



INSTITUTO SUPERIOR MINERO METALÚRGICO

“Dr. Antonio Núñez Jiménez”.

Facultad de Metalurgia - Electromecánica

TRABAJO DE DIPLOMA EN OPCIÓN AL TÍTULO DE INGENIERO INFORMÁTICO

**SISTEMA DE GESTIÓN ENERGÉTICA ASISTIDA PARA EL ISMMM-
MÓDULO DE CONTROL DEL CONSUMO ENERGÉTICO.**

Autor:

Alexei Cala Hinojosa

Tutor:

Ing. José Rolando Pérez Sandó

Consultante:

Dr. Arístides Alejandro Legrá Lobaina

Moa, Holguín, Cuba

“Año del 53 Aniversario del Triunfo de la Revolución”

DECLARACIÓN DE AUTORÍA

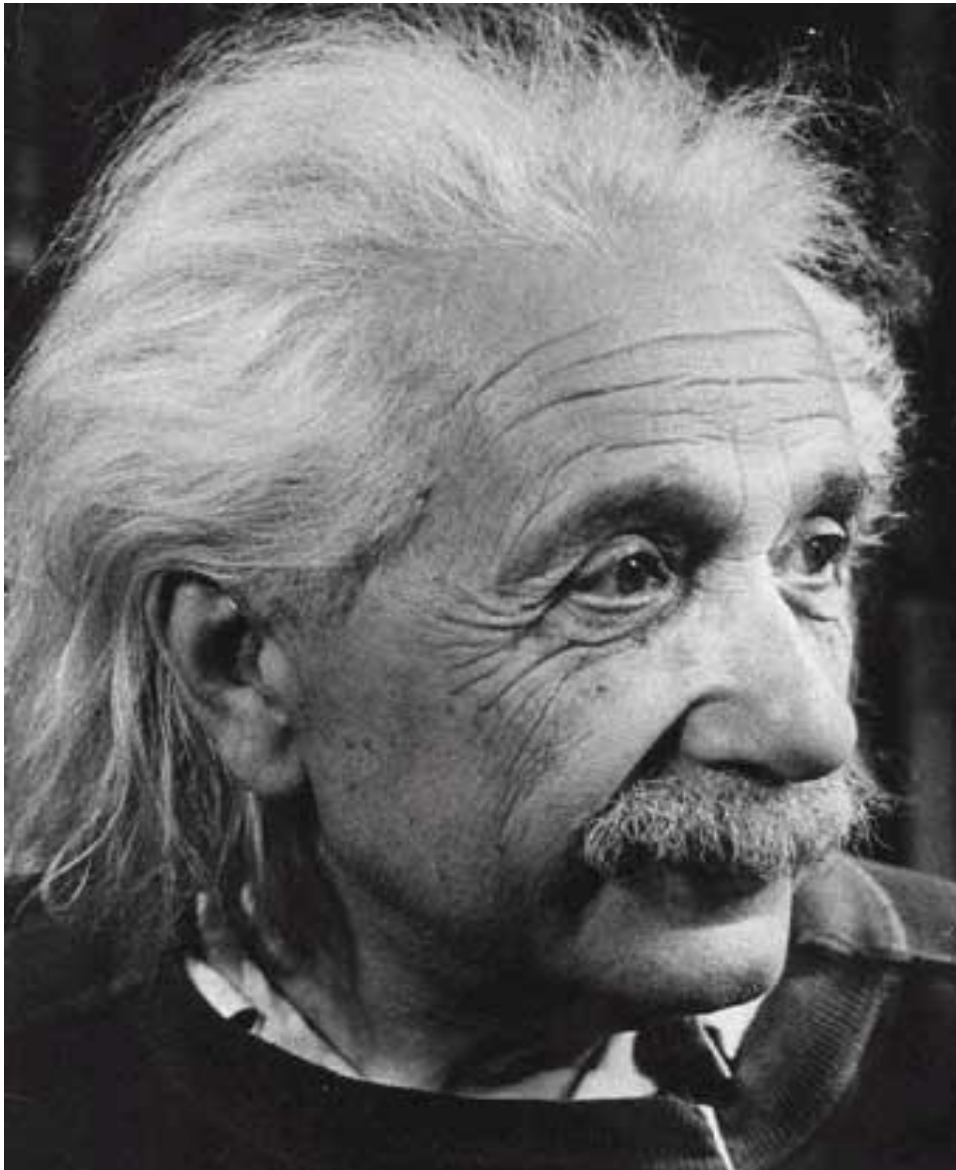
Declaro que soy el único autor de este trabajo y autorizo al Instituto Superior Minero Metalúrgico de Moa para que hagan el uso que estimen pertinente con el mismo.

Para que así conste firmo la presente a los ____ días del mes _____ del año 2011.

Firma del autor

Firma del tutor

FRASE



“La energía no se crea ni se destruye, se transforma”

Albert Einstein

DEDICATORIA

*Este trabajo va dedicado especialmente a mi madre
, Nini, por haberme inculcado los deseos de seguirme superando
y dar siempre lo mejor de si para que este sueño sea posible y
a mi padre, Nello, porque me enseñó que todo el mundo tiene algo que
aprender
y que brindar a los demás.*

A mi abuela, Marcia, por haberme inculcado siempre los buenos principios

A mi tía Chelo, por darme siempre buenos consejos.

*A mi tío Tico, porque se que es una de las personas que más confía en mi
educación.*

*A mi hermano Alexander, que siempre me ha brindado apoyo cuando lo he
necesitado.*

*A mi hermana Amarilis, quien siempre ha confiado en mí y si siempre ha
estado a mi lado en todo momento.*

A mi cuñada Yari, a quien quiero como si fuera mi hermana también.

A Liamelita, la sobrinita más linda del mundo.

