos (SGBD) es un conjunto de programas que permiten crear y mantener una Base de Dato, asegurando su integridad, confidencialidad y seguridad. Por tanto, debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD.
- Manipular la base de datos: realizar consultas, actualizarla, generar informes. (Sistema Gestor de base de datos SGBD, 2004)

PostgreSQL 8.4+

PostgreSQL 8.4+ es el sistema de gestión de base de datos relacional orientada a objeto publicado bajo la licencia BSD más avanzado de todos, y en esta nueva versión incluye algunas novedades como:

- Permisos de usuario por columna.
- Nuevas herramientas de monitoreo de consultas para administradores.

CAPITULO 1: FUNDAMENTACION TEORICA

- Soporte de collation por base de datos, útil en entornos de varios idiomas.
- La carga que imponía VACUUM fue ampliamente reducida gracias al VisibilityMap.
- pg-migrator permite actualizaciones desde PostgreSQL 8.3 con un downtime mínimo.
- Restauración en paralelo que aumenta la velocidad de recuperación de backups hasta 8 veces. (PostgreSQL 8.4, 2009)

Protocolos de comunicación

Un protocolo es un conjunto de reglas usadas por computadoras para comunicarse unas con otras a través de una red. Un protocolo es una regla o estándar que controla o permite la comunicación en su forma más simple, un protocolo puede ser definido como las reglas que dominan la sintaxis, semántica y sincronización de la comunicación. Los protocolos pueden ser implementados por hardware, software, o una combinación de ambos. A su más bajo nivel, un protocolo define el comportamiento de una conexión de hardware.

Websocket

Se utilizará WebSocket ya que es una tecnología que proporciona un canal de comunicación bidireccional y full-dúplex sobre un único socket TCP. Está diseñada para ser implementada en navegadores y servidores web, pero puede utilizarse por cualquier aplicación cliente/servidor. La API de WebSocket está siendo normalizada por el W3C, y el protocolo WebSocket, a su vez, está siendo normalizado por el IETF (Internet EngineeringTaskForce).

Como las conexiones TCP ordinarias sobre puertos diferentes al 80 son habitualmente bloqueadas por los administradores de redes, el uso de esta tecnología proporcionaría una solución a este tipo de limitaciones proveyendo una funcionalidad similar a la apertura de varias conexiones en distintos puertos, pero multiplexando diferentes servicios WebSocket sobre un único puerto TCP (a costa de una pequeña sobrecarga del protocolo).

Arquitectura de Software

A medida que crece la complejidad de las aplicaciones, y que se extiende el uso de sistemas distribuidos y sistemas basados en componentes, los aspectos arquitectónicos del desarrollo de software están recibiendo un interés cada vez mayor, tanto desde la comunidad científica como desde la propia industria del software.

Sin pretender establecer una definición completa ni definitiva, se considera como Arquitectura la estructura de alto nivel de un sistema de software, lo que incluye sus componentes, las propiedades observables de dichos componentes y las relaciones que se establecen entre ellos.

Esta definición se centra en aspectos puramente descriptivos, y determina que cualquier sistema de software, o al menos cualquiera que tenga una cierta complejidad, tiene una arquitectura, independientemente de si esta arquitectura está representada en algún lugar de forma explícita, o incluso de si quienes desarrollaron el sistema eran conscientes de ella. (Addison Wesley., 1998.)

El módulo para la Dirección de Inspección está conformado por una arquitectura en n-capas, es un estilo de programación donde el objetivo principal es separar los diferentes aspectos del desarrollo tales como:

- Capa de Presentación.
- Capa de Negocio.
- Capa de Acceso a Datos.
- Capa de datos.

La capa de presentación es la que ve el usuario. Le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado

CAPITULO 1: FUNDAMENTACION TEORICA

previo para comprobar que no hay errores de formato). También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario. Esta capa se comunica únicamente con la capa de negocio.

La capa del negocio fue la que facilitó la realización del módulo para la Dirección de Inspección ya que es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Además se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos almacenar o recuperar información de él.

La capa de acceso a datos es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Capa de datos: Es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Sistemas de Control de Versiones(CVS)

RapidSVN 0.12

Es software libre que se utiliza para gestionar los datos del repositorio. A cada integrante del proyecto se le asignan diferentes permisos según los documentos y la información que se necesitan para el rol que desempeñe. También es una aplicación web que permite gestionar repositorios Maven y además:

- Archiva
- Resuelve los problemas que pueden aparecer en organizaciones
- Resude el tiempo de construcción
- Mayor control de los artefactos descargados.

Características:

CAPITULO 1: FUNDAMENTACION TEORICA

- Autenticación/Autorización (roles) en Archivo y repositorios gestionados.
- Gestión de usuarios.
- Gestión de repositorios internos.
- Navegación sobre los artefactos.
- Identificación de los artefactos anónimos.
- Búsqueda de artefacto en los repositorios definidos.

1.5 Conclusiones Parciales

En este capítulo se abordó sobre algunos sistemas de gestión de la información existentes. Se abordaron también conceptos que facilitaron la comprensión del trabajo y características de las herramientas, tecnologías y metodología a utilizar para el desarrollo del “Módulo para la Dirección de Inspección de la Administración Provincial de Artemisa”.

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

Capítulo 2. Características, Análisis y Diseño del Sistema

Introducción

En el presente capítulo para definir las características del sistema, se describen las principales funcionalidades de la solución a implementar, se elaboran las historias de los usuarios del negocio, las tareas de ingeniería, además se incluye el diseño con metáforas y el diagrama de componentes. También se realiza el modelado del dominio y se hace la propuesta del sistema describiendo como debe funcionar.

2.1 Características y Principales Funcionalidades del Sistema

El sistema a desarrollar tiene como objetivo llevar a cabo la correcta gestión de la información en la Administración Provincial de Artemisa centrándonos en este caso en el Órgano de la Dirección de Inspección, asegurando una correcta organización y seguridad en el órgano de dirección, además de ayudar a que la información se encuentre centralizada para así hacer más fácil el trabajo a la hora de la toma de decisiones.

La gestión de la información de los datos consiste en Insertar, Eliminar, Modificar y Buscar datos referentes a la información mensual a rendir por cada uno de los municipios del mes y acumulada hasta el mes. En la misma cuando se gestiona la información de las actividades del registro de la tabla 1 se recogen datos como municipio, mes, centros violadores, total de inspecciones. Cuando se gestiona la información de los indicadores del registro para la tabla 2 con decreto 274 y resolución 4272 se recogen datos como actualizar los modelos del registro, cantidad de infractores, el valor promedio por multas. Además se cuenta con la opción de generar reportes y se pueden modificar si es necesario realizar algún cambio.

