



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD REGIONAL DE MÁRTIRES DE ARTEMISA

Trabajo de Diploma para Optar por el Título de

Ingeniero en Ciencias Informáticas

Módulo Multas y Patrimonio de la Dirección de Finanzas y
Precios de la Administración Provincial de Artemisa.

Autor:

Reddy García Collazo

Tutor: Ing. Dania Fernández Aguilar

Artemisa, 2012



“El futuro de nuestra patria tiene que ser necesariamente un futuro de hombres de ciencia, tiene que ser un futuro de hombres de pensamiento, porque precisamente es lo que más estamos sembrando; lo que más estamos sembrando son oportunidades a la inteligencia...”

Módulo Multa y Patrimonio

Declaración de Auditoría

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

Para que así conste firmamos la presente a los ____ días del mes de _____ del Año _____

Tutor: Ing. Dania Fernández Aguilar

Autor: Reddy García Collazo

Módulo Multa y Patrimonio

Datos de Contacto

Tutor: Nombre y Apellidos: Ing. Dania Fernández Aguilar.

Sexo: F. Institución: UCI.

Correo electrónico: daguilar@hab.uci.cu

Teléfono del trabajo: 365619

Título de la especialidad de graduado: Ingeniero en Ciencias Informáticas.

Año de graduación: 2009

Institución donde se graduó: Universidad de las Ciencias Informáticas.

Módulo Multa y Patrimonio

AGRADECIMIENTOS

Reddy García Collazo

Agradecer a la Revolución por darme la oportunidad de estudiar en esta Universidad, a mi familia que me apoyo en todo momento. A mi novia, por estar siempre a mi lado y darme ese aliento positivo que uno necesita. También le agradezco Armando Simón, Neybis Lago Clara, Dariel Hernández y Víctor Barzana por su ayuda incondicional en los momentos que la necesité. A mis amigos de la escuela especialmente los de la Brigada 101 por compartir conmigo momentos de tristeza y alegrías, a los profesores que de una forma u otra me ayudaron en mi formación y a lograr mis metas especialmente a José Carlos Martínez y los Vicedecanos José Ramón y Juana Elena que fui como un hijo para ellos. Agradecer a mi compañera de tesis Jennifel López Álvarez por las largas madrugadas de trabajos juntos. Ya mi tutora Dania Fernández Aguilar por brindarnos su apoyo y ayuda para lograr los resultados alcanzados con este trabajo. En fin agradecerles a todas las personas maravillosas que conocí en mi carrera universitaria.

Gracias o todos de verdad

Módulo Multa y Patrimonio

DEDICATORIA

Reddy García Collazo

El presente trabajo de diploma ha sido resultado de varios años de estudio, esfuerzo y sacrificio y con mucho amor se lo dedico a mi familia que es un poco extensa, especialmente a:

A mi madre por darme su amor y cariño, por ser de mí la persona que soy, por apoyarme en todo momento sin esperar nada a cambio, por sus buenos consejos siempre oportunos, por sacrificarse y dar todo para que en un futuro fuese un profesional, por sentirse orgullosa de verme cumplir mis metas. Te quiero mucho.

A mi padre porque a pesar de sus problemas siempre me dio su amor, siempre me ha apoyado en todo, se ha sacrificado para darme todo lo que necesito para cumplir mis metas y debe estar orgulloso de verme cumplirlas. Te quiero mucho.

A mi abuela querida por darme su amor y cariño, por apoyarme siempre en esos momentos difíciles que tuve, por tenerme presente en lo mas profundo de su corazón y se que se sentirá orgullosa de mi al verme hecho un ingeniero. Por ser ella para mi una persona muy especial. Te quiero mucho mucho y que dures muchos años más.

A mi hermano menor por darme su mano y apoyo en momentos difíciles de mi carrera. A los demás familiares que también quiero muchísimo y que lo han dado todo para apoyarme, por sus buenos consejos y doy gracias por tener esta hermosa familia.

Módulo Multa y Patrimonio

Resumen

El objetivo principal de este trabajo es mejorar los procesos que se llevan a cabo en los departamentos de Multa y Patrimonio y darle solución mediante el desarrollo de un módulo que contribuya a la gestión de la información de manera que sea más confiable y eficiente. Para esto se utilizó la metodología SXP apoyado en el Lenguaje de Modelado Unificado (UML). Fue desarrollado mediante el IDE NetBeans, Java como lenguaje de programación y JWebSocket como Frameworks, obteniéndose como resultado el Módulo Multa y Patrimonio como parte del Sistema Informativo de la Administración Provincial de Artemisa mejorando la gestión de la información en los departamentos de Multa y Patrimonio de esta entidad.

Palabras claves: Módulo, Sistema, Gestión.

Módulo Multa y Patrimonio

Introducción	1
Capítulo 1: Fundamentación Teórica	7
1.1 Marco Conceptual	7
1.2 Tendencias Actuales	10
1.2.1 Ámbito Internacional	10
1.2.2 Ámbito Nacional	12
1.2.3 En la UCI	12
1.3 Metodología de desarrollo de software	13
1.3.1 Extreme Programming (XP)	14
1.3.2 Scrum	14
1.3.3 SXP	15
1.4 Herramientas a utilizar.	15
1.4.1 Lenguaje de Modelado Unificado (UML)	16
1.4.2 Herramienta CASE	16
1.4.3 Entorno de Desarrollo Integrado.....	17
1.4.4 Herramientas de control de versiones.....	18
1.4.5 Maven	19
1.4.6 Pruebas unitarias.....	20
1.5 Tecnologías utilizadas	21
1.5.1 Lenguaje de programación	21
1.5.2 Frameworks.....	22
1.5.3 Sistema Operativo	23
Conclusiones	26
Capítulo 2. Características, Análisis y Diseño del Sistema.	27
2.1 Solución Propuesta.	27
2.2 Información a automatizar.	27
2.4 Modelo de Dominio	29
2.5 Lista de Reserva del Producto (LRP)	30
2.6 Historias de Usuarios y Tareas de Ingeniería	36
2.7 Plan de Release	46
2.8 Diseño con Metáfora.	46
2.9 Diagrama de Componentes.	47
Conclusiones	49
CAPÍTULO 3: Implementación y Validación del Sistema	50
3.1 Estándar de Código	50
3.2 Distribución de la Implementación.	50

Módulo Multa y Patrimonio

3.3 Pruebas Unitarias con JUnit.....	52
3.4 Resultados Obtenidos.....	55
3.5 Funcionalidades Obtenidas.....	55
3.6 Aporte Social y Económico.	55
Conclusiones.....	56
CONCLUSIONES.....	57
RECOMENDACIONES.....	58
REFERENCIA BIBLIOGRÁFICA.....	59
BIBLIOGRAFÍA.....	64
GLOSARIO DE TÉRMINOS.....	<i>¡Error! Marcador no definido.</i>
Anexos.....	<i>¡Error! Marcador no definido.</i>

Módulo Multa y Patrimonio

Introducción

Con el surgimiento de las Tecnologías de la Información y las Comunicaciones el hombre ha intentado automatizar los procesos de trabajo para mejorar la calidad y productividad de sus actividades. El siglo XXI ha sido denominado como el momento de la Tecnología en la Información, alcanzando niveles de desarrollo impresionantes.

Los cambios de la modernidad en el orden político, económico, cultural y tecnológico han originado que la gestión de la información se desarrolle, por tanto se hace necesaria la creación de aplicaciones capaces de almacenar grandes cantidades de información y procesar enormes volúmenes de datos, llamados sistemas gestores de la información que ayudan a las organizaciones a llegar a sus objetivos mediante estrategias, que incluyen la optimización de procesos y el pensamiento disciplinado.

Su implementación puede mejorar la efectividad operativa, reducir costos, aumentar la satisfacción de clientes y partes interesadas, lograr mejoras continuas y potenciar la innovación. Los sistemas gestores de información son más que un subsistema informático ya que puede definirse como un conjunto de componentes interrelacionados que permiten capturar, procesar y almacenar información para apoyar la toma de decisiones y llevar a cabo el control de una institución.

En Cuba a partir de 1959, comienza el desarrollo de la Informática y las Comunicaciones con los programas de formación del personal informático. Por tales motivos surgieron por iniciativa del presidente cubano Fidel Castro Ruz instituciones como la Universidad de las Ciencias Informáticas (UCI), que inició su funcionamiento en el curso 2002 – 2003.

En el año 2007 se crea la Facultad Regional (UCI) de Artemisa una de las más jóvenes instituciones de este tipo en el país, tiene como propósito fundamental la informatización de la sociedad a través de distintas aplicaciones entre las que se encuentran los sistemas gestores los cuales han venido a

resolver los problemas en todos los ámbitos tanto políticos, sociales, culturales y económicos.

Situación problemática: Por la gran cantidad de municipios que contenía la provincia de La Habana, en el 2011 se decide dividir en dos provincias, Mayabeque y Artemisa. En esta última en el municipio cabecera se encuentra ubicado el Gobierno Provincial, el cual consta de 32 direcciones, las que se deciden informatizar, una de ellas es precisamente una de las líneas de gran importancia para el desarrollo económico del país, la dirección de Finanzas y Precios, la misma cuenta con 5 departamentos, entre ellos se encuentran los Departamento de Multas y Patrimonio.

El Departamento de Multas maneja un conjunto de información referente a las gestiones que se realizan a nivel provincial en cuanto al tema verificar, controlar y evaluar el trabajo de las oficinas de cobro y control de multas y garantizar el cumplimiento de los indicadores de eficacia en el cobro de las multas. En el Departamento de Patrimonio se maneja un conjunto de información referente al control de inmuebles estatales en la provincia.

La información, se gestiona de forma manual, por vía telefónica, por correo electrónico o mediante mensajeros que entregan la información de las entidades que se encuentran en los municipios de la provincia, en formato duro o digital.

Esta entidad no cuenta con una adecuada forma de gestionar la información por lo que la realización de este proceso provoca pérdida y duplicado de la misma, debido a que se necesita mucho tiempo para manejarla, igualmente cuando se buscan datos de forma manual dentro de grandes volúmenes de información, trayendo consigo problemas de eficiencia. Todo esto a su vez ocasiona problemas de confiabilidad, posibilitando que en ocasiones no se tenga constancia real de la información o que la misma no sea la correcta.

Todas estas dificultades ocasionan problemáticas para la generación de reportes inmediatos, así como su entrega al Presidente de la Administración

Provincial, en el tiempo determinado y con la calidad requerida. La entrega incorrecta y tardía de esta información afecta la toma de decisiones de la Provincia Artemisa.

A partir de lo antes expuesto y para dar solución a la necesidad existente, se plantea el siguiente **problema científico**: ¿Cómo mejorar el proceso de gestión de la información en el Departamento de Multas y Patrimonio que eleve la eficiencia y confiabilidad de la información generada en la Dirección de Finanzas y Precios de la Administración Provincial de Artemisa?

Se define como **objeto de estudio**: el proceso de gestión de la información. Donde el **campo de acción** está enmarcado en el proceso de gestión de la información en las esferas de Multas y Patrimonio.

Idea a defender: La implementación del Módulo Multas y Patrimonio contribuye a elevar la eficiencia y confiabilidad en el proceso de gestión de la información en la Dirección de Finanzas y Precios de la Administración Provincial de Artemisa.

Se plantea como **objetivo general**: Desarrollar el Módulo Multas y Patrimonio de la Dirección de Finanzas y Precios que contribuya a elevar la eficiencia y confiabilidad en el proceso de gestión de la información en la Administración Provincial de Artemisa.

Objetivos específicos:

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

Para dar cumplimiento a los objetivos generales se plantean las siguientes

Tareas de la Investigación:

1. Establecimiento de los fundamentos teórico-metodológicos para el desarrollo de los procesos de gestión de información.
2. Caracterización del proceso de gestión de la información en los Departamentos Multas y Patrimonio.
3. Desarrollo del módulo para los Departamentos Multas y Patrimonio.
4. Validación de la propuesta de solución.

Con lo que obtendremos los siguientes **Aporte Práctico:**

1. Módulo Multas y Patrimonio de la Dirección de Finanzas y Precios que contribuya a elevar la eficiencia y confiabilidad en el proceso de gestión de la información en la Administración Provincial de Artemisa.

Para el desarrollo del trabajo se utilizaron los siguientes **métodos científicos:**

Métodos Teóricos

◆ **Análisis – Síntesis:** Se realizó un estudio con profundidad de la información acerca de las tecnologías, metodologías y herramientas posibles a ser utilizadas en el desarrollo del Sistema de Gestión propuesto, pudiendo definir con mayor certeza las mismas, sintetizando sus características y analizando la viabilidad de cada una.