AGRADECIMIENTOS

Primeramente quisiera agradecer a la luz de mi vida; mis padres, que sin ellos nada de esto hubiera sido posible.

A mi familia en general por darme siempre apoyo cuando lo necesité.

A mi amigo de infancia Eidier por demostrarme que los verdaderos amigos se prueban en los momentos difíciles y en esos momentos él me supo extender su mano.

A mi colega y amigo Yanquiel y a su novia Eilen por ayudarme en todo lo que pudieron, a ellos les voy a estar eternamente agradecido por su hospitalidad y confianza.

A mi grupo, que es sin duda el mejor grupo del mundo

A Pikirí, por su tutoría ejemplar

Al Mello por su ayuda desinteresada

A Legrá por darme la oportunidad de participar en este proyecto

A los vecinos del barrio: Eider, Magalis, Doralis, Arelis, Jadier, Daimis, la Moña, Robin, Shaili, Gotero, Yoa, Jorgito, Lisbeth, Pachi, Indira, Coco, Leo, Rubiel, Reinier.

A todos los que de una forma u otra aportaron su granito de arena en la realización de este trabajo.

RESUMEN

El mundo actual gira y se sustenta sobre el flujo constante de información. Las necesidades de acelerar este flujo impulsan a todos al uso de las TIC (Tecnologías de la Informática y las Comunicaciones). El ISMMM (Instituto Superior Minero Metalúrgico de Moa) no está exento de este proceso evolutivo de las ciencias informáticas. La Vicerrectoría de Aseguramiento del ISMMM, se desenvuelve en un entorno de niveles considerables de información provenientes del control del consumo energético en la institución, esta información es manejada de forma manual. La gestión de la información se ha convertido en un tema de amplia significación, lo que propicia la necesidad de crear un sistema automatizado que permita llevar el control del consumo energético monitoreados por la Vicerrectoría de Aseguramiento. El objetivo de esta investigación es obtener un módulo que permita realizar el control del consumo energético en el ISMMM, garantizando la gestión de la información de manera ágil y segura. El uso de esta aplicación facilitará el trabajo en Vicerrectoría de Aseguramiento, durante el Proceso de control del consumo energético. Además permitirá la creación y refinamiento de historiales, sin los errores que se cometen actualmente, contando con datos fiables para la toma de futuras decisiones.

ABSTRACT

The current world draws and it nourishes on the constant flux of information. The needs to hurry this flux impels to all according to custom of the tic (technologies of the computer science and the communications). The ISMMM (Superior Institute Metallurgical Miner of Moa) is not exempt of this evolutionary process of the computer's sciences. It would vice-rectoring it of securing of the ISMMM, develops to him in an environment of considerable levels of coming information of the control of the energy consumption in the institution, this information is managed in a manual way. The step of the information has converted in a topic of enlarges meaning, which propitiates the need to create an automatized system that permits take the control of the energy consumption monitoreados for vice-rectoring it of securing. The objective of this investigation is to obtain a module that it permits carry out the control of the energy consumption in the ISMMM, by guaranteeing the step of the information in an agile way and sure. The use of this application will facilitate the work in vice-rectoring of securing, during the process of control of the energy consumption. Moreover it will permit the creation and refinement of histories, without the errors that it is committed at present, counting on reliable data for takes it of decisions reversions.

INDICE

ÍNDICE DE FIGURAS	X
ÍNDICE DE TABLAS	X
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	6
1.1. ESTADO DEL ARTE	6
1.1.1. Antecedentes en el ISMMM	6
1.2. EL PROCESO DE CONTROL DEL CONSUMO ENERGÉTICO EN EL ISMMM	7
1.3. SISTEMAS AUTOMATIZADOS EXISTENTES VINCULADOS AL CAMPO DE ACCIÓN.....	8
1.4. TENDENCIAS Y TECNOLOGÍAS ACTUALES.....	9
1.4.1. Aplicaciones WEB. Ventajas y Desventajas	10
1.4.2. Servidores para aplicaciones WEB	12
1.4.3. Lenguajes de Programación WEB	14
1.4.3. Entornos de programación Web	16
1.4.4. Protocolo TCP/IP	17
1.4.5. Sistemas Gestores de Base de Datos (SGBD)	19
1.4.6. AjaxCRUD	24
1.5. ARQUITECTURA	24
1.5.1. Patrones Arquitectónicos.....	24
1.5.2. Arquitectura en capas	25
1.5.3. Modelo Cliente-Servidor	28
1.5.4. Dispositivos de Hardware de Control Automático	30
1.5.5. Arquitectura de los Sistemas Informáticos de Control utilizando NPort Server Lite.	30
1.6. METODOLOGÍA DE DESARROLLO DE SOFTWARE.....	31
1.7. METODOLOGÍA XP	32
1.7.1. Valores de la XP.....	32
1.7.2. Proceso XP	32
1.7.3. Prácticas XP.....	33
1.7.4. ¿Por qué elegir XP?	35
1.7.5. Fases de la metodología XP.....	36
1.8. HERRAMIENTAS CASE (COMPUTER – AIDED SOFTWARE ENGINEERING)	38
1.9. LENGUAJE UNIFICADO DE MODELADO (UML)	39
1.10. ARQUITECTURA A UTILIZAR	40
1.11. TECNOLOGÍAS Y HERRAMIENTAS A UTILIZAR	41
CAPÍTULO 2: PLANEACIÓN Y DISEÑO	43
2.1. PERSONAL RELACIONADO CON EL SISTEMA.....	43

