



Facultad Regional “Mártires de Artemisa”

Título: Módulo energía para el servidor del Sistema Informativo de la Administración Provincial de Artemisa.

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

Autor: Dunaikelys Mesa Martínez

Tutor: Ing. Frank Rosales Muñoz

Cotutora: MSc. Rita M. Concepción Bravo

Artemisa, Junio 2012

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

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

Dunaikelys Mesa Martínez

MSc. Rita M. Concepción Bravo

Ing. Frank Rosales Muñoz

“Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber”

Albert Einstein.

A mis tutores Rita y Frank, por guiarme y orientarme durante el desarrollo del trabajo de diploma. A Jose, por sus consejos.

A mis amigos que estuvieron en los momentos de alegrías, pero también de tristezas dándome ánimo y apoyo: Eleidys, Regla, Evelyn, Dunia, Cuni, y Danelys.

A todos aquellos profesores que colaboraron de una manera u otra en mi formación profesional y en la confección de este trabajo.

A Liliana mi compañera de tesis por todos los momentos difíciles que hemos pasado juntas en la confección de este proyecto.

A toda mi familia en especial a mi hermano Iovanys a mi abuela Marina y a mis tíos.

A mis papitos del alma, por estar siempre allí con ese amor tan grande que enternece cualquier corazón. A ellos le debo todo lo que soy.

A las personas que se han interesado en saber cómo iba la tesis, a los que de una forma u otra ayudaron a terminarla, a los que me han ayudado y apoyado en los cinco años de universidad y a lo largo de mi vida, a todos:

¡MUCHASGRACIAS!

Le dedico este trabajo de diploma a las dos personas más importantes de mi vida: Nancy y Diosdado, mis padres.

A mi mamá, por ese amor incondicional que solo una madre puede dar, por su apoyo, por su paciencia con mis malcriadeces, por sacrificarse tanto para que no me falte nada.

A mi papá, por ese amor tan grande que pasa por encima de cualquier cosa, por el gran esfuerzo que ha realizado durante los cinco años de mi carrera para complacerme en todo, por creer siempre en mí y darme las fuerzas necesarias para soportar el dolor de solo verlo una vez al año.

Por guiarme con comprensión y cariño a cumplir mis sueños, que se los entrego porque son de ustedes también.

Los procesos de gestión de la información ocupan, cada vez más, un espacio mayor en la economía de los países a nivel mundial resultando un aspecto estratégico para las instituciones que se introducen en el entorno actual y aceptan las nuevas tecnologías. Su utilización permite un mejoramiento en la consulta y modificación de la información.

Como parte de la necesidad que tiene la Administración Provincial de Artemisa de automatizar sus procesos, la “Facultad Regional Mártires de Artemisa” tiene entre sus proyectos de investigación el desarrollo del Sistema Informativo de la Administración Provincial (SINAP) de Artemisa.

El módulo energía para el servidor del SINAP tiene como objetivo mejorar el proceso de gestión de la información del Departamento Independiente de Energía de la Administración Provincial de Artemisa, trayendo consigo mayor rapidez en la búsqueda de la información.

Para guiar el desarrollo del módulo se utilizó la metodología SXP, empleándose diferentes herramientas y tecnologías. El módulo fue sometido a valoración y avalado por especialistas del centro de desarrollo de la Facultad Regional “Mártires de Artemisa”, los cuales determinaron su calidad.

El módulo energía permite controlar la información de energía eléctrica que se maneja en el Departamento Independiente de Energía de la Administración Provincial de Artemisa insertando a los profesionales en las más modernas prácticas tecnológicas de estos tiempos.

Índice

Índice	7
Introducción.....	1
Capítulo 1: Fundamentos teórico-metodológicos de los procesos de gestión de información.....	8
1.1 Información y datos	8
1.2 La gestión de la información por procesos.....	9
1.2.1 Componentes para la gestión de la información.....	10
1.3 Los módulos en los sistemas informáticos.....	12
1.3.1 Sistemas informáticos de gestión de información. Soluciones internacionales.....	13
1.3.2 Sistemas informáticos de gestión de información. Soluciones nacionales	15
1.4 Metodologías de desarrollo de software	16
1.5 Herramientas y tecnologías	21
1.5.1 Lenguajes empleados para el desarrollo	25
1.5.2 Framework para el desarrollo.....	26
Conclusiones del Capítulo	28
Capítulo 2: Características, análisis y diseño del módulo	29
2.1 Concepción del sistema	29
2.2 Procesos y flujos de información del Departamento.	29
2.2.1 Planificación del proyecto por roles.....	33
2.3 Modelo de Dominio	33
2.4 Captura de requisitos	34
2.5 Lista de riesgos	39
2.6 Diseño con metáforas	40
2.7 Tareas de ingeniería	43

2.8 Plan de Releases	46
Conclusiones del Capítulo	47
Capítulo 3: Validación del módulo.	48
3.1 Casos de pruebas	48
3.1.2 Caso de prueba unitaria para la clase servicio Plan del Consumo Energético.	48
3.2 Pruebas Unitarias automatizadas con JUnit.	53
3.2.1 Resultado de las pruebas con JUnit	53
3.3 Resultado obtenido.....	56
3.4 Funcionalidades Obtenidas	56
3.5 Aporte Práctico	56
Conclusiones del Capítulo	57
Conclusiones generales	58
Recomendaciones	59
Referencias Bibliográficas	60
Bibliografía	64
Anexo	69
Glosario de Términos	82

Introducción

Desde los inicios del hombre existió la información y con ella la necesidad de comunicarla y transferirla. Se idearon diferentes métodos para hacer perdurar los conocimientos aprovechando las herramientas y el entorno.

Con el devenir de los siglos, durante la última revolución industrial, impulsada por el amplio desarrollo de la ciencia y la tecnología, surgió la necesidad de que se actuara sobre la información para su control. La información se convirtió en un recurso valioso en manos de los agentes económicos y por la importancia que adquirió se comenzó a gestionar como recurso.

La gestión de la información ocupa, cada vez más, un espacio mayor en la economía de los países a nivel mundial resultando un aspecto estratégico para las instituciones que se introducen en el entorno actual y aceptan las nuevas tecnologías. Son muchas las entidades que recurren al desarrollo de espacios e infraestructuras para disponer de su propia información, distribuir sus recursos y poseer canales de comunicación rápidos y eficientes que colaboren con el desarrollo del trabajo y la toma de decisiones.

Existe hoy una mayor integración en la gestión de la información debido al trabajo constante de las empresas cuyo soporte es la red. Nunca antes se lograron estas ventajas en las empresas debido a la escasez de una tecnología integradora. La era de la información y del conocimiento arrastra a la empresa tradicional a una nueva dinámica de trabajo.

El desarrollo avanzado de la tecnología, acompañado de la progresiva industria del software y la incorporación de coherentes sistemas para la gestión de información, aportan novedosas soluciones para favorecer valores a los denominados recursos intangibles, aumentar tácticas de administración y fortificar niveles de eficiencia. “La elección de tecnología es una de las elecciones más decisivas a las que se enfrenta cualquier país, grande o pequeño, rico o pobre. Es una elección con una influencia generalizada. Determina lo que se produce, cómo y dónde es producido...” (McRobie, 2011)

La informatización de la sociedad en Cuba es uno de los puntos con mayor prioridad en la dirección del Partido y el Estado; tarea en la que están

comprometidos, en mayor o menor medida, todas las instituciones a nivel nacional. Se hace todo lo posible por corresponder al Programa Rector que persigue dar publicidad al uso masivo de la tecnología de la información a escala nacional.

En el 2007 entró en vigor el Decreto-Ley No.252 “Sobre la continuidad y el fortalecimiento del sistema de dirección y gestión empresarial cubano”, que aprobó las bases generales para la gestión de la información y el perfeccionamiento empresarial, con vistas a su aplicación progresiva en la economía nacional. Se propuso, de manera ordenada, la realización de las transformaciones necesarias en la gestión empresarial en el país; así como de las entidades gubernamentales encargadas de su supervisión, donde se plantea: “La innovación tecnológica y la generalización de los logros de la ciencia y la técnica, son elementos básicos en la elevación de la eficiencia y en el incremento de los aportes a la sociedad socialista (...)”. (Decreto ley No. 252, 2007)

Para innovar en temas de gestión de la información el estado cubano se apoya, entre otras entidades, en instituciones universitarias tales como el Instituto Superior Politécnico "José A. Echeverría" (CUJAE), Universidad de la Habana (UH), Universidad de las Ciencias Informáticas (UCI). Esta última constituye el pilar fundamental del avance tecnológico en el país, es líder en productos de software y soluciones informáticas. Es centro de referencia nacional de calidad de software. Destacada en el desarrollo científico técnico en informática, cuenta con una red de centros de investigación, desarrollo e innovación a nivel nacional.

Las facultades regionales de la UCI deben cumplir funciones de producción-investigación para resolver problemas locales, de ahí que a partir de la insuficiente modernización tecnológica en las nuevas y reordenadas entidades estatales de occidente, puedan trabajar la UCI y el Gobierno, en conjunto, para cumplir las políticas de desarrollo fijadas por el país en materia de gestión de la información.

Como parte de la necesidad que tiene la Administración Provincial de Artemisa de automatizar sus procesos, la “Facultad Regional Mártires de Artemisa” tiene

entre sus proyectos de investigación el desarrollo del Sistema Informativo de la Administración Provincial (SINAP) de Artemisa.

El Gobierno Provincial de Artemisa, institución donde se desarrolla esta investigación, surgida el 1ro de enero del año 2011 con la nueva distribución política administrativa, aprobada por la Asamblea Nacional del Poder Popular, cuenta con 32 direcciones. Dentro de la “Dirección de Economía” se encuentra el “Departamento Independiente de Energía” que se encarga de gestionar el cumplimiento de las políticas, normas y disposiciones legales establecidas por el Estado, respecto al uso racional y eficiente de la energía y los portadores energéticos en el territorio en la provincia de Artemisa.

La gestión de información que se realiza en este departamento es costosa en cuanto a tiempo y esfuerzo, debido a que el Jefe de Regulación debe llenar todos los modelos de forma manual.

En el Departamento Independiente de Energía no existe una adecuada forma de almacenar y recuperar información. Se dificulta el análisis de los reportes y los datos se presentan en formato “Excel”. Debido a los problemas de organización y seguridad del “Departamento Independiente de Energía”, personas no autorizadas pueden tener acceso a información clasificada pudiendo modificar o borrar cualquier contenido de los documentos.

Cuando se necesita una información urgente resulta engorroso buscar de forma manual y dentro de grandes volúmenes de datos. La información con la que se toman las decisiones muchas veces no se encuentra centralizada en una misma computadora lo que resulta riesgoso ante cualquier fallo de hardware.

Esta actual gestión de la información ocasiona pérdidas y duplicado de los datos, problemas de organización y de seguridad con respecto a los documentos que estos generan tales como: el “Consumo en Barras Diario” y “Consumo del Sector Estatal”. Trayendo consigo que la información no esté íntegra, haya mala calidad en los reportes, poca confiabilidad en la documentación, y retrasos en la entrega de información actualizada al presidente del gobierno para la toma de decisiones en la provincia. Estos elementos se traducen en baja eficiencia y confiabilidad en el trabajo con la información.

A partir de la situación problemática descrita se plantea el siguiente **problema de investigación**: ¿Cómo contribuir a mejorar la confiabilidad y eficiencia del proceso de gestión de la información en el Departamento Independiente de Energía de la Administración Provincial de Artemisa?

Definiendo como **objeto de estudio**: el proceso de gestión de la información.

Como **campo de acción**: el procesamiento de los datos en el proceso de gestión de la información

Se plantea como **objetivo general**: Desarrollar un módulo para el servidor del SINAP que contribuya a mejorar la confiabilidad y eficiencia del proceso de gestión de la información en el Departamento Independiente de Energía de la Administración Provincial de Artemisa.

Idea a Defender: el desarrollo de un módulo para el servidor del SINAP contribuirá a mejorar la confiabilidad y eficiencia del proceso de gestión de información en el Departamento Independiente de Energía de la Administración Provincial de Artemisa.

Se plantean las siguientes **tareas de la investigación**:

- Establecimiento de los fundamentos teórico-metodológicos para el desarrollo de los procesos de gestión de información.
- Caracterización el proceso de gestión de la información en el Departamento Independiente de Energía de la Administración Provincial de Artemisa.
- Desarrollo del módulo energía para el servidor del Sistema Informativo de la Administración Provincial de Artemisa.
- Validación del módulo para el servidor del SINAP mediante pruebas unitarias.

Para la realización de esta investigación se utilizaron diferentes métodos científicos.

Métodos teóricos

- Analítico-Sintético: Empleado un estudio de las tecnologías, metodologías y herramientas posibles a ser utilizadas en el desarrollo del módulo propuesto.
- Histórico-Lógico: Empleado para estudiar la evolución y desarrollo de los sistemas de gestión y comprender lógicamente cuales son las tendencias actuales.
- Modelación: Aplicado para mostrar los diferentes diagramas que se construyen como resultado del proceso de ingeniería de software.
- Enfoque de sistema: Este método permitió el estudio de los fenómenos en su integridad: cliente - servidor - base de datos, formada por componentes relacionados entre sí y con el medio, lo que manifiesta su dinámica y desarrollo.

Métodos empíricos

- Entrevista: Utilizado para entrevistar al personal del Departamento Independiente de Energía de la Administración Provincial de Artemisa, con el objetivo de obtener información para el desarrollo del producto.(Ver anexo 1)
- Análisis documental: Utilizado para sistematizar la bibliografía y los documentos metodológicos disponibles, posibilitando el acercamiento a buen número de referentes teóricos, en su mayoría de la actual década.

Método Estadístico - matemático

- Análisis porcentual: Contribuyen a determinar la muestra de sujetos a estudiar, tabular los datos empíricos obtenidos desde la recolección hasta su organización, análisis e interpretación y establecer las generalizaciones apropiadas a partir de ellos.

Población tomada para la investigación: Seis especialistas que representan un 100% del Departamento Independiente de Energía.

Muestra tomada para la investigación: está compuesta por el 100% de los trabajadores del Departamento Independiente de Energía. Se entrevistaron a los especialistas y dirigentes de este departamento, la técnica que se utilizó fue el

Muestreo Intencional.

Declaración de Variables

Variable independiente: módulo para el servidor del SINAP.

Variables dependientes: eficiencia, confiabilidad.

Actualidad: La gestión de la información actualmente es muy utilizada en las organizaciones ya sea de forma manual o informatizada. Muchas empresas están en busca de lograr una mayor optimización de los procesos de gestión que sean capaces de responder a sus necesidades actuales.

Necesidad: El volumen de información que manejan las organizaciones cada día aumenta considerablemente y es por este motivo que demora la obtención de la información. Las organizaciones se desenvuelven en un escenario cada vez más complejo donde surge la necesidad de lograr un entorno informatizado que agilice el proceso de gestión de manera segura.

El **aporte práctico** de la presente investigación consiste en el desarrollo de un módulo para el servidor del SINAP que permita mejorar el proceso de gestión de la información del Departamento Independiente de Energía de la Administración Provincial de Artemisa.

El presente Trabajo de Diploma está estructurado en:

Capítulo 1: Fundamentos teórico-metodológicos de los procesos de gestión de información

Comprende los conceptos generales y básicos que permiten el entendimiento de temas relacionados con los procesos de gestión de la información. Se precisa el estado del arte en el ámbito nacional e internacional, así como el análisis de las tendencias del uso de las diferentes herramientas informáticas que pueden ser empleadas.

Capítulo 2: Características, análisis y diseño del módulo energía.

Se realiza una propuesta de solución, se describe cómo debe funcionar y se destaca sus características distintivas; se especifican los requisitos funcionales y no funcionales.

Capítulo 3: Validación del módulo.

En este capítulo se plasman los casos de pruebas o test de aceptación a las que fue sometida la aplicación en cada una de las iteraciones. Se exponen los resultados obtenidos y se muestran las funcionalidades alcanzadas en el período de desarrollo.