◆ **Histórico - Lógico:** Para poder realizar este trabajo primeramente se analizó lo referente al surgimiento de la informática. Se realizó un análisis de la trascendencia de los Sistemas de Gestión en el mundo y en Cuba, más específicamente en la Universidad de las Ciencias Informáticas.

◆ **Inducción – Deducción:** Este método se aplicó para seleccionar el lenguaje de programación que se escogió, pues son las formas de razonamiento que permiten llegar a un grupo de conocimientos

generalizadores, tanto desde el análisis de lo particular a lo general, como desde el análisis de elementos generalizadores a uno de menor nivel de generalización.

◆ **Modelación:** Se realizaron los modelos correspondientes al ciclo de vida del desarrollo del Sistema de Gestión propuesto, esto propició facilidades para cumplir con las tareas de análisis y diseño de los procesos que intervinieron en la aplicación, así como para la implementación del sistema.

Métodos Empíricos

◆ **Entrevista:** Se utilizaron para diagnosticar y detectar los problemas actuales en los Departamentos de Multas y Patrimonio de la Dirección de Finanzas y Precios de la Administración Provincial de Artemisa.

Como **población** se consideró a los 2 jefes de los Departamentos de Multas y Patrimonio de la Dirección de Finanzas y Precios de la Administración Provincial de Artemisa. La **muestra**, seleccionada de modo intencional, representa el 100% de la población.

El desarrollo de este documento se encuentra compuesto por tres capítulos donde se refleja todo el trabajo investigativo, el diseño del sistema y la implementación de la solución propuesta, distribuido de la siguiente manera:

El **Capítulo I** Fundamentación Teórica: se hace un análisis del estado del arte del objeto de estudio, se investiga acerca de los sistemas informáticos vinculados al campo de acción, se fundamentan las metodologías, tecnologías y herramientas utilizadas para el desarrollo del sistema de gestión.

En el **Capítulo II** Características, Análisis y Diseño del Sistema: se define el negocio y se describe la solución propuesta para la situación problemática. Se presentan las características y funcionalidades del sistema a partir de los requisitos funcionales y no funcionales capturados. Se realiza el análisis y diseño del sistema.

En el **Capítulo III** Implementación y validación del Sistema: incluye la programación realizada a partir de los requerimientos y pruebas utilizadas para la validación de la misma.

Módulo Multa y Patrimonio

Capítulo 1: Fundamentación Teórica

En el desarrollo de este capítulo se realiza un estudio a nivel mundial y en nuestro país de aplicaciones web que manejan todo lo referente a la gestión de información y una valoración de las posibles herramientas, tecnologías y metodología posibles a utilizar para la realización del análisis, diseño e implementación del problema planteado, de forma tal que quede especificado el marco teórico de la investigación.

1.1 Marco Conceptual

Información

La información es considerada, actualmente, un recurso que se encuentra al mismo nivel que los recursos financieros, materiales y humanos, que hasta el momento habían constituido los ejes sobre los que había girado la gestión empresarial. Si la teoría económica tradicional mantenía al capital, la tierra y el trabajo como elementos primarios de estudio, la información se ha convertido, ahora, en el cuarto recurso a gestionar. (PALMERO BERBERENA, 2011)

Es un conjunto de datos con un significado, o sea, que reduce la incertidumbre o que aumenta el conocimiento de algo. En verdad, la información es un mensaje con significado en un determinado contexto, disponible para uso inmediato y que proporciona orientación a las acciones por el hecho de reducir el margen de incertidumbre con respecto a nuestras decisiones. (CHIAVENATO, 2006)

El autor se acoge a lo planteado por Idalberto Chiavenato por ser la más apropiada para la presente investigación.

Gestión

Se referirá a la acción y al efecto de administrar o gestionar un negocio. En la gestión se realizan un conjunto de trámites que se llevan a cabo para resolver un asunto o concretar un proyecto. También y a la par de esto, en una gestión habrá que dirigir, gobernar, disponer, organizar y ordenar para lograr los objetivos propuestos. De lo dicho se desprende que la gestión es una tarea que requiere de mucha conciencia, esfuerzo, recursos y buena voluntad para ser

llevada a cabo no satisfactoriamente. (SAVIGNE, 2011)

Una gestión, entonces, podrá estar orientada a resolver un problema específico, a concretar un proyecto, un deseo, pero también puede referir a la dirección y administración que se realiza en una empresa, una organización, un negocio, e incluso a nivel de gobierno. (LEVISY, 2011)

El autor adopta para la investigación lo expresado por Dorroman Savigne y Leyva Velázquez en el artículo, "El proceso de gestión y la gestión económica en las empresas".

Sistema

Un sistema es un conjunto de elementos relacionados. Puede estructurarse de conceptos, objetos y sujetos, se compone de otros sistemas a los que llámanos subsistemas. (OSCAR, 2011)

Es una unión de partes o componentes, conectados en una forma organizada. Las partes se afectan por estar en el sistema y se cambian si lo dejan. La unión de partes hace algo. (VAN GIGCH, 2011)

Para la presente investigación el autor adopta el concepto del autor John P. Van Gigch, ya que es el más apropiado para el tema.

Sistema de Gestión

Es un conjunto de etapas unidas en un proceso continuo, que permite trabajar ordenadamente hasta lograr mejoras. Establece cuatro etapas que hacen de este sistema un proceso circular, a medida que se repite el ciclo se logra obtener una mejora.

Las cuatro etapas del sistema de gestión son:

Etapa de Ideación

Etapa de Planeación

Etapa de Implementación

Etapa de Control (VERGARA, 2009)

Es la gestión (planeamiento, organización, operaciones y control) de los recursos (humanos y físicos) que tienen que ver con el apoyo a sistemas (desarrollo, mejoría y mantenimiento) y servicios (procesamiento,

transformación, distribución, almacenamiento y recuperación) de la información (datos, textos, voz e imagen) para una empresa. (ARCEO, 2008) El autor de esta investigación asume que el concepto más apropiado para sistema de gestión es el planteado por G. Vergara.

Sistema de gestión de la información

Es el proceso que se encarga de suministrar los recursos necesarios para la toma de decisiones, así como para mejorar los procesos, productos y servicios de la organización, y cobra especial importancia en las empresas con la aparición de la informática. (BERBERENA, 2011)

Son el conjunto de componentes interrelacionados que recogen, procesan, almacenan y distribuyen la información para dar soporte a la toma de decisiones y al control de la organización. Además del apoyo a la toma de decisiones, coordinación y control, los sistemas de información pueden ayudar a los gestores a analizar problemas y a desarrollar nuevos productos o servicios. (LAUDON, 2007).

El autor de la presente investigación coincide con lo planteado por R. Berberena acerca de que es Sistema Gestión de la Información.

Módulo

Es una pieza o un conjunto unitario de piezas fáciles de ensamblar. Por otra parte, el producto final o sistema puede ser reparado si se repara el módulo o componente que no funciona. Se conoce como modularidad a la capacidad de un sistema para ser entendido como la unión de varios componentes que interactúan entre sí, cada uno cumple con una tarea en pos de un objetivo común. (GÓMEZ, 2010)

Es un componente auto controlado de un sistema, el cual posee una interfaz bien definida hacia otros componentes; algo es modular si es construido de manera tal que facilite su ensamblaje, acomodamiento flexible y reparación de sus componentes. (LASTRE, 2008)

El autor se acoge a lo planteado por M. A. Gómez por ser el concepto de módulo más apropiado para la presente investigación.

Confiabilidad

Capacidad de un elemento de desempeñar una función requerida, en condiciones establecidas. (ELLMANN, 2008)

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

Se coincide con lo expresado por el autor Enrique Ellmann acerca de que es confiabilidad.

Eficiencia

Implica operar de una determinada forma en la cual todos los recursos se utilicen de la manera más adecuada posible. (DA. SILVA, 2000)

Es la capacidad de lograr los objetivos y metas programadas con los recursos disponibles en un tiempo predeterminado. Capacidad para cumplir en el lugar, tiempo, calidad y cantidad las metas y objetivos establecidos. (CHIAVENATO, 2006)

El autor coincide con lo expresado por Idalberto Chiavenato acerca de que es eficiencia.

1.2 Tendencias Actuales

Hoy por hoy se pueden encontrar diversos sistemas informáticos que han incluido soluciones que gestionan la información en cuanto al control del patrimonio y el cobro de las multas. A continuación se mostrarán ejemplos:

1.2.1 Ámbito Internacional

InmoComputer: Gestión de Ventas Inmobiliarias es un software para inmobiliarias con el que podrá controlar todos sus inmuebles, clientes y propietarios. Es un software de gestión muy intuitivo y de fácil manejo. Los programas para inmobiliarias le permitirán realizar un seguimiento exhaustivo sobre cada factor que interviene en su negocio inmobiliario, para realizar satisfactoriamente una óptima gestión y administración de este negocio.

Mediante el uso del software de gestión de inmobiliarias podrá trabajar fácilmente tanto en ventas como en alquileres. El software para inmobiliarias representa una herramienta muy completa que, además de poder ofrecerle una

gestión básica de una inmobiliaria, le ofrece aplicaciones avanzadas, para que usted pueda tener cubierto en todos los aspectos su gestión. **El sistema ERP Solmicro-eXpertis Gestión de Inmuebles:** Además de todas las funcionalidades de gestión empresarial (comercial, logística y financiero) cuenta con módulos específicamente desarrollados para empresas con Gestión de Inmuebles. Permitiendo realizar tareas, actividades, obras y proyectos por inmueble (construcción, reformas, mantenimiento). Alquiler, compra y venta de inmuebles, contratos y facturación. Presupuesto por inmueble y patrimonio, movimientos entre almacenes, inventarios, regularizaciones, Control de activos, elementos, números de serie y ubicaciones, Información asociada a cada activo.

Sistema de Gestión de Expedientes de Multas de Tráfico (MULTAPOL): es el programa ideal para la Gestión de Multas de Tráfico, contiene entre otras estas características:

1. Gestión y configuración del fichero de Agentes denunciantes.
2. Ordenanza de Tráfico de la ciudad
3. Libro registro de Entradas y Salidas de documentos.
4. Sistema de SMS integrado,
5. Sistema de Fax integrado.
6. Fichero con todos los Códigos Postales de España.
7. Integrado con la Web de Correos para la impresión automática de certificados.
8. Sistema de importación de denuncias desde otras aplicaciones para su tramitación a través de MULTAPOL.
9. Word integrado para notificaciones.
10. Posibilidad de configurar los modelos de notificaciones.
11. Posibilidad de definir los plazos para las distintas notificaciones.
12. Sistema Automático de Notificaciones.
13. Emisión de Carta de Pago de multa para su presentación a Bancos o Cajas.

14. Códigos de barras integrados en las notificaciones.
15. Posibilidad de incluir la fotografía de la denuncia en las notificaciones.
16. Acuses de recibos de forma masiva.
17. Integrado con nuestra aplicación PolMobile para recogida de infracciones en la vía pública con PDAS.(TECNI-SOFT, 2010)

1.2.2 Ámbito Nacional

Sistema Automatizado para el Control y Cobro de Multas: El país anualmente por concepto de multas ingresa al presupuesto del estado más de 150 millones de pesos y de ellos 3 millones en CUC, por lo que la Actividad de Multas es una importante fuente de ingresos para el país. El sistema tiene como principal objetivo ahorrar al país en un año más de 60 000 CUC y 190 000 CUP. Facilitará la creación de una red nacional a fin de viabilizar su pago y control.

La aplicación, confeccionada por la División de Desarrollo de Software (DESOFT). Entre sus ventajas también se destaca el registro digital del pago de la sanción en cualquier municipio, independientemente de donde la impusieron. Consecuentemente, el sistema elimina el traslado actual de matrices y documentación entre oficinas, lo cual evita la posibilidad de extravíos o pérdidas, además de los gastos que origina ese trasiego.

Igualmente aporta herramientas de apoyo a la toma de decisiones, diseño de reportes avanzados a nivel municipal, provincial y nacional, así como la evaluación de las multas por organismos impositores. Este instrumento de informatización agiliza los trámites, contribuye a elevar la eficiencia y evita indisciplinas en el pago de las multas. (GARCÍA SANTOS, 2009)

1.2.3 En la UCI

Sistema de Gestión de Inmuebles (SIM): debido al gran número de inmuebles en las Universidades e Instituciones que juegan un papel vital para su funcionamiento, los mismos están localizados indistintamente en cualquier lugar, cada uno tiene sus características propias que lo identifican como entidad única, lo cual genera un apreciable caudal de información que resulta precisa almacenar para su uso posterior en la toma de decisiones.