2.2.	LISTA DE RESERVA	43
2.3.	HISTORIAS DE USUARIO.....	44
2.4.	PLANIFICACIÓN DE ENTREGAS	46
2.4.1.	<i>Estimación de Esfuerzo por Historias de Usuario</i>	<i>47</i>
2.4.2.	<i>Planificación de Iteraciones.....</i>	<i>48</i>
2.4.3.	<i>Plan de duración de las Iteraciones</i>	<i>49</i>
CAPÍTULO 3: DESARROLLO Y PRUEBAS		51
3.1	MODELO DE DATOS	51
3.2	CLASES, RESPONSABILIDADES Y COLABORADORES.....	52
3.3	DESARROLLO DE LAS ITERACIONES.....	55
3.3.1	<i>Tareas por Historias de Usuario</i>	<i>55</i>
3.4	PRUEBAS	57
3.4.1	<i>Desarrollo dirigido por Pruebas.....</i>	<i>57</i>
3.4.2	<i>Pruebas de Aceptación</i>	<i>58</i>
CAPÍTULO 4: ESTUDIO DE FACTIBILIDAD		60
4.1.	EVALUACIÓN COSTO-BENEFICIO	60
4.2.	EFFECTOS ECONÓMICOS.	61
4.3.	ELEMENTOS PARA IDENTIFICAR LOS COSTOS Y BENEFICIOS DEL PROYECTO	62
4.4.	FACTIBILIDAD ECONÓMICA.....	63
4.5.	EVALUACIÓN ECONÓMICA.....	64
4.5.1.	<i>Beneficios tangibles</i>	<i>66</i>
4.5.2.	<i>Beneficios y Costos Intangibles en el proyecto</i>	<i>66</i>
CONCLUSIONES		69
RECOMENDACIONES		70
BIBLIOGRAFÍA.....		71
GLOSARIO DE TÉRMINOS		75
ANEXOS.....		76
ANEXO 1: HISTORIAS DE USUARIOS.....		76
ANEXO 2: TAREAS DE INGENIERÍA		80
ANEXO 3: TARGETAS CRC		95
ANEXO 4: PRUEBAS DE ACEPTACIÓN.....		99

ÍNDICE DE FIGURAS

Figura 1.1. Modelo vista controlador	25
Figura 1.2. Arquitectura en tres capas.	27
Figura 1.3 Arquitectura de los Sistemas Informáticos de Control utilizando NPort Server Lite	30
Figura. 3.1. Modelo de datos del módulo de control del consumo energético	52
Figura 4.1 Punto de equilibrio.	68

ÍNDICE DE TABLAS

Tabla 1 Personal relacionado con el Sistema	43
Tabla 2 Planilla de historia de usuario.	46
Tabla. 3 HU No.1: Gestionar el registro de control por Locales.	46
Tabla 4 Estimación de esfuerzo por historias de usuario.....	47
Tabla 5 Plan de duración de las iteraciones	49
Tabla 6 Tarjeta CRC para la clase Mediciones	53
Tabla 7 Tarjeta CRC para la clase MedicionDAO.....	54
Tabla 8 Tarjeta CRC para la clase MedicionServicio	54
Tabla 9 Tareas por Historias de Usuario	56
Tabla 10 Historias abordadas en la primera iteración	56
Tabla 11 Historias abordadas en la segunda iteración	56
Tabla 12 Historias abordadas en la tercera iteración	57
Tabla 13 Planilla de prueba de aceptación.	58
Tabla 14 Costos en moneda libremente convertible	65
Tabla 15 Costos en Moneda Nacional	65
Tabla16. HU No.16: Gestión de control de actividades de transportación.....	76
Tabla 17. HU No.3: Gestión de mediciones.....	77
Tabla 18. HU No.4: Gestión de contadores.	77
Tabla 19. HU No.5: Conexión con MULTILIN.	78
Tabla 20. HU No.6: Reportes de consumo mensual	78
Tabla 21. HU No.7: Crear Filtro de Locales.	79
Tabla 22. Tarea de Ingeniería No.1: Insertar Registro de control del local	80
Tabla 23. Tarea de Ingeniería No.2: Mostrar registro de control del local	81

Tabla 24. Tarea de Ingeniería No.3: Modificar registro de control del local -----	81
Tabla 25. Tarea de Ingeniería No.4: Eliminar registro de control del local -----	82
Tabla 26 Tarea de Ingeniería No.5: Insertar control de actividades de transportación-----	82
Tabla 27. Tarea de Ingeniería No.6: Mostrar control de actividades de transportación----	83
Tabla 28. Tarea de Ingeniería No.7: Modificar control de actividades de transportación---	83
Tabla 29. Tarea de Ingeniería No.8: Eliminar control de actividades de transportación ---	84
Tabla 30. Tarea de Ingeniería No.9: Actualización del consumo mensual de combustible	84
Tabla 31. Tarea de Ingeniería No.10: Insertar mediciones de contadores -----	85
Tabla 32. Tarea de Ingeniería No.11: Mostrar mediciones de contadores -----	85
Tabla 33. Tarea de Ingeniería No.12: Modificar mediciones de contadores -----	86
Tabla 34. Tarea de Ingeniería No.13: Eliminar mediciones de contadores -----	86
Tabla 35. Tarea de Ingeniería No.14: Calcular el consumo del día-----	87
Tabla 36. Tarea de Ingeniería No.15: Insertar contadores-----	87
Tabla 37. Tarea de Ingeniería No.16: Mostrar contadores-----	88
Tabla 38. Tarea de Ingeniería No.17: Modificar contadores -----	88
Tabla 39. Tarea de Ingeniería No.18: Eliminar contadores -----	89
Tarea 40. de Ingeniería No.19: Conectar el sistema con MULTILIN -----	89
Tabla 41. Tarea de Ingeniería No.20: Saber estado de la conexión con MULTILIN -----	90
Tabla 42. Tarea de Ingeniería No.21: Lista de locales por áreas administrativas -----	90
Tabla 43. Tarea de Ingeniería No.22: Lista de locales por subáreas administrativas -----	91
Tabla 44. Tarea de Ingeniería No.23: Lista de locales por edificios -----	91
Tabla 45. Tarea de Ingeniería No.24: Lista de locales por tipos de locales-----	92
Tabla 46. Tarea de Ingeniería No.25: Lista de locales por áreas de consumo final-----	92
Tabla 47. Tarea de Ingeniería No.26: Reporte de consumo mensual por locales -----	93
Tabla 48. Tarea de Ingeniería No.27: Generar reporte de consumo mensual de combustible en transportación -----	93
Tabla 49. Tarea de Ingeniería No.28: Generar reporte de consumo por Áreas -----	94
Tabla 50. Tarjeta CRC No.1: Control Local -----	95
Tabla 51. Tarjeta CRC No.2: Control Local Acceso a Datos -----	96
Tabla 52. Tarjeta CRC No.3: Servicio ControlLocal -----	96
Tabla 53. Tarjeta CRC No.4: Contadores-----	96
Tabla 54. Tarjeta CRC No.5: Tarjeta CRC No.5: Transportación-----	97
Tabla 55. Tarjeta CRC No.5: MULTILIN-----	97
Tabla 56. Tarjeta CRC No.6: Filtro -----	97
Tabla 57. Tarjeta CRC No.7: Medición -----	98