Este trabajo cuenta con un Resumen, Introducción, Capítulos, seguido de Conclusiones, Recomendaciones, Bibliografía, Glosario de Términos y Anexos.

Capítulo 1: Fundamentos teórico-metodológicos de los procesos de gestión de información.

En el presente capítulo se precisan los conceptos generales que permiten la mejor comprensión de temas relacionados con la gestión de la información, así como las aplicaciones que presentan relación con la investigación a nivel mundial y nacional. Se abordan las tecnologías y metodología necesarias para el correcto desarrollo de la propuesta de solución.

1.1 Información y datos

La Ciencia de la Información está interesada en un conjunto de conocimientos relacionados con el origen, colección, organización, almacenamiento, recuperación, interpretación, transmisión, transformación y utilización de la información. Incluye la investigación de las representaciones de información en los sistemas naturales y artificiales, la utilización de códigos para la transmisión eficiente del mensaje, el estudio de instrumentos y técnicas de procesamiento de la información, tales como computadoras y sistemas de programación.

Existen muchos autores que han expuesto su criterio acerca de la información. La definen como “un conjunto de datos acerca de algún suceso, hecho, fenómeno o situación, que organizados en un contexto determinado, tienen su significado y cuyo propósito puede ser el de reducir la incertidumbre o incrementar el conocimiento acerca de algo”. (Thompson, 2008)

Un análisis importante lo realizan Davenport y Prusack al decir que la información “son datos dotados de pertinencia y propósito, con un bien marcado carácter contextual y de lectura del entorno”. (Davenport y Prusack, 2001)

Los datos expuestos fríamente sin elaboración de significado no aportan nada a la toma de decisiones, constituyendo solamente una representación simbólica.

Se parte del último concepto para plantear que la información es un conjunto de datos que adquieren significado, importancia y propósito; en su ciclo de vida pasa por fases de generación, reproducción, distribución, adquisición, procesamiento, almacenamiento y asimilación.

Eficiencia y Confiabilidad

Una organización es eficiente si está estructurada para ayudar al logro de los objetivos de la empresa con un mínimo de consecuencias o costos no deseados. La eficiencia está vinculada a utilizar los medios disponibles de manera racional para llegar a una meta. Se trata de la capacidad de alcanzar un objetivo fijado con anterioridad en el menor tiempo posible y obtener los mayores resultados con la mínima inversión. (Robbins y Coulter, 2005)

La confiabilidad se refiere a la seguridad y estabilidad en los resultados al realizar un proceso de obtención de datos garantizando su consistencia, permitiendo una correcta realización de las operaciones y mantenimiento de las funcionalidades previstas, en un tiempo determinado y bajo condiciones específicas.

1.2 La gestión de la información por procesos

La gestión de la información puede considerarse, tal como lo definen Rodríguez y Domínguez, “como aquel proceso que se encarga de gestionar la información necesaria para la toma de decisiones y un mejor funcionamiento de los procesos, productos y servicios de la organización” (Rodríguez y Domínguez, 2007).

Según Ponjuán, un proceso puede definirse como un “... símbolo de actividad. Quiere esto decir que todo proceso implica una actividad, una transferencia, un movimiento, un cambio” (Ponjuán, 2005). Los procesos de la gestión de información se manifiestan de diferentes formas en las organizaciones y pueden identificarse por medio de su funcionamiento, objetivos y estrategias.

En esta investigación se asume el concepto planteado en la Norma ISO 9000:2000, en su apartado 3.4.1, que define un proceso como el “conjunto de actividades mutuamente relacionadas o que interactúan para transformar elementos de entrada en resultados”. (Norma Internacional ISO 9000:200). Luego, puede añadirse que los elementos de entrada para un proceso son generalmente el resultado de otros procesos.

Cada proceso tiene un punto de inicio (entradas) y un punto final (salidas) en los que se puede producir una relación entre el sistema y el ambiente con otros procesos unidos con determinadas áreas de la organización, sean

departamentos o grupos de trabajo. "Las organizaciones son tan eficientes como lo son sus procesos." (Navarro, 2002)

1.2.1 Componentes para la gestión de la información

Un análisis de la literatura consultada permitió identificar cuatro componentes importantes del enfoque por procesos en relación a la gestión de la información; estos son: los procesos, las entradas, las salidas y los flujos de información. A continuación, se define cada uno de ellos:

- **Procesos:** Son cada una de las acciones que intervienen y se interrelacionan en el sistema y que permiten la evolución del ciclo de vida de la información, donde las entradas a un proceso del sistema pueden constituir la salida de otro y a la inversa.
- **Entradas:** Se definen por las necesidades de las personas y las fuentes de información procedentes, tanto internas como externas.
- **Salidas:** Constituyen la conclusión del ciclo de vida de la información, posibilitan disponer de productos y servicios de información con valor añadido y deben garantizar la satisfacción de las necesidades de la comunidad de usuarios a la que se vincula el sistema con las exigencias de calidad que ellos demandan o necesitan.
- **Flujo de información:** Es el tránsito de la información, desde las entradas por cada uno de los procesos, hasta las salidas. En el paso de la información, desde las entradas a las salidas, intervienen una serie de procesos ordenados que se relacionan estrechamente por medio de diversos flujos, con vista a que el usuario obtenga una nueva información de valor añadido.

Cualquiera de estos cuatro componentes se vincula con diversos recursos: humanos, físicos, materiales y tecnológicos (hardware y software).

La habilidad de la organización para identificar y manejar los flujos de información le permite, en gran medida, la eficiente utilización de los recursos y le concede fortalezas para elevar la confiabilidad de sus procesos, satisfacer las necesidades internas y externas, así como la posibilidad de prepararse para enfrentar los cambios que se presenten. "La incorporación de las Nuevas

Tecnologías de la Información permite redefinir los procesos alcanzando grados de eficacia y eficiencia inimaginables.” (Navarro b, 2002).

Las tecnologías para gestionar flujos de información

Las instituciones deben recopilar y organizar información de modo que se pueda acceder a ella fácilmente y garantizar que cualquier usuario pueda disponer de los datos necesarios cuando realmente los requiera, lo que se conoce como “valor de oportunidad”.(Lorenzo, 2005). Las instituciones deben promocionar el uso correcto de los recursos informativos disponibles y con ello intentar que los usuarios puedan ser cada vez más independientes a la hora de buscar información.

Es por esto que muchas instituciones se apoyan en las tecnologías para gestionar sus flujos de información utilizando sistemas informáticos, definiéndose estos como: “un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar la toma de decisiones y el control de una organización” (Emery,1990)

Los sistemas informáticos están compuestos por elementos lógicos, físicos y humanos que se relacionan entre sí y realizan procesos con un fin específico dentro de la organización donde se encuentran.

- Componente físico: el hardware del sistema informático, los ordenadores, los periféricos y el sistema de comunicaciones.
- Componente lógico: el software del sistema informático (los programas, las estructuras de datos y la documentación asociada).
- Componente humano: todas las personas participantes en las fases de la vida de un sistema informático (diseño, desarrollo, implantación, explotación).

El software se encuentra distribuido en el hardware y lleva a cabo el proceso lógico que requieren los datos mientras que el componente humano determina el alcance de los medios empleados y su utilidad futura.

Los sistemas informáticos se adaptan al entorno que les rodea, por lo que para su implantación se requiere analizar tanto los datos como el contexto, es decir,

realizar un estudio de las necesidades y usos de información de los “clientes” a los va dirigido este sistema. La complejidad de un sistema informático está dada por el proceso de gestión de la información al cual se desea aplicar; por esta causa los sistemas informáticos se dividen en módulos, para adaptarse a los flujos de información de cada proceso.

1.3 Los módulos en los sistemas informáticos

En la actualidad los sistemas informáticos se han desarrollado y poseen más funcionalidades; son más grandes y difíciles de implementar por lo que se hace conveniente dividir el programa en pequeños pedazos llamados módulos.

En programación, un módulo es un fragmento de un programa que se desarrolla de forma independiente del resto. Esta independencia hace posible un mecanismo de compilación por separado que limita la complejidad del programa que se está desarrollando.

Al implementar el módulo por separado la persona que lo desarrolla solo debe preocuparse de él, prescindiendo en parte de cómo se utiliza este módulo dentro del programa. Quien escriba el resto del programa no debe preocuparse de los detalles del módulo sino solo de cómo utilizarlo. Los módulos dentro de un programa pueden llegar a realizar varias tareas o funcionalidades. (Visbal, 2009)

Los módulos suelen estar (aunque no necesariamente) organizados jerárquicamente en niveles de forma que haya un módulo principal que realice las llamadas oportunas a los módulos de nivel inferior. Cada uno de los módulos de un programa, idealmente, debería cumplir las siguientes características:

- Tamaño relativamente pequeño: Esto facilita aislar el impacto que pueda tener la realización de un cambio en el programa, ya sea para corregir un error, para el rediseño del algoritmo correspondiente.
- Independencia modular: Cuanto más independientes son los módulos entre sí más fácil y flexiblemente se trabajará con ellos, esto implica que para desarrollar un módulo no es necesario conocer detalles internos de otros módulos.

La independencia modular mejora el rendimiento humano y permite realización del programa en equipo o el desarrollo de módulos de forma paralela. También contribuye a la reutilización de software.

Servidores en los sistemas informáticos

Un servidor es todo proceso que proporciona un servicio a otros siendo “el encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él; (...) normalmente maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos.” (Wiley, 1993)

El servidor es un programa que recibe una solicitud, realiza el servicio requerido y devuelve los resultados en forma de una respuesta. Generalmente puede tratar múltiples peticiones (múltiples clientes) al mismo tiempo.

1.3.1 Sistemas informáticos de gestión de información. Soluciones internacionales.

ContaLuzWeb: Es un sistema de apoyo a los encargados públicos en el proceso de control y toma de decisiones relativas al consumo y despacho de electricidad. Se encuentra implantado en la Universidad de San Pablo y se tornó una herramienta imprescindible para reducir las despensas y el consumo de energía eléctrica. Consta de una estructura de n-capas donde se encuentra una interfaz de usuario, un servidor donde se desarrolla la lógica del negocio y una base de datos para guardar información relevante.

Muestra una interface de entrada para las informaciones contenidas en una factura de energía eléctrica. A través de “ContaLuzWeb” la Universidad puede controlar que unidades consumidoras están sujetas a las tarifas más elevadas por consumo de exceso de energía. La base de datos generada por el histórico de informaciones almacenadas en el “ContaLuzWeb” permite al funcionario prever el crecimiento futuro del consumo de energía y de demanda.

Para su confección se estudió la lógica de negocio y fueron evaluados cuales serían los informes y cuáles las informaciones que podrían ser relevantes para la toma de decisiones en el proceso de la gestión de energía. Después de la

entrada de datos, el sistema trabaja con las informaciones y genera informes gerenciales para el usuario.

Se utilizó para la implementación un software libre (lenguaje JAVA con base de datos MySQL), dada la necesidad de mayor robustez de la solución debido al volumen de cuentas. (Saidel y Rossi y Galvão; 2008)

SENTRON powermanager V 2.0: Procesa y controla de forma fiable y sencilla los datos recogidos. El paquete estándar del SENTRON powermanager cubre los requerimientos básicos más usuales. Con este paquete los usuarios reciben un software de gestión de energía fácil de usar que permite detectar datos, presentarlos, archivarlos y procesarlos en una base de datos integrada, a través de los instrumentos de medida.

La opción "Web" se ofrece para satisfacer las demandas en relación a presentaciones web del mercado. El servidor de SENTRON powermanager está diseñado para la recogida, control, evaluación, entrega y registro de datos energéticos.

Objetivos generales del sistema de gestión de energía:

- Identificar posibles ahorros.
- Reducir costes de energía.
- Asegurarse de la disponibilidad de energía. (SiEMES., 2012)

Business News Americas (Bnamericas): Es un sistema informático de gestión de información que entrega y produce en tiempo real reportes, datos estadísticos, perfiles de empresas, proyectos, eventos y noticias diarias de la región, cubriendo 12 sectores industriales: que conforman los diferentes módulos por los que está compuesto el sistema.

- Telecomunicaciones / Tecnologías de la Información
- Banca / Seguros / Privatización / Metales
- Minería Residuos / Infraestructura
- Aguas y Residuos / Infraestructura

- Energía Eléctrica / Petróleo y Gas / Petroquímicos

La lógica de negocio de este sistema está centrada en la búsqueda de noticias, perfiles de compañías, análisis de la industria, eventos regionales, entrevistas, columnas de opinión e informes externos en cualquier fecha, sector, actividad, país y compañía. (Jímenez y Umaña, 2011)

1.3.2 Sistemas informáticos de gestión de información. Soluciones nacionales

Inspección Estatal Energética: Esta basado en una plataforma web que tiene como razón social la de realizar diagnósticos energéticos a entidades y organizaciones cubanas apoyadas en los grupos territoriales existentes a lo largo del país que radican en la Direcciones Provinciales de Planificación y el Grupo Nacional que se encuentra en el Ministerio de Economía y Planificación, los cuales, con ayuda de instrumentos portátiles de medición sofisticados y programas técnicos de computación orientados al perfil energético, dictaminan las posibilidades de ahorro existentes y las recomendaciones para poderlas llevar a cabo.

Gestiona los boletines técnicos de años anteriores, refleja diagnósticos del comportamiento energético de las fábricas de cemento cubanas durante el decenio de 1989 a 1999, muestra análisis de los indicadores energéticos y medioambientales comparándolos con tecnologías existentes en el mundo y permite la petición de diagnósticos de los usuarios vía on-line.

Redenerg:

Es el portal de la red del Sistema Nacional de Información de la Energía (SNIE) en Cuba. Constituye elemento de primera prioridad para garantizar la distribución de la información que sobre Energía se genera tanto por instituciones nacionales como extranjeras, dedicadas al desarrollo de este tema; lo cual ofrece el justo medio para la búsqueda rápida, eficaz y oportuna de la información de carácter relevante, como soporte a la toma de decisiones por directivos y especialistas del sector energético en Cuba y para potenciar la creación de una elevada cultura y conciencia energética en la población.

El diseño e instalación de la plataforma tecnológica para el soporte de la red ofrece una adecuada y eficaz visibilidad y accesibilidad de alcance nacional a los servicios de información, posibilitando la oportuna obtención y distribución de la información de carácter relevante, por parte de los proveedores y clientes respectivamente, al menor costo e independiente de su ubicación territorial. (González y Arencibia, 2005).

1.3.3 Resultado del estudio realizado a los sistemas informáticos de gestión de información

Son muchos los sistemas de gestión de información de energía eléctrica a nivel internacional, sin embargo en ocasiones resultan demasiado genéricos para ser implantados en una empresa con procesos específicos, sumándole a esto que la mayoría son consideradas software propietario por lo que habría que tener en cuenta el costo de obtención.

Las soluciones mencionadas a nivel nacional cumplen con las expectativas de los centros para los cuales fueron desarrollados, pero resultan muy específicos para ser implantados en una empresa con procesos distintos a los que estos manejan.

Después de haberse realizado un estudio de algunas soluciones informáticas para la gestión de información de energía eléctrica, se ha llegado a la conclusión de que los sistemas anteriores no cumplen con los requerimientos para su implantación en el Departamento Independiente de Energía de la Administración Provincial de Artemisa. Por lo que se justifica la creación de una solución informática para esta entidad.

Con el análisis realizado se evidencia que un sistema informático que gestione la información es una excelente herramienta de trabajo para cualquier empresa en la actualidad. Continuar la producción de sistemas que gestionen la información de las empresas cubanas es una necesidad.

1.4 Metodologías de desarrollo de software

Una metodología de desarrollo de software no es más que un conjunto de herramientas, técnicas, soporte documental y procedimientos a seguir para

desarrollar un software. Las metodologías deben especificar: las tareas que se llevan a cabo en cada etapa del ciclo de vida del proyecto; las restricciones que deben aplicarse, las técnicas y herramientas a emplear; y cómo se controla y gestiona un proyecto.