El sistema tiene como objetivo el almacenamiento de dicha información, utiliza una pasarela de autenticación que permite que el personal responsable de trabajar con el sistema tenga un usuario y contraseña que será autorizado a acceder al sistema luego de la respectiva verificación contra el servidor de identificación de la Institución.

Los usuarios con permisos y encargados de gestionar toda la información de los inmuebles tendrán las opciones para organizar, distribuir, registrar y crear reportes basándose en criterios de selección, además de poder actualizar la información de cada inmueble en caso de ser necesario gracias a algoritmos optimizados que utiliza el sistema para acceder a la información almacenada en la base de datos del sistema. La implementación del producto se apoya en un sistema modular, programable y extensible y con opciones ágiles.

Permite automatizar el proceso, apoyándose en una red con una base de datos accesible desde los distintos nodos de la misma involucrados en este. Mediante una interfaz web amigable, el personal responsable de trabajar con los datos del sistema podrá acceder a la base de datos y gestionar toda la información necesaria para la posterior toma de decisiones de manera rápida y segura. (PÉREZ ESCALONA; BERTOT VIEITO, 2009)

CedruX: basado en los principios de independencia tecnológica y en las particularidades de la economía cubana. Esta solución nacional incluye un subsistema para el control de los activos fijos tangibles (AFT), que pretende servir de apoyo a la toma de decisiones en las entidades.

El subsistema es capaz de gestionar activos de forma dinámica, tributa mediante operaciones al sistema contable de la empresa y estandariza el tratamiento de los AFT a nivel nacional. Incluye además temas como la multimoneda, la agrupación de activos y la generación de documentos, prestaciones del sistema que responden a necesidades del país permitiendo que el control de los AFT sea confiable, estable y legalmente correcto. (GONZÁLEZ ALVAREZ, 2009)

1.3 Metodología de desarrollo de software

No se puede hablar de un software de calidad sin tener en cuenta las

tendencias y tecnologías informáticas existentes a nivel mundial. Siendo para ello necesario hacer una correcta selección de las herramientas y metodología a utilizar para lograr que el producto cumpla con la mayor calidad los requisitos.

Metodología de Desarrollo de Software. Surge ante la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental a la hora de desarrollar un producto (software). Se clasifican en dos tipos. Las más utilizadas son: La Programación Extrema (XP), Scrum y el Proceso Unificado de Software (RUP).

1.3.1 Extreme Programming (XP)

La metodología XP consiste básicamente en ajustarse estrictamente a una serie de reglas que se centran en las necesidades del cliente para lograr un producto de buena calidad en poco tiempo. La Programación Extrema es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software. Promueve el trabajo en equipo, preocupándose en todo momento del aprendizaje de los desarrolladores y estableciendo un buen clima de trabajo.

Este tipo de método se basa en una realimentación continuada entre el cliente y el equipo de desarrollo con una comunicación fluida entre todos los participantes, también busca simplificar las soluciones implementadas y coraje para los múltiples cambios. Este tipo de programación es la adecuada para los proyectos con requisitos imprecisos, muy cambiantes y con un riesgo técnico excesivo. (ALLEGUI FERNÁNDEZ, 2011)

La metodología de XP se basa en:

1. Pruebas Unitarias
2. Re fabricación
3. Programación en pares

1.3.2 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. Scrum

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.(CORRAL, 2007)

Los roles principales en Scrum son el ScrumMaster, que mantiene los procesos y trabaja de forma similar al director de proyecto, el ProductOwner, que representa a los stakeholders (interesados externos o internos), y el Team que incluye a los desarrolladores.

1.3.3 SXP

SXP: Es un híbrido cubano de metodologías ágiles que tiene como base las metodologías SCRUM y XP que permiten actualizar los procesos de desarrollo de software para el mejoramiento de su producción. Consta de 4 fases: Planificación-Definición, Desarrollo, Entrega y Mantenimiento.

Esta metodología ayuda a fortalecer el trabajo en equipo, enfocados en una misma dirección, permitiendo además seguir de forma clara el avance de las tareas a realizar, a partir de la inserción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la producción, aumentando el nivel de interés del equipo.

Porque utilizar SXP, pues es una metodología 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 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.

SXP está especialmente indicada para proyectos de pequeños equipos de trabajo, rápido cambio de requisitos o requisitos imprecisos, muy cambiantes, donde existe un alto riesgo técnico y se orienta a una entrega rápida de resultados y una alta flexibilidad. Ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro, permitiendo además seguir de forma clara el avance de las tareas a realizar, de forma que los jefes pueden ver día a día cómo progresa el trabajo.(PEÑALVER ROMERO, 2008)

1.4 Herramientas a utilizar.

1.4.1 Lenguaje de Modelado Unificado (UML)

UML: Lenguaje Unificado de Modelado (LUM o UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML permite la creación de los diferentes modelos que ofrecen las vistas necesarias para la construcción de un software de calidad y permite la comprensión del sistema que se quiere realizar tanto por parte de los usuarios finales, como de los desarrolladores que implementarán la solución.

1.4.2 Herramienta CASE

Las herramientas CASE son aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software, reduciendo tiempo y dinero. Estas herramientas ayudan en el ciclo de vida de desarrollo del software en tareas como, el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática y documentación o detección de errores.

Rational Rose Data Modeler: Es una herramienta de modelado visual que posibilita que los diseñadores de bases de datos, analistas, arquitectos, desarrolladores y todos los demás miembros del equipo de desarrollo trabajen juntos, capturando y compartiendo los requerimientos de negocio y dándoles seguimiento a medida que cambian a través del proceso.

Visual Paradigm: Es una herramienta UML profesional que 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. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Además, la herramienta es colaborativa, es decir, soporta múltiples usuarios trabajando sobre el mismo proyecto y permite control de versiones. Cabe destacar igualmente su robustez, usabilidad y portabilidad. En definitiva, es una herramienta muy a tener en cuenta a la hora de ponerse manos a la obra con

un proyecto importante. (DELGADO PORRES; REYES MATOS, 2011)

¿Por qué utilizar Visual Paradigm?

Esta herramienta es estable en cuanto a su ejecución en diferentes sistemas operativos y la facilidad de abrir y trabajar con un modelo. Es una herramienta que guarda todo el modelo en un solo fichero por lo que basta con copiarse solo ese fichero para estar seguro de que tiene todo el trabajo encapsulado en él.

Además el equipo de desarrollo presenta conocimientos básicos de la herramienta lo que posibilita un mejor desempeño con la misma. Finalmente, Visual Paradigm es una herramienta que trabaja bastante decente en ordenadores poco potentes.

1.4.3 Entorno de Desarrollo Integrado.

Los IDEs (Integrated Development Environment) son un conjunto de herramientas para el programador, que suelen incluir en una misma suite, un buen editor de código, administrador de proyectos y archivos, enlace transparente a compiladores y integración con sistemas controladores de versiones o repositorios. (LUCIANO, 2010)

Eclipse: Es un entorno de desarrollo integrado, de Código Abierto y Multiplataforma. Mayoritariamente se utiliza para desarrollar lo que se conoce como “Aplicaciones de Cliente Enriquecido”, opuesto a las aplicaciones “Cliente Liviano” basadas en navegadores. Es una potente y completa plataforma de Programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java.

No es más que un entorno de desarrollo integrado (IDE) en el que encontrarás todas las herramientas y funciones necesarias para tu trabajo, recogidas además en una atractiva interfaz que lo hace fácil y agradable de usar.(GONZÁLES PINO, 2010)

NetBeans: Permite que las aplicaciones sean desarrolladas a partir de un conjunto de Componentes de software llamados módulos. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados

independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software. NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento.

¿Por qué utilizar NetBeans?

Cuenta con excelente soporte para aplicaciones web, es decir J2EE. Es una plataforma flexible. Permite que las aplicaciones sean desarrolladas a partir de un conjunto de módulos que pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

1.4.4 Herramientas de control de versiones

El control de versiones ayuda virtualmente en todos los aspectos al dirigir un proyecto, comunicación entre los desarrolladores, manejo de los lanzamientos, administración de fallos, estabilidad entre el código y los esfuerzos de desarrollo experimental y atribución y autorización en los cambios de los desarrolladores. El sistema de control de versiones permite a una fuerza coordinadora central abarcar todas estas áreas.

El núcleo del sistema es la gestión de cambios, identificar cada cambio a los ficheros del proyecto, anotar cada cambio con meta-data como la fecha y el autor de la modificación y disponer esta información para quien sea y como sea. Es un mecanismo de comunicación donde el cambio es la unidad básica de información. (FOGEL, 2007)

CVS: Conocido como Concurrent Versioning System, es una aplicación informática que implementa un sistema de control de versiones que mantiene el registro de todo el trabajo y los cambios en los ficheros de un proyecto y permite que distintos desarrolladores colaboren.

CVS se ha hecho popular en el mundo del software libre pero tiene algunas desventajas como no ofrecen facilidad para hacer referencia a cambios en múltiples ficheros, no permite renombrar o copiar ficheros dentro del sistema de control, no trabaja muy bien con ficheros muy grandes o con ficheros binarios y algunas operaciones son lentas cuando muchos ficheros están involucrados.

SVN: Es un sistema de control de versiones que se ha popularizado bastante, en especial dentro de la comunidad de desarrolladores de software libre. Está preparado para funcionar en red, y se distribuye bajo una licencia libre de tipo Apache. Permite gestionar proyectos editados de maneras concurrentes por muchos usuarios, resolviendo conflicto y controlando las versiones. Surge con la intención de sustituir y mejorar al conocido CVS.

RapidSVN: Es un cliente gráfico para Subversion, un programa de control de versiones sustituto de CVS. Es fácil de usar, tanto por quienes ya conocen Subversion como para quien empieza, pudiendo acceder a direcciones SVN, subir y descargar contenido y sincronizarlo con el servidor original, comprobar su estado, crear y fusionar direcciones.(LÓPEZ, 2010)

¿Por qué SVN y RapidSVN?

SVN promueve el trabajo y la colaboración en equipo. Permite administrar las versiones de los archivos de un proyecto, observar el trabajo, realizar cambios y guardar estos. RapidSVN proporciona una interfaz fácil de usar para las características de Subversion, simple para los principiantes pero lo suficientemente flexible como para aumentar la productividad para los usuarios de Subversion con experiencia, se ejecuta en cualquier plataforma.

1.4.5 Maven

Es una herramienta de gestión de proyectos que se basa en la “convención sobre configuración”. Es decir, asume un comportamiento por defecto que permite empezar a trabajar sin necesidad de configuración. Por ejemplo, Maven asume la estructura de los ficheros del proyecto, con lo cual a la hora de compilar no se pierde tiempo indicando donde se encuentra el código fuente o las librerías.

Entre los aspectos más importantes a destacar se encuentra la gestión de dependencias que nos permite saber qué librerías y qué versiones son necesarias en el proyecto para ejecutarse. Mediante plugins, permite generar los mapas de Hibernate a partir de una base de datos.

Maven define, entre otras cosas:

1. Un modelo de objeto del proyecto.
2. Un sistema de gestión de dependencias.
3. El ciclo de vida del proyecto.
4. La lógica para ejecutar nuevas tareas en determinadas fases del ciclo de vida.

1.4.6 Pruebas unitarias

Se basan en las pruebas realizadas a los principales procesos determinando fallas que pudieran ocurrir, como adelantarse a obtener los posibles errores.

¿Por qué pruebas unitarias?

1. Asegura calidad del código entregado. Es la mejor forma de detectar errores tempranamente en el desarrollo.
2. Ayuda a definir los requerimientos y responsabilidades de cada método en cada clase probada.
3. Constituye una buena forma de ejecutar pruebas de concepto. Cuando es necesario hacer pruebas de conceptos sin integrar usar pruebas unitarias se convierte en un método efectivo.
4. Permite hacer refactorización tempranamente en el código. No es necesario todo un ciclo de integración para hacer refactorización en la aplicación, basta con ver como se comporta un caso de prueba para hacer refactorización unitario sobre la clase que estamos probando en cuestión.
5. Permite incluso hacer pruebas de estrés tempranamente en el código. Por ejemplo un método que realice una consulta SQL que exceda los tiempos de aceptación es posible optimizarla antes de integrar con la aplicación.
6. Permite encontrar errores tempranamente en el desarrollo. Y está demostrado que mientras más temprano se corrijan los errores, menos costará corregirlos. (*RODRIGUEZ JORGE, 2006*).