Tabla 58. Tarjeta CRC No.8: MediciónDAO -----	98
Tabla 59. Tarjeta CRC No.9: Connect -----	98
Tabla 60. Prueba de aceptación No 1. Gestión de mediciones -----	99
Tabla 61. Prueba de aceptación No 2. Filtro-----	99
Tabla 62. Prueba de aceptación No 3. Actividades de transportación-----	100
Tabla 63. Prueba de aceptación No 4. Contadores -----	100
Tabla 64. Prueba de aceptación No 5. Control Local -----	101
Tabla 65. Prueba de aceptación No 6. Conexión con MULTILIN -----	102
Tabla 66. Prueba de aceptación No 7. Reportes de consumo mensual -----	102



INTRODUCCIÓN

La informatización de sistemas es tarea común hoy en día. Automatizar cada actividad es un camino que conduce al desarrollo de nuevas tecnologías. La informática y las telecomunicaciones se imponen si se aspira a un futuro de operaciones rápidas y fiables, donde aparecen los sistemas automatizados como parte esencial en los procesos de manejo de información. En el mundo actual las TIC se abren paso en cada sector de la vida, ya sea empresarial, social, comercial u otro cualquiera.

Nuestro país actualmente se encuentra enfrascado en un proceso de perfeccionamiento de su modelo económico, para ello se hace de suma importancia el uso eficiente de las tecnologías actuales en función de lograr un avance significativo en el control de los recursos energéticos y su uso racional.

El ISMMM realiza de manera sistemática la planificación de las actividades a realizar en un período de tiempo determinado, repercutiendo esta planificación sobre los recursos energéticos asignados. En este proceso los representantes administrativos de las distintas áreas hacen peticiones basados en sus necesidades específicas. Debido a que al momento de realizarse dichas peticiones ya el ISMMM cuenta con un presupuesto asignado para gastos, en cuestión de recursos energéticos, no siempre se pueden satisfacer todas las demandas. Por ello la institución debe seguir una estrategia de forma tal que se garantice la solución de aquellas actividades reconocidas como una prioridad.

De la realización de este proceso de asignación de recursos se determina como uno de los factores a enfrentar, realizar una asignación de recursos eficiente, teniendo en cuenta los factores actuales y bajo el análisis correcto de los historiales. Actualmente dicho análisis no puede ser llevado a cabo, pues no se cuenta con un registro confiable de historiales para identificar las necesidades actuales debido a que no está establecido un sistema de control eficiente sobre el consumo energético, que defina los aspectos necesarios para el posterior análisis de los datos capturados. Esto imposibilita que el procesamiento de los datos se realice a través de modelos que conlleven a la mejora de las tomas de decisiones, pues la inexistencia de historiales legibles hace imposible la



aplicación de modelos estadísticos y métodos de estimación.

El energético del ISMMM necesita realizar las mediciones a los contadores energéticos las cuales son tabuladas en una hoja de Excel de difícil acceso desde otro puesto de trabajo, estas mediciones son necesarias para saber el consumo de electricidad al finalizar el mes. Con estas mediciones se hace posible calcular el consumo total de electricidad pero no se puede conocer el consumo aproximado de cada local en base a lo que este se planificó debido a que el trabajo debe hacerse de forma manual lo cual constituye una tarea tediosa y con grandes posibilidades de cometerse errores. Al no poder hacer una estimación del consumo de cada local del ISMMM, se hace imposible la estimación del consumo desglosado por cada estructura de consumo energético (áreas administrativas, edificios, etc.). Actualmente el ISMMM cuenta con dos instrumentos de control automático, los cuales no se están utilizando porque no existe una herramienta que proporcione el acceso al los mismos y el instituto no cuenta con un trabajador que realice este trabajo. En las condiciones económicas actuales de nuestro país, se hace imprescindible el ahorro y control del combustible consumido, por lo que el ISMMM necesita obtener un registro de control de todas las actividades de transportación que se realizan durante el mes, las cuales se registran en papel con la posibilidad de que se cometan errores o pérdidas de información por deterioro de las fuentes. Toda esta problemática constituye el **Problema Social** de esta investigación.

En el instituto se realizó un pequeño sistema sobre una hoja electrónica de cálculo en Excel el cual se está utilizando actualmente a pesar de presentar varios problemas. También se realizaron varios trabajos de tesis en los cuales se desarrollaron partes de un sistema informático para la gestión energética, una de estas partes comprendía el control. En estos momentos no se encuentra en explotación debido a la presencia de diversos errores en la presentación de la información, errores de integración entre los módulos implementados y la no consistencia con el sistema de información actual de gestión energética en el ISMMM.