El sistema cuenta con una interfaz agradable y asequible para todo tipo de usuario

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

que interactúe con ella. Facilitando así una correcta gestión de la información y un mejor trabajo.

2.2 Planificación del Proyecto por Roles.

ROL	RESPONSABILIDAD	NOMBRE
Gerente	Es el responsable de tomar las decisiones finales, acerca de estándares y convenciones a seguir durante el proyecto.	<u>Dania Fernández</u>
Jefe de proyecto	Organiza y guía las reuniones. Asegura condiciones adecuadas para el proyecto.	Leosmel Zayas
Programador	Es el encargado de producir el código y escribir las pruebas unitarias. Debe existir una comunicación y coordinación adecuada entre los programadores y otros miembros del equipo.	Daymi Duque Eleydis Agramonte
Cliente	El cliente participa en las tareas que involucran la lista de reserva del producto.	Trabajadores del área de inspección CAP.
Analista	Es el encargado de escribir las historias de usuario y las pruebas funcionales para validar su implementación.	Daymi Duque Eleydis Agramonte
Diseñadores	Encargados del diseño del	Daymi Duque Eleydis Agramonte

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

	sistema; así como el de los prototipos de interfaces, máximos responsables de la realización del diseño de las metáforas y supervisan el proceso de construcción.	
Encargado de Pruebas	Es el encargado de ayudar al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.	Daymi Duque Eleydis Agramonte
Arquitecto	Se vincula directamente con el analista y el diseñador debido a que su trabajo tiene que ver con la estructura y el diseño en grande del sistema. Ayuda en el diseño de las metáforas.	Daymi Duque Eleydis Agramonte

Tabla 1. Planificación del proyecto por roles

2.3 Modelo de Dominio

Dentro de las actividades más importantes definidas en la metodología SXP se encuentra la definición del

Modelo de Historias de Usuario del Negocio, en el cual se hace una detallada descripción del negocio en cuestión. Pero si dicho negocio no está bien definido entre los clientes y los ejecutores del proyecto; entonces es generado el llamado Modelo de Dominio:

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

Diagrama de Dominio del sistema:

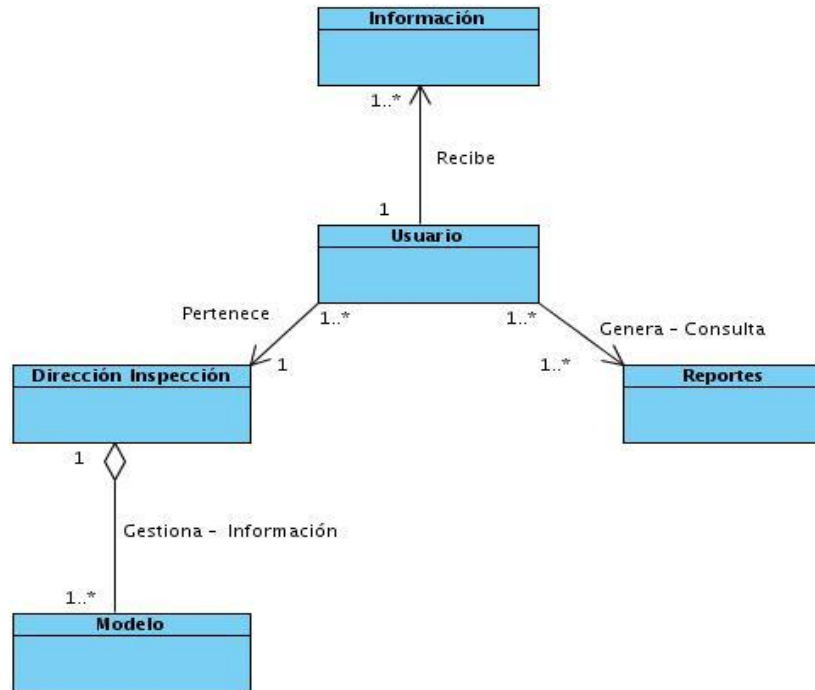


Figura 2: Modelo de dominio

Dirección Inspección: Es la representación de la Dirección de Inspección de la Administración Provincial de Artemisa.

Usuario: Especialista de la Dirección de Inspección que se encarga de gestionar toda la información referente a dicha dirección.

Información: Información que maneja la Dirección de Inspección para realizar su trabajo.

Reporte: muestra de toda la información en general.

Modelo: contiene la información referente a la dirección de inspección.

2.4 Lista de Reserva del Producto (LRP)

En la metodología SXP se cuenta con la actividad Lista de Reserva del Producto (LRP) en la cual se recoge una lista priorizada con todo el trabajo a desarrollar en

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

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

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

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

Lista de Reserva del Producto (LRP)

Prioridad	Ítem *	Descripción	Estimación	Estimador
Muy Alta				
	1	Insertar información sobre las actividades del registro para la Tabla1.	2días	Analista
	2	Modificar información sobre las actividades del registro para la Tabla1.	2 días	Analista
	3	Eliminar información sobre las actividades del registro para la Tabla1.	1día	Analista
	4	Buscar información sobre las actividades del registro para la Tabla1.	1día	Analista
	5	Insertar información sobre los indicadores del registro para la Tabla 2 con decreto 274 y resolución 4272.	2días	Analista

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

	6	Modificar información sobre los indicadores del registro para la Tabla 2 con decreto 274 y resolución 4272.	2 días	Analista
	7	Eliminar información sobre los indicadores del registro para la Tabla 2 con decreto 274 y resolución 4272.	1 día	Analista
	8	Buscar información sobre los indicadores del registro para la Tabla 2 con decreto 274 y resolución 4272.	1 día	Analista
	9	Insertar información sobre los indicadores del registro para la Tabla 2.	2 días	Analista
	10	Modificar información sobre los indicadores del registro para la Tabla 2.	1 día	Analista
	11	Eliminar información sobre los indicadores del registro para la Tabla 2.	1 día	Analista
	12	Buscar información sobre los indicadores del registro para la Tabla 2.	1 día	Analista
	13	Insertar información sobre el control de las inspecciones internas del registro.	2 días	Analista
	14	Modificar información sobre el control de las inspecciones internas del registro.	1 día	Analista
	15	Eliminar información sobre el control de las inspecciones internas del registro.	1 día	Analista