1.5 Tecnologías utilizadas

1.5.1 Lenguaje de programación

Lenguaje de programación: Es aquel elemento dentro de la informática que nos permite crear programas mediante un conjunto de instrucciones, operadores y reglas de sintaxis; que pone a disposición del programador para que este pueda comunicarse con los dispositivos hardware y software existentes. (ARIAS MARÍN, 2008)

PHP: Es un lenguaje de programación interpretado, diseñado para la creación de páginas web dinámicas. Se usa principalmente para la interpretación del lado del servidor. El hecho de que el código fuente sea invisible al navegador web y al cliente hace que la programación en PHP sea segura y confiable.

Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. Permite aplicar técnicas de programación orientada a objetos. Utiliza el patrón de diseño Modelo Vista Controlador (MVC), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes.

JAVA: Es un lenguaje de programación que junta lo mejor de varios mundos como es C, C++, etc. Es un lenguaje orientado a objetos, lo que significa que toda la programación JAVA gira en torno a clases. Es un lenguaje multiplataforma, esto significa que no está enlazado a un sistema operativo en concreto y los programas desarrollados con él funcionarán correctamente tanto en Windows como en Linux. Java es un lenguaje dinámico.

La máquina virtual de Java enlaza los programas Java en tiempo de ejecución, eliminando la necesidad de enlazarlos con las librerías en tiempo de compilación. Java tiene la característica de que libera la memoria automáticamente, por lo cual uno no se tiene que preocupar por la pérdida de esta.

¿Porque utilizar JAVA?

Java es productivo porque usando los frameworks y librerías que ofrece es relativamente rápido y sencillo construir aplicaciones a medida. Tiene una gigantesca comunidad Java que hay detrás, con blogs, foros, ejemplos y

soluciones.

La versatilidad y eficiencia de la tecnología Java, la portabilidad de su plataforma y la seguridad que aporta, la han convertido en la tecnología ideal para su aplicación a redes. De portátiles a centros de datos, de consolas de juegos a súper equipos científicos, de teléfonos móviles a Internet, Java está en todas partes.

1.5.2 Frameworks

Los Frameworks ayudan en el desarrollo de software, proporcionan una estructura definida la cual ayuda a crear aplicaciones con mayor rapidez. Ayuda a la hora de realizar el mantenimiento del sitio gracias a la organización durante el desarrollo de la aplicación. Los Frameworks son desarrollados con el objetivo de brindarles a los programadores y diseñadores una mejor organización y estructura a sus proyectos.

Es un concepto sumamente genérico, se refiere a “ambiente de trabajo y ejecución. Es un marco de aplicación o conjunto de bibliotecas orientadas a la reutilización a muy gran escala de componentes software para el desarrollo rápido de aplicaciones. En general los framework son soluciones completas que contemplan herramientas de apoyo a la construcción y motores de ejecución.

JUnit: Es un “framework” para automatizar las pruebas unitarias de aplicaciones Java. Se utiliza en la fase de desarrollo mediante un conjunto de clases que permite realizar la ejecución de clases Java de manera controlada, para poder evaluar si el funcionamiento de cada uno de los métodos de la clase se comporta como se espera. (FUENTES PEREZ, 2011).

JWebSocket: Es un puro Java / JavaScript solución de alta velocidad de comunicación bidireccional para la Web, segura, fiable y rápido. Se utiliza para crear aplicaciones basadas en la comunicación en la web. Es una nueva tecnología ultra flexible y de alta velocidad de comunicación bidireccional. jWebSocket es una implementación de código abierto de Java y JavaScript del protocolo HTML5 WebSocket con un enorme conjunto de extensiones.

¿Porque utilizar jWebSocket y JUnit?

Ofrece innovadoras funciones de comunicación. Permite crear fácilmente aplicaciones, redes sociales y juegos en línea. Ofrece una amplia gama de funcionalidad básica. Con jWebSocket puede configurar rápidamente una solución independiente con un mínimo de instalación y configuración.

Mientras que JUnit, es una herramienta simple, fácil de usar y de configurar. Esta diseñado para reportar fallos o éxitos sobre código sin necesidad de interpretar el reporte. Ejecuta casos de prueba (Test Cases) contra un conjunto de objetos y provee formas de inicializar y configurar cada Caso.

1.5.3 Sistema Operativo

Un Sistema Operativo es un conjunto de programas que controla la ejecución de programas de aplicación y actúa como una interfaz entre el usuario y el hardware de una computadora, explota y administra los recursos de hardware de la computadora con el objeto de proporcionar un conjunto de servicios a los usuarios del sistema.

Características de los Sistemas Operativos.

1. Permite que los recursos de la computadora se usen de la manera más eficiente posible.
2. Se encarga de manejar de una mejor manera los recursos de la computadora en cuanto a hardware se refiere, esto es, asignar a cada proceso una parte del procesador para poder compartir los recursos.
3. Se encarga de comunicar a los dispositivos periféricos, cuando el usuario así lo requiera.
4. Organizar datos para acceso rápido y seguro.
5. Facilita al usuario el acceso y manejo de los dispositivos de Entrada/Salida de la computadora.
6. Permite que se puedan compartir el hardware y los datos entre los usuarios.

Ubuntu: Ubuntu es una distribución de GNU/Linux, enfocado a computadoras personales (escritorio y laptops), es una de las más importantes distribuciones

de Linux a nivel mundial. Siempre será de **libre de costo es decir** no pagas por una licencia de uso. Puedes descargar, usar y compartir Ubuntu con todos. Ubuntu está diseñado pensando en la seguridad.

Cumple con 4 libertades esenciales para sus usuarios

1. La libertad de ejecutar el programa, para cualquier propósito.
2. La libertad de estudiar cómo trabaja el programa y cambiarlo para que haga lo que el usuario quiera. El acceso al código fuente es una condición necesaria para ello.
3. La libertad de redistribuir copias para que el usuario pueda ayudar al prójimo.
4. La libertad de distribuir copias de sus versiones modificadas a terceros. Si alguien lo hace, puede dar a toda la comunidad una oportunidad de beneficiarse de sus cambios. El acceso al código fuente es una condición necesaria para ello.

Windows: Es un sistema operativo gráfico para computadoras personales cuyo propietario es la empresa Microsoft. Las distintas versiones de Windows, que ofrecen un entorno gráfico amigable y sencillo, principalmente desde la versión Windows 95, han convertido a Windows en el sistema operativo más utilizado del mundo. Por eso la mayoría de las empresas fabricantes de hardware y software en el mundo prefieren desarrollar sus aplicaciones basadas en este sistema operativo.

Windows ha incorporado a través de sus diferentes versiones múltiples herramientas que se han convertido en estándares en la mayoría de los usuarios en el mundo. Así, Windows incorpora, entre otro software, herramientas como Internet Explorer y el Reproductor de Windows Media, los cuales se han convertido en el navegador de Internet y reproductor multimedia, respectivamente.

Algunas de las características más relevantes son:

1. Mayor información y más asequible.
2. Permite ejecutar varias aplicaciones al mismo tiempo.
3. Posibilidad de integrar recursos multimedia.

4. Herramienta para el trabajo en red, transmisión de información y comunicación entre usuarios.
5. Incorporación de importantes programas para diversos usos.

¿Porque utilizar Ubuntu?

Lo mejor de todo es que es software libre, que se puede utilizar sin ninguna restricción en su licencia, se puede copiar e instalar en cualquier número de computadoras y lo mejor es que en este caso Ubuntu es gratuito. No es necesario ser un experto para probar este sistema debido a su simplicidad. Ubuntu ha ampliado las fronteras del territorio de Linux.

Conclusiones

En el presente capítulo se han dado a conocer los conceptos fundamentales relacionados con la comprensión de la presente investigación. Se realizó un análisis de las Metodología, tecnologías y herramientas que serán utilizadas a lo largo del desarrollo del sistema propuesto.

De esta forma queda seleccionado como IDE el NetBeans, como herramienta de CASE, VisualParadigm, lenguaje de programación; Java, para el control de versiones; Subversion (SVN) y RapidSVN, el framework a utilizar será jWebSocket y como sistema operativo Ubuntu.

Módulo Multa y Patrimonio

Capítulo 2. Características, Análisis y Diseño del Sistema.

En este capítulo se realiza el análisis de las características y principales funcionalidades del sistema. Se dará a conocer la planificación del proyecto por roles, los requisitos funcionales y no funcionales en los que es necesario basarse para la solución del problema y los diagramas generados durante este flujo de trabajo.

2.1 Solución Propuesta.

El Departamento de Multas y Patrimonio de la dirección de Finanzas y Precio de la Administración Provincial de Artemisa mostraron dificultades en cuanto a eficiencia y confiabilidad de la información generada, debido a que se gestiona de forma manual, por vía telefónica, por correo electrónico o mediante mensajeros que entregan la información de las entidades que se encuentran en los municipios de la provincia, en formato duro o digital.

Todas estas dificultades ocasionan problemáticas para la generación de reportes inmediatos, así como su entrega al Presidente en el tiempo y con la calidad requerida.

El módulo a realizar busca satisfacer los requerimientos de la Dirección de Finanzas y Precios de la Administración Provincial de Artemisa mediante el proceso de gestión de la información para que de esta forma contribuya a elevar la eficiencia y confiabilidad en los Departamentos de Multas y Patrimonio.

2.2 Información a automatizar.

El departamento de **Multas** se encarga de verificar, controlar y evaluar el trabajo de las oficinas de cobro y control de multas, así como el cumplimiento de los indicadores de eficacia en el cobro de las multas a través de los modelos:

1. Indicadores Seleccionados
2. Multas Canceladas
3. Cantidad de Operaciones
4. Análisis de las Devoluciones

5. Decreto Ley 171
6. Préstamo a Estudiantes
7. Gestión de Cobro

El Departamento de **Patrimonio** se encarga de manejar información referente a los inmuebles estatales existentes en la provincia, a través de los siguientes modelos:

1. Viviendas
2. Total de Expedientes Iniciados
3. Total de Inmuebles

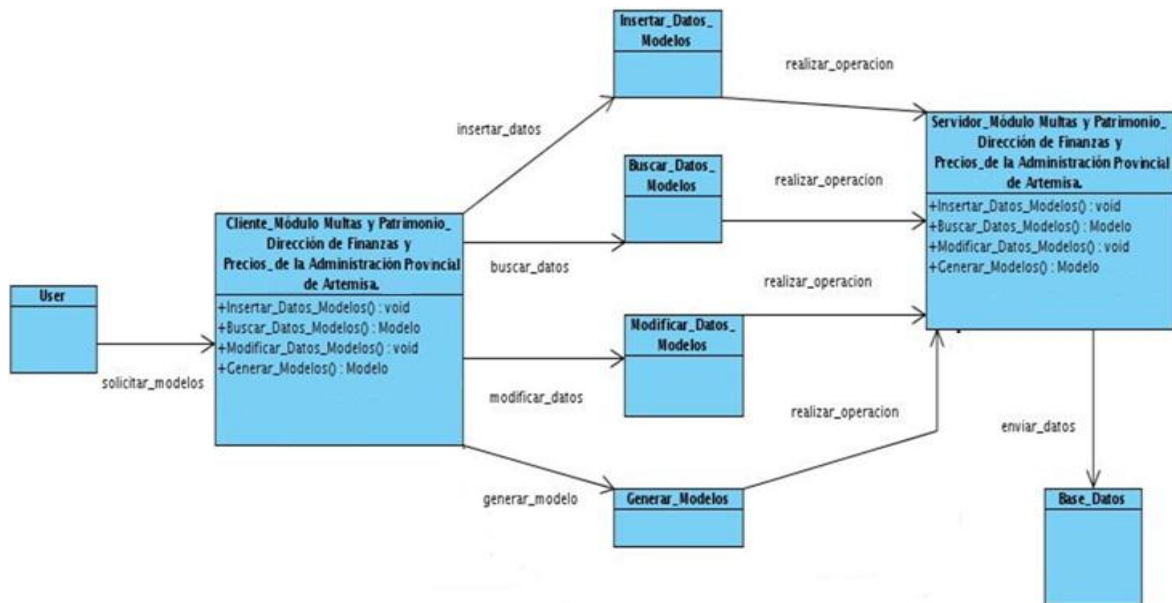
2.3 Planificación del Proyecto por roles.