Las metodologías se podrían clasificar en dos grupos: las metodologías Ligeras o Ágiles y las metodologías Pesadas, estas últimas orientadas al control de los procesos, estableciendo rigurosamente las actividades a desarrollar, herramientas a utilizar y notaciones que se usarán; requieren una extensa documentación, ya que pretende prever todo de antemano.

Las metodologías Ligeras o Ágiles están orientadas a interactuar con el cliente y el desarrollo incremental del software, especialmente preparados para cambios durante el proyecto mostrando versiones parcialmente funcionales del software al cliente en intervalos cortos de tiempo, para que pueda evaluar y sugerir cambios en el producto según se va desarrollando (Canós y Letelier y Penadés, 2010)

En la actualidad existen disímiles metodologías, esto se debe a que los requisitos de un software a otro son variados y cambiantes, por lo que se aplica la que más se adecue a las expectativas en ocasiones del cliente o de la experiencia del equipo de trabajo.

Un factor fundamental a tener en cuenta para escoger una metodología de desarrollo de software es la complejidad del sistema a desarrollar, la cantidad de requisitos que deben ser implementados en el sistema y la cantidad de información que se maneja en el proceso.

Una de las metodologías pesadas más conocidas y utilizadas es la Metodología RUP (Rational Unified Process). RUP posee un proceso de desarrollo pesado basado en la documentación y la justificación de este proceso es que gracias a la mucha documentación se pueden reconocer los problemas y fallos de forma temprana y corregirlos. La dificultad de esta metodología radica en que debe aplicarse a equipos de trabajo grandes pues posee 32 roles y genera muchos artefactos finales. En equipos pequeños donde se deben distribuir los 32 roles significaría un incremento de tiempo y costo.

Otra de las más exitosas es SCRUM. Es una metodología de desarrollo de software para la autogestión de los equipos de trabajo. Entre sus ventajas se tiene que el equipo de producción trabaja en la misma dirección, con un objetivo claro, reuniéndose día tras días para recordar a los desarrolladores cual es la tarea de hoy, fijando objetivos a corto plazo, y dándole fin a los problemas que vayan surgiendo. Proporciona una visión esclarecedora del avance de las actividades y permite que se pueda evaluar diariamente el progreso del trabajo.

SCRUM plantea qué pasos deben seguir para hacer un software en equipo, de hecho, podría aplicarse a tareas que no son de software como la jerarquía. La mejor forma de utilizar SCRUM es complementándose con otras metodologías de desarrollo que sean más explícitas con respecto a las herramientas a utilizar. (Scrum. 2011)

XP (Programación Extrema) es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, según Calero XP “se basa en la simplicidad, la comunicación y el reciclado continuo de código, para algunos no es más que aplicar una pura lógica”. (Calero, 2011). Consiste en aplicar un conjunto de reglas estrictamente establecidas que promueven el trabajo en equipo, preocupándose en todo momento del aprendizaje del mismo y estableciendo un buen clima de trabajo.

La Programación Extrema es adecuada para proyectos con requisitos imprecisos, muy cambiantes y con un riesgo técnico excesivo. Se realimenta continuamente de las exigencias del clientes y las capacidades del equipo de desarrollo con una comunicación fluida entre todos los participantes, busca simplificar las soluciones implementadas y coraje para los múltiples cambios.

SXP es una metodología de desarrollo de software compuesta por las metodologías SCRUM y XP siendo una metodología híbrida cubana ofrece lo mejor de ambas partes, XP dice cómo tenemos que hacer el software (desarrollo) y SCRUM describe como procede diariamente la planificación.

Esta metodología consta de 4 fases principales para el perfeccionamiento del producto:

1. Planificación-Definición: Se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.
2. Desarrollo: Se realiza la implementación del sistema hasta que esté listo para ser entregado. El tiempo puede disminuir a medida que se está refinando el producto.
3. Entrega: Se realiza la entrega de la documentación, el entrenamiento al usuario e instalación del producto.
4. Mantenimiento: Se realiza el soporte para el cliente y se genera la plantilla de Gestión de cambios.

De cada una de estas fases se despliegan siete flujos de trabajo: concepción inicial, captura de requisitos, diseño con metáforas, implementación, prueba, entrega de la documentación, soporte e investigación; esta última se puede realizar en cualquier punto del ciclo de vida del proyecto ya que los desarrolladores deben investigar y estudiar. Las entregas son frecuentes, lo que permite mejorar el diseño cada vez que se le añade una nueva funcionalidad. (Peñalver, 2008)

La metodología SXP define los siguientes roles:

- Líder del Proyecto (Scrum Máster): Su principal trabajo es remover impedimentos y reducir riesgos del producto.
- Gerente (Management): Es el responsable de tomar las decisiones finales, acerca de estándares y convenciones a seguir durante el proyecto.
- Especialistas: Es necesario que conozca a fondo el proceso, ya sea de la metodología utilizada o cualquier otro proceso o elementos de gran importancia para el desarrollo de software.
- Consultor: Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto, en el que puedan sugerir problemas, además aportan ideas y experiencias para el beneficio del sistema en desarrollo.

- Cliente (Customer): El cliente presenta la Lista de Reserva del producto al equipo, enfatizando el valor y prioridades del mismo, define la meta de la iteración y aprueba las modificaciones en la Reserva del producto.
- Miembros del Proyecto (Scrum Team): Sus tareas son estimar esfuerzo, crear la reserva del Sprint, revisar la Lista de Reserva del Producto y sugerir obstáculos que deban ser removidos para cumplir con las tareas que aparecen.
- Programadores (Programmers): Es el encargado de producir el código y escribir las pruebas unitarias.
- Analista (Analyst): Es el encargado de escribir las historias de usuario y las pruebas funcionales para validar su implementación.
- Diseñadores (Designers): Encargados del diseño del sistema.
- Encargado de Pruebas (Tester): Es el encargado de ayudar al cliente a escribir las pruebas funcionales.
- Arquitecto (Architect): Su trabajo tiene que ver con la estructura y el diseño en grande del sistema.
- Gestor de Investigaciones: Persona encargada de gestionar todas las tareas investigativas que se desarrollan.

A pesar de aplicar este tipo de organización por roles en los proyectos productivos, no se debe de crear un esquematismo, pues en la mayoría de los casos ocurre un solapamiento de roles, es decir una persona puede desempeñar en diferentes etapas, en dependencia de su responsabilidad inicial, varios roles.

SXP permite una integración continua del sistema. Todos los integrantes del equipo de trabajo conocen algo sobre todas las partes y muy bien aquellas en las que trabajan. Todo el código se escribe en parejas.

Metodología de desarrollo de software a utilizar

A partir de este análisis sobre las posibles metodologías de desarrollo de software que podrían haber guiado el proceso de desarrollo, se escoge la

metodología ágil SXP, debido a los beneficios que puede aportar en los proyectos de creación de software, por ser adaptable al desarrollo de esta investigación, porque permite reducir el tiempo y los posibles riesgos a lo largo de todo el ciclo de vida del proyecto, además ofrece una gran flexibilidad ante los cambios de requerimiento, promueve el trabajo en equipo, evidencia rápidos resultados y permite actualizaciones de los procesos de desarrollo.

1.5 Herramientas y tecnologías

Ingeniería de Software asistida por computadoras (CASE)

Las herramientas CASE (Ingeniería de Software Asistida por Computadora), son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero.

Existen alrededor de treinta y siete herramientas CASE, todas con sus particularidades, unas más costosas que otras, más complejas, con más o menos opciones y dedicadas a diferentes tipos de proyectos. Entre estas se encuentran Visual Paradigm, Rational Rose, Umbrello, ArgoUML. Estas herramientas pueden proveer muchos beneficios en todas las etapas del proceso de desarrollo de software, algunas de ellas son:

- Verificar el uso de todos los elementos en el sistema diseñado.
- Automatizar el dibujo de diagramas.
- Ayudar en la documentación del sistema.
- Ayudar en la creación de relaciones en la Base de Datos.
- Generar estructuras de código.

La principal ventaja de la utilización de una herramienta CASE, es la mejora de la calidad de los desarrollos realizados y, en segundo término, el aumento de la productividad. Para conseguir estos dos objetivos es conveniente contar con una organización y una metodología de trabajo, además de la propia herramienta.

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

Visual Paradigm 6.4 es una de las herramientas más utilizadas en el mercado, fue creada para el ciclo vital completo del desarrollo de software; captura de requisitos, análisis, diseño e implementación, automatizándolo y acelerando el proceso de producción.

Es una herramienta profesional enfocada al negocio, muy sencilla de usar, fácil de instalar y actualizar. Permite dibujar todos los tipos de diagramas de clases, generar código inverso y documentación. Posee soporte para UML, el modelo y código permanecen sincronizados en todo el ciclo de desarrollo, existen múltiples versiones para cada necesidad, disponibilidad en múltiples plataformas. (Barrientos, 2012). Es por estas características que la autora decide utilizar esta herramienta para el modelado de los diagramas.

Maven

Maven es una herramienta para la gestión de proyectos de software, que se basa en el concepto de POM (Project Object Model). A pesar de no contar con una interfaz gráfica, ayuda a crear los directorios del proyecto y permite compilar, empaquetar, generar documentación, pasar los test, y preparar las construcciones.

El proceso de construcción permite trabajar de una forma fácil con múltiples proyectos al mismo tiempo. Maven 2.2.1 viene con un mecanismo por el cual el proyecto puede descargar otras dependencias requeridas para el proyecto desde un repositorio central de librerías. Esto permite al usuario de Maven la reutilización de librerías entre proyectos e impulsa la comunicación entre proyectos. (Apache Maven, 2012).

Herramienta de control de versiones Subversion (SVN)

El sistema de control de versiones (SVN) 0.12.0 es un software libre que administra el acceso a un conjunto de ficheros y mantiene una historia de

cambios realizados. El control de versiones es útil para guardar cualquier documento que se cambie con frecuencia. Consiste en una copia muestra en un repositorio central y un programa cliente con el que cada usuario sincroniza su copia local, lo que permite compartir los cambios sobre un mismo conjunto de ficheros. El repositorio guarda registro de los cambios realizados por cada usuario y permite volver a un estado anterior en caso de necesidad.

Ventajas

- Actualización de ficheros modificados.
- Copias de seguridad centralizadas.
- Historial de Cambios.
- Brinda acceso remoto.
- Provee seguridad al sistema.

RapidSVN Se utiliza para gestionar los datos del repositorio SVN. Es una herramienta libre que permite darle a cada integrante del proyecto diferentes permisos según los documentos y la información que sea necesaria para el rol que desempeñe.

Entre sus principales características se destacan:

- Simple - proporciona una interfaz fácil de usar para las características de Subversion.
- Eficiente - simple para los principiantes pero lo suficientemente flexible como para aumentar la productividad para los usuarios de Subversion con experiencia.
- Portable - se ejecuta en cualquier plataforma.
- Rápido - completamente escrito en C ++.

Entorno de Desarrollo Integrado (IDE)

Un entorno de desarrollo integrado (IDE) es un programa compuesto por varias herramientas que son utilizadas por los programadores para desarrollar código. Las IDE poseen un editor de texto, un constructor de interfaz gráfica, un compilador y un depurador. Pueden especializarse en un lenguaje de programación o a varios lenguajes como C, C++ o java.

Eclipse es una de estas herramientas a las que se conoce como entornos de desarrollo integrado. En ella se pueden desarrollar aplicaciones en diferentes lenguajes de programación entre los que encontramos C++, Java, Ruby, PHP; constituyendo así una potente plataforma de programación, desarrollo y compilación.

Si bien Eclipse es multiplataforma no todos los plugins lo son lo que obliga en ciertos casos a depender de una plataforma específica para desarrollar. Los plugins que se desarrollan por la comunidad no tienen todas las funcionalidades que presentan en otras herramientas comerciales. Eclipse consume gran cantidad de recursos en la PC lo que conlleva al desarrollador a depender de un buen hardware para su uso.

NetBeans es de código abierto y multiplataforma. Es una plataforma pensada para escribir, compilar, depurar y ejecutar programas. La programación en esta IDE se realiza a través de componentes modulares o módulos. Las aplicaciones construidas a partir de módulos pueden ser extendidas. Los programadores de software desarrollan los módulos independientemente, permitiendo la integración de estos a la aplicación ampliando las funcionalidades de la misma, de ahí que sean flexible/extendida. Por esta cualidad NetBeans IDE es muy utilizada en aplicaciones grandes. (NetBeans, 2012)

Tecnologías que soporta Netbeans 7.0.1:

- Java EE 6, Java EE 5 and J2EE 1.4
- Spring 3.0, 2.5
- Hibernate 3.2.5
- Apache Maven 3.0.3
- Soporta HTML5.

Después de estudiar las características de los Entornos de Desarrollo Integrado (IDE) se llega a la conclusión de que el IDE con el que se le daría solución óptima al módulo para el SINAP es el Netbeans. Las versiones más recientes permiten integrar el framework Spring, Hibernate, JUnit posibilitando desarrollar de forma más eficiente y organizada el módulo.

Por ser usado por una gran parte de los desarrolladores de aplicaciones web del mundo posee mucha documentación en diversos formatos, tanto audiovisuales como en texto. Al poseer una interfaz agradable e intuitiva los desarrolladores se sienten identificados y a gusto. Por todo lo expuesto se propone la utilización de NetBeans para el desarrollo de la propuesta de solución.

1.5.1 Lenguajes empleados para el desarrollo

Lenguaje de programación Java

Java es un lenguaje de programación de alto nivel orientado a objetos; es muy próximo a la forma de pensar humana, esto permite crear programas modulares y código reutilizable. Los compiladores de Java son capaces de detectar errores que aparecen durante el tiempo de ejecución en otros lenguajes. La tecnología Java está compuesta básicamente por dos elementos: el lenguaje Java y su plataforma.

La plataforma es la máquina virtual de Java (Java Virtual Machine). Es independiente de la plataforma: el mismo código java que funciona en un sistema operativo, funcionará en cualquier otro sistema operativo que tenga instalada la máquina virtual java.

Lenguaje unificado de modelado (UML)

UML es un lenguaje de notación orientada a objetos. Proporciona un vocabulario y reglas para permitir una comunicación entre los desarrolladores al implementar un modelado común para todos. Soluciona el problema de comunicación: cualquier desarrollador con conocimientos de UML será capaz de entender, independientemente del lenguaje utilizado para el desarrollo. Los principales objetivos de UML son:

- Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: a partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Permite dividir el proyecto en una amplia variedad de diagramas para visualizar el sistema desde varias perspectivas, dando una idea de la implementación del proyecto. (Unified Modeling language, 2011)

1.5.2 Framework para el desarrollo

Un framework es un marco de aplicación o conjunto de bibliotecas orientadas a la reutilización de componentes software para el desarrollo rápido de aplicaciones, es una tecnología o modelo de programación que contiene máquinas virtuales, compiladores, bibliotecas de administración de recursos en tiempo de ejecución y especificaciones de lenguajes.

Los framework permiten el desarrollo rápido de aplicaciones. Los componentes incluidos en un framework constituyen una capa que librería al programador, es decir el programador no necesita plantearse una estructura global de la aplicación, sino que el framework le proporciona un esqueleto que hay que llenar. Entre los objetivos principales que persigue un framework están: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones.

Framework jWebSocket

jWebSocket es un marco de trabajo de código abierto para el desarrollo de aplicaciones web y móviles, basado en Java en el lado del servidor y en JavaScript del lado del cliente. Utiliza el protocolo webSocket, este permite que la comunicación sea más rápida y eficiente entre el cliente y el servidor. La

comunicación basada en webSockets garantiza menos sobrecarga en la red en el orden de 400 veces y solamente 1/3 de la latencia en la red. (jWebSocket, 2012)

jWebsocket establece un modelo de token. Los tokens son datos abstractos que a través de una estructura jerárquica y una API proporcionan métodos de acceso a los contenidos. Con el objetivo de realizar una abstracción en la manipulación de los diferentes formatos, el marco de trabajo convierte los paquetes de datos entrantes y salientes en tokens. El servidor jWebSocket está diseñado para funcionar como servidor de comunicaciones o como servidor Web.

jWebSocket es compatible con las últimas versiones de Chrome, Safari y Firefox; mientras que para las versiones de navegadores más antiguos ofrece un plug-in Flash, que es compatible con cualquier navegador y totalmente transparente para la aplicación.