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

	16	Buscar información sobre el control de las inspecciones internas del registro.	1día	Analista
	17	Insertar información sobre el control de inspecciones con decreto 274 del registro.	2días	Analista
	18	Modificar información sobre el control de inspecciones con decreto 274 del registro.	1día	Analista
	19	Eliminar información sobre el control de inspecciones con decreto 274 del registro.	1día	Analista
	20	Buscar información sobre el control de inspecciones con decreto 274 del registro.	1día	Analista
	21	Insertar información sobre el control de la situación del personal de las DIM-CAM y DIP-CAP.	2días	Analista
	22	Modificar información sobre el control de la situación del personal de las DIM-CAM y DIP-CAP.	1día	Analista
	23	Eliminar información sobre el control de la situación del personal de las DIM-CAM y DIP-CAP.	1día	Analista
	24	Buscar información sobre el control de la situación del personal de las DIM-CAM y DIP-CAP del registro.	1día	Analista
	25	Insertar información sobre el control de gasto de salario y otros gastos corrientes de las DIM-CAM y DIP-CAP.	2días	Analista

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

	26	Modificar información sobre el control de gasto de salario y otros gastos corrientes de las DIM-CAM y DIP-CAP.	1día	Analista
	27	Eliminar información sobre el control de gasto de salario y otros gastos corrientes de las DIM-CAM y DIP-CAP.	1día	Analista
	28	Buscar información sobre el control de gasto de salario y otros gastos corrientes de las DIM-CAM y DIP-CAP del registro.	1día	Analista
	29	Insertar información sobre el acta de conciliación con las oficinas de cobro de multas por las DIM-CAM y DIP-CAP.	2días	Analista
	30	Modificar información sobre el acta de conciliación con las oficinas de cobro de multas por las DIM-CAM y DIP-CAP.	1día	Analista
	31	Eliminar información sobre el acta de conciliación con las oficinas de cobro de multas por las DIM-CAM y DIP-CAP.	1día	Analista
	32	Buscar información sobre el acta de conciliación con las oficinas de cobro de multas por las DIM-CAM y DIP-CAP del registro.	1día	Analista
	33	Insertar información sobre la conciliación de multas por incisos.	2días	Analista

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

	34	Modificar información sobre la conciliación de multas por incisos.	1día	Analista
	35	Eliminar información sobre la conciliación de multas por incisos.	1día	Analista
	36	Buscar información sobre la conciliación de multas por incisos de la Tabla 7 del registro.	1día	Analista
	37	Insertar información sobre la conciliación de multas por incisos de la Tabla 7.1 del registro.	2días	Analista
	38	Modificar información sobre la conciliación de multas por incisos de la Tabla 7.1 del registro.	1día	Analista
	39	Eliminar información sobre la conciliación de multas por incisos de la Tabla 7.1 del registro.	1día	Analista
	40	Buscar información sobre la conciliación de multas por incisos de la Tabla 7.1 del registro.	1día	Analista
	41	Insertar información sobre la conciliación de multas por incisos de la Tabla 7.2 del registro.	2días	Analista
	42	Modificar información sobre la conciliación de multas por incisos de la Tabla 7.2 del registro.	1día	Analista
	43	Eliminar información sobre la conciliación de multas por incisos de la Tabla 7.2 del registro.	1día	Analista
	44	Buscar información sobre la conciliación de multas por incisos de la Tabla 7.2 del registro.	1día	Analista

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

	45	Insertar información sobre la conciliación de multas por incisos de la Tabla 7.3 del registro.	2días	Analista
	46	Modificar información sobre la conciliación de multas por incisos de la Tabla 7.3 del registro.	1día	Analista
	47	Eliminar información sobre la conciliación de multas por incisos de la Tabla 7.3 del registro.	1día	Analista
	48	Buscar información sobre la conciliación de multas por incisos de la Tabla 7.3 del registro.	1día	Analista
Alta				
	49	Generar reporte sobre las actividades del registro para la Tabla 1.	1día	Analista
	50	Generar reporte sobre los indicadores del registro para la Tabla 2 con decreto 274 y resolución 4272.	1día	Analista
	51	Generar reporte sobre los indicadores del registro para la Tabla 2.	1día	Analista
	52	Generar reporte sobre el control de las inspecciones internas del registro.	1día	Analista
	53	Generar reporte sobre el control de inspecciones con decreto 274 del registro.	1día	Analista
	54	Generar reportes sobre el control de la situación del personal de las DIM-CAM y DIP-CAP del modelo de	1día	Analista

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

		registro.		
	55	Generar reportes sobre el control de gasto de salario y otros gastos corrientes de las DIM-CAM y DIP-CAP del modelo de registro.	1día	Analista
	56	Generar reportes sobre el acta de conciliación con las oficinas de cobro de multas por las DIM-CAM y DIP-CAP del modelo de registro.	1día	Analista
	57	Generar reportes sobre la conciliación de multas por incisos de la tabla 7 del modelo de registro.	1día	Analista
	58	Generar reportes sobre la conciliación de multas por incisos de la tabla 7.1 del modelo de registro.	1día	Analista
	59	Generar reportes sobre la conciliación de multas por incisos de la tabla 7.2 del modelo de registro.	1día	Analista
	60	Generar reportes sobre la conciliación de multas por incisos de la tabla 7.3 del modelo de registro.	1día	Analista
Media				
Baja				
Requisitos no funcionales				
	61	Tener un ordenador con más de 1 GB de memoria RAM.		
	62	Interfaz amigable.		

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

	63	El sistema debe proveer de forma ordenada y detallada las funcionalidades del sistema.		
	64	Las salidas del sistema deben de ser exactas evitando los errores en el procesamiento de los datos.		
	65	El sistema debe ser escalable de modo que futuras funcionalidades puedan ser implementadas e incorporadas.		
	66	La información manejada por el sistema estará protegida de acceso no autorizado y divulgación a terceras personas y aquellos usuarios del sistema deben autenticarse antes de realizar cualquier actividad.		

Tabla 2. Lista de reserva del producto

2.5 Historias de Usuario y Tarea de Ingeniería

Cuando se hace uso de la metodología SXP, para especificar los casos de uso como correspondería en RUP, se hace mediante historias de usuarios, descriptoras de las tareas que el sistema debe hacer, cuestión que depende en gran medida de las especificaciones realizadas por el cliente. Se escriben con un lenguaje natural y con palabras concisas para no exceder su tamaño en unas pocas líneas de texto. Van a ser la guía para la construcción posterior de las pruebas de aceptación comprobando de esta manera la correcta implementación de las historias de usuario.