Rol	Responsabilidad	Nombre
<i>Gerente</i>	Es el responsable de tomar las decisiones finales, acerca de estándares y convenciones a seguir durante el proyecto. Participa en la definición de objetivos y requerimientos, así como en la selección de los miembros del Equipo del Proyecto. Tiene la responsabilidad de controlar el progreso del software y trabaja junto con el Líder de Proyecto en la reducción de la Lista de reserva del producto, así como en la de riesgos.	<i>Dania Aguilar Fernández</i>
<i>Cliente</i>	Es el encargado de facilitar toda la información necesaria para el desarrollo del sistema.	<i>SIGOB</i>

<i>Analista</i>	Dirigir los recursos hacia el resultado deseado, deducción de requisitos, para determinar el comportamiento que se espera del software, garantía de calidad, para garantizar las perspectivas del cliente, diseño, para que exista una mínima certeza de que el software es viable y eficaz con la tecnología existente, gestión de configuración, para controlar el caos a medida que el software crece.	<i>Reddy García Collazo</i>
<i>Programador</i>	La implementación de requisitos mediante un lenguaje de programación escribiendo, depurando y manteniendo el código fuente del programa seleccionado para desarrollar el proyecto.	<i>Reddy García Collazo</i>
<i>Diseñador de BD</i>	El diseño lógico y físico de las bases de datos	<i>Nagyara Martínez</i> <i>Fernández</i>
<i>Administrador de BD</i>	Los aspectos ambientales de una base de datos. Crear y probar, verificar o ayudar a la verificación en la integridad de los datos, definir o implementar controles de acceso a los datos, ayudar a los programadores e utilizar eficientemente la base de datos.	<i>Nagyara Martínez</i> <i>Fernández</i>

2.4 Modelo de Dominio.

El modelo de dominio puede ser tomado como el punto de partida para el diseño del sistema y tiene como objetivo capturar lo necesario para comprender donde va a funcionar el sistema y los conceptos que utilizan los usuarios.



2.5 Lista de Reserva del Producto (LRP).

La Lista de Reserva del Producto (LRP) recoge con un orden lógico el trabajo a realizar en el proyecto. Puede crecer y modificarse mientras que se van obteniendo información del producto y del cliente.

Prioridad	Ítem *	Descripción	Estimación	Estimado por
Muy Alta				
	1	Insertar datos de Inmuebles por Municipios	2 días	Analista
	2	Buscar datos de Inmuebles por Municipios	1 día	Analista
	3	Insertar datos del total de Expedientes iniciados por	2 días	Analista

		Municipios		
	4	Buscar datos de total del Expedientes iniciados por Municipios	1 día	Analista
	5	Insertar datos de Viviendas por Municipios	2 días	Analista
	6	Buscar datos de Viviendas por Municipios	1 día	Analista
	7	Insertar datos de Indicadores por Municipios	2 días	Analista
	8	Buscar datos de Indicadores por Municipios	1 día	Analista
	9	Insertar datos de Multas Canceladas por Organismos	1 día	Analista
	10	Buscar datos de Multas Canceladas por Organismos	1 día	Analista
	11	Insertar datos de Operaciones por Municipios	2 días	Analista
	12	Buscar datos de Operaciones por Municipios	1 día	Analista
	13	Insertar datos de Devoluciones de Org.Impositores	2 días	Analista

	14	Buscar datos de Devoluciones de Org.Impositores	1 día	<i>Analista</i>
	15	Insertar datos de información sobre la situación de las Multas del Decreto Ley 171	2 días	<i>Analista</i>
	16	Buscar datos de información sobre la situación de las Multas del Decreto Ley 171	1 día	<i>Analista</i>
	17	Insertar datos de Contratos de Préstamos a Estudiantes	2 días	<i>Analista</i>
	18	Buscar datos de Contratos de Préstamos a Estudiantes	1 día	<i>Analista</i>
	19	Insertar datos de información sobre la Gestión de Cobro	2 días	<i>Analista</i>
	20	Buscar datos de información sobre la Gestión de Cobro	2 días	<i>Analista</i>
	21	Modificar datos de Inmuebles por Municipios	1 día	<i>Analista</i>
	22	Modificar datos del total de Expedientes iniciados por Municipios	1 día	<i>Analista</i>

	23	Modificar datos de Vivienda por Municipios	1 día	<i>Analista</i>
	24	Modificar datos de Indicadores por Municipios	1 día	<i>Analista</i>
	25	Modificar datos de Multas Canceladas por Organismos	1 día	<i>Analista</i>
	26	Modificar datos de Operaciones por Municipios	1 día	<i>Analista</i>
	27	Modificar datos de Devoluciones de Org. Impositores	1 día	<i>Analista</i>
	28	Modificar datos de información sobre la situación de las Multas del Decreto Ley 171	1 día	<i>Analista</i>
	29	Modificar datos de Contratos de Préstamos a Estudiantes	1 día	<i>Analista</i>
	30	Modificar datos de información sobre la Gestión de Cobro	1 día	<i>Analista</i>
Alta				
	31	Generar Reporte de Inmuebles por Municipios	1 día	<i>Analista</i>

	32	Generar Reporte total de Expedientes iniciados	1 día	Analista
	33	Generar Reporte Viviendas	1 día	Analista
	34	Generar Reporte Indicadores por Municipios	1 día	Analista
	35	Generar Reporte de Multas Canceladas	1 día	Analista
	36	Generar Reporte de la Cantidad de Operaciones	1 día	Analista
	37	Generar Reporte del Análisis de las Devoluciones por Organismos Impositores	1 día	Analista
	38	Generar Reporte sobre la situación de las Multas del Decreto Ley 171	1 día	Analista
	39	Generar Reporte del Informe sobre los Contratos de Préstamos a Estudiantes	1 día	Analista
	40	Generar Reporte del Informe sobre la Gestión de Cobro	1 día	Analista
Media				

Baja			
	41	Se necesita 1 Gb de Ram como mínimo tanto en el servidor Web como en el servidor de Base de Datos.	
	42	Se debe contar con un Disco Duro de 80 Gb (o más) en el servidor Web y en el de Base de Datos.	
	43	Los Navegadores Web a utilizar pueden ser Firefox 7.0, Internet Explorer4 10.0, Chrom.	
	44	Se debe tener instalados y configurados los Repositorios de Maven en el servidor web.	
	45	Se debe tener instalado y configurado el IDE Netbeans 7.0.1 en los servidores Web.	
	46	Se debe tener instalada y configurada la máquina virtual de Java OpenJDK 7 en el servidor Web.	
	47	Se debe utilizar para realizar las salvas de todos los cambios que se realicen sobre la información el	

	<i>Gestor de Base de Datos</i> <i>Postgres SQL.</i>	
--	--	--

2.6 Historias de Usuarios y Tareas de Ingeniería.

Las historias de usuario es una forma de administrar los requisitos sin tener que elaborar gran cantidad de documentos y sin requerir de mucho tiempo para administrarlos.

Historia de Usuario	
Número: <i>HU_7</i>	Nombre Historia de Usuario: Gestionar datos del reporte Indicadores por Municipios.
Modificación de Historia de Usuario Número: <i>Ninguna</i>	
Usuario: Reddy García Collazo(Servidor) Jennifel López Alvarez(Cliente)	Iteración Asignada: 2
Prioridad en Negocio: <i>Muy Alta</i>	Puntos Estimados: <i>0,6 semanas</i>
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: <i>0,6 semanas</i>
Descripción: La presente historia de usuario tiene como objetivo insertar, modificar y buscar toda la información referente al reporte Indicadores por Municipios en el departamento Multas.	
Observaciones: Ninguna	
<u>Prototipo de interfaz: Ver Anexo #2 Gestionar datos del reporte Indicadores por Municipios.</u>	

Tarea de Ingeniería	
Número Tarea: 7.1	Número Historia de Usuario: HU_7
Nombre Tarea: Implementación de la funcionalidad Insertar datos del reporte Indicadores por Municipios.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2/6
Fecha Inicio: 13/02/12	Fecha Fin: 15/02/12
Programador Responsable: Reddy García Collazo, Jennifel López Alvarez	
Descripción: A partir de la presente Tarea de Ingeniería se realiza la inserción de los datos que se manejan en el reporte Indicadores por Municipios del Departamento de Multa de la Dirección de Finanzas y Precios.	

Tarea de Ingeniería	
Número Tarea: 7.2	Número Historia de Usuario: HU_7
Nombre Tarea: Implementación de la funcionalidad Modificar datos del reporte Indicadores por Municipios.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1/6
Fecha Inicio: 15/02/12	Fecha Fin: 16/02/12
Programador Responsable: Reddy García Collazo, Jennifel López Alvarez	
Descripción: A partir de la presente Tarea de Ingeniería se realiza la modificación de los datos ya insertados en el reporte Indicadores por Municipios del Departamento de Multa de la Dirección de Finanzas y Precios.	

Tarea de Ingeniería	
Número Tarea: 7.3	Número Historia de Usuario: HU_7
Nombre Tarea: Implementación de la funcionalidad Buscar datos del reporte Indicadores por Municipios.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1/6
Fecha Inicio: 16/02/12	Fecha Fin: 17/02/12
Programador Responsable: Reddy García Collazo, Jennifel López Alvarez	
Descripción: A partir de la presente Tarea de Ingeniería se realiza la búsqueda de los datos que se manejan en el reporte Indicadores por Municipios del Departamento de Multa de la Dirección de Finanzas y Precios.	

Historia de Usuario	
Número: HU_8	Nombre Historia de Usuario: Generar Reporte Indicadores por Municipios
Modificación de Historia de Usuario Número: <i>Ninguna</i>	
Usuario: Reddy García Collazo(Servidor) Jennifel López Alvarez(Cliente)	Iteración Asignada: 3
Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: <i>0,2 semanas</i>
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: <i>0,2 semanas</i>
Descripción: La presente historia de usuario tiene como objetivo generar un reporte con toda la información referente a Indicadores por Municipios.	
Observaciones: Para que sea posible se debe realizar anteriormente la inserción de los datos de al menos un Indicador por Municipio.	

Prototipo de interfaz: [Ver Anexo #3 Generar Reporte Indicadores por Municipios](#)

Tarea de Ingeniería	
Número Tarea: 8.1	Número Historia de Usuario: HU_8
Nombre Tarea: Implementación de la funcionalidad generar reportes sobre Indicadores por Municipios.	
Tipo de Tarea : Desarrollo.	Puntos Estimados: 1/6
Fecha Inicio: 19/03/12	Fecha Fin: 20/03/12
Programador Responsable: Reddy García Collazo Jennifel López Alvarez	
Descripción: En la presente Tarea de Ingeniería a partir de los datos insertados se genera el reporte Indicadores por Municipios del Departamento de Multa de la Dirección de Finanzas y Precios.	

Historia de Usuario	
Número: HU_9	Nombre Historia de Usuario: Gestionar datos del reporte Multas Canceladas por Organismos
Modificación de Historia de Usuario Número: <i>Ninguna</i>	
Usuario: Reddy García Collazo(Servidor) Jennifel López Alvarez(Cliente)	Iteración Asignada: 2
Prioridad en Negocio: <i>Muy Alta</i>	Puntos Estimados: <i>0,6 semanas</i>
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: <i>0,6 semanas</i>
Descripción: La presente historia de usuario tiene como objetivo insertar,	

modificar y buscar toda la información referente al reporte Multas Canceladas por Organismos en el departamento Multas.

Observaciones: Ninguna

[Prototipo de interfaz: Ver Anexo #4 Gestionar datos del reporte Multas Canceladas por Organismos](#)

Tarea de Ingeniería

Número Tarea: 9.1

Número Historia de Usuario: HU_9

Nombre Tarea: Implementación de la funcionalidad Insertar datos del reporte Multas Canceladas por Organismos

Tipo de Tarea: Desarrollo.

Puntos Estimados: 1/6

Fecha Inicio: 17/02/12

Fecha Fin: 18/02/12

Programador Responsable: Reddy García Collazo
Jennifel López Alvarez

Descripción: A partir de la presente Tarea de Ingeniería se realiza la inserción de los datos que se manejan en el reporte Multas Canceladas por Organismos en el Departamento de Multa de la Dirección de Finanzas y Precios.

Tarea de Ingeniería

Número Tarea: 9.2

Número Historia de Usuario: HU_9

Nombre Tarea: Implementación de la funcionalidad Modificar datos del reporte Multas Canceladas por Organismos

Tipo de Tarea: Desarrollo.