La solución que propone jWebSocket se compone de:

- jWebSocket server: Un servidor desarrollado en Java para llamadas cliente a servidor (C2S), servidor cliente (S2C) y llamadas cliente a cliente (C2C).
- jWebSocket clients: Clientes JavaScript y Java para interacciones con el servidor jWebSocket proporcionando altos niveles de abstracción.

Framework Spring 3.0

Spring es un framework basado en la Inversión de Control y en la Programación Orientada a Aspectos. Se distribuye de forma libre y su código es abierto. Permite configurar complejas aplicaciones a partir de componentes simples. En Spring los objetos de la aplicación se declaran en ficheros (normalmente en formato xml) y el framework se encarga de instanciarlos y configurarlos correctamente a través de la inyección de dependencias. Mediante esta técnica los objetos reciben pasivamente sus dependencias sin necesidad de crearlas o buscarlas, proporcionando un bajo acoplamiento entre los componentes de la aplicación. (Johnson y Hoeller, 2010).

Spring ofrece servicios y recursos para las conexiones de bases de datos, transacciones de bases de datos, la seguridad, de mensajería, almacenamiento en caché, y mucho más. Para gestionar la seguridad de la aplicación con todos los grupos, los roles, y los permisos, sólo se tiene que inyectar un poco de clases en el framework de seguridad de Spring.

Framework JUnit

JUnit es un framework para la plataforma Java, que permite definir los casos de prueba unitarios para las clases importantes del sistema. Permite ejecutar las clases de forma controlada, para poder evaluar si se comporta de la manera esperada.

JUnit ejecutará las pruebas sobre la clase y comparará contra los resultados esperados, si los resultados son correctos indicará que las pruebas fueron exitosas, en caso que alguno de los resultados no sea el esperado JUnit indicará que se produjo un fallo e indicará el o los métodos que no pasaron sus evaluaciones. Es importante destacar que JUnit seguirá con las pruebas aunque ya se haya detectado una falla.

En la actualidad las herramientas de desarrollo como NetBeans y Eclipse cuentan con plugins que permiten la generación de las plantillas necesarias para la creación de pruebas de manera automática. Las pruebas automáticas permiten al programador enfocarse en el resultado esperado, dejando a la herramienta la creación de las clases que permiten coordinar las pruebas. (Malfará y Cukerman y Cócaro, 2006)

Conclusiones del Capítulo

El análisis de los sistemas existentes a nivel nacional e internacional permitió identificar la necesidad de desarrollar una solución capaz de gestionar la información. Finalmente se propusieron las herramientas, tecnologías y metodología necesaria para el correcto desarrollo de la propuesta de solución.

Capítulo 2: Características, análisis y diseño del módulo

En este capítulo se realiza la propuesta de solución y se definen sus características. Se presenta el modelado del dominio con el objetivo de comprender el flujo de la información del Departamento Independiente de Energía; se especifican los requisitos funcionales y no funcionales, se elaboran las historias de usuarios y las tareas de ingeniería asociadas a la metodología SXP.

2.1 Concepción del sistema

Al comienzo de todo proyecto, y para su correcta planificación, se realizan un conjunto de encuentros entre el cliente y el equipo de desarrollo para poder tener una visión de lo que el cliente desea y así poder definir las características del producto. Se especifica la necesidad de desarrollar el producto y se especifican las actividades para su desarrollo.

2.2 Procesos y flujos de información del Departamento.

Los especialistas del Departamento Independiente de Energía de la Administración Provincial de Artemisa son los encargados de gestionar el cumplimiento de la política, normas y disposiciones legales establecidas por el Estado respecto al uso racional y eficiente de la energía en el territorio de la provincia de Artemisa. Este departamento cuenta con un total de seis trabajadores, entre ellos: especialistas, secretarios y dirigentes, que tienen acceso a esta información de los cuales solo los especialistas deben poder gestionar este tipo de información.

Procesos y flujos de información que se maneja en el Departamento independiente de Energía.

Para la gestión de la información del departamento independiente de energía se realizan tres procesos principales: recibir la información, procesar la información, y entregar la información en este orden. El primer proceso comienza cuando la información es enviada desde los municipios y la empresa eléctrica al Departamento Independiente de Energía de la Administración Provincial de Artemisa. Esta información es guardada en diferentes medios de almacenamiento; a partir de aquí comienza el segundo proceso donde se

consultan, buscan y generan documentos en diferentes computadoras con las que trabajan los especialistas del departamento. Como resultado de estos procesos se generan los reportes. El tercer proceso inicia al terminar los reportes que son enviados a la dirección para la toma de decisiones.

Consumo en Barras del Sector Estatal: es un documento que representa el consumo de energía mensual de la provincia de Artemisa por los centros seleccionados y el consumo del sector estatal de cada municipio. Por cada centro se recogen datos como: municipio en la que se encuentra ubicado, organismo al que pertenece, el plan de consumo, el consumo real, el desglose de los planes por organismos, los centros cortados por incumplir el plan y la cantidad de día que serán cortados por este incumplimiento.

Los especialistas realizan un cálculo para definir los días que se corta la energía de un centro y los centros que reinciden en el incumplimiento del plan quedan registrados para un posterior análisis. Estos datos se presentan en diferentes documentos con formato Excel, lo que resulta engorroso al realizar una búsqueda de la información necesaria. Muchas veces los documentos se encuentran distribuidos en diferentes máquinas, dándose el caso de documentos duplicados y con datos erróneos. (Ver anexo 1)

Cumplimiento del Plan por Municipios: Documento que recoge la demanda eléctrica de un organismo para un mes dado, tanto de sus centros seleccionados como no seleccionados, desagregados por municipios. Por cada municipio de la provincia se recoge el consumo real diario y se compara con el plan de consumo establecido; así hasta cumplir con el plan del mes.

Descripción de la propuesta de solución

Para contribuir a la gestión de la información del Departamento Independiente de Energía se propone un módulo para el servidor SINAP, que automatice el flujo de la información del segundo proceso de este departamento que se encarga de almacenamiento de datos, consulta y creación de reportes.

Con esta solución los procesos de almacenamiento, consulta y modificación de los datos, se realizan de forma eficiente y cómoda para el trabajador permitiendo generar reportes en tiempo real. El sistema brinda la posibilidad de buscar e

insertar los datos necesarios para generar los reportes. De modo que los tiempos de respuesta deben ser generalmente rápidos (no debe exceder los 10 segundos). El módulo debe poder alojar hasta 10 usuarios al mismo tiempo.

Diseño general del Sistema Informativo de la Administración Provincial de Artemisa.

El SINAP presenta una arquitectura basada en n-capas, que permite distribuir sus funcionalidades por módulos. Dichas partes se complementan entre sí, haciendo más fácil la gestión de la información en la Administración Provincial de Artemisa por departamentos o direcciones según sea el caso, puesto que aumentan la reutilización de código.

Lo que se conoce como arquitectura en capas es en realidad un estilo arquitectónico donde el objetivo principal es separar los diferentes aspectos del desarrollo: las cuestiones de presentación, lógica de negocio y mecanismos de almacenamiento (ver figura 2.1). La capa de presentación representa la interfaz de usuario, la capa lógica de negocio y acceso a dato se encuentran en el servidor del SINAP y la Base de Datos es donde se guardan los datos que persisten.



Fig.2.1 Arquitectura n-capas.

Esta arquitectura se empleó por la necesidad de contar con porciones de la aplicación que se puedan cambiar sin tener que modificar el resto de la aplicación. Esto facilita el trabajo con módulos teniendo en cuenta sus características: tamaño relativamente pequeño e independencia modular.

Cada capa está diseñada por módulos que le dan respuesta a funciones diferentes, los módulos de diferentes capas pero con los mismos propósitos tienen un punto de encuentro donde se comunican (ver figura 2.2).

El proyecto para informatizar la Administración Provincial de Artemisa, con el objetivo de una mejor organización, decidió distinguir los módulos de diferentes capas destinados a las mismas funciones. Para un mejor entendimiento se plantea un ejemplo: los módulos a1, a2 y a3 que se encuentran ubicados en capa de presentación, capa lógica de negocio y capa de acceso a datos respectivamente. Con un mismo propósito se les llama módulo (a). El desarrollo del módulo energía que pertenece a la capa lógica de negocio.

Descripción general del módulo energía para el servidor del SINAP

El módulo energía es el encargado de insertar, modificar, buscar y realizar reportes estadísticos referentes a la información que se maneja en el Departamento Independiente de Energía de la Administración Provincial de Artemisa. Este módulo forma parte del servidor del SINAP y con respecto a otros módulos que se encuentra en el servidor tiene independencia modular; tiene relación con al menos una de las interfaz de usuario que se encargue de activar las solicitudes realizadas por el cliente, y a su vez tiene acceso a la base de datos del SINAP.

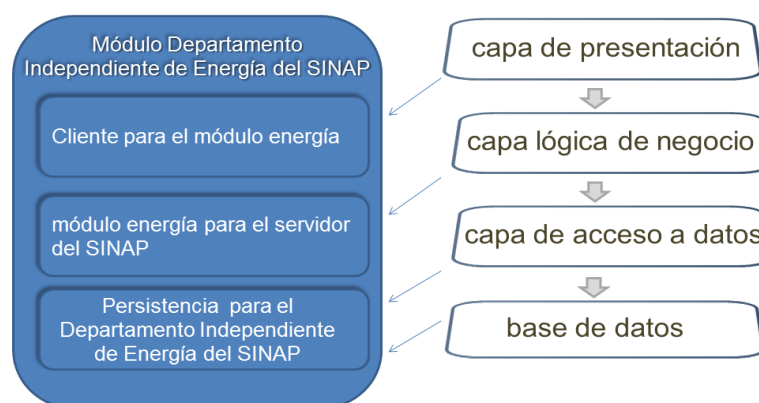


Fig.2.2 Ubicación del módulo energía en la arquitectura n-capa.

El módulo energía se considera una parte importante del SINAP, es el encargado de gestionar la información del Departamento Independiente de

Energía y se integra con los diferentes módulos que conformen el servidor del sistema informático.

2.2.1 Planificación del proyecto por roles

Antes de realizar el diseño es necesario que el equipo de desarrollo sea conformado. Según los roles de la metodología SXP el equipo de desarrollo queda así:

Rol	Responsabilidad	Nombre
Gerente	Profesor	Dania Fernández
Jefe proyecto	Tutor	Frank Rosalez
Miembro del equipo		
Programador	Estudiante	Liliana Hernández Dunaikelys Mesa
Diseñador BD	Estudiante	Liamelys Herrera
Analista	Estudiante	Liliana Hernández Dunaikelys Mesa

Tabla: 2.1 Planificación del proyecto por roles.

2.3 Modelo de Dominio

Dentro de las actividades definidas en la metodología SXP se encuentra la definición del modelo de historias de usuario del negocio, donde se hace una detallada descripción del negocio en cuestión. Si el negocio no está bien definido entre los clientes y los ejecutores del proyecto; entonces se genera el modelo de dominio.

El modelo de dominio se utiliza para capturar y expresar el entendimiento en un área bajo análisis como paso previo al diseño de un sistema. Los analistas lo utilizan como un medio para comprender el sector industrial o de negocios al cual el sistema va a servir.

El propósito fundamental de este modelo es generar una terminología común y sentar las bases del entendimiento del desarrollo. Debido a que la Administración Provincial de Artemisa carece de un cliente en el negocio y el sistema está altamente centrado en tecnologías informáticas, se propone un modelo del dominio que permite de manera visual mostrar al usuario y a los desarrolladores los principales conceptos que se manejan en el dominio del sistema en desarrollo. En la figura 2.1 se representan los principales conceptos definidos dentro del dominio.

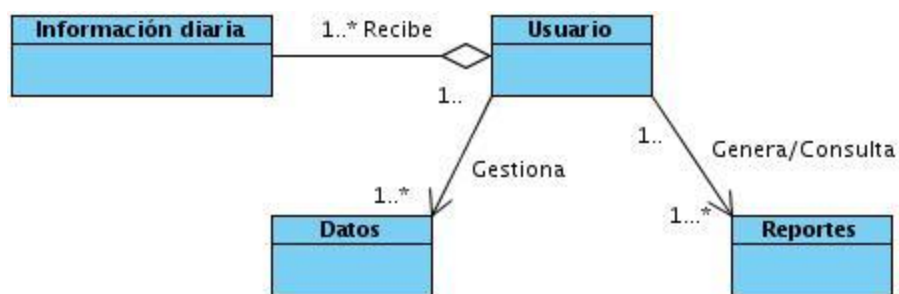


Figura 2.1: Modelo de Dominio de la Propuesta de Solución.

Conceptos del modelo de dominio:

Información diaria: es la información que recibe el usuario desde la empresa eléctrica que contiene los datos del consumo energético.

Usuario: persona encargada dentro del Departamento Independiente de Energía (técnico de consumo o jefe de regulación y demanda) de recibir la información diaria y almacenarla.

Datos: son los datos que maneja el sistema del Departamento Independiente de Energía. El usuario es capaz de introducir, modificar y buscar los datos en el sistema.

Reportes: estos reportes se generan a partir de los datos diarios y semanales almacenados en el sistema. El usuario puede consultar o generar estos reportes.

2.4 Captura de requisitos

El proceso de captura de requerimientos es una etapa de suma importancia dentro del proceso de desarrollo de software. Éste se preocupa de descubrir y analizar las necesidades del usuario del sistema a construir. Es el punto de

partida para las siguientes actividades del proyecto, sirviendo de base para verificar si se alcanzaron los objetivos establecidos.

La captura de requisitos permite gestionar las necesidades del proyecto en forma estructurada, cada actividad tendrá varios pasos a seguir. Mejora de la capacidad de predecir cronogramas de proyecto proporcionando un punto de partida para controlar actividades específicas. Mejora de la calidad del software pues si se cumple con todos los requisitos el software poseerá lo que el cliente desea por lo tanto tendrá buena calidad. Evita rechazo de usuarios finales debido a que obliga a los usuarios a considerar sus requerimientos cuidadosamente.

Listado de reserva del producto (LRP)

En la LRP se especifica los requisitos funcionales y no funcionales que se tienen en cuenta en el sistema. Es creada por las sugerencias del cliente y la experiencia del analista del proyecto. En la lista de reserva del producto se clasifica los requisitos por prioridad que van desde muy alta hasta la prioridad baja.