1. Gestionar información de las actividades del registro para la Tabla 1.

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

Historia de Usuario	
Número: HU_1	Nombre Historia de Usuario: Gestionar información de las actividades del registro para la Tabla 1.
Modificación de Historia de Usuario Número : ninguna	
Usuario: Eleidys Agramonte Díaz Daymi Duque Carrodegua	IteraciónAsignada: 2
Prioridad en Negocio: Muy Alta	PuntosEstimados: 1/2
Riesgo en Desarrollo: Muy Alto	PuntosReales: 1/2
Descripción: La presente historia de usuario tiene como objetivo insertar, modificar, eliminar y buscar toda la información referente a las actividades, precios, comercio, el MEP (Ministerio de Economía y Planificación) que se distribuye en Ornato Público e Higiene Comunal, entre otras cosas que se encuentran plasmadas en el modelo de registro para la tabla número 1.	
Observaciones: Ninguna	
Prototipo de Interfaz: Ver anexo #2 Gestionar información de las actividades del registro para la Tabla 1.	

Tabla 3. Historia de usuario 1

Tarea de Ingeniería	
Número Tarea: 1.1	Número Historia de Usuario: HU_1
Nombre Tarea: Investigar cómo implementar las funcionalidades Insertar, Eliminar, Modificar y Buscar para la gestión de la información en la dirección de inspección.	
Tipo de Tarea : Investigación	Puntos Estimados: 1/6
Fecha Inicio: 30/01/2012	Fecha Fin: 31/01/2012
Programador Responsable: Daymi Duque Carrodegua Eleydis Agramonte Díaz	
Descripción: Se llevará a cabo un estudio sobre el proceso de cómo realizar las diferentes funcionalidades insertar, eliminar, modificar y buscar para la gestión de la	

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

información en la dirección de inspección.

Tabla 4. Tarea de ingeniería 1.1

Tarea de Ingeniería	
Número Tarea: 1.2	Número Historia de Usuario: HU_1
Nombre Tarea: Implementación de la funcionalidad Insertar información sobre las actividades del registro de la dirección de inspección.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1/12
Fecha Inicio: 31/01/2012	Fecha Fin: 01/02/2012
Programador Responsable: Daymi Duque Carrodegas Eleydis Agramonte Díaz	
Descripción: A partir de la investigación realizada se sientan las bases para la implementación de esta funcionalidad sobre la dirección de inspección.	

Tabla 5. Tarea de ingeniería 1.2

Tarea de Ingeniería	
Número Tarea: 1.3	Número Historia de Usuario: HU_1
Nombre Tarea: Implementación de la funcionalidad Eliminar información sobre las actividades del registro de la dirección de inspección.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1/12
Fecha Inicio: 31/01/2012	Fecha Fin: 01/02/2012
Programador Responsable: Daymi Duque Carrodegas Eleydis Agramonte Díaz	
Descripción: A partir de la investigación realizada se sientan las bases para la implementación de esta funcionalidad sobre la dirección de inspección.	

Tabla 6. Tarea de ingeniería 1.3

Tarea de Ingeniería	
Número Tarea: 1.4	Número Historia de Usuario: HU_1

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

Nombre Tarea: Implementación de la funcionalidad Modificar información sobre las actividades del registro de la dirección de inspección.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1/12
Fecha Inicio: 01/02/2012	Fecha Fin: 02/02/2012
Programador Responsable: Daymi Duque Carrodegua Eleydis Agramonte Díaz	
Descripción: A partir de una investigación realizada se sientan las bases para la implementación de esta funcionalidad sobre la dirección de inspección.	

Tabla 7. Tarea de ingeniería 1.4

Tarea de Ingeniería	
Número Tarea: 1.5	Número Historia de Usuario: HU_1
Nombre Tarea: Implementación de la funcionalidad Buscar información sobre las actividades del registro de la dirección de inspección.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1/12
Fecha Inicio: 01/02/2012	Fecha Fin: 02/02/2012
Programador Responsable: Daymi Duque Carrodegua Eleydis Agramonte Díaz	
Descripción: A partir de una investigación realizada se sientan las bases para la implementación de esta funcionalidad sobre la dirección de inspección.	

Tabla 8. Tarea de ingeniería 1.5

2.6 Plan de Releases

Releases	Descripción de la iteración	Orden de la HU a implementar	Duración total
----------	-----------------------------	------------------------------	----------------

CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

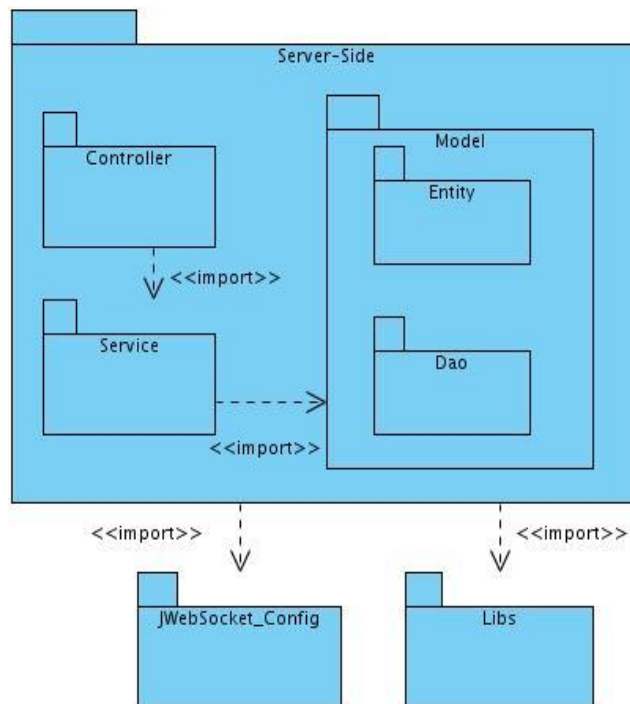
1.	En esta iteración se implementarán las historias de usuario que tienen prioridad muy alta.	HU_1, HU_3, HU_5, HU_7, HU_9, HU_11, HU_13, HU_15, HU_17, HU_19, HU_21, HU_23.	10 Semanas
2.	En esta iteración se desarrollarán las historias de usuario de prioridad alta y se integrarán con las historias de usuario ya implementadas.	HU_2, HU_4, HU_6, HU_8, HU_10, HU_12, HU_14, HU_16, HU_18, HU_20, HU_22, HU_24.	2 Semanas

Tabla 9. Plan de Releases

2.7 Diseño con Metáforas.

El Diseño con metáforas es sencillamente el diseño de la solución más simple que pueda funcionar y ser implementado en un momento dado del proyecto; lo cual genera el artefacto conocido como Modelo de Diseño, que a su vez está compuesto por un diagrama de paquetes, el cual expone dicho diseño.