Por los anteriormente expuesto y analizado, acorde con las necesidades existentes y



experiencias obtenidas en la Vicerrectoría de Aseguramiento, se evidencia que el Proceso de control del consumo energético no se realiza de manera eficiente, identificándose como **Problema Científico** a resolver: *La necesidad de un módulo para el sistema GEA que permita realizar el control del consumo energético en el ISMMM.*

Este problema se enmarca en el **Objeto de Estudio**: *Proceso de Informatización del Control del Consumo Energético.* El **Campo de Acción** abarcado es: *Informatización del Proceso de control del consumo energético en el ISMMM.*

El **Objetivo General** de la investigación consiste en el desarrollo de un módulo para el sistema GEA que permita realizar la gestión del control del consumo energético en el ISMMM.

Las **Tareas** que se llevarán a cabo para darle cumplimiento a los objetivos trazados son:

- Estudiar de los diferentes Sistemas de control del consumo energético.
- Estudiar las tecnologías seleccionadas para el desarrollo de la aplicación.
- Diseñar el sistema para garantizar que el producto satisfaga las necesidades de los usuarios.
- Estudiar detalladamente el proceso de control del consumo energético.
- Identificar los requerimientos.
- Diseñar la base de datos.
- Implementar el Módulo de Control de Consumo Energético.
- Realizar pruebas funcionales.
- Realizar Manual de Usuario.

Para dar respuesta al problema científico se plantea como **Idea a Defender** que la creación de un sistema automatizado para el control del consumo energético, agilizará y mejorará el desarrollo de esta actividad en la entidad, y por consiguiente la optimización de la gestión energética del centro.

Los **Métodos de Investigación** utilizados son los teóricos y empíricos. Los métodos empíricos empleados fueron la observación y el análisis de documentos para la



recopilación de la información. La observación se utilizó para conocer la funcionalidad de la Vicerrectoría de Aseguramiento, el CEETAM y el comportamiento del problema. Mediante el análisis de documentos se pudo conocer el funcionamiento actual del proceso de control del consumo energético en el ISMMM.

Los métodos teóricos utilizados fueron: el método histórico y lógico para la búsqueda de los antecedentes del software. El análisis y síntesis para la recopilación y el procesamiento de la información obtenida en los métodos empíricos y arribar a las conclusiones de la investigación. Mediante el método hipotético deductivo se pudo elaborar la idea a defender y se verificó la misma.

Las etapas de la investigación son:

- El estudio de las distintas metodologías de desarrollo de software, de las herramientas, tecnologías y patrones arquitectónicos para hacer una selección de las mejores, y utilizarlos en la confección del software de acuerdo a sus características.
- La contextualización del problema, enmarcando los objetivos específicos requeridos por el cliente.
- Análisis y diseño del software con todas las especificaciones requeridas.
- Realización del estudio de factibilidad.

El presente trabajo consta de introducción, 4 capítulos, conclusiones, recomendaciones, bibliografía, glosario de términos y anexos.

En el Capítulo uno **Fundamentación Teórica**, se analizan aspectos relacionados con el Control del Consumo Energético, conceptualización y características, se realiza un estudio acerca de los diferentes sistemas existentes vinculados al campo de acción, además se presenta la metodología y las herramientas a utilizar en el desarrollo del sistema propuesto.

En el Capítulo dos **Planeación y Diseño**, se hace uso de la metodología expuesta en el



capítulo inicial para el desarrollo del proyecto, abordando en detalles cada una de sus fases.

El Capítulo tres **Desarrollo y Pruebas**, se presentan los principales métodos y definiciones dentro de la implementación de los flujos de trabajo. Se describen además las pruebas realizadas y sus resultados.

En el Capítulo cuatro **Estudio de Factibilidad y Sostenibilidad**, se realiza un estudio de los esfuerzos requeridos para la realización del sistema, y se valora la sostenibilidad del producto.



CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En el mundo actual, las TIC han evolucionado cuantiosamente y junto con ella la producción de diversos productos de software. En este capítulo se realiza un esbozo de las características del control del consumo energético, además se hace una descripción sobre las aplicaciones Web. Se muestran algunos de los sistemas automatizados existentes unido al campo de acción y se exponen tanto la metodología a seguir en el desarrollo de la aplicación, como las herramientas a emplear en la misma.

1.1. Estado del arte

Importancia del control del consumo energético

El Control del consumo de energía nos permite:

- Conocer si las variables evaluadas están bajo control o no.
- Conocer los límites en que se puede considerar la variable bajo control.
- Identificar los comportamientos que requieren explicación e identificar las causas no aleatorias que influyen en el comportamiento de los consumos.
- Conocer la influencia de las acciones correctivas sobre los consumos o costos energéticos.

1.1.1. Antecedentes en el ISMMM

Hoja de cálculos en Excel del energético del ISMMM

En la Vicerrectoría de Aseguramiento del ISMMM, existe un pequeño sistema diseñado en una hoja electrónica de cálculo de Excel. El energético del ISMMM realiza las mediciones de los contadores y las registra en esta hoja de Excel de difícil acceso desde otro puesto de trabajo, estas mediciones son necesarias para saber el consumo de electricidad al finalizar el mes. Con estas lecturas se hace posible calcular el consumo total de electricidad pero no se puede conocer el consumo aproximado de cada local en base a lo que este se planificó. Al no poder hacer una estimación del consumo de cada local del



ISMMM, se hace imposible la estimación del consumo desglosado por cada estructura de consumo energético (áreas administrativas, edificios, etc.).