Ítem *	Descripción	Estimación	Estimado por
Prioridad Muy Alta			
1	Insertar Datos del Plan del Consumo Energético	2 días	Analista
2	Insertar Datos del Real del Consumo Energético	2 días	Analista
3	Insertar Información de los Consumos Acumulados	2 días	Analista
4	Insertar Datos de los Centros Seleccionados	2 días	Analista
5	Insertar Datos de los Incumplidores del	2 días	Analista

	Plan Acumulado de los Centros Seleccionados		
6	Insertar Datos de los Organismos de la Administración Central del Estado (OACE) de los Centros con Deficiencias	2 días	Analista
7	Insertar Datos del Desglose de los Planes	2 días	Analista
8	Insertar Datos de los Centros Cortados por Incumplimiento	2 días	Analista
9	Modificar Datos del Plan del Consumo Energético	2 días	Analista
10	Modificar Datos del Real del Consumo Energético	2 días	Analista
11	Modificar Información de los Consumos Acumulados	2 días	Analista
12	Modificar Datos de los Centros Seleccionados	2 días	Analista
13	Modificar Datos de los Incumplidores del Plan Acumulado de los Centros Seleccionados	2 días	Analista
14	Modificar Datos de los Organismos de la Administración Central del Estado (OACE) de los Centros con Deficiencias	2 días	Analista
15	Modificar Datos del Desglose de los Planes	2 días	Analista
16	Modificar Datos de los Centros Cortados	2 días	Analista

	por Incumplimiento		
17	Buscar Datos del Plan del Consumo Energético	2 día	Analista
18	Buscar Datos del Real del Consumo Energético	2 día	Analista
19	Buscar Información de los Consumos Acumulados	2 día	Analista
20	Buscar Datos de los Centros Seleccionados	2 día	Analista
21	Buscar Datos de los Incumplidores del Plan Acumulado de los Centros Seleccionados	1 día	Analista
22	Buscar Datos de los Organismos de la Administración Central del Estado (OACE) de los Centros con Deficiencias	1 día	Analista
23	Buscar Datos del Desglose de los Planes	1 día	Analista
24	Buscar Datos de los Centros Cortados por Incumplimiento	1 día	Analista
Prioridad Alta			
25	Generar Reporte del Consumo en Barras Diario	5 días	Analista
26	Generar Reporte de los Consumos Acumulados	5 días	Analista
27	Generar Reporte de los Centros	5 días	Analista

	Seleccionados		
28	Generar Reporte de los Incumplidores del Plan Acumulado de los Centros Seleccionados	5 días	Analista
29	Generar Reporte de los Organismos de la Administración Central del Estado (OACE) de los Centros con Deficiencias	5 días	Analista
30	Generar Reporte del Desglose de los Planes	5 días	Analista
31	Generar Reporte de los Centros Cortados por Incumplimiento	5 días	Analista
RNF (Requisitos No Funcionales)			
47	Sistema operativo GNU/Linux (Ubuntu 10.10 o superior), Windows xp, vista, 7.		
48	Memoria RAM 1GB (Mínimo).		
49	CPU 2.4 GHz (Mínimo).		
50	10GB de Disco duro		
51	Navegadores (Firefox 5 en adelante y Chromer o Chromiun 13 en adelante).		
52	Máquina virtual de java versión jdk7		

Tabla 2.2: Lista de Reserva del Producto.

Historias de usuario

Las historias de usuarios las definen los propios clientes juntos con el equipo de desarrollo. Son cortas, concretas y describen tareas que el sistema debe hacer. Se define un tiempo estimado que no debe ser muy prolongado para su

implementación.

Para ilustrar se pone como ejemplo la primera historia de usuario definida por los desarrolladores y el cliente.

Historia de Usuario	
Número: HU_1	Nombre historia de usuario: Gestionar Plan del Consumo Energético
Modificación de historia de usuario número: ninguna	
Usuario: Liliana Hernández Suárez y Dunaikelys Mesa	Iteración asignada: 2
Prioridad en negocio: Muy Alta	Puntos estimados: 1
Riesgo en desarrollo: Alto	Puntos reales: 1
Descripción: La presente historia de usuario tiene como objetivo gestionar la información referente al cumplimiento del plan de consumo energético a nivel provincial para cada una de los municipios de la provincia de Artemisa. Debe permitir insertar, modificar y buscar los datos relacionadas con este proceso.	
Observaciones: Para que esto sea posible el usuario ya debe de estar logueado, para poder manejar dicha información.	

Tabla 2.3: Historia de Usuario Gestionar Plan del Consumo Energético.

2.5 Lista de riesgos

Los riesgos del proyecto son eventos, que si ocurren, poden tener efectos positivos o negativos sobre los objetivos del proyecto. El riesgo tiene una causa y, si se produce, un impacto. El riesgo incluye una amenaza para el cumplimiento de los objetivos del proyecto y, a la vez, una oportunidad para mejorar estos objetivos.

La lista de riesgo es el documento que se genera de la actividad de valoración de riesgos. Se elabora para tener constancia y evitar los posibles riesgos a los que se puede enfrentar el proyecto y medir el posible impacto que tendría este

sobre el mismo, el efecto y la probabilidad. El objetivo de esta lista es mantener todos ellos dentro de los límites definidos y aceptados en el desarrollo del proyecto. Estos posibles riesgos se mitigan y en caso de que ocurran se procede a ejecutar el plan de contingencias.

Riesgos

1. El equipo de desarrollo no cuenta con la suficiente experiencia en las tecnologías `jQuery`.
2. Poco conocimiento de la metodología de desarrollo SXP.
3. Las computadoras destinadas a la implementación del producto no cuentan con las prestaciones necesarias.
4. Fallos continuos del fluido eléctrico.
5. Poca coordinación entre los integrantes del equipo y el personal de dirección.

Mitigación de riesgo

1. Brindar capacitación a los integrantes del equipo de desarrollo.
2. Proporcionar documentación de la metodología y capacitación
3. Priorizar las computadoras que tienen mayor necesidad de mejorar las prestaciones.
4. Realizar salvallas continuas a la información.
5. Coordinar las tareas y realizar reuniones para ver los progresos que se han realizado.

2.6 Diseño con metáforas

En SXP no se especifica una arquitectura temprana para el sistema, por lo que se asume una arquitectura evolutiva y los posibles problemas que se generarían por no contar con ella explícitamente en el comienzo del proyecto se solventan con la existencia de una metáfora.

Las metáforas conforman el vocabulario para realizar la descripción del problema; como debería funcionar el sistema con la solución más simple a

implementar. A partir de esto se genera el artefacto “Modelo de Diseño” que está compuesto por el diagrama de paquetes que expone el diseño realizado, donde se muestran las dependencias lógicas entre paquetes de software, ya se traten de componentes de código fuente o librerías.

Los diagramas de paquetes describen los elementos físicos del sistema y sus relaciones. El diagrama de paquetes que se muestra a continuación se usa para reflejar la organización de los paquetes y sus elementos en el módulo de energía. A continuación se presenta el diagrama de paquetes que se propone con una síntesis de lo que representa cada paquete

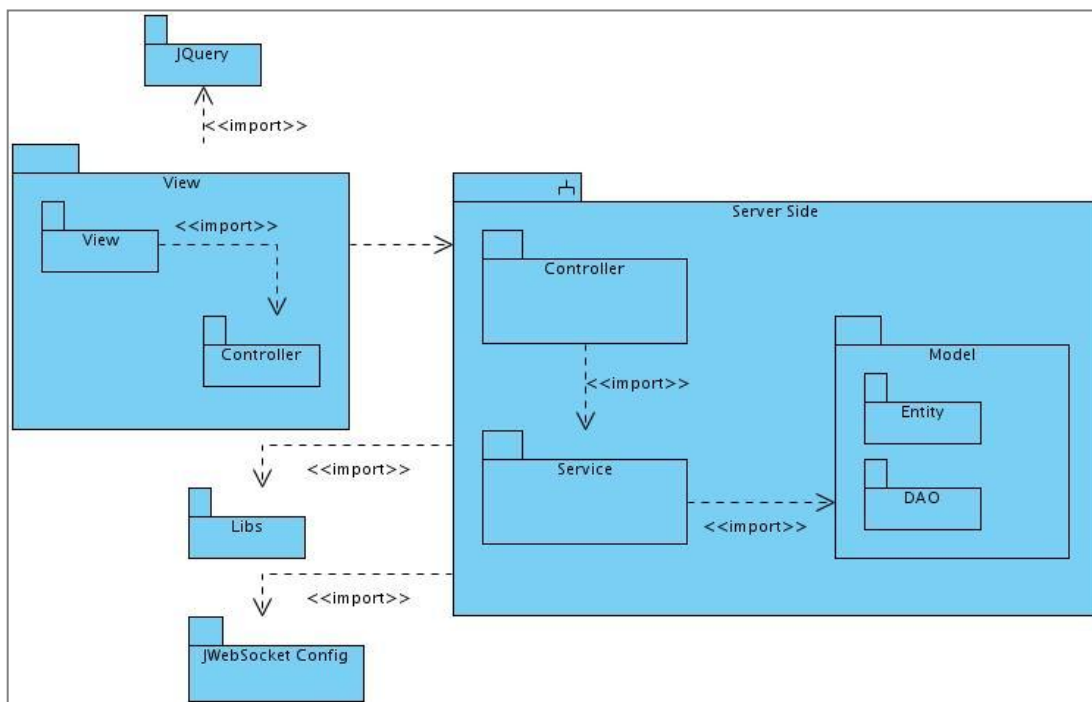


Fig2.3: Diagrama de paquete

Descripción

- En el paquete View se encuentran los componentes que representan la interfaz de usuario. Es importante destacar que en esta capa se utiliza el patrón arquitectónico Modelo-Vista-Controlador (MVC) el cual separa la presentación de los datos.
- En el paquete Server Side se representan los componentes que son ubicados en diferentes paquetes, que darán cumplimiento a los requisitos funcionales del sistema. Depende de dos paquetes; uno de ellos que

contiene las librerías y el otro, Jwebsocket Confging, que contiene los xml de configuración del sistema los cuales permitirán vincular las vistas de la aplicación con el Server Side.

- Server Side está conformado por varios paquetes. Cada uno de ellos tiene funcionalidades específicas en el desarrollo del sistema, aquí se encuentran ubicados los Controller y los Service para poder dar cumplimiento a cada uno de los requisitos. Estos forman parte de la capa lógica del negocio de la arquitectura del SINAP.
 - En el paquete Controller es donde se representan los componentes que darán cumplimiento a los requisitos funcionales del sistema.
 - En el paquete Service es donde se encuentran los eventos correspondientes para cada componente.
- En el paquete Model se encuentran los componentes Entity y Dao. En el paquete Entity se guardan los datos que son persistentes, por lo tanto sus datos permanecerán guardados en la base de datos del sistema y en paquete Dao se encuentran los archivos que nos permiten la interacción con el paquete Service.

Diagrama de componente

Los diagramas de componentes describen elementos físicos del sistema, así como las relaciones existentes entre ellos. Pueden estar conformados por archivos, paquetes, bibliotecas, entre otros.

Se representa todos los tipos de elementos de software que van a ser parte de la aplicación. A continuación se muestra el diagrama de componentes para el módulo que se propone en esta investigación.

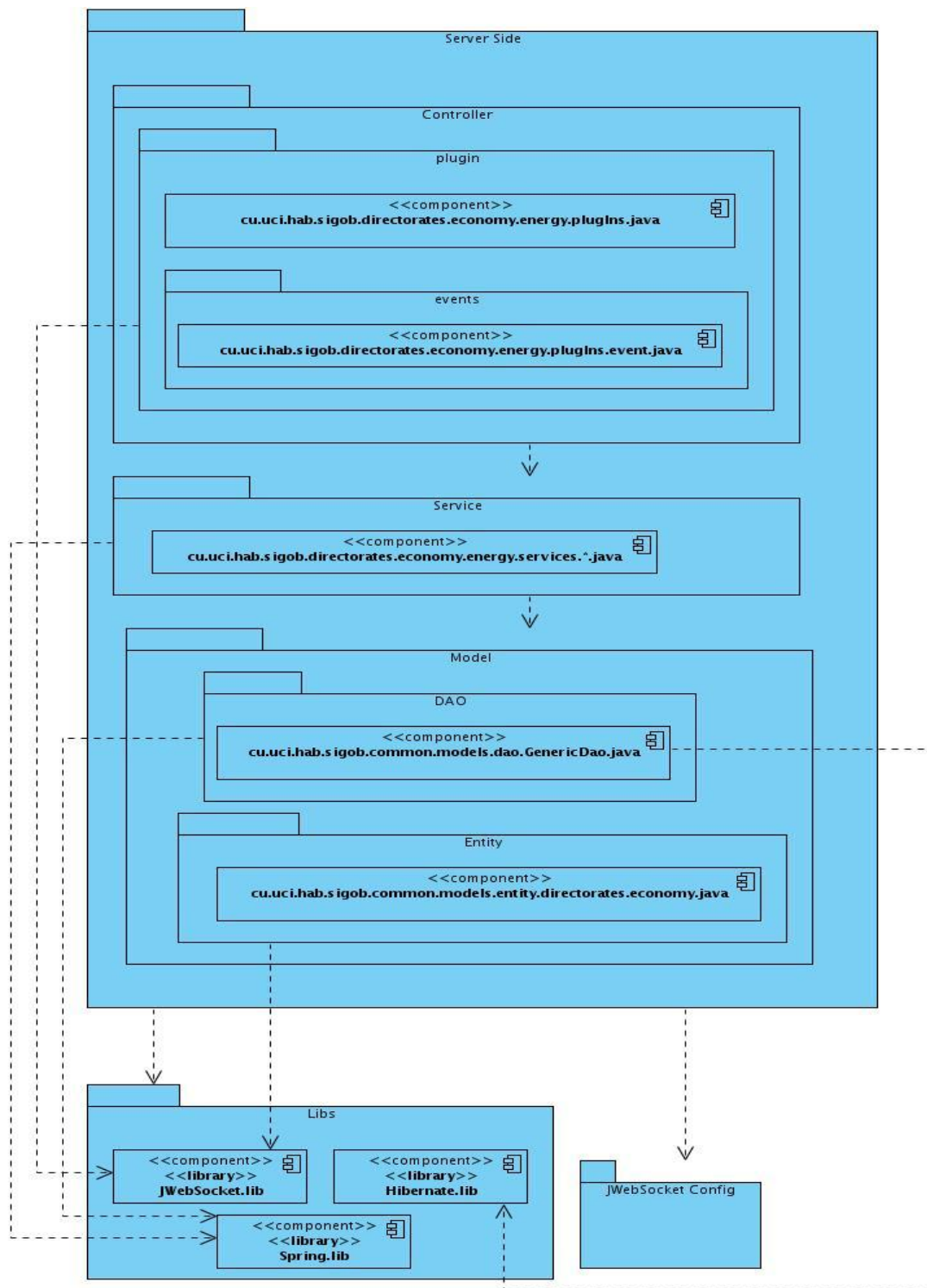


Fig 2.4: Diagrama de componente.

2.7 Tareas de ingeniería

Las tareas de ingeniería facilitan la definición de cada una de las actividades que se describen en las historias de usuarios. Permite conocer el tiempo que

durara cada una y quién debe desarrollar dicha tarea facilitando poder realizar una buena estimación del tiempo de implementación.

Durante la etapa de desarrollo se generaron 35 tareas de ingenierías. Las tareas se generan de las historias de usuario. Cada historia de usuario tiene una o varias tareas de ingeniería. A continuación se presentan las tareas de ingeniería de las historia de usuario “Gestionar la Información del Plan del Consumo Energético” y “Generar reporte de la Información del Plan del Consumo Energético”.

Tarea de Ingeniería	
Número tarea: 1.1	Número historia de usuario: HU_1
Nombre tarea: Implementar los requerimientos para procesar los datos insertados de la Información del Plan del Consumo Energético.	
Tipo de tarea : Desarrollo	Puntos estimados: 1/3
Fecha inicio: 30/01/2012	Fecha fin: 31/01/2012
Programador responsable: Dunaikelys Mesa.	
Descripción: Realizar un estudio sobre el proceso de como insertar. A partir del estudio realizado se establecen las bases para la implementación de esta funcionalidad. Se insertarán el plan del consumo por municipio y por fecha.	

Tabla 2.5 Tarea de Ingeniería insertar datos del Plan del Consumo Energético.

Tarea de Ingeniería	
Número tarea: 1.2	Número historia de usuario: HU_1
Nombre tarea: Implementar los requerimientos para procesar los datos buscados de la Información del Plan del Consumo Energético.	
Tipo de tarea : Desarrollo	Puntos estimados: 1/3

Fecha inicio: 21/02/2012	Fecha fin: 22/02/2012
Programador responsable: Dunaikelys Mesa.	
Descripción: Realizar un estudio sobre el proceso de como buscar. A partir del estudio realizado se establecen las bases para la implementación de esta funcionalidad. Se buscará por el municipio y la fecha.	

Tabla 2.6 Tarea de Buscar datos del Plan del Consumo Energético.

Tarea de Ingeniería	
Número tarea: 1.3	Número historia de usuario: HU_1
Nombre tarea: Implementar los requerimientos para procesar los datos que fueron resultados de la búsqueda de la Información del Plan del Consumo Energético.	
Tipo de tarea : Desarrollo	Puntos estimados: 1/3
Fecha inicio: 07/03/2012	Fecha fin: 08/03/2012
Programador responsable: Dunaikelys Mesa.	
Descripción: Realizar un estudio sobre el proceso mostrar información. A partir del estudio realizado se establecen las bases para la implementación de esta funcionalidad. En este proceso se deben mostrar una lista de datos. Teniéndose en cuenta que para mostrar información antes se debe haber realizado el proceso de búsqueda. Por lo que los procesos de búsqueda y listado de información están relacionados. Esta funcionalidad está presente en todos los buscar.	