A continuación se presenta el diagrama de paquetes para el sistema que se propone:



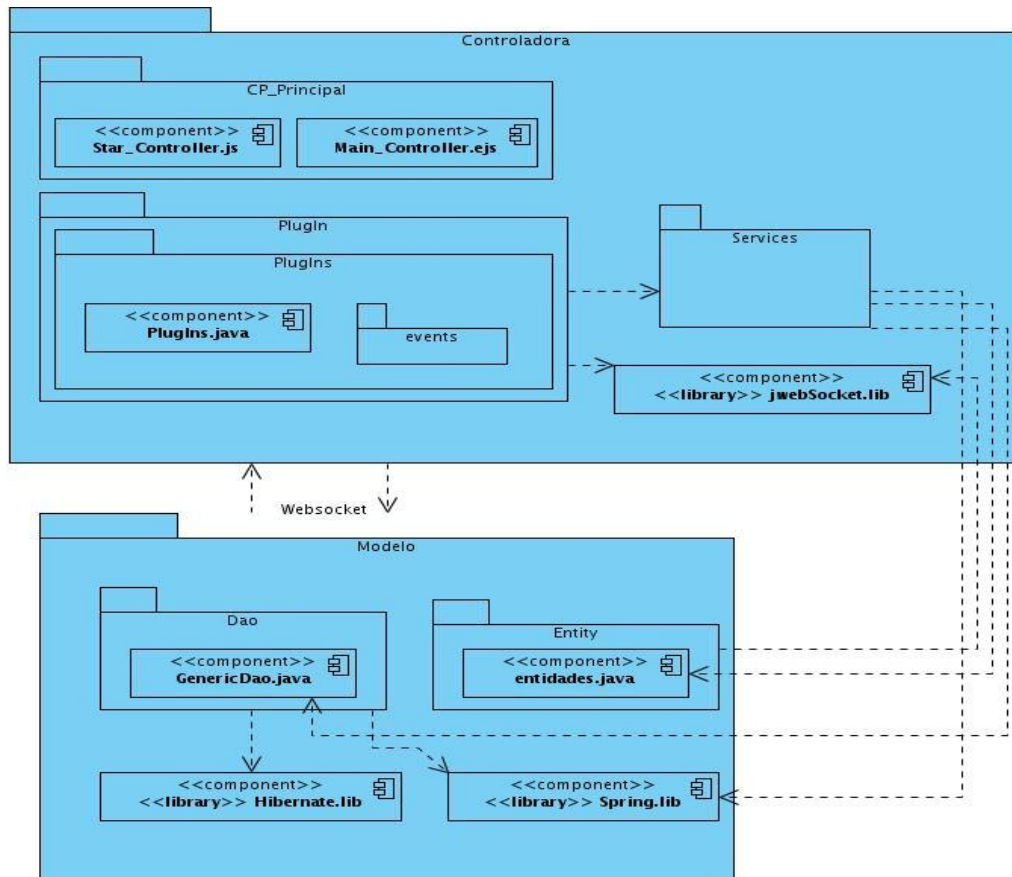
CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

Figura 3. Diagrama de paquetes

2.8 Diagrama de Componentes

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

A continuación se presenta el diagrama de componentes para el sistema que se propone:



CAPITULO 2: CARACTERISTICAS, ANALISIS Y DISEÑO DEL SISTEMA

Figura 4. Diagrama de componente

2.9 Conclusiones Parciales

En este capítulo resultaron aprobados los requisitos funcionales necesarios para obtener un sistema eficaz. Se definió el modelo conceptual que identifica los conceptos más importantes del dominio del problema. Además se reflejan claras las tareas que el sistema debe realizar, de esta forma se obtiene el módulo para la Dirección de Inspección de la Administración Provincial de Artemisa.

CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA

Capítulo 3. Adquisición y Validación de los Resultados del Sistema

Introducción

En este capítulo se describen los casos de prueba de aceptación, con los resultados obtenidos y las funcionalidades que realiza el módulo para la dirección de inspección. También se trata sobre el aporte social y económico que ofrece el sistema informativo a la Administración Provincial de Artemisa.

3.1 Pruebas unitarias realizadas

La etapa de pruebas es una de las fases del ciclo de vida de los proyectos. Se la podría ubicar después del análisis, el diseño y la programación, pero dependiendo del proyecto en cuestión y del modelo de proceso elegido, su realización podría ser en forma paralela a las fases citadas o inclusive repetirse varias veces durante la duración del proyecto.

La importancia de esta fase será mayor o menor según las características del sistema desarrollado, llegando a ser vital en sistemas de tiempo real u otros en los que los errores sean irre recuperables.

Las pruebas no tienen el objeto de prevenir errores sino de detectarlos. Se efectúan sobre el trabajo realizado y se deben encarar con la intención de descubrir la mayor cantidad de errores posible.

Aunque no hay una clasificación oficial o formal acerca de los diversos tipos de pruebas de software, existen dos vertientes fundamentales:

- Pruebas de tipo Caja Negra (Black Box testing): cuando una aplicación es probada usando su interfaz externa, generalmente la GUI.
- Pruebas de tipo Caja Blanca (White Box testing): cuando una aplicación es probada desde dentro, usando su lógica aplicativa.

Una prueba de tipo Caja Negra se lleva a cabo sin tener conocimiento de la

CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA

estructura/funcionamiento interno del sistema, de ahí su nombre. Quien realiza la prueba sólo conoce las entradas apropiadas que deberá recibir la aplicación, así como las correspondientes salidas, sin llegar a saber cómo es que se realiza este proceso.

La prueba de tipo Caja Blanca utiliza datos para realizar la tarea derivados de un análisis directo del código a ser probado; a diferencia de la prueba de tipo Caja Negra, se necesita conocimiento específico del código para analizar los resultados. Para este trabajo se han realizado las pruebas de caja blanca debido a que es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivar los casos de prueba.

Las pruebas de caja blanca intentan garantizar que:

- Se ejecutan al menos una vez todos los caminos independientes de cada módulo.
- Se utilizan las decisiones en su parte verdadera y en su parte falsa.
- Se ejecuten todos los bucles en sus límites.
- Se utilizan todas las estructuras de datos internas.