Al estar diseñado en Excel no permite acceder desde cualquier puesto de trabajo sin transportarlo, no es seguro, no cumple con la filosofía de software libre, no permite visualizar el consumo de un determinado portador en un local determinado, no permite determinar las actividades de mayor consumo energético, entre otros inconvenientes.

Sistema para el control y gestión de portadores energéticos

En el instituto se desarrolló un sistema informático para el control del consumo de los portadores energéticos el cual no se está utilizando por el siguiente inconveniente. Actualmente en el instituto se está llevando a cabo un proceso de perfeccionamiento de la gestión energética con la creación de un Sistema para la Gestión Energética Asistida (GEA) el cual comprende tres componentes fundamentales; la planificación, el control y el pronóstico del consumo energético. El sistema informático para el control y gestión de portadores energéticos actualmente no se está utilizando fundamentalmente porque presenta inconsistencia con la nueva base de datos creada para el sistema GEA. Además esta aplicación no se puede integrar con dicho sistema porque no está basada en el control de la fase de planificación del consumo.

Se determinó diseñar un sistema que se ajuste a las necesidades y requerimientos específicos del centro. Con el presente trabajo se pretende desarrollar una aplicación Web que permita realizar el control del consumo energético basado en el chequeo de las actividades planificadas durante el mes, así como la gestión de los equipos de control y sus mediciones, con un entorno agradable, flexible, adaptable y sencillo, permitiendo un acceso rápido y seguro a la información desde cualquier puesto de trabajo, que cumpla con la ética de software libre.

1.2. El proceso de control del consumo energético en el ISMMM

En el ISMMM se lleva a cabo el control del consumo de cuatro portadores energéticos fundamentalmente: diesel, gasolina, electricidad y Fuel-oil. En el caso de la electricidad, el instituto cuenta con un energético, el cual realiza la lectura de los tres contadores



presentes en la institución, dichas mediciones son tabuladas en una hoja electrónica de cálculo las cuales serán usadas para determinar el consumo total de electricidad al concluir el mes. El control del combustible consumido por los vehículos es llevado a cabo de la siguiente forma, primeramente se determina el índice de consumo de cada vehículo para el mes que se desea controlar. Para realizar el control de combustible consumido diariamente, se utiliza un modelo de control diario, el cual es actualizado todos los días al finalizar la actividad de transportación. Una actividad de consumo energético importante realizada en el ISMMM es la cocción de alimentos. Para realizar el control del consumo energético en esta actividad, se registra la fecha de llenado de un depósito de 1.75 t, presente en el local.

El control de consumo automático

Este tipo de control es realizado por un dispositivo de control automático. Actualmente existen en el mercado diversos tipos de dispositivos de control de este tipo, entre los más utilizados se encuentra el MULTILIN. Nuestro centro cuenta con dos dispositivos de este tipo, los cuales no se encuentran en funcionamiento actualmente. En este trabajo se pretende realizar una contribución al control automático en el centro, brindando la posibilidad de conectar el sistema con MULTILIN.

1.3. Sistemas automatizados existentes vinculados al campo de acción

CeleradorS2C (DESOFT, 2009)

Descripción resumida del producto

Permite el control del consumo de combustible por tarjeta magnética. Emite una serie de informes estadísticos por centros de costo, unidades de costo y vehículos, además de los comprobantes contables de gastos e ingresos generados por la actividad.

Descripción amplia del producto

Está destinado al control del consumo de combustible por tarjeta magnética en las empresas cubanas. A partir de estas informaciones primarias se pueden obtener una serie de datos estadísticos que son de gran utilidad para saber el gasto de combustible de la



entidad. Puede llevar el control del gasto por unidades de costo, centros de costo y vehículos. Adicionalmente, emite un grupo de comprobantes contables a partir de cada una de las operaciones que se ejecutan tanto de gastos como de ingresos.

Este software no se puede utilizar en el ISMMM porque no se adapta a la estructura energética de la entidad, además el sistema no es tecnología Web lo que dificulta la integración con el nuevo sistema de gestión energética asistida que se está realizando en el ISMMM, sumándole a esto el problema de acceso desde cualquier parte al mismo por parte de los usuarios implicados. (Héctor, 2009)

PQMPC

El PQMPC es un programa basado en Windows para el PQM. Se puede usar para introducir ajustes, leer valores de medida, monitorizar estados y evaluar la calidad de la red. Toda la información que continuamente recoge el PQM puede ser transferida a un suministrador externo para el display, control o análisis a través del interfaz de comunicaciones.

Existen pantallas para supervisar todos los valores de medida como intensidad, tensión o potencia. Se pueden representar también el estado de las alarmas y los ajustes de control.

Este software no se utiliza en el instituto porque no es tecnología Web lo cual dificulta su integración con el Sistema GEA (Gestión Energética Asistida), además el módulo que se va a desarrollar no pretende implementar todas las funcionalidades que brinda este programa porque no se encuentran dentro de las necesidades que se plantean actualmente. (Moxa.com, 2009)

1.4. Tendencias y tecnologías actuales

Ante el incesante avance de las tecnologías, la sociedad, ávida de nuevas herramientas y funcionalidades, exige a los desarrolladores de software nuevos retos y nuevas concepciones para satisfacer sus exigencias, cada vez más ambiciosas. Para satisfacer estas exigencias, los desarrolladores deben buscar nuevas ideas surgiendo así nuevas



metodologías y formas de desarrollo que permiten confeccionar productos cada vez más complejos.