Tabla 2.7 Tarea ingeniería mostrar listado de datos.

Tarea de Ingeniería	
Número tarea: 1.5	Número historia de usuario: HU_1

Nombre tarea: Implementar los requerimientos para procesar los datos modificados de la Información del Plan del Consumo Energético.	
Tipo de tarea : Desarrollo	Puntos estimados: 1/3
Fecha inicio: 07/03/2012	Fecha fin: 08/03/2012
Programador responsable: Dunaikelys Mesa.	
Descripción: Realizar un estudio sobre el proceso de como modificar. A partir del estudio realizado se establecen las bases para la implementación de esta funcionalidad.	

Tabla 2.8 Tarea de modificar datos del Plan del Consumo Energético.

2.8 Plan de Releases

Planificar las actividades a efectuar en un proyecto es una vía utilizada para estimar el tiempo de duración. En SXP se elabora el Plan de Releases por iteraciones para planificar el tiempo de las actividades a desarrollar en un proyecto.

En este plan se recogen las iteraciones a realizar con sus características, el orden por prioridad de las historias de usuario y su planificación estimada para ser implementadas. Cada iteración permite transformar un subconjunto de la “Reserva del producto” en un incremento en la funcionalidad del producto que sea potencialmente entregable a los usuarios. Como resultado de la priorización de historias se llegó a la siguiente planificación:

iteración	Descripción de la iteración	Orden de la HU a implementar	Duración total
2	El objetivo de esta iteración es implementar las funcionalidades encargadas de insertar, modificar, eliminar y buscar lo referente al consumo energético.	HU_1, HU_2, HU_5, HU_8, HU_11, HU_14, HU_17, HU_20	4 semanas
3	El objetivo de esta iteración es implementar las funcionalidades encargadas de generar los reportes referentes al consumo energético.	HU_3, HU_6, HU_9, HU_12, HU_15, HU_18, HU_21.	8 semanas

Tabla 2.9 Plan de Releases.

Conclusiones del Capítulo

Se analizaron en este capítulo los procesos fundamentales que están relacionados con la solución informática a desarrollar. Resultaron claras las tareas que el sistema debe realizar. Quedaron aprobados los requisitos funcionales necesarios para obtener un módulo eficiente. Se obtuvo el desarrollo del módulo para el Departamento Independiente de Energía del Sistema Informativo del Consejo de la Administración de Artemisa. Su implementación se desarrolló satisfactoriamente a partir del diseño planteado.

Capítulo 3: Validación del módulo.

En el presente capítulo se exponen los casos de pruebas o test que se le realizaron al módulo. Las pruebas tienen como objetivo validar que el “módulo de energía” cumpla con las expectativas esperadas desde el punto de vista de su confiabilidad y eficiencia.

3.1 Casos de pruebas

Las pruebas son preparadas por el equipo de desarrollo, para comprobar las funcionalidades, forman parte del ciclo de vida del software y se deben realizar a lo largo de su desarrollo. La realización de las pruebas, garantiza la entrega de un producto de mayor calidad y que responda siempre a las necesidades del cliente.

Se definieron casos de prueba donde se incluyen todas las historias de usuario. A continuación se dan a conocer las pruebas que se realizaron, con una de las historias de usuario con las que cuenta el módulo energía.

3.1.2 Caso de prueba unitaria para la clase servicio Plan del Consumo Energético.

Caso de Prueba Unitaria	
Código Caso de Prueba:PU-01	Nombre historia de usuario: buscar Plan del Consumo Energético.
Nombre de la persona que realiza la prueba: Dunaikelys Mesa	
Descripción de la prueba: El objetivo de este caso de prueba es realizar la búsqueda referente a los datos del plan del consumo energético con sus campos correspondientes.	
Condiciones de ejecución: Para poder ejecutar el archivo de prueba debe existir el archivo de configuración persistence-config.xml en la dirección descrita en el archivo de prueba ConsumptionPlanTest.java.	
Descripción Archivo Prueba: El archivo de prueba contiene la carga de las	

<p>configuraciones de la conexión de la base de datos y los objetos de la clase servicio a la cual se le inyecta las configuraciones de la conexión a la base de datos a partir del objeto del tipo GenericDao. En este archivo se crea el evento para buscar la información de los modelos y se les pasan datos de pruebas.</p>
<p>Entrada / Pasos de ejecución: Ejecutar el archivo con el nombre ConsumptionPlanTest.java.</p>
<p>Resultado Esperado: Si la prueba se realizó con éxito se espera el mensaje (true), en caso que el municipio y la fecha que se desea buscar no se encuentre en la base de datos se espera el mensaje (no existe registros que mostrar).</p>
<p>Evaluación de la Prueba: Satisfactoria</p>

Tabla 3.1 Caso de prueba unitaria para la función búsqueda del Plan del Consumo Energético.

Caso de Prueba Unitaria	
Código Caso de Prueba: PU-02	Nombre historia de usuario: modificar Plan del Consumo Energético.
Nombre de la persona que realiza la prueba: Dunaikelys Mesa	
Descripción de la prueba: El objetivo de este caso de prueba es realizar la modificar referente a los datos del plan del consumo energético con sus campos correspondientes.	
Condiciones de ejecución: Para poder ejecutar el archivo de prueba debe existir el archivo de configuración persistence-config.xml en la dirección descrita en el archivo de prueba ConsumptionPlanTest.java.	
Descripción archivo prueba: El archivo de prueba contiene la carga de las configuraciones de la conexión de la base de datos y los objetos de la clase servicio a la cual se le inyecta las configuraciones de la conexión a la base de	

datos a partir del objeto del tipo GenericDao. En este archivo se crea el evento para modificar la información de los modelos y se les pasan datos de pruebas.
Entrada / Pasos de ejecución: Ejecutar el archivo con el nombre ConsumptionPlanTest.java.
Resultado Esperado: Si la prueba se realizó con éxito se espera el mensaje (Se modificó correctamente), en caso que el municipio que se desea modificar coincide con un municipio que tenga la misma fecha se espera el mensaje (no se puede modificar el registro).
Evaluación de la prueba: Satisfactoria

Tabla 3.2 Caso de prueba unitaria para la función modificar del Plan del Consumo Energético.

Caso de Prueba Unitaria	
Código Caso de Prueba: PU-03	Nombre Historia de Usuario: generar Plan del Consumo Energético.
Nombre de la persona que realiza la prueba: Dunaikelys Mesa	
Descripción de la prueba: El objetivo de este caso de prueba es realizar el reporte de la información del plan del consumo energético con sus campos correspondientes.	
Condiciones de ejecución: Para poder ejecutar el archivo de prueba debe existir el archivo de configuración persistence-config.xml en la dirección descrita en el archivo de prueba ConsumptionPlanTest.java.	
Descripción archivo prueba: El archivo de prueba contiene la carga de las configuraciones de la conexión de la base de datos y los objetos de la clase servicio a la cual se le inyecta las configuraciones de la conexión a la base de datos a partir del objeto del tipo GenericDao. En este archivo se crea el evento para generar la información de los modelos y se les pasan datos de pruebas.	

Entrada / Pasos de ejecución: Ejecutar el archivo con el nombre ConsumptionPlanTest.java.
Resultado esperado: Si la prueba se realizó con éxito se esperan todos los datos del reporte generado, en caso contrario se espera una lista vacía.
Evaluación de la prueba: Satisfactoria

Tabla 3.3 Caso de prueba unitaria para la función generar reporte del Plan del Consumo Energético.

Caso de Prueba Unitaria	
Código Caso de Prueba:PU-04	Nombre Historia de usuario: Cargar nomencladores.
Nombre de la persona que realiza la prueba: Dunaikelys Mesa	
Descripción de la prueba: El objetivo de la siguiente prueba unitaria es cargar los nomencladores de la base de datos.	
Condiciones de ejecución: Para poder ejecutar el archivo de prueba debe existir el archivo de configuración persistence-config.xml en la dirección descrita en el archivo de prueba Load.java.	
Descripción archivo prueba: El archivo de prueba contiene la carga de las configuraciones de la conexión de la base de datos y los objetos de la clase servicio a la cual se le inyecta las configuraciones de la conexión a la base de datos a partir del objeto del tipo GenericDao. En este archivo se crean los eventos para cargar los nomencladores.	
Entrada / Pasos de ejecución: Ejecutar el archivo con el nombre Load.java.	
Resultado esperado: Si la prueba se realizó con éxito se esperan todos los datos de los nomencladores, en caso contrario se espera una lista vacía	
Evaluación de la prueba: Satisfactoria	

Tabla 3.4 Caso de prueba unitaria para la función cargar nomencladores del Plan del Consumo Energético.

Caso de Prueba Unitaria	
Código Caso de Prueba:PU-05	Nombre historia de usuario: insertar Plan del Consumo Energético.
Nombre de la persona que realiza la prueba: Dunaykelis Mesa	
Descripción de la prueba: El objetivo de este caso de prueba es realizar la inserción referente a los datos del plan del consumo energético con sus campos correspondientes.	
Condiciones de ejecución: Para poder ejecutar el archivo de prueba debe existir el archivo de configuración persistence-config.xml en la dirección descrita en el archivo de prueba ConsuptionPlanTest.java.	
Descripción archivo prueba: El archivo de prueba contiene la carga de las configuraciones de la conexión de la base de datos, los objetos de la clase servicio a la cual se le inyecta las configuraciones de la conexión a la base de datos a partir del objeto del tipo GenericDao. En este archivo se crean los eventos para insertar la información de los modelos y se les pasan datos de pruebas. Se realiza el llamado al método que se desea probar.	
Entrada / Pasos de ejecución: Ejecutar el archivo con el nombre ConsuptionPlanTest.java.	
Resultado esperado: Si la prueba se realizó con éxito se espera el mensaje (Se insertó correctamente), en caso que el municipio que se desea insertar se encuentre en la base de datos se espera el mensaje (ya está insertado ese municipio).	
Evaluación de la prueba: Satisfactoria	

Tabla 3.5 Caso de prueba unitaria para la función insertar información del Plan del Consumo Energético.

3.2 Pruebas Unitarias automatizadas con JUnit.

El objetivo de las pruebas unitarias es aislar cada parte del módulo y mostrar que las partes individuales son correctas. Las pruebas unitarias automatizadas aseguran que una determinada clase cumpla con un comportamiento esperado.

Cuando la prueba se ejecuta de forma aislada debe resultar 100% efectivas antes de ser integrada al sistema. En caso contrario hay que solucionar los errores y ejecutar nuevamente los casos de prueba hasta lograr que funcione correctamente. La automatización de las pruebas unitarias permite al programador ejecutar todas las pruebas cuantas veces sea necesario.

Estas pruebas se realizan cuando: la interfaz de un método no es clara, la implementación es complicada, para probar entradas, condiciones inusuales o luego de modificar algo.

Las principales ventajas de la utilización de pruebas unitarias automáticas en el desarrollo son:

Fomentan el cambio: Las pruebas unitarias facilitan que el programador cambie el código para mejorar su estructura (lo que se ha dado en llamar refactorización), puesto que permiten hacer pruebas sobre los cambios y así asegurarse de que los nuevos cambios no han introducido errores.

Simplifica la integración: Puesto que permiten llegar a la fase de integración con un grado alto de seguridad de que el código está funcionando correctamente. De esta manera se facilitan las pruebas de integración

Documenta el código: Las propias pruebas son documentación del código puesto que ahí se puede ver cómo utilizarlo.

Los errores están más acotados y son más fáciles de localizar: dado que tenemos pruebas unitarias que pueden desenmascararlos.

3.2.1 Resultado de las pruebas con JUnit

Las pruebas se realizaron durante y al final de cada iteración para comprobar si el método funciona según lo esperado en los casos de prueba. Se ejecutaron pruebas unitarias con JUnit a cada uno de los métodos de las clases servicios. A continuación se muestra un ejemplo del método adicionar “Plan del Consumo Energético” al que se le realizó la prueba unitaria para verificar su correcto funcionamiento.

```

/**
 * Test of update method, of class ConsumptionPlanService.
 */
public void testModifyConsumptionPlan() throws Exception {
    System.out.println("update");
    ModifyConsumptionPlan event = new ModifyConsumptionPlan();
    event.setIdReal("2");
    event.setDate("2012-02-02");
    event.setId("2");
    event.setMunicipality(3);
    event.setPlan(3.4);
    event.setRecordId("2");
    consption.update(event);
}

/**
 * Test of add method, of class ConsumptionPlanService.
 */
public void testAddConsumptionPlan() throws Exception {
    System.out.println("add");
    AddConsumptionPlan addPlan = new AddConsumptionPlan();
    addPlan.setDate("2012-02-02");
    addPlan.setMunicipality(5);
    addPlan.setPlan(20.3);
    consption.add(addPlan);
}

```

Fig.3.1: Código de Pruebas Unitarias funcionalidad adicionar Plan del Consumo Energético.

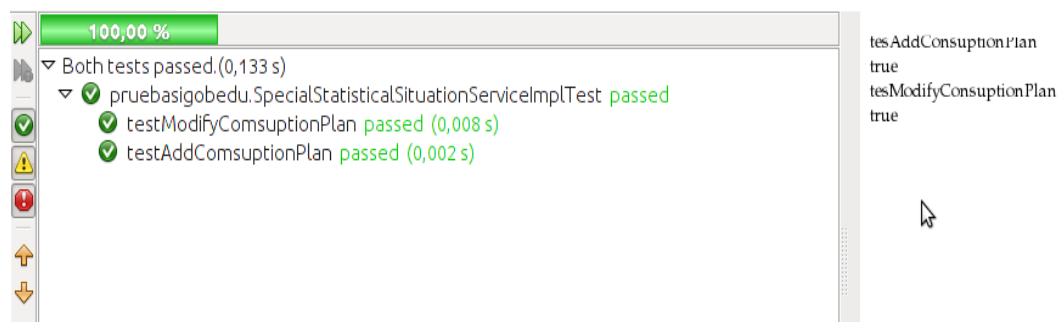


Fig. 3.2: Resultado Prueba Unitaria buscar Plan del Consumo Energético

La interfaz de usuario en modo gráfico de JUnit presenta una barra de progreso la cual toma color verde si todas las pruebas ejecutadas son 100% satisfactorias, o color rojo si falló alguna. En este último caso se despliega en una lista la descripción de las fallas o errores generados.

Los elementos que se garantizaron con la realización de estas pruebas son la eficiencia y la confiabilidad del módulo energía. Se realizaron pruebas donde se

apreció el tiempo de procesamiento y se evidenció que el tiempo es mucho menor que el utilizado por un especialista para realizar la búsqueda de información manualmente. (Ver anexo 2).

Con las pruebas se intentaron encontrar errores en las siguientes categorías:

- Funciones incorrectas o ausentes.
- Errores de estructuras de datos.
- De acceso a bases de datos.

Las pruebas unitarias fueron realizadas para cada uno de los requisitos funcionales (Ver anexo 3). La gráfica que se muestra a continuación muestra el porcentaje de los resultados obtenidos durante y al final de cada iteración de las pruebas realizadas a los requisitos funcionales. De modo que al llegar a la última iteración todas las funcionalidades quedaron probadas al 100% y se integraron satisfactoriamente.

El módulo energía desarrollado fue sometido a valoración y avalado por especialistas del centro de desarrollo de la Facultad Regional “Mártires de Artemisa”, los cuales determinaron su calidad (Ver anexo 4).