A continuación se muestran las pruebas realizadas a 4 métodos programados:

```
public void processEvent(InsertActivitiesEvents events, C2SResponseEvent response){
```

```
1   Boolean v =getService().insertActivities(events);
```

```
2   if(v){
```

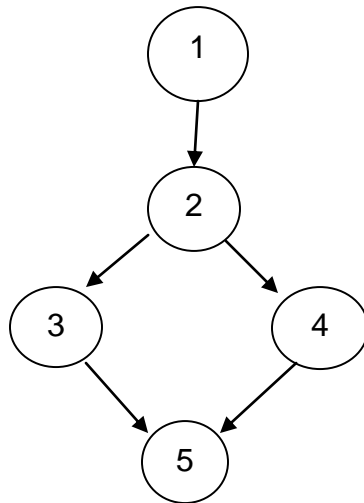
```
3   response.getArgs().setString("response", "La actividad y el" +  
"decreto, mes, municipio" + "insertado ya existe");
```

CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA

```

4  else {
    response.getArgs().setString("response", "Se ha insertado" +
    "los datos correctamente");
5  }

```



Complejidad Ciclomática = 2

Camino: C1 1 – 2 – 3 – 5 C2 1 – 2 – 4 – 5

Caso de Prueba para C1	
v	false
response	“los datos no se han insertado correctamente”
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que el organismo no ha sido insertado.

Caso de Prueba para C2	
v	true
response	“los datos se han insertado correctamente”
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que el organismo ha sido insertado.

CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA

```
public void processEvent(ModifyActivitiesEventsm Events, C2SResponseEvent response) {
```

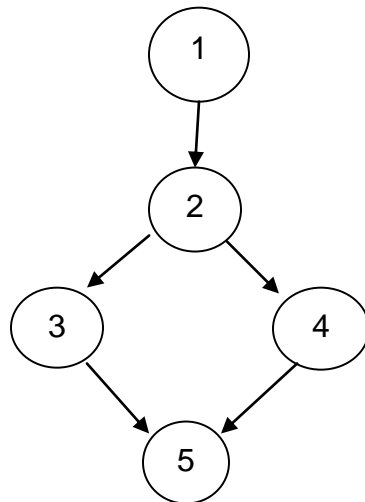
```
1 Boolean v =getService().modifyActivities(events);
```

```
2 if(v){
```

```
3 response.getArgs().setString("response", "los datos" + " no se han podido modificar");
```

```
4 else {
response.getArgs().setString("response", "los datos" + "se han modificado correctamente");
```

```
5 }
```



Complejidad Ciclomática = 2

Camino: C1 1 – 2 – 3 – 5

C2 1 – 2 – 4 – 5

Caso de Prueba para C1	
v	false
response	“los datos no se han modificado correctamente”
Resultado	El método ha recorrido el camino adecuado devolviendo un

CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA

	mensaje que el organismo no ha sido modificado.
--	---

Caso de Prueba para C2	
v	true
response	"los datos se han modificado correctamente"
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que el organismo ha sido modificado.

```
public void processEvent(SearchActivitiesEventsaEvents, C2SResponseEvent  
response) {
```

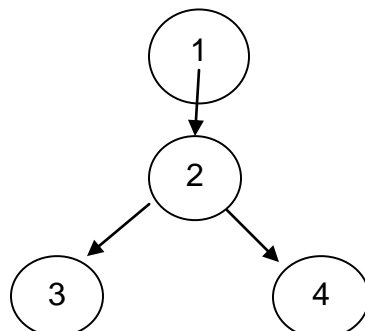
```
1 List<Activity>activitiesToSearch=activitiesService.searchActiviti  
es (searchActivities);
```

```
2 if(!activitiesToSearch.isEmpty()){
```

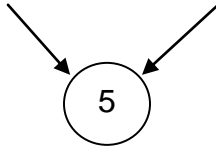
```
3 response.getArgs().setList("activitiesToSearch",  
activitiesToSearch); r  
esponse.getArgs().setBoolean("v", true);  
}
```

```
4 else{  
response.getArgs().setList("activitiesToSearch",  
activitiesToSearch);  
response.getArgs().setBoolean("v", false);  
}
```

```
5 }
```



CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA



Complejidad Ciclomática = 2

Caminos: C1 1 – 2 – 3 – 5

C2 1 – 2 – 4 – 5

Caso de Prueba para C1	
activitiesToSearch.isEmpty()	false
response	activitiesToSearch
v	true
Resultado	El método ha recorrido el camino adecuado devolviendo un listado con los resultados de la búsqueda en la variable activitiesToSearch.

Caso de Prueba para C2	
activitiesToSearch.isEmpty()	true
response	activitiesToSearch
v	false
Resultado	El método ha recorrido el camino adecuado devolviendo un listado vacío en la variable activitiesToSearch.

```
public void processEvent(DeleteActivitiesEventsdEvents, C2SResponseEvent response) {
```

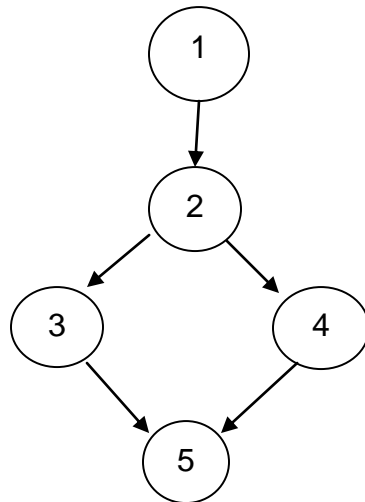
```

1   Boolean v =getService().deleteActivities(events);
2   if(v){
3       response.getArgs().setString("response", "los datos" + " no se han eliminado");
4       else {
           response.getArgs().setString("response", "los datos" + "se han eliminado correctamente");
       }
    }

```

CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA

5 }



Complejidad Ciclomática = 2

Caminos: C1 1 – 2 – 3 – 5 C2 1 – 2 – 4 – 5

Caso de Prueba para C1	
v	false
response	“los datos no se han eliminado correctamente”
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que el organismo no ha sido eliminado.

Caso de Prueba para C2	
v	true
response	“los datos se han eliminado correctamente”
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que el organismo ha sido eliminado.

3.2 Resultados Obtenidos

Como resultados obtenidos tenemos que con el resultado del trabajo de diploma queda disponible en su versión 1.0 el “Módulo para la Dirección de Inspección de la

CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA

Administración Provincial de Artemisa”, se logró obtener un sistema que cumple con todos los requerimientos especificados en el levantamiento de requisitos y le brinda la posibilidad a los usuarios de acceder a todas sus funcionalidades.