Puntos Estimados: 1/6

Fecha Inicio: 18/02/12

Fecha Fin: 20/02/12

Programador Responsable: Reddy García Collazo, Jennifel López Alvarez

Descripción: A partir de la presente Tarea de Ingeniería se realiza la modificación de los datos ya insertados en el reporte Multas Canceladas por Organismos del Departamento de Multa de la Dirección de Finanzas y Precios.

Tarea de Ingeniería

Número Tarea: 9.3

Número Historia de Usuario: HU_9

Nombre Tarea: Implementación de la funcionalidad Buscar datos del reporte Multas Canceladas por Organismos

Tipo de Tarea : Desarrollo.

Puntos Estimados: 1/6

Fecha Inicio: 20/02/12

Fecha Fin: 21/02/12

Programador Responsable: Reddy García Collazo
Jennifel López Alvarez

Descripción: A partir de la presente Tarea de Ingeniería se realiza la búsqueda de los datos que se manejan en el reporte Multas Canceladas por Organismos del Departamento de Multa de la Dirección de Finanzas y Precios.

Historia de Usuario

Número: HU_10

Nombre Historia de Usuario: Generar Reporte de Multas Canceladas.

Modificación de Historia de Usuario Número: Ninguna

Usuario: Reddy García Collazo(Servidor)
Jennifel López Alvarez(Cliente)

Iteración Asignada: 3

Prioridad en Negocio: <i>Alta</i>	Puntos Estimados: <i>0,2 semanas</i>
Riesgo en Desarrollo: <i>Alto</i>	Puntos Reales: <i>0,2 semanas</i>
Descripción: La presente historia de usuario tiene como objetivo generar un reporte con toda la información referente a Multas Canceladas.	
Observaciones: Para que sea posible se debe realizar anteriormente la inserción de los datos de al menos una Multa Cancelada.	
<u>Prototipo de interfaz: Ver Anexo #5 Generar Reporte de Multas Canceladas.</u>	
Tarea de Ingeniería	
Número Tarea: 10.1	Número Historia de Usuario: HU_10
Nombre Tarea: Implementación de la funcionalidad generar reportes sobre Multas Canceladas por Organismos	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1/6
Fecha Inicio: 20/03/12	Fecha Fin: 21/03/12
Programador Responsable: Reddy García Collazo Jennifel López Alvarez	
Descripción: En la presente Tarea de Ingeniería a partir de los datos insertados se genera el reporte Multas Canceladas por Organismos del Departamento de Multa de la Dirección de Finanzas y Precios.	

Historia de Usuario	
Número: HU_11	Nombre Historia de Usuario: Gestionar datos del reporte Operaciones por Municipios.

Modificación de Historia de Usuario Número: *Ninguna*

Usuario: Reddy García Collazo(Servidor)
Jennifel López Alvarez(Cliente)

Iteración Asignada: 2

Prioridad en Negocio: *Muy Alta*

Puntos Estimados: *0,6
semanas*

Riesgo en Desarrollo: *Alto*

Puntos Reales: *0,6 semanas*

Descripción: La presente historia de usuario tiene como objetivo insertar, modificar y buscar toda la información referente al reporte Operaciones por Municipios en el departamento Multas.

Observaciones: Ninguna

[Prototipo de interfaz: Ver Anexo #6 Gestionar datos del reporte Operaciones por Municipios.](#)

Tarea de Ingeniería

Número Tarea: 11.1

Número Historia de Usuario: HU_11

Nombre Tarea: Implementación de la funcionalidad Insertar datos del reporte Operaciones por Municipios.

Tipo de Tarea: Desarrollo.

Puntos Estimados: 2/6

Fecha Inicio: 21/02/12

Fecha Fin: 23/02/12

Programador Responsable: Reddy García Collazo, Jennifel López Alvarez

Descripción: A partir de la presente Tarea de Ingeniería se realiza la inserción de los datos que se manejan en el reporte Operaciones por Municipios del Departamento de Multa de la Dirección de Finanzas y Precios.

Tarea de Ingeniería	
Número Tarea: 11.2	Número Historia de Usuario: HU_11
Nombre Tarea: Implementación de la funcionalidad Modificar datos del reporte Operaciones por Municipios.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1/6
Fecha Inicio: 23/02/12	Fecha Fin: 24/02/12
Programador Responsable: Reddy García Collazo, Jennifel López Alvarez	
Descripción: A partir de la presente Tarea de Ingeniería se realiza la modificación de los datos ya insertados en el reporte Operaciones por Municipios del Departamento de Multa de la Dirección de Finanzas y Precios.	

Tarea de Ingeniería	
Número Tarea: 11.3	Número Historia de Usuario: HU_11
Nombre Tarea: Implementación de la funcionalidad Buscar datos del reporte Operaciones por Municipios.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1/6
Fecha Inicio: 24/02/12	Fecha Fin: 25/02/12
Programador Responsable: Reddy García Collazo , Jennifel López Alvarez	
Descripción: A partir de la presente Tarea de Ingeniería se realiza la búsqueda de los datos que se manejan en el reporte Operaciones por Municipios del Departamento de Multa de la Dirección de Finanzas y Precios.	

Historia de Usuario	
Número: HU_12	Nombre Historia de Usuario: Generar Reporte de la Cantidad de Operaciones.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Reddy García Collazo(Servidor) Jennifel López Alvarez(Cliente)	Iteración Asignada: 3
Prioridad en Negocio: Alta	Puntos Estimados: 0,2 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 0,2 semanas
Descripción: La presente historia de usuario tiene como objetivo generar un reporte con toda la información referente a Cantidad de Operaciones.	
Observaciones: Para que sea posible se debe realizar anteriormente la inserción de los datos de al menos una Operación.	
Prototipo de interfaz: Ver Anexo #7 Generar Reporte de la Cantidad de Operaciones.	
Tarea de Ingeniería	
Número Tarea: 12.1	Número Historia de Usuario: HU_12
Nombre Tarea: Implementación de la funcionalidad generar reportes sobre la Cantidad de Operaciones.	
Tipo de Tarea: Desarrollo.	Puntos Estimados: 1/6
Fecha Inicio: 21/03/12	Fecha Fin: 22/03/12
Programador Responsable: Reddy García Collazo, Jennifel López Alvarez	

Descripción: En la presente Tarea de Ingeniería a partir de los datos insertados se genera el reporte Cantidad de Operaciones del Departamento de Multa de la Dirección de Finanzas y Precios.

Departamento Patrimonio Ver Anexo #17

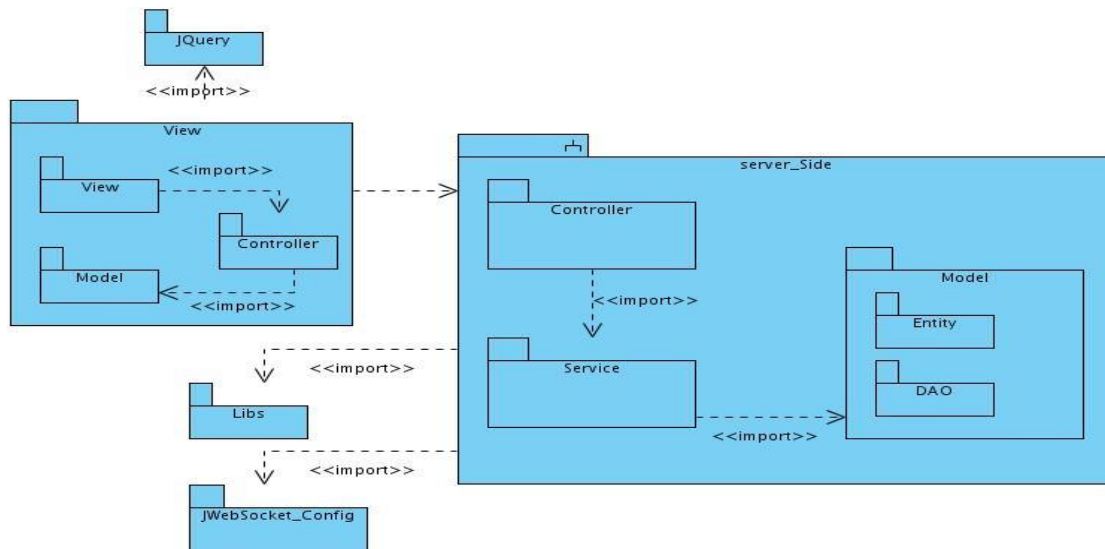
2.7 Plan de Release.

El plan de release define las iteraciones a realizar. Tiene como entrada las Historias de Usuario definidas previamente la prioridad que definida en cada historia.

Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
2	En esta iteración se desarrollarán las historias de usuario Gestionar los datos que contienen los reportes.	HU_1, HU_3, HU_5, HU_7, HU_9, HU_11, HU_13, HU_15, HU_17, HU_19	6,6 semanas
3	En esta iteración se desarrollarán las historias de usuario Generar los Reportes y se irán integrando con las ya realizadas.	HU_2, HU_4, HU_6, HU_8, HU_10, HU_12, HU_14, HU_16, HU_18 ,HU_20	1,6 semanas

2.8 Diseño con Metáfora.

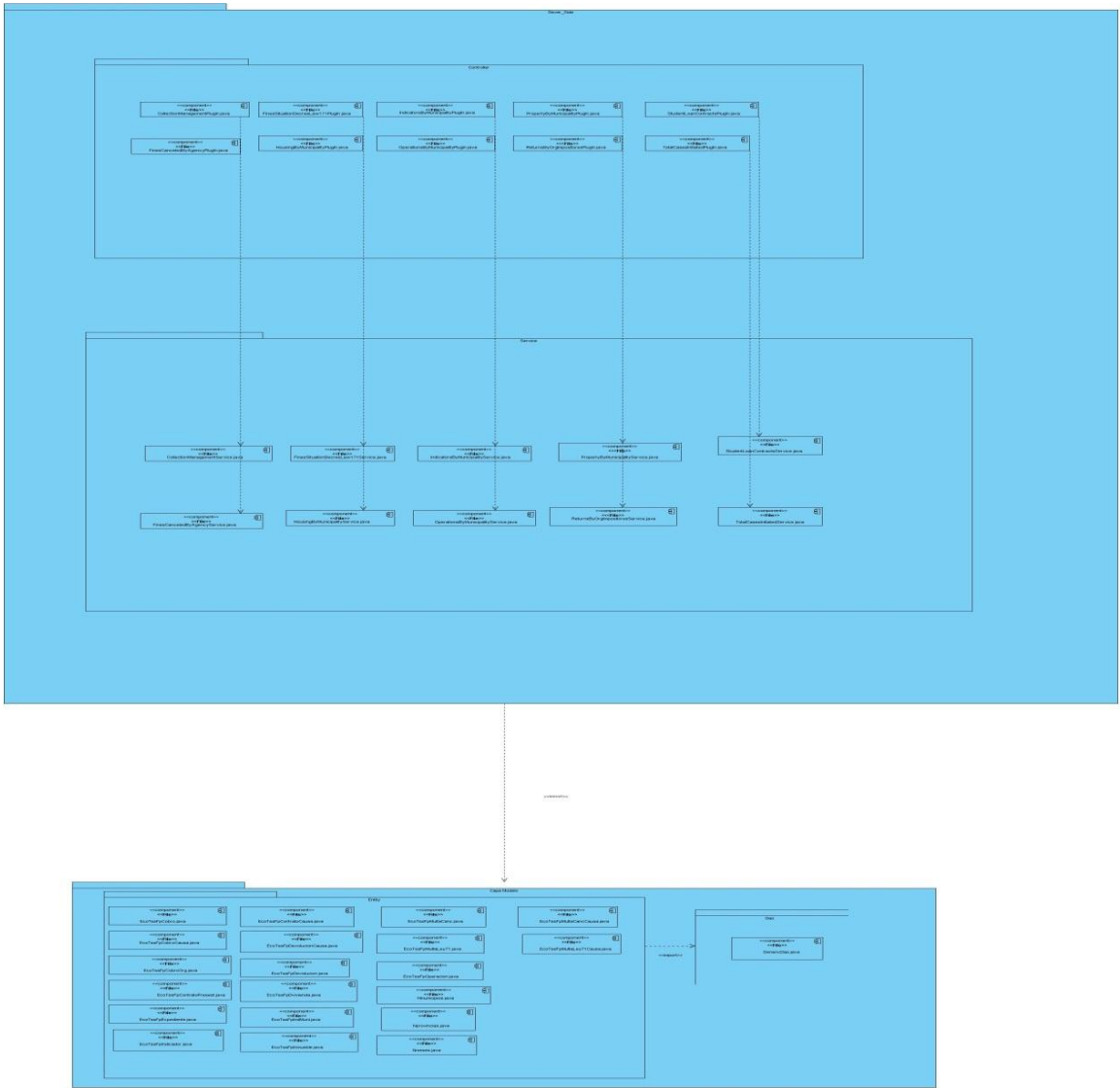
El Diseño con metáforas es un diseño de solución más simple que pueda funcionar y ser implementado en un momento dado, es una historia compartida que describe cómo debería funcionar el sistema de lo cual se genera el artefacto Modelo de Diseño.