1.4.1. Aplicaciones WEB. Ventajas y Desventajas

Con la llegada de la Web 2.0, propiciada esta por el desarrollo de las Tecnologías de la Información y las Comunicaciones y los nuevos retos de Internet, se abre paso una nueva idea en la concepción del software; el software orientado a servicios o SaaS (acorde a sus siglas en inglés). SaaS es un modelo de distribución del software que proporciona a los clientes el acceso al mismo a través de la red (generalmente Internet), de manera que les libra del mantenimiento de las aplicaciones, de operaciones técnicas y de soporte. Las aplicaciones distribuidas en la modalidad SaaS pueden llegar a cualquier tipo de empresa sin importar su tamaño o su ubicación geográfica. Se trata de un modelo que une el producto (software) al servicio, para dotar a las empresas de una solución completa que permita optimizar sus costes y sus recursos. (CIIN, 2005)

Ventajas WEB: (Graham, 2001)

Compatibilidad multiplataforma: Las aplicaciones Web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones de software descargables. Varias tecnologías incluyendo Java, Flash, ASP y Ajax permiten un desarrollo efectivo de programas soportando todos los sistemas operativos principales.

Actualización: Las aplicaciones basadas en Web están siempre actualizadas con el último lanzamiento sin requerir que el usuario tome acciones pro-activas, y sin necesitar llamar la atención del usuario o interferir con sus hábitos de trabajo, pues no se hace necesario iniciar nuevas descargas y/o procedimientos de instalación (algunas veces imposible cuando usted está trabajando dentro de grandes organizaciones).

Inmediatez de acceso: Las aplicaciones basadas en Web no necesitan ser descargadas, instaladas y configuradas. Usted accede a su cuenta online y están listas para trabajar sin importar cuál es su configuración o su hardware.



Menos requerimientos de memoria: Las aplicaciones basadas en Web tienen menos demandas de memoria RAM de parte del usuario final que los programas instalados localmente. Al residir y correr en los servidores del proveedor, esas aplicaciones basadas en Web usan en muchos casos la memoria de las computadoras donde ellas corren, dejando más espacio para correr múltiples aplicaciones sin incurrir en frustrantes deterioros en el rendimiento.

Menos Bugs: Las aplicaciones basadas en Web deberían ser menos propensas a colgarse y crear problemas técnicos debido a software o conflictos de hardware con otras aplicaciones existentes, protocolos o software personal interno. Con aplicaciones basadas en Web, todos utilizan la misma versión, y todos los bugs pueden ser corregidos tan pronto como son descubiertos.

Precio: Las aplicaciones basadas en Web no requieren la infraestructura de distribución, soporte técnico y marketing requerido por el software descargable tradicional. Esto permite que las aplicaciones online cuesten una fracción de sus contrapartes descargables y no totalmente gratuitas, mientras que ofrecen componentes adicionales y servicios Premium como una opción.

Los datos también van online: El hecho de que el manejo de los datos sea realizado de forma remota libra al usuario de la responsabilidad en la protección de los mismos, y al mismo tiempo logra que los recursos sean accesibles en cualquier momento.

Múltiples usuarios concurrentes: Las aplicaciones basadas en Web pueden ser utilizadas por múltiples usuarios al mismo tiempo. No hay más necesidad de compartir pantallas o enviar instantáneas cuando múltiples usuarios pueden ver e incluso editar el mismo documento de manera conjunta.

Los datos son más seguros: Si bien la ruptura de discos no va a desaparecer, es probable que los usuarios escuchen mucho menos del tema. A medida que las compañías se hagan cargo del almacenamiento de los datos del usuario, granjas de almacenamiento de datos redundantes, altamente fiables, los usuarios van a tener mucho menos riesgo de



perder sus datos debido a una ruptura de disco impredecible o a un virus de la computadora. Las compañías que proveen aplicaciones basadas en Web van a brindar amplios servicios de resguardo de datos ya sea como una parte integral del servicio básico o como una opción paga.

Desarrollar aplicaciones en el lenguaje que usted quiera: Una vez que las aplicaciones han sido separadas de computadoras locales y sistemas operativos específicos, pueden también ser escritas en prácticamente cualquier lenguaje de programación. Debido a que las aplicaciones Web son esencialmente una colección de programas más que un simple programa, ellas podrían ser escritas en cualquier lenguaje de programación existente.

Desventajas WEB: (Masternewmedia.org, 2005)

- Acceso limitado, la necesidad de conexión permanente y rápida a Internet hacen que el acceso a estas aplicaciones no esté al alcance de todos.
- La interactividad no se produce en tiempo real, en las aplicaciones Web cada acción del usuario conlleva un tiempo de espera hasta que se obtiene la reacción del sistema.
- Elementos de interacción muy limitados. En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones Web (mediante formularios principalmente) son muy escasas.
- Diferencias de presentación entre plataformas y navegadores. La falta de estándares ampliamente soportados dificulta el desarrollo de las aplicaciones.

1.4.2. Servidores para aplicaciones WEB

El servidor Web es un programa que corre sobre el servidor que escucha las peticiones HTTP que le llegan y las satisface. Dependiendo del tipo de la petición, el servidor Web buscará una página Web o bien ejecutará un programa en el servidor. De cualquier modo, siempre devolverá algún tipo de resultado HTML al cliente o navegador que realizó la petición. (Vegas, 2002)

Apache



Apache, sustancialmente, es un proyecto nacido para crear un servidor de web estable, fiable y veloz para plataformas Unix. Apache nace, por una parte, de un código ya existente y de una serie de parches (patch) para mejorar su fiabilidad y sus características; de ahí su nombre. (Htmlpoint.com, 2008)

Características: (Ciberaula.com, 2005)

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Es una tecnología gratuita con un código fuente disponible. El hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto. Esta característica le ofrece al software un grado de transparencia tal que es posible determinar en todo momento qué es lo que se está instalando, sin secretos ni puertas traseras.
- Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este, y están ahí para que se instalen cuando se necesiten. Otra cosa importante es que cualquiera que posea alguna experiencia en la programación de C o Perl puede escribir un módulo para realizar una función determinada.
- Trabaja con Perl, PHP y otros lenguajes de script. Perl destaca en el mundo del script y Apache utiliza su parte del pastel de Perl tanto con soporte CGI como con soporte mod perl. También trabaja con Java y páginas JSP. Teniendo todo el soporte que se necesita para tener páginas dinámicas.
- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurarlo para que ejecute un determinado script cuando ocurra un error en concreto.
- Tiene una alta configurabilidad en la creación y gestión de logs³⁰. Permite la creación de ficheros de log a la medida del administrador, de este modo se puede tener un mayor control sobre lo que sucede en el servidor.