3.3 Funcionalidades Obtenidas

Entre las funcionalidades que posee el sistema informativo para la dirección de inspección se encuentran:

Nota # 1: La palabra gestionar lleva explícita las funcionalidades: insertar, eliminar, modificar y buscar.

- Gestionar información de las actividades del registro para la Tabla 1.
- Gestionar información de los indicadores del registro para la tabla 2 con decreto 274 y resolución 4272.
- Gestionar información de los indicadores del modelo de registro para la tabla 2.
- Gestionar información del control de las inspecciones internas del registro.
- Gestionar información del control de inspecciones con decreto 274 del registro.
- Gestionar información del control de la situación del personal de las DIM-CAM y DIP-CAP del modelo de registro.
- Generar reportes sobre las actividades del modelo de registro para la tabla 1.
- Generar reportes sobre los indicadores del registro para la tabla 2 con decreto 274 y resolución 4272.
- Generar reportes sobre los indicadores del registro para la Tabla 2.
- Generar reportes sobre el control de las inspecciones internas del registro.
- Generar reportes sobre el control de inspecciones con decreto 274 del registro.
- Generar reportes sobre el control de la situación del personal de las DIM-CAM y DIP-CAP del modelo de registro.

CAPITULO 3: ADQUISICIÓN Y VALIDACION DE LOS RESULTADOS DEL SISTEMA

3.4 Aporte social y económico.

El módulo para la dirección de inspección del Sistema Informativo de la Administración Provincial de Artemisa, le brindará a esta dirección una herramienta eficaz que se ajusta a sus necesidades. De igual manera le proporcionará una mejor efectividad en el manejo del flujo y procesamiento de datos que en ella se gestionan. Además el sistema fue desarrollado con herramientas libres, lo cual aporta grandes beneficios económicos no solo al Sistema Informativo de la Administración Provincial de Artemisa sino al país en general.

3.5 Conclusiones Parciales

En el presente capítulo se definieron las pruebas unitarias del módulo para la dirección de inspección del Sistema Informativo de la Administración Provincial de Artemisa donde se obtuvo en cada una de ellas resultados satisfactorios, además se plasmaron los resultados obtenidos, y el aporte social y económico que brinda el módulo para la dirección de inspección.

CONCLUSIONES GENERALES

Conclusiones Generales

Se realizó un estudio sobre los sistemas de gestión existentes en el mundo y en Cuba, se ha analizado su funcionamiento y características fundamentales. Se implementó y probó la solución a los requerimientos de las aplicaciones clientes para la gestión de la información de la dirección de Inspección de la AP de Artemisa. Se realizó la validación mediante pruebas funcionales de los resultados obtenidos con la solución siendo estos satisfactorios. Por todo lo anteriormente expuesto se concluye que los objetivos trazados para el presente trabajo se han cumplido satisfactoriamente.

RECOMENDACIONES

Recomendaciones

Con el desarrollo del módulo para la Dirección de Inspección de la Administración Provincial de Artemisa se recomienda:

- Ampliar las funcionalidades del módulo según las necesidades del cliente.
- Tener en cuenta este trabajo para proyectos futuros, siempre y cuando se cumpla con las normas de confidencialidad establecidas.

REFERENCIAS BIBLIOGRAFICAS

Referencias Bibliográficas

- Ing. Nancy Oña Aldama, Idania Hernández Oramas. Sistema de inspección de la planta de reactivos clínicos Última actualización: diciembre 29, 1999. Disponible en: http://bvs.sld.cu/revistas/far/vol34_2_00/far05200.htm.
- Sistema de Gestión Informativa de Vehículos (VIMS) Última actualización: 2011, 1997. Disponible en: http://www.es.sgs.com/es/vehicle_information_management_system_vims?serviceld=13389&lobld=14933.
- Gustavo Rebolledo Saavedra, "Gestión". Última actualización: March 22, 1993. Disponible en: <http://b3.bibliotecologia.cl/ar-gestion.htm>.
- Conceptodegestión- Definición, Significado y Qué es. Última actualización: 2012. (Definición.de). Disponible en: <http://definicion.de/gestion/>.
- Bass, L., Clements, P., y Kazman, R. Software Architecture in Practice. Addison Wesley. Última actualización: 1998.
- AUGIER ESCALONA, ALEJANDRO. Gestión de la Información. Última actualización: febrero 13, 2012. Disponible en: http://www.ecured.cu/index.php/Gesti%C3%B3n_de_la_Informaci%C3%B3n
- SEGUROS. novamens Última actualización: 2010. Disponible en: <http://www.novamens.com/novamens/esp/compania/casos-de-estudio/casos/iones.html?uri=urn:kbee:54afb980-a434-11dd-9a48004854883d6f&page-uri=urn:kbee:517c64c0-a772-11dd-b6bd-004854883d6f>.
- ROMERO PEÑALVER. Metodología Ágil de Desarrollo SXP - EcuRed Última actualización: 2008. Disponible en: http://www.ecured.cu/index.php/Metodologia_Agil_de_Desarrollo_SXP.
- Juan Pablo Gómez Gallego. Universidad Tecnológica de Pereira.

REFERENCIAS BIBLIOGRAFICAS

- Fundamentos de La Metodología RUP. Última actualización: septiembre 16, 2007. Disponible en: <http://es.scribd.com/doc/297224/RUP>.
- García Santillán Arturo, Landa Víctor Alberto. Diseño e implementación de un sistema del Programa operativo anual SISPOA. Última actualización: 2000. Disponible en: <http://www.eumed.net/libros/2007c/332/Confabilidad.html>
 - Concepto de información - Definición, Significado y Qué es. Última actualización: 2008-2012. Disponible en: <http://definicion.de/informacion/>.
 - Definición de Sistema Operativo. Última actualización: 1999, 2012. Disponible en: <http://www.masadelante.com/faqs/sistema-operativo>.
 - Ingenieriasoftware dos - Grupo RUP. Última actualización: 2010. Disponible en: <http://ingenieriasoftware.dos.wikispaces.com/Grupo+RUP>.
 - NetBeans IDE 7.1 - Un entorno de código abierto de desarrollo integrado de los desarrolladores de software. Última actualización: 9 de enero 2012. Disponible en: <http://www.filecluster.es/programas/NetBeans-IDE-144777.html>.
 - PostgreSQL 8.4 » VivaLinux!. Última actualización: 2009. Disponible en: <http://www.vivalinux.com.ar/software/postgresql-8.4>.
 - ¿Qué son los sistemas de gestión?. Última actualización: 2012. Disponible en: <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion>.
 - SCRUM. Última actualización: febrero 4, 2007. Disponible en: <http://www.chuidiang.com/ood/metodologia/scrum.php>.
 - Omitsis Consulting. SCRUM como metodología de desarrollo. Última actualización: marzo 7, 2010. Disponible en: <http://www.omitsis.com/scrum-como-metodologia-de-desarrollo>.
 - Sistema Gestor de base de datos SGBD. Última actualización: noviembre 1, 2004. Disponible en: http://www.error500.net/garbagecollector/archives/categorias/bases_de_datos/sistema_gestor_de_base_de_datos_sgbd.php.
 - Ubuntu 11.04: Información, características, novedades y descarga | Qportatil.