1. En el paquete de presentación se encuentran las clases donde se reciben y muestran los datos de la aplicación a los clientes por cada requisito funcional observado en el negocio y las relaciones entre ellas.
2. En el paquete Controlador se encuentran los Plugins y los servicios donde se manejan y procesan los datos de la aplicación de cada uno de los requisitos funcionales observados en el negocio y las relaciones de entre ellos.
3. El paquete de Modelo tiene las principales entidades donde se encuentra almacenada la información que será gestionada.

2.9 Diagrama de Componentes.

El diagrama de componentes ilustra los componentes que serán usados para construir el sistema y satisfacer sus requisitos. El Diagrama de Componente muestra la relación entre los componentes y sus dependencias. Son usados para estructurar los componentes en los sistemas del software. Además examinan y controlan las dependencias entre componentes o interfaces. Estos diagramas muestran elementos físicos los cuales incluyen archivos, bibliotecas, módulos o paquetes.



Conclusiones

Al concluir este capítulo se realizó la descripción de los procesos del negocio, obteniéndose una perspectiva del sistema que se desea desarrollar. Partiendo de los requerimientos funcionales y no funcionales se definen las capacidades y cualidades que la aplicación debe tener. De esta forma se sientan las bases para una exitosa implementación a partir del diseño planteado anteriormente.

CAPÍTULO 3: Implementación y Validación del Sistema.

En el presente capítulo se expone el estándar de código utilizado en el desarrollo de la implementación realizada a partir de los requerimientos; las pruebas efectuadas a las diferentes funcionalidades del sistema con el fin de lograr un producto con calidad, así como los resultados obtenidos hasta el momento y su aporte económico y social.

3.1 Estándar de Código

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Al comenzar un proyecto de software se establece un estándar de codificación para asegurarse de que todos los programadores del proyecto trabajen de forma coordinada.

La legibilidad del código fuente repercute directamente en lo bien que un programador comprende un sistema de software. La mantenibilidad del código es la facilidad con que el sistema de software puede modificarse para añadirle nuevas características, modificar las ya existentes, depurar errores, o mejorar el rendimiento.

En el desarrollo de este módulo se utilizó para la definición de los nombres de los identificadores el convenio establecido por CamelCase. Para las clases, enumeradores, interfaces se hizo uso de la variante UpperCamelCase. Usando 4 espacios de indentación en el código y cada línea no debe sobrepasar los 80 caracteres. Se declararon todos los atributos de la clase al principio de esta. Los comentarios en bloque deben ser utilizados al comienzo de cada archivo y antes de cada método.

3.2 Distribución de la Implementación.

La implementación esta compuesta por varios elementos que a continuación se definen:

Eventos: Son Las acciones del usuario sobre el programa. Cada vez que se produce un evento, arranca una determinada función o procedimiento que realiza la acción programada por el usuario para ese evento concreto. Es un mensaje de software que indica que algo ha ocurrido.

```

    */
    public class GenerateCollectionManagementEvent extends C2SEvent {

        private Integer month;
        private Integer year;
        private Integer townName;

        /**
         * @return the townName
         */
        public Integer getTownName() {
            return townName;
        }

        /**
         * @param townName the townName to set
         */
        @ImportFromToken
        public void setTownName(Integer townName) {
            this.townName = townName;
        }

        /**
         * @return the month
         */
        public Integer getMonth() {
            return month;
        }
    }

```

Servicios: Permiten realizar la operación solicitada por el cliente, estas clases son las ordenan la ejecución de consultas en la bases de datos mediante el uso del GenericDao.

```

    */
    public class CollectionManagementService {

        private GenericDao daoFines;

        /**
         * Este método Inserta los datos del modelo Gestion de Cobro
         * @param managementInsert
         */
        @Transactional
        public boolean insertCollectionManagement(InsertCollectionManagementEvent managementInsert) {

            EcoTssFpCobro collectionInsert = new EcoTssFpCobro();
            EcoTssFpCobroCausa collectionCause = new EcoTssFpCobroCausa();
            EcoTssFpCobroOrg collectionOrg = new EcoTssFpCobroOrg();
            Boolean var = false;
            List<EcoTssFpCobroCausa> list = daoFines.find(EcoTssFpCobroCausa.class);
            List<EcoTssFpCobroCausa> result = new ArrayList<EcoTssFpCobroCausa>();
            for (int i = 0; i < list.size(); i++) {
                if (list.get(i).getEcoTssFpCobro().getEcoTssFpMes() == managementInsert.getMonth()
                    && (list.get(i).getEcoTssFpCobro().getAno() == managementInsert.getYear())
                    && (list.get(i).getEcoTssFpCobro().getIdMunicipio() == managementInsert.getTownName())
                    && (list.get(i).getEcoTssFpCausa().getId() == managementInsert.getCause().longValue())) {

                    result.add(list.get(i));
                }
            }
            if (result.isEmpty()) {
                var = true;
            }
        }
    }

```

Plug-ins: Extensiones para dar soporte a los acontecimientos en el lado del servidor, básicamente son para escuchar o notificar a los eventos relacionados en el mismo.

```
public class CollectionManagementPlugIn extends EventModelPlugIn {  
    private CollectionManagementService service;  
  
    /**  
     * Este método se encarga de procesar el evento insertar del modelo Gestión  
     * de Cobro  
     * @param aEvent  
     * @param aResponseEvent  
     */  
    public void processEvent(InsertCollectionManagementEvent aEvent, C2SResponseEvent aResponseEvent) {  
        Boolean flag = service.insertCollectionManagement(aEvent);  
        aResponseEvent.getArgs().setBoolean("response", flag);  
    }  
  
    /**  
     * Este método se encarga de procesar el evento modificar del modelo Gestión  
     * de Cobro  
     * @param aEvent  
     * @param aResponseEvent  
     */  
    public void processEvent(ModifyCollectionManagementEvent aEvent, C2SResponseEvent aResponseEvent) {  
        service.modifyCollectionManagement(aEvent);  
        aResponseEvent.getArgs().setString("response", "Ha sido modificado correctamente");  
    }  
}
```

3.3 Pruebas Unitarias con JUnit.

Las pruebas unitarias son comprobaciones que se le realizan al software para garantizar su correcto funcionamiento de las clases y sus principales funcionalidades durante la implementación. Su objetivo principal es garantizar que cada método realizado sea un elemento funcional para de esta garantizar el paso a una nueva etapa sin errores anteriores.

Las pruebas que se realizan al Módulo Multa y Patrimonio de la Dirección de Finanzas y Precios se realizaron como parte de las pruebas ejecutadas por el procedimiento para la ejecución automática de pruebas unitarias en los proyectos de la Facultad Regional “Mártires de Artemisa”. La comprobación de datos se dividió en dos momentos mostrándose resultados diferentes y ascendentes siendo estos el 50% y el 100% respectivamente. A continuación se muestra un ejemplo:

```
CollectionManagementServiceTest.java x CollectionManagementService.java x
70 @Test
71 public void testModifyCollectionManagement() {
72     System.out.println("modifyCollectionManagement");
73     ModifyCollectionManagementEvent managementModify = new ModifyCollectionManagementEvent();
74     managementModify.setAgencyAmount("32.2");
75     managementModify.setAgencyName(9);
76     managementModify.setAgencyNumber(8);
77     managementModify.setAmount("65.2");
78     managementModify.setCause(4);
79     managementModify.setDepartment(5);
80     managementModify.setMoney(56);
81     managementModify.setMonth(6);
82     managementModify.setNameId(32);
83     managementModify.setNumber(14);
84     managementModify.setProvince(3);
85     managementModify.setTownName(21);
86     managementModify.setYear(2011);
87     CollectionManagementService instance = new CollectionManagementService();
88     boolean expectedResult = true;
89     boolean result = instance.modifyCollectionManagement(managementModify);
90     assertEquals(expectedResult, result);
91     // TODO review the generated test code and remove the default call to fail.
92     System.out.println(result);
93 }
94 }
95
```

Test Results

pruebasigobArt x

50,00 %

1 test passed, 1 test caused an error.(0,111 s)

- pruebasigobart.CollectionManagementServiceTest FAILED
 - testInsertCollectionManagement caused an ERROR: For input string: "habana"
 - For input string: "habana"
 - java.lang.NumberFormatException
 - at sun.misc.FloatingDecimal.readJavaFormatString(FloatingDecimal.java:1224)
 - at java.lang.Double.parseDouble(Double.java:510)
 - at pruebasigobart.CollectionManagementService.insertCollectionManagement(CollectionManagementService.java:12)
 - at pruebasigobart.CollectionManagementServiceTest.testInsertCollectionManagement(CollectionManagementServiceTest.java:12)
 - testModifyCollectionManagement passed (0,002 s)

modifyCollectic
true

Breakpoints Output Test Results

```
CollectionManagementServiceTest.java x CollectionManagementService.java x
41  */
42  @Test
43  public void testInsertCollectionManagement() {
44
45      InsertCollectionManagementEvent managementInsert = new InsertCollectionManagementEvent();
46      managementInsert.setAgencyAmount("3.5");
47      managementInsert.setAgencyName(1);
48      managementInsert.setAgencyNumber(2);
49      managementInsert.setAmount("3.3");
50      managementInsert.setCause(3);
51      managementInsert.setDepartment(4);
52      managementInsert.setMoney(5);
53      managementInsert.setMonth(6);
54      managementInsert.setNumber(7);
55      managementInsert.setProvince(8);
56      managementInsert.setTownName(9);
57      managementInsert.setYear(2012);
58      CollectionManagementService instance = new CollectionManagementService();
59      boolean expectedResult = true;
60      boolean result = instance.insertCollectionManagement(managementInsert);
61      assertEquals(expectedResult, result);
62      // TODO review the generated test code and remove the default call to fail.
63      System.out.println("insertCollectionManagement");
64      System.out.println(result);
65  }
66
67  /**
68   * Test of modifyCollectionManagement method, of class CollectionManagementService.
69   */
70  @Test
```

Test Results

pruebasigobArt x

100,00 %

Both tests passed.(0,115 s)

- pruebasigobart.CollectionManagementServiceTest passed
 - testInsertCollectionManagement passed (0,011 s)
 - testModifyCollectionManagement passed (0,001 s)

```
insertCollectionManagement
true
modifyCollectionManagement
true
```

Breakpoints Output Test Results

3.4 Resultados Obtenidos.

Como resultado de este trabajo se obtuvo el Módulo Multas y Patrimonio de la Administración Provincial de Artemisa.

3.5 Funcionalidades Obtenidas.

Este sistema está conformado por varias funcionalidades que serán mencionadas a continuación:

1. Insertar los datos que conforman los modelos de los Departamentos de Multa y Patrimonio de la dirección de Finanzas y precios de la Administración Provincial de Artemisa.
2. Modificar los datos que conforman los modelos de los Departamentos de Multa y Patrimonio de la dirección de Finanzas y precios de la Administración Provincial de Artemisa.
3. Buscar los datos que conforman los modelos de los Departamentos de Multa y Patrimonio de la dirección de Finanzas y precios de la Administración Provincial de Artemisa.
4. Generar los datos que conforman los modelos de los Departamentos de Multa y Patrimonio de la dirección de Finanzas y precios de la Administración Provincial de Artemisa.

3.6 Aporte Social y Económico.

El Módulo Multas y Patrimonio trae ventajas para el desarrollo económico y social de la Administración Provincial de Artemisa pues le facilita el trabajo a los especialistas y disminuye el tiempo en cuanto al procesamiento de la información en los Departamento de Multas y Patrimonio. Además le posibilita a la Administración Provincial de Artemisa un mejor control de la información que se maneja en estos departamentos. Este sistema le ahorra a la entidad el costo de la compra de un software para la gestión de sus actividades.