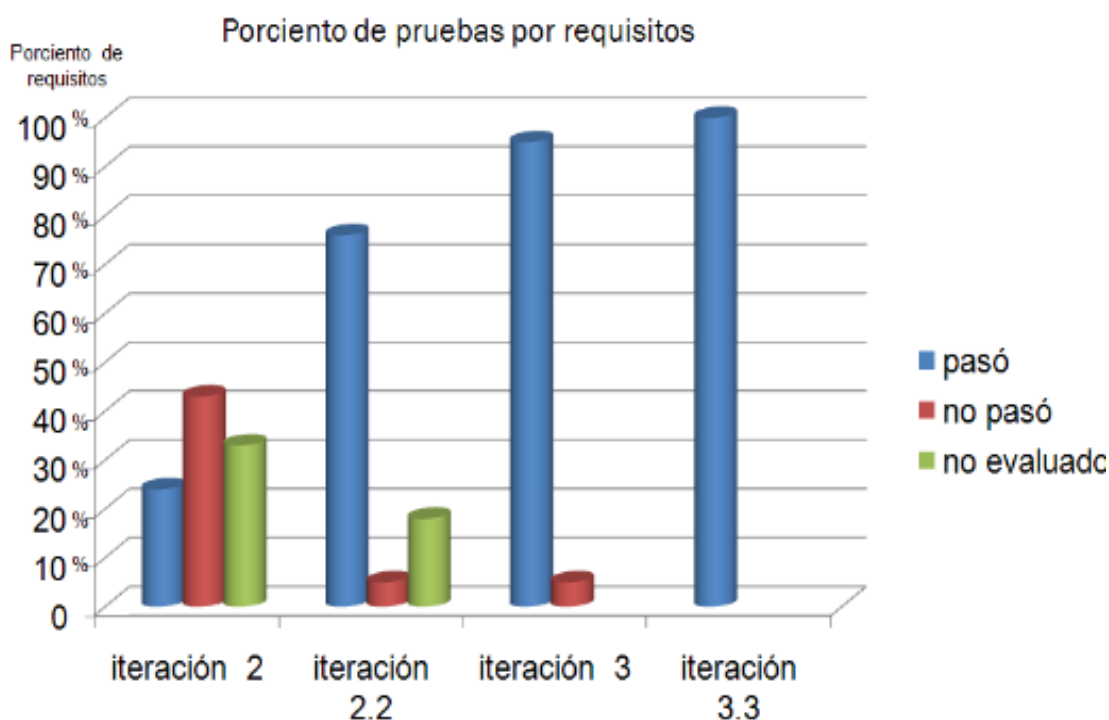


Fig. 3.3: Gráfica de las pruebas realizadas a la clases servicios

3.3 Resultado obtenido

Como resultado de este trabajo se obtuvo el módulo energía para el servidor del SINAP disponible en su versión 1.0. El módulo cumple con todas las especificaciones necesarias para procesar la información del Departamento Independiente de Energía de la Administración Provincial de Artemisa.

3.4 Funcionalidades Obtenidas

Entre las principales funcionales que posee el módulo energía para el servidor del Sistema Informativo de la Administración Provincial de Artemisa en su versión 1.0 se pueden mencionar:

- Permite obtener de toda la información del consumo energético de la provincia en el sector estatal.
- Permite realizar búsqueda por varios criterios de selección.
- Permite modificaciones de datos previamente almacenados.
- Permite al usuario generar Reporte por un criterio establecido.

3.5 Aporte Práctico

El ejemplo de las nuevas tecnologías en la dirección de los procesos constituye hoy una realidad en el mundo desarrollado. A medida que nuestro país y sus instituciones de gobierno se modernicen en este sentido los resultados en materia de organización serán potencialmente superiores. Se han mencionado ejemplos de la utilización de estos sistemas en este trabajo lo que evidencia su inserción en la dinámica de trabajo de los centros cubanos.

El módulo energía constituye un aporte atendiendo al proceso de informatización de la sociedad en particular al proceso de gestión de la información del Departamento Independiente de la Administración Provincial de Artemisa. Es importante señalar que la investigación está enfocada a apoyar la labor empresarial de la institución brindando la posibilidad gestionar más confiable y eficientemente la información. A través de los reportes que brinda el módulo se pueden realizar análisis del consumo energético de la provincia y se

logra una mayor veracidad y exactitud de la información que se emplea en la toma de decisiones.

El aporte económico se manifiesta en el control del recurso información y la disminución del tiempo empleado en la gestión de este recurso permitiendo minimizar los riesgos a los que están expuestos. La contratación de un proyecto con la UCI le permite hoy a esta institución acceder a un producto terminado y certificado completamente gratis con un mínimo de gastos logísticos.

Conclusiones del Capítulo

En el presente capítulo se elaboraron y aplicaron los casos de pruebas unitarias a cada clase para dar validez y veracidad a la propuesta de solución. Se obtuvo un módulo (prototipo funcional) con todos los algoritmos capaces de solucionar los problemas detectados en el Gobierno Provincial de Artemisa.

Conclusiones generales

Se cumplió el objetivo trazado a través de las tareas que guiaron el proceso de desarrollo del módulo energía para el servidor del Sistema Informativo de la Administración Provincial de Artemisa. Resumiendo lo tratado en cada una de las tareas de investigación se concluyen que:

1. Se realizó un estudio de las herramientas a nivel nacional e internacional, evidenciándose la necesidad de crear una solución que cumpla con los requisitos funcionales del Departamento Independiente de Energía.
2. Se caracterizó el proceso de gestión de la información en el Departamento Independiente de Energía evidenciándose que el proceso es costoso en cuanto a tiempo y esfuerzo.
3. Se desarrolló el módulo energía para el servidor del Sistema Informativo de la Administración Provincial de Artemisa, obteniendo como resultado las funcionalidades que dan solución a los requisitos definidos.
4. Se validaron todas las funcionalidades del módulo energía para el servidor del Sistema Informativo de la Administración Provincial de Artemisa demostrando que cumple con la expectativa para el cual fue creado.

Con este módulo el proceso de gestión de la información es más eficientemente y confiable: evitando errores en los documentos, duplicidad de la información, generando reportes en tiempo real y humanizando el trabajo del especialista.

Recomendaciones

Se recomienda para otras iteraciones del producto:

- Realizar las pruebas de aceptación del producto.
- Realizar la funcionalidad exportar a formato pdf.

Referencias Bibliográficas

1. McRobie, G. Tecnología para el desarrollo humano y sostenible. [En línea] septiembre 2011. Disponible en: <http://www.uclm.es/profesorado/igarrido/tecnocooperacion/McRobie.pdf>. [Fecha de consulta 30 de noviembre de 2011].
2. Decreto – ley No. 252 Sobre La Continuidad y el Fortalecimiento del sistema de dirección y gestión empresarial cubano, Ciudad de La Habana, Cuba, 7 de agosto 2007. 3 p. [Fecha de consulta 14 de noviembre de 2011].
3. Thompson I. Qué es Información. [En línea] Octubre de 2008. Disponible en: <http://www.promonegocios.net/mercadotecnia/que-es-informacion.html>. [Fecha de consulta 30 de noviembre de 2011].
4. Davenport TH; Prusack L. Conocimiento en acción. Como las organizaciones manejan lo que saben. Buenos Aires. Prentice Hall. [En línea] 2001. Disponible en: http://books.google.com.cu/books/about/Conocimiento_en_accion.html. [Fecha de consulta 30 de noviembre de 2011].
5. Robbins S. Coulter M, del libro: “Administración”, Octava Edición, de, Pearson Educación, 2005, Págs. 7.
6. Rodriguez Y, Dominguez A. La gestión del conocimiento: un nuevo enfoque en la gestión empresarial. [En línea] Octubre 2007, Disponible en: <http://cis.sld.cu/E/monografias/gestion/cap1.htm>, [Fecha de consulta 30 de noviembre de 2011].
7. Ponjuán Dante G, Villardefrancos Alvarez MC, Leon Santos M. Principios y metodos para el mejoramiento organizacional. La Habana: Felix Varela; 2005.
8. ISO. Norma Internacional ISO 9000:2000. Sistemas de gestión de la calidad - Fundamentos y vocabulario. Ginebra: ISO; 2000.

9. Navarro E. Calidad, gestión de procesos y tecnologías de la información. [En línea] 2002. Disponible en: gestionPolis.com. [Fecha de consulta 5 de febrero de 2012].
10. Lorenzo A; Miralles R. Servicios de documentación en entornos gubernamentales: estudio de un caso. [En línea]: 02-07-2005. Disponible en: <http://www.hipertex.net/web/pag234htm>. [Fecha de consulta 5 de febrero de 2012].
11. Emery J. (1990). "Sistemas de información para la dirección". Editorial Díaz de Santos S. Madrid, España. ISBN: 8487189636.
12. Visbal S, La gestión documental, de información y el conocimiento en la empresa. El caso de Cuba. ed. Última actualización:(2009).Disponible en: http://bvs.sld.cu/revistas/aci/vol19_5_09/aci02509.html.
13. Wiley John. Introduction to Client / Server Systems: A Practical Guide for Systems Professionals. Páginas 105-120. [Fecha de consulta 5 de febrero de 2012].
14. Saidel, Marco A; Rossi, Luís N; Galvão, Luiz C. R; Paula, Sílvio. [En línea] 2008-01-01. Disponible en: <http://worldwidescience.org/topicpages/multi/ES/e/energy+facilities.html>
Sistema para la Gestión del Uso de la Energía en Instituciones Públicas. [Fecha de consulta 5 de febrero de 2012]
15. SIEMES. SENTRON powermanage. [En línea] 1996. <http://www.buildingtechnologies.siemens.com/bt/low-voltage/EN/product-portfolio/software/software-sentron/powermanager/Pages/powermanager.aspx>. [Fecha de consulta 5 de febrero de 2012]
16. Jiménez H y Umaña L. 2011. Business News Americas. Energía. [En línea] 2011. [Citado el: 10 de enero de 2012.] <http://member.bnameicas.com/news/energielectrica>.

17. El cambio organizacional en el sistema nacional de información de la energía en Cuba .González A y Arencibia A. 2005. La Habana: s.n., 2005].
18. Canós J, Letelier P, Penadés C. Metodologías Ágiles en el Desarrollo de Software DSIC. Universidad Politécnica de Valencia.:2010. [Página citada 10] Disponible en: <http://www.bibdigital.epn.edu.ec/bitstream/15000/1118/1/CD-1963.pdf>. . [Fecha de consulta 8 de febrero de 2012].
19. Scrum. [En línea]. Disponible en: <http://scrum.es> 20/08/2011. [Fecha de consulta 5 de febrero de 2012].
20. Calero, S. Una explicación de la programación extrema (XP). Una explicación de la programación extrema (XP). [En línea] [Fecha de consulta 30 de noviembre de 2011]. <http://www.willydev.net/descargas/prev/ExplicaXp.pdf>.)
21. Peñalver, G. MA-GMPR-UR2 Metodología ágil para proyectos de software libre. Universidad de las Ciencias Informáticas. Ciudad de La Habana, Cuba: 2008. [Página citada 94.] http://bibliodoc.uci.cu/TD/TD_1309_08.pdf. [Fecha de consulta 8 de febrero de 2012].
22. Barrientos E. El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML. El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML. [En línea]. Disponible en: <http://www.monografias.com/trabajos16/lenguaje-modelado-unificado/lenguaje-modelado-unificado.shtml>. [Fecha de consulta 8 de febrero de 2012].
23. Apache Maven. [En línea] Disponible en: <http://maven.apache.org>. [Fecha de consulta 5 de febrero de 2012].
24. Netbeans. [En línea].2012. Disponible en:<http://netbeans.org/community/releases/70/>. [Fecha de consulta 5 de febrero de 2012].
25. Unified Modeling language. [En línea]. 7-25-2011. Disponible en:

- <http://www.uml.org>. [Fecha de consulta 30 de noviembre de 2011].
26. jWebsocket. Disponible en: <http://www.jwebsocket.org>. [Fecha de consulta 5 de febrero de 2012].
27. Johnson R, Hoeller.J. Reference Documentaion. Spring Framework [En línea]. 2006.
<http://www.http://static.springsource.org/spring/docs/3.0.x/spring-framework-reference/html/>. [Fecha de consulta 5 de febrero de 2012].
28. Malfará D, Cukerman D, Cócáro F, Cassinelli J. Séttimo. Gestión de Software 2006 Testing en eXtreme Programming, [Página citada 94.][Fecha de consulta 5 de febrero de 2012]

Bibliografía

- Barrientos E. El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML. El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML. [En línea]. Disponible en: <http://www.monografias.com/trabajos16/lenguaje-modelado-unificado/lenguaje-modelado-unificado.shtml>. [Fecha de consulta 8 de febrero de 2012].
- Beck, K. Extreme Programming Explained. Embrace Change. Pearson Education, 1999. p. Traducido al español como: “Una explicación de la programación extrema. Aceptar el cambio”, Addison Wesley, 2000. [Fecha de Consulta: 31 de enero de 2012]
- Bienvenido a NetBeans. [En línea]. <http://www.netbeans.org>. [Fecha de consulta 5 de febrero de 2012].
- Business New Americas [En línea]. Disponible en: <http://www.bnamericas.com/news/energiaelectrica>.
- Calero, S. Una explicación de la programación extrema (XP). Una explicación de la programación extrema (XP). [En línea] [Fecha de consulta 30 de noviembre de 2011]. <http://www.willydev.net/descargas/prev/ExplicaXp.pdf>.)
- Contreras, Y. Rivero, S. Diseño del Sistema de Gestión de información del Centro de Estudios de Medio Ambiente y Recursos Naturales (CEMARNA) de la Universidad de Pinar del Río. GestioPolis.com. [En línea] 4 de 10 de 2007. Disponible en: <http://www.gestiopolis.com/administracion-estrategia/sistemas-de-gestion-de-informacion-en-estudio-de-medio-ambiente.htm>. [Fecha de consulta 30 de noviembre de 2011].
- Canós J, Letelier P y Penadés C. Metodologías Ágiles en el Desarrollo de Software DSIC. Universidad Politécnica de Valencia. [En línea] 2 Mar 2010. Disponible en: <http://www.bibdigital.epn.edu.ec/bitstream/15000/1118/1/CD-1963.pdf>

-
- Davenport TH; Prusack L. Conocimiento en acción. Como las organizaciones manejan lo que saben. Buenos Aires. Prentice Hall. [En línea] 2001. Disponible
 - Decreto – ley no. 252 Sobre La Continuidad y el Fortalecimiento del sistema de dirección y gestión empresarial cubano, Ciudad de La Habana, Cuba, 7 de agosto 2 007. 3 p.[Fecha de consulta 14 de noviembre de 2011].
 - Delgado M. Localización: Ingeniería informática, ISSN 0717-4195, No. 8, 2002 <http://dialnet.unirioja.es/servlet/articulo>.
 - Emery, J. (1990). "Sistemas de información para la dirección". Editorial Díaz de Santos S. Madrid, España. ISBN: 8487189636.
 - El cambio organizacional en el sistema nacional de información de la energía en Cuba. González A y Arencibia Alois. 2005. La Habana : s.n., 2005].
 - García, J.; Cayero U., De la gestión de la información a la gestión del conocimiento, en Investigación Bibliotecología: Archivonomía, Bibliotecología e Información, (2003) 17(4), Enero-Junio, pp. 54-69. Disponible en: <http://www.ejournal.unam.mx/iibiblio/vol17-34/IBI03404.pdf> [Fecha de consulta: 26 de enero de 2012]
 - Geronimo G "Informática e Ingeniería, Programación extrema (Extreme Programming). [En línea]. Disponible en: junio 18, 2008, <http://www.geronet.com>.
 - Gregorio C, Aplicaciones y perspectivas de los estudios métricos de la información en la gestión de información y el conocimiento en las organizaciones. [En línea]. 2008 Disponible en: <http://www.cnpt.embrapa.br/RevistaAIBDA/v29/v29n1d01.pdf>. [Fecha de consulta 30 de noviembre de 2011].
 - Hibernate. [En línea]. Disponible en: <http://www.Hibernate.org>. [Fecha de consulta 5 de febrero de 2012].