REFERENCIAS BIBLIOGRAFICAS

- Última actualización: junio 13, 2011. Disponible en: <http://www.qportatil.com/2011/06/ubuntu-11-04-informacion-caracteristicas-novedades-y-descarga/>.
- J. G EMERY. Sistemas de Planeamiento y Control en la Empresa. Buenos Aires: El Ateneo. Última actualización: 1977.
 - CASTILLA PLAZA, C. Implicaciones de las tecnologías de la información en la gestión del sistema empresa. Tutor: Dr. Santo Domingo Garachana, A. Departamento de Economía Financiera y Contabilidad I. Universidad Complutense de Madrid. Última actualización: 1999.
 - Phil Bartle. Información para la gestión y gestión de la información. Última actualización: 5 de febrero de 2009. Disponible en: <http://www.scn.org/mpfc/modules/mon-miss.htm>
 - Metodología Extreme Programming. Última actualización: 9 de Febrero de 2009. Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_softwa_re_07062004.
 - Diego Gómez. Lo Mejor de XP. Última actualización: 10 de Febrero de 2009. Disponible en: www.dosideas.com/metodologias/546-lo-mejor-de-xp.html.
 - Ferré Grau, Xavier & Sánchez Segura, María Isabel. Desarrollo Orientado a Objetos con UML. Última actualización: 23 de febrero 2009. Disponible en: <http://www.clikear.com/manuales/uml/index.aspx>.
 - Características de Java. Última actualización: febrero 2007. Disponible en: http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/I_3.htm.
 - R Martín, D Enríquez, H Viltres. Trabajo Investigativo sobre Sistemas de Gestión de Contenidos CMS Drupal. Universidad de las Ciencias Informáticas, Ciudad de la Habana. Última actualización 2006.
 - Francois Zaninotto Potencier, Fabien. librosweb.es. Symfony, la guía definitiva. Última actualización: 24 de Enero de 2009. Disponible en: http://www.librosweb.es/symfony_1_2/. ISBN-13: 978-1590597866.
 - Lois Marx Peña, Achang González, José Rafael Viera. LIMS de Calidad del

REFERENCIAS BIBLIOGRAFICAS

Centro de Ingeniería Genética y Biotecnología: Implementación del Módulo de Análisis Químico. Ciudad de la Habana : s.n. Última actualización: 2008.

- Oktaba, Hanna. Unam Posgrado Ciencia e Ingeniería de la Computación. Última actualización: 25 de abril 2009. Disponible en: <http://www.mcc.unam.mx/~cursos/Algoritmos/javaDC99-2/patrones.html>.
- Herramientas Case. Última actualización: 2006. Disponible en: <http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5.ht>.

BIBLIOGRAFIA

Bibliografía

- ALEJANDRO A. PAVEZ SALAZAR. LA GESTION DEL CONOCIMIENTO EN LAS ORGANIZACIONES. Última actualización: 2000. Disponible en: <http://www.gestiondelconocimiento.com/documentos2/apavez/gdc.htm>.
- ARMANDO DUANY DANGEL. Gestión del Conocimiento y la Información. Última actualización: 2001. Disponible en: <http://www.econlink.com.ar/gestion-conocimiento/cuba>.
- M.C. GABINO ESTEVEZ DELGADO ING. EDUARDO OCHOA HERNÁNDEZ. Gestión de la información. Última actualización: Septiembre 2001. Disponible en: <http://dieumsnh.qfb.umich.mx/gesinfo/>.
- Definición de Java - Qué es, Significado y Concepto. Última actualización: Disponible en: <http://definicion.de/java/>.

Anexos

Anexo #1. Guía de Entrevista.

Preguntas realizadas al personal de la Dirección de Inspección:

- ¿Cuáles son los grupos de usuarios o roles involucrados?
- ¿Todos los trabajadores acceden a la misma información?
- ¿Cómo tienen organizado los niveles de acceso, si es por departamento, por categoría de la información?

ANEXOS

Anexo #4. Aval de Calidad de Software



Centro de Desarrollo

Aval de Calidad de Software


El grupo de Calidad de Software del Centro de Desarrollo de la Facultad Regional "Mártires de Artemisa" conformado por:

- **Asesora de Calidad:** Ing. Maidel Ojeda Cruz
- **Asesor de Tecnología:** Ing. Domma Moreno Dager
- **Especialista de Calidad:** Ing. Yenisleydi Rodríguez Martínez


emite el presente **Aval de Calidad de Software** en colaboración con los especialistas del Centro de Desarrollo a: Daymí Duque Carrodegua, como resultado satisfactorio de su desempeño en la tareas asociadas al proyecto: Sistema Informativo de la Administración Provincial de Artemisa.

Para emitir el presente aval se valoraron un conjunto de elementos evaluados de manera individual teniendo en cuenta los parámetros de calidad de software del proyecto. A continuación se presenta los resultados en cada uno de los aspectos valorados:

Elementos evaluados	Resultado
Estandarización del código fuente del proyecto	Satisfactorio
Limpieza, organización y estructuración del código fuente	Satisfactorio
Funcionalidad e integración del sistema	Satisfactorio
Validación y seguridad de la información gestionada	Satisfactorio
Generación de todos los artefactos de la metodología SXP	Satisfactorio
Cumplimiento de las plantillas establecidas para cada artefacto	Satisfactorio
Ortografía, concordancia y redacción de cada uno de los artefactos	Satisfactorio
Modelación de diagramas con el uso de UML	Satisfactorio
Trazabilidad de los requisitos funcionales y no funcionales	Satisfactorio
Efectividad de los casos de pruebas definidos en el proyecto	Satisfactorio


Ing. Maidel Ojeda Cruz
Asesora de Calidad de SW




Msc. Yamila Vigil Regalado
Directora Centro de Desarrollo