Conclusiones

Al concluir el desarrollo de las pruebas al sistema se demuestra con el análisis de los resultados obtenidos que el sistema ha cumplido con las funcionalidades descritas en la etapa inicial, en el tiempo establecido y cumpliendo con el estándar de código propuesto. Funcionalidades que son de gran importancia para elevar la eficiencia y confiabilidad en el proceso de gestión de la información de los departamentos de Multas y Patrimonio de la Dirección de Finanzas y Precios de la Administración Provincial de Artemisa.

Módulo Multa y Patrimonio

CONCLUSIONES

Mediante el presente trabajo se realizó un análisis de la metodología, las herramientas y tecnologías seleccionadas, para crear un módulo que contribuya a elevar la eficiencia y confiabilidad en el proceso de gestión de la información relacionada con los Departamentos Multas y Patrimonio de la Dirección de Finanzas y Precios. A través del análisis y diseño de la solución propuesta permitió una mejor organización, reduciendo la complejidad de implementación de las historias de usuarios.

Se le dio respuesta a los requisitos funcionales propuestos para el desarrollo del Módulo Multas y Patrimonio, contribuyendo a que este producto eleve la eficiencia y confiabilidad, mejorando la calidad de servicio en la dirección de Finanzas y Precios.

Módulo Multa y Patrimonio

RECOMENDACIONES

De manera general se puede plantear que el Módulo Multas y Patrimonio cumple con los objetivos descritos al inicio del trabajo, sin embargo, se proponen las siguientes recomendaciones:

1. Continuar optimizando las funcionalidades para mejorar la gestión de la información en los Departamento de Multas y Patrimonio.
2. Extender la solución propuesta a otras entidades gubernamentales provinciales.
3. Desplegar a las entidades municipales, para de esta manera garantizar y optimizar la llegada de la información a los departamentos Multa y Patrimonio.

Módulo Multa y Patrimonio

REFERENCIA BIBLIOGRÁFICA

ALLEGUI FERNÁNDEZ, Y.. *Metodologías de desarrollo de software - EcuRed* . ed. Última actualización: 2011. (). Disponible en: http://www.ecured.cu/index.php/Metodolog%C3%ADas_de_desarrollo_de_software.

ARCEO ANACELI. Sistema de gestión como estrategia dentro de la administración. ed. Última actualización: 2008. (). Disponible en: <http://www.gestiopolis.com/administracion-estrategia/sistemas-de-gestion-como-estrategia-dentro-de-la-administracion.htm>.

ARIAS MARÍN, M. D.. *Definición de lenguaje de programación. Tipos. Ejemplos* . ed. Última actualización: 2008. (). Disponible en: <http://catedraprogramacion.foroactivo.net/t83-definicion-de-lenguaje-de-programacion-tipos-ejemplos>.

BERBERENA, R.. *Los sistemas integrados de gestión de la información - Ilustrados!* . ed. Última actualización: 2011. (). Disponible en: <http://www.ilustrados.com/tema/13101/sistemas-integrados-gestion-informacion.html>.

CHIVENATO IDALBERTO. Definición información ed. Última actualización: 2006. (). Disponible en: <http://www.promonegocios.net/mercadotecnia/definicion-informacion.html>

CORRAL, R.. *SCRUM* . ed. Última actualización: 2007. (). Disponible en: <http://www.chuidiang.com/ood/metodologia/scrum.php>.

DA SILVA REINALDO O. ed. *Última actualización: 2000.* (). Disponible en: <http://definicion.de/eficiencia/>.

DELGADO PORRES, M.; REYES MATOS, E. Y. *Versión 2.0 del Sistema para la Gestión de la Información asociada a las Investigaciones y Postgrado*. Tutor: Ing. Delvis Ricardo Álvarez. . Universidad de las Ciencias Informáticas, 2011.

DENIS VALDIVIA, J. A.. *Ministerio de Finanzas y Precios de Cuba - EcuRed* . ed. Última actualización: 2011. (). Disponible en: http://www.ecured.cu/index.php/Ministerio_de_Finanzas_y_Precios_de_Cuba.

EGUÍLUZ, J.. *¿Qué es Symfony? | Symfony.es* . ed. Última actualización: 2012. (). Disponible en: <http://www.symfony.es/que-es-symfony>.

ELLMANN ENRIQUE. *Confianza*. de. *Última Actualización: 2008.* (). Disponible en <http://es.scribd.com/doc/88792227/confianza>.

FUENTES PEREZ HECTOR. *JUnit: Manual Básico* de. *Última Actualización: 2011* (). Disponible en: <http://fuenteperez.es/blog/junit-manual-basico>.

GARCIA SANTILLAN ARTURO; LAMADRID LANDA VICTOR ALBERTO. *Diseño e Implementación de un sistema del Programa operativo anual SISPOA*. de. *Última Actualización: 2000* (). Disponible en <http://www.eumed.net/libros/2007c/332/Confianza.htm>

GARCÍA SANTOS, N. *Crean software para control de multas - Cuba - Juventud Rebelde - Diario de la juventud cubana*. ed. Última actualización: 2009. (). Disponible en: <http://www.juventudrebelde.cu/cuba/2009-10-03/crean-software-para-control-de-multas/>.

GÓMEZ, A. M. eumed.net. [En Línea]. 04,2010. [Citado: 18 01, 2012.] <http://www.eumed.net/rev/cccss/08/bgsv.htm>.

GONZÁLES PINO, M.. *Eclipse, entorno de desarrollo integrado - EcuRed* . ed. Última actualización: 2010. (). Disponible en: http://www.ecured.cu/index.php/Eclipse,_entorno_de_desarrollo_integrado.

GONZÁLEZ ALVAREZ, L.; RIVERO ÁLVAREZ, T. et al.. *Gestión del subsistema Activos Fijos Tangibles del sistema Cedrux*. ed. publicado el: 2009 última actualización: 2009.

IDALBERTO, CHIAVENATO. 2006. *Introducción a la Teoría General de la Administración*. Séptima. s.l.: McGraw-Hill Interamericana, 2006. pág. 110.

LASTRE REYNALDO, Reyvis.E. Análisis y diseño del Módulo de Reportes para el portal de la UJC en la UCI. Ing. Susana Reina Consuegra. Universidad de las Ciencias en Informática de la Ciudad Habana, 2008.

LAUDON. *Sistemas De Información Para Gestión Y Toma De Decisiones*. [En Línea] 2007 [Citado: 01 02, 2012.] <http://www.mitecnologico.com/Main/SistemasDeInformacionParaGestionYTomaDeDecisiones>.

LEVISY. Gestión del Conocimiento. ed. Última actualización: 2011. ().
Disponible en: <http://www.buenastareas.com/ensayos/Gestion-Del-Conocimiento/1524762.html>

LÓPEZ, J. M.. *RapidSVN - Descargar* . ed. Última actualización: 2010. ().
Disponible en: <http://rapidsvn.softonic.com/>.

LUCIANO. *Entornos de Desarrollo Integrado para Java* . ed. Última actualización: 2010. (). Disponible en: <http://luauf.com/2008/05/13/entornos-de-desarrollo-integrado-para-java/>.

OSCAR. *Sistemas-y-diseño-de-sistemas*. ed. Última actualización: 2011. ().
Disponible en: <http://oscaring.bligoo.com.mx/unidad-2-sistemas-y-diseno-de-sistemas>

PEÑALVER ROMERO, G. .. *SXP, METODOLOGÍA ÁGIL PARA EL DESARROLLO DE SOFTWARE*. ed. publicado el: 2008 , última actualización: 2008.

PÉREZ ESCALONA; BERTOT VIEITO. *Contribución a la conservación del Patrimonio Cultural en Cuba a través de la producción de software* . ed. Última actualización: 2009. (). Disponible en: http://www.informatica-juridica.com/trabajos/Contribucion_conservacion_Patrimonio_Cultural_en_Cuba_a_traves_produccion_software.asp.

RODRIGUEZ JORGE. *Pruebas unitarias*. de. 'Ulmina Actualización 2006' ().
Disponible en <http://blog.continuum.cl/wp-content/uploads/2008/08/pruebas-unitarias.pdf>

SAVIGNE DARROMAN Y LEYVA VELAZQUEZ. "El proceso de gestión y la gestión económica en las empresas" en Observatorio de la Economía Latinoamericana, N° 145, 2011. Texto completo en <http://www.eumed.net/cursecon/ecolat/cu/2011/>

SOLANO OBDULIO. Tomado de: Newsletter "Inside Industry", *Publicación de Industry Sector, Siemens Colombia. Ed. 3, Abril de 2010.*

TECNI-SOFT. *MULTAPOL Gestión de Multas de Tráfico.* . ed. Última actualización: 2010. (). Disponible en: <http://www.pol-es.com/multapol.html>.

KARL FOGEL. *Producir Software de Código Abierto.* ed. publicado el: 2007 , última actualización: 2007.

VAN GIGCH JOHN P. Teoría general de sistemas, Págs 16 , 17.ed. Última actualización: 2010. (). Disponible en:<http://rub212ingdesistemas.blogspot.com/2011/09/unidad-i-teoria-general-de-sistemas-su.html>

VERGARA, G. *¿Que es un SISTEMA de GESTION? | Blog de MejoraTuGestion* . ed. Última actualización: 2009. (). Disponible en: <http://mejoratugestion.com/mejora-tu-gestion/que-es-un-sistema-de-gestion/>.

Módulo Multa y Patrimonio

BIBLIOGRAFÍA

ANGGEW. *Sistemas Operativos I - Monografias.com* . ed. Última actualización: 2000. (). Disponible en: <http://www.monografias.com/trabajos5/sisop/sisop.shtml>.

ARÉCHIGA ROCHA, J. L.. *Información JAVA* . ed. Última actualización: 2007. (). Disponible en: <http://arechiga.50megs.com/tpoo2/javah1.html>.

BARRA, R.. *Qué es Ubuntu, por qué usarlo y cómo probarlo | Placer Digital* . ed. Última actualización: 2011. (). Disponible en: <http://placerdigital.net/que-es-ubuntu-por-que-usarlo-y-como-probarlo/>.

EGUÍLUZ, J.. *¿Qué es Symfony? | Symfony.es* . ed. Última actualización: 2012. (). Disponible en: <http://www.symfony.es/que-es-symfony/>.

FLORENCIA. *Definición de Gestión » Concepto en Definición ABC* . ed. Última actualización: . (). Disponible en: <http://www.definicionabc.com/general/gestion.php>.

GARCIA, L.. *Sistema de control de versiones: subversion* . ed. Última actualización:2008. (). Disponible en: <http://recursostic.educacion.es/observatorio/web/es/software/software-general/548-luis-garcia>.

GRACÍA JEREZ, A.. *Gestores de repositorios Maven* . ed. Última actualización: 2012. (). Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=repositorios->

maven.

MANGRAR. *Gestores de repositorios Maven* . ed. Última actualización: 2011. (). Disponible en: <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=repositorios-maven>.

MORA SOTO, J. A.. *Herramientas UML... ¿cuál utilizar?* | Jose Arturo Mora Soto . ed. Última actualización: 2010. ().

Disponible en: <http://www.jams.name/2010/04/18/herramientas-uml-cual-utilizar/>.

PEREZ, O.. *Historia de la Informática en Cuba - EcuRed* . ed. Última actualización: 2010. (). Disponible en: http://www.ecured.cu/index.php/Historia_de_la_Inform%C3%A1tica_en_Cuba#La_UCI.

PÉREZ VALDÉS, D.. *Los Frameworks de PHP agilizan tu trabajo* . ed. Última actualización: 2007. (). Disponible en: <http://www.maestrosdelweb.com/editorial/los-frameworks-de-php-agilizan-tu-trabajo/>.

VERGARA, G. *¿Que es un SISTEMA de GESTION?* | Blog de MejoraTuGestion . ed. Última actualización: 2009. (). Disponible en: <http://mejoratugestion.com/mejora-tu-gestion/que-es-un-sistema-de-gestion/>.

jWebSocket - The Open Source Java WebSocket Server .(). Disponible en: <http://jwebsocket.org/>.

Programas para inmobiliarias Software de gestion de inmobiliarias Software para inmobiliarias . ed. Última actualización: 2012. (). Disponible en: <http://www.inmocomputer.com/>.

Sistemas Gestion de Inmuebles | Sistema de gestion para Gestion de Inmuebles | Sistemas ERP Gestión de Inmuebles Solmicro-eXpertis . ed. Última actualización: 2008. (). Disponible en: <http://www.solmicro.com/programas-gestion/programas-erp-gestion-inmuebles/modulos-erp-gestion-inmuebles.html>.