- Inspección Estatal Energética. Rodríguez, Fernando. 2004. la Habana: s.n., 2004, Vol. 14.
- ISO. Norma Internacional ISO 9000:2000. Sistemas de gestión de la calidad - Fundamentos y vocabulario. Ginebra: ISO; 2000.
- jWebsocket. Disponible en: <http://www.jwebsocket.org>. [Fecha de consulta 5 de febrero de 2012].
- Lenguaje de programación. [En línea] 2010. <http://www.alegsa.com.ar/Dic/lenguaje%20de%20programacion.php>. [Fecha de consulta 30 de noviembre de 2011]. [26]. springframework.org. Disponible en: <http://www.springframework.org>. [Fecha de consulta 5 de febrero de 2012].
- Lorenzo A; Miralles R. Servicios de documentación en entornos gubernamentales: estudio de un caso. [En línea]: 02-07-2005. Disponible en:<http://www.hipertex.net/web/pag234htm>. [Fecha de consulta 5 de febrero de 2012].
- McRobie, G. Tecnología para el desarrollo humano y sostenible. [En línea] septiembre 2011. Disponible en: <http://www.uclm.es/profesorado/igarrido/tecnocooperacion/McRobie.pdf>. [Fecha de consulta 30 de noviembre de 2011].
- Navarro E. Calidad, gestión de procesos y tecnologías de la información. [En línea] 2002. Disponible en: gestionPolis.com. [Fecha de consulta 5 de febrero de 2012].
- Saidel, Marco A; Rossi, Luís N; Galvão, Luiz C. R; Paula, Sílvio. [En línea] 2006-01-01. Disponible en:<http://worldwidescience.org/topicpages/multi/ES/e/energy+facilities.htm>
I Sistema para la Gestión del Uso de la Energía en Instituciones Públicas. [Fecha de consulta 5 de febrero de 2012].
- Organización Internacional para la Estandarización (ISO). Sistema para dirigir y controlar una organización con respecto a la calidad. [En línea]

1946. Disponible en: <https://docs.google.com>. [Fecha de consulta 30 de noviembre de 2011].

- Ponjuan, G., (1997). “El desarrollo profesional en ciencias de la información y sus aportes al cambio” en revista Ciencia de la Información. Vol.28. No.2. La Habana, Cuba. pp.127-134
- Peñalver, G. MA-GMPR-UR2 Metodología ágil para proyectos de software libre. Universidad de las Ciencias Informáticas. Ciudad de La Habana, Cuba: 2008. [Página citada 94.] http://bibliodoc.uci.cu/TD/TD_1309_08.pdf
- Rodríguez Y, Domínguez A. La gestión del conocimiento: un nuevo enfoque en la gestión empresarial. [En Línea] Octubre 2007, Disponible en: <http://cis.sld.cu/E/monografías/gestión/cap1.htm>, [Fecha de consulta 30 de noviembre de 2011].
- Rodríguez G, Metodología ágil para proyectos de software libre. 2008 [Fecha de consulta: 31 de enero de 2012]
- Scrum. [En línea]. Disponible en: <http://scrum.es> 20/08/2011. [Fecha de consulta 5 de febrero de 2012].
- Johnson R, Hoeller J. Reference Documentaion. Spring Framework [En línea]. 2006. <http://www.http://static.springsource.org/spring/docs/3.0.x/spring-framework-reference/html/>. [Fecha de consulta 5 de febrero de 2012].
- SiEMES. [En línea] 06-feb-2012. Disponible en: <https://support.automation.siemens.com>. [Fecha de consulta 5 de febrero de 2012]
- Thompson, I. Qué es Información. [En línea] Octubre de 2008. Disponible en: <http://www.promonegocios.net/mercadotecnia/que-es-informacion.html>. [Fecha de consulta 30 de noviembre de 2011].
- Unified Modeling language. [En línea]. 7-25-2011. Disponible en: <http://www.uml.org>. [Fecha de consulta 30 de noviembre de 2011]

-
- Welcome to Apache Maven. [En línea] Disponible en:<http://maven.apache.org>. [Fecha de consulta 5 de febrero de 2012].
 - Sardiñas Y, Importancia De La Captura De Requerimientos En El Proceso Productivo Del Proyecto Ciscop De La Facultad Regional De Artemisa. [En línea] 10/11/2008. Disponible en: <http://www.articuloz.com/comunicaciones-articulos/importancia-de-la-captura-de-requerimientos-en-el-proceso-productivo-del-proyecto-ciscop-de-la-facultad-regional-de-artemisa-importance-of-the-captu-636738.html>

Anexo

Anexo1

Esta entrevista está dirigida al personal del Departamento Independiente de Energía como parte del levantamiento de información para el desarrollo del módulo energía del servidor del Sistema Informativo de la Administración Provincial de Artemisa. Le rogamos que conteste las preguntas siguientes. Las identidades de los entrevistados serán reservadas una vez que se generalice esta investigación. Gracias por su cooperación.

Nombre y Apellidos: _____.

Años de experiencia laboral: _____ .

Cargo que desempeña: _____.

Objetivo: Conocer las deficiencias existentes para gestionar toda la información referente a la energía.

1. ¿Con cuántos modelos de información relevante trabaja su departamento?
2. ¿Cuáles son estos modelos?
3. ¿Puede usted clasificarlos según su prioridad en alta medio o bajo?
4. ¿A través de qué vías recibe la información?
5. ¿En qué formato reciben y/o gestionan la información?
6. ¿Tienen algún problema para procesar la información?
7. (De ser positiva la respuesta anterior) ¿Cuáles son los principales problemas que tienen con la información? Y la frecuencia.
8. ¿Cómo se lleva a cabo el llenado de los modelos?
9. ¿Cómo funciona el flujo de información en el departamento?

Análisis de los resultados

Se entrevistó a las 6 personas del Departamento Independiente de Energía y a partir de las respuestas que brindaron se elaboran las tablas siguientes que resumen los resultados obtenidos:

1. Los modelos relevantes con que se trabaja y su prioridad

Los 6 entrevistados que constituyen el 100% de la población entrevistada coinciden que existen 9 modelos de información relevante trabaja su departamento.

2 y 3. Los modelos de información relevante y su prioridad.

Los 6 entrevistados que constituyen el 100% de la población entrevistada coinciden que los modelos de información relevante (incluyendo su prioridad) son:

Modelos	Prioridad		
	Alta	Media	Baja
Consumo en barras del sector Estatal	x		
✦ Órganos de la Administración Central		x	
✦ Acumulado de los incumplidores por plan		x	
✦ Desglose de los planes		x	
✦ Centros cortados		x	
✦ Consumo acumulado		x	
Cumplimiento del Plan por Municipios	x		
29. Plan del Consumo		x	
30. Consumo del Real		x	

--	--	--	--

4. Vías por las que se recibe y/o se procesa la información.

Los 6 entrevistados que constituyen el 100% de la población entrevistada coinciden en que Vías por las que se recibe y/o se procesa la información son:

Modelos	Correo electrónico	Manual	Vía telefónica	Mensajeros
Consumo en barras del sector Estatal	x	x	x	x
Órganos de la Administración Central		x		
Acumulado de los incumplidores por plan		x		
Desglose de los planes		x		
Centros cortados		x		
Consumo acumulado		x		
Cumplimiento del Plan por Municipios	x	x	x	x
Plan del Consumo		x		
Consumo del Real		x		

5. Formato en que reciben y/o gestionan la información.

Los 6 entrevistados que constituyen el 100% de la población entrevistada coinciden en que el formato en que reciben y/o gestionan la información son:

Modelos	Archivos XML	Archivos Excel	Documentos en Word
Consumo en barras del sector Estatal		x	
Órganos de la Administración Central		x	
Acumulado de los incumplidores por plan		x	
Desglose de los planes		x	
Centros cortados		x	
Consumo acumulado		x	
Cumplimiento del Plan por Municipios		x	
Plan del Consumo		x	
Consumo del Real		x	

6. Existencia de problemas en el procesamiento de la información.

De los 6 entrevistados, 4 de ellos (66%) opina que existen problemas con el procesamiento de la información. Un entrevistado (Sujeto C y D)(33%) opina que no existen problemas.

Respuesta	Si	No
Sujeto A	x	
Sujeto B	x	
Sujeto C		x
Sujeto D		x
Sujeto E	x	
Sujeto F	x	

7. Los principales problemas que tienen con la información.

Los 4 sujetos que plantean los problemas coinciden en las siguientes dificultades:

Dificultad para analizar	Sujeto A	Sujeto B	Sujeto E	Sujeto F
Dificultad para analizar	x	x	x	x
Pérdidas de información			x	x
Duplicados de información	x			x
Falta de seguridad		x		x

8. Frecuencia con la que se recibe y/o se procesa la información relevante.

Los 3 entrevistados que constituyen el 100% de la población entrevistada coinciden en que la frecuencia con la que se recibe y/o se procesa la información relevante son:

Modelos	diaria	semanal	mensual	anual
Consumo en barras del sector Estatal	x			
Órganos de la Administración Central			x	
Acumulado de los incumplidores por plan			x	x
Desglose de los planes			x	
Centros cortados			x	
Consumo acumulado			x	
Cumplimiento del Plan por Municipios	x			
Plan del Consumo	x	x		
Consumo del Real	x	x		

Anexo 2

Resultado de la observación a las pruebas realizadas.

Funcionalidades	Velocidad de procesamiento del módulo.
Insertar Consumo del Plan Energético	00.2s
Insertar Consumo del Real Energético	00.2s
Insertar Información de los Consumos Acumulados	00.2s
Insertar Centros Seleccionados	00.2s
Insertar Incumplidores del Plan Acumulado de los Centros Seleccionados	00.2s
Insertar OACE de los Centros con Deficiencias	00.2s
Insertar Desglose de los Planes	00.2s
Insertar Centros Cortados por Incumplimiento	00.2s
Generar Reporte del Consumo en Barras Diario	00.357s
Generar Reporte de los Consumos Acumulados	00.355s
Generar Reporte de los Centros Seleccionados	00.355s

Generar Reporte de los Incumplidores del Plan Acumulado de los Centros Seleccionados	00.355s
Generar Reporte de los OACE de los Centros con Deficiencias	00.355s
Generar Reporte del Desglose de los Planes	00.355s
Generar Reporte de los Centros Cortados por Incumplimiento	00.355s
Modificar Consumo del Plan Energético	00.8s
Modificar Consumo del Real Energético	00.8s
Modificar Información de los Consumos Acumulados	00.8s
Modificar Centros Seleccionados	00.8s
Modificar Incumplidores del Plan Acumulado de los Centros Seleccionados	00.8s
Modificar OACE de los Centros con Deficiencias	00.8s
Modificar Desglose de los Planes	00.8s
Modificar Centros Cortados por Incumplimiento	00.8s
Buscar Consumo del Plan Energético	00.325s

Buscar Consumo del Real Energético	00.325s
Buscar Información de los Consumos Acumulados	00.325s
Buscar Centros Seleccionados	00.325s
Buscar Incumplidores del Plan Acumulado de los Centros Seleccionados	00.325s
Buscar OACE de los Centros con Deficiencias	00.325s
Buscar Desglose de los Planes	00.325s
Buscar Centros Cortados por Incumplimiento	00.325s

Estos valores pueden variar en dependencia de la cantidad de datos almacenados en la Base de Datos (volumen de información).

Anexo 3

	Requisitos	2	2.2	3
1	Insertar Consumo del Plan Energético	No Pasó	Pasó	Pasó
2	Insertar Consumo del Real Energético	No Pasó	Pasó	Pasó
3	Insertar Información de los Consumos Acumulados	Pasó	Pasó	Pasó
4	Insertar Centros Seleccionados	Pasó	Pasó	Pasó
5	Insertar Incumplidores del Plan Acumulado de los Centros Seleccionados	Pasó	Pasó	Pasó
6	Insertar OACE de los Centros con Deficiencias	Pasó	Pasó	Pasó
7	Insertar Desglose de los Planes	Pasó	Pasó	Pasó
8	Insertar Centros Cortados por Incumplimiento	Pasó	Pasó	Pasó
9	Generar Reporte del Consumo en Barras Diario	No Evaluado	No Evaluado	No Pasó
10	Generar Reporte de los Consumos Acumulados	No Evaluado	No Evaluado	No Pasó
11	Generar Reporte de los Centros Seleccionados	No Evaluado	No Evaluado	Pasó
12	Generar Reporte de los Incumplidores del Plan	No	No	Pasó

	Acumulado de los Centros Seleccionados	Evaluado	Evaluado	
13	Generar Reporte de los OACE de los Centros con Deficiencias	No Evaluado	No Evaluado	Pasó
14	Generar Reporte del Desglose de los Planes	No Evaluado	No Evaluado	Pasó
15	Generar Reporte de los Centros Cortados por Incumplimiento	No Evaluado	No Evaluado	Pasó
16	Modificar Consumo del Plan Energético	Pasó	Pasó	Pasó
17	Modificar Consumo del Real Energético	Pasó	Pasó	Pasó
18	Modificar Información de los Consumos Acumulados	No Pasó	No Pasó	Pasó
19	Modificar Centros Seleccionados	No Pasó	Pasó	Pasó
20	Modificar Incumplidores del Plan Acumulado de los Centros Seleccionados	No Pasó	Pasó	Pasó
21	Modificar OACE de los Centros con Deficiencias	No Pasó	Pasó	Pasó
22	Modificar Desglose de los Planes	No Pasó	Pasó	Pasó
23	Modificar Centros Cortados por Incumplimiento	No Pasó	Pasó	Pasó
24	Buscar Consumo del Plan Energético	No Pasó	Pasó	Pasó

25	Buscar Consumo del Real Energético	No Pasó	Pasó	Pasó
26	Buscar Información de los Consumos Acumulados	Pasó	Pasó	Pasó
27	Buscar Centros Seleccionados	No Pasó	Pasó	Pasó
28	Buscar Incumplidores del Plan Acumulado de los Centros Seleccionados	No Pasó	Pasó	Pasó
29	Buscar OACE de los Centros con Deficiencias	No Pasó	Pasó	Pasó
30	Buscar Desglose de los Planes	No Pasó	Pasó	Pasó
31	Buscar Centros Cortados por Incumplimiento	No Pasó	No Pasó	Pasó

Anexo 4



Centro de Desarrollo

Aval de Calidad de Software

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

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

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

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

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

Ing. Maidel Ojeda Cruz
Asesora de Calidad de SW



Msc. Yamila Vigil Regalado
Directora Centro de Desarrollo

Glosario de Términos

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

Código abierto: en inglés: Open Source, es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código.

Eventos: en la informática existe la programación orientada a eventos, en la que un evento es un mensaje de software que indica que algo ha ocurrido, como un tecleo o un click de un mouse. En el control de procesos, un evento es una ocurrencia que ha ocurrido y que ha sido registrado.

Excel: Microsoft Excel es una aplicación para manejar hojas de cálculo. Este programa es desarrollado y distribuido por Microsoft, y es utilizado normalmente en tareas financieras y contables.

Framework estructura conceptual y tecnológica de soporte definida, normalmente, con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros programas para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Full-duplex: es utilizado en las telecomunicaciones para definir a un sistema que es capaz de mantener una comunicación bidireccional, enviando y recibiendo mensajes de forma simultánea.

J2EE: es una plataforma de programación, parte de la Plataforma Java, para desarrollar y ejecutar software de aplicaciones en el lenguaje de programación Java con arquitectura de N capas distribuidas y que se apoya ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones.

JSP: es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

ORM: es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, utilizando un motor de persistencia.

Plugins: es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica. Esta aplicación adicional es ejecutada por la aplicación principal e interactúan por medio de la API.

POO: es programación orientada a objetos o POO (OOP según sus siglas en inglés) es un paradigma de programación que usa objetos y sus interacciones, para diseñar aplicaciones y programas informáticos.

TCP: en español (Protocolo de Control de Transmisión), es uno de los protocolos fundamentales en Internet. Da soporte a muchas de las aplicaciones más populares de Internet (navegadores, intercambio de ficheros, clientes ftp) y protocolos de aplicación HTTP, SMTP, SSH y FTP.

XML son las siglas de Extensible Markup Language, una especificación/lenguaje de programación desarrollada por el W3C. XML es una versión de SGML, diseñado especialmente para los documentos de la web.