Se pueden extender las características de Apache hasta donde la imaginación y los conocimientos lleguen, debido a que el equipo de desarrollo está formado por voluntarios, diseminados por todo el mundo, que sigue manteniendo este servidor de web libre.

XAMPP

Es un paquete formado por un servidor web Apache, una base de datos MySQL y los intérpretes para los lenguajes PHP y Perl. El nombre proviene de X (para cualquier sistema operativo), A (Apache), M (MySQL), P(PHP), P (Perl). El programa esta liberado bajo la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris, y MacOS X. (BLANCO CRIADO, 2008)

XAMPP es regularmente actualizado para incorporar las últimas versiones de Apache/MySQL/PHP y Perl. También incluye otros módulos como OpenSSL, y PhpMyAdmin. Para instalar XAMPP requiere solamente una pequeña fracción del tiempo necesario para descargar y configurar programas por separado.

Oficialmente, los diseñadores de XAMPP solo pretendían su uso como una herramienta de desarrollo, para permitir a los diseñadores de sitios webs y programadores testear su trabajo en sus propios ordenadores sin ningún acceso a Internet. En la práctica sin embargo, XAMPP es utilizado actualmente para servidor de sitios webs en WWW, y con algunas modificaciones es generalmente lo suficientemente seguro para serlo. Una herramienta especial es suministrada para proteger fácilmente las partes más importantes del paquete.

1.4.3. Lenguajes de Programación WEB

PHP

Es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de



comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK34.

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools). Fue creado originalmente por Rasmus Lerdorf en 1994. (Achour, 2005)

Ventajas de PHP

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.

Desventajas de PHP (Dondo, 2005)

- No posee una abstracción de base de datos estándar, sino bibliotecas especializadas para cada motor (a veces más de una para el mismo motor).
- No posee adecuado manejo de internacionalización, unicode, etc.
- Por su diseño dinámico no puede ser compilado y es muy difícil de optimizar.
- Por sus características promueve la creación de código desordenado y complejo de mantener.



- Está diseñado especialmente para un modo de hacer aplicaciones web que es ampliamente considerado problemático y obsoleto (mezclar el código con la creación de la página web).

JSP

La tecnología Java para la creación de páginas web con programación en el servidor. JSP es un acrónimo de Java Server Pages, que en castellano vendría a decir algo como Páginas de Servidor Java. Es una tecnología orientada a crear páginas web con programación en Java. Java Server Pages es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. (Universidad de Valparaíso, 2006)

Esta tecnología es un desarrollo de la compañía Sun Microsystems. La Especificación JSP 1.2 fue la primera que se liberó y en la actualidad está disponible la Especificación JSP 2.1.

Las JSP's permiten la utilización de código Java mediante scripts. Además es posible utilizar algunas acciones JSP predefinidas mediante etiquetas. Estas etiquetas pueden ser enriquecidas mediante la utilización de Librerías de Etiquetas (TagLibs o Tag Libraries) externas e incluso personalizadas.

Con JSP se puede crear aplicaciones web que se ejecuten en variados servidores web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java. Por tanto, las JSP podrán escribirse con el editor HTML/XML habitual. (Millán Tejedor, 2005)

1.4.3. Entornos de programación Web

Dreamweaver: Sin lugar a dudas, es una de las herramientas más utilizadas por los webmasters para el trabajo con aplicaciones visuales en este caso en el diseño e implementación de páginas Web. Se adapta increíblemente a las necesidades de todo tipo de profesional de diseño Web, tanto para lo que prefieren programar el código



directamente en el editor de texto como para los que gustan del ambiente visual. Se trata de un editor de texto especialmente diseñado para trabajar con documentos Web como HTML, PHP, ASP, JavaScript, entre otros.

PhpDesigner 2008 v6.0.0: Es un completo entorno de desarrollo y programación especialmente diseñado para los gurús de PHP, aunque también permite trabajar con comodidad en otros lenguajes de programación como HTML, XHTML, CSS y SQL.

Ofrece toda una serie de asistentes y diálogos integrados que facilitan en todo momento las tareas, además de acceso directo a librerías de código o scripts de uso habitual, utilidades diversas y toda suerte de herramientas, todo ello en una interfaz de diseño sencillo y elegante que se puede personalizar con nada menos que dieciocho temas distintos.

Cuenta con cliente de FTP y navegador de ficheros integrado, utilidades de corrección y autocompletado, búsqueda integrada en Google y soporte para proyectos, además de usar un práctico esquema de color para la sintaxis del código fuente que facilita enormemente la programación.

PhpDesigner soporta: PHP, HTML, XHTML, CSS, Java, Perl, JavaScript, VB, C# y SQL. (Citado en: <http://www.intercambiosvirtuales.org/software/php-designer-2008-v6020-professional>)

1.4.4. Protocolo TCP/IP

En realidad no es un protocolo sino la familia de protocolos TCP/IP (Transmission Control Protocol (TCP) y el Internet Protocol (IP).) de Internet, que permiten la transmisión de información en redes de computadoras. Algunos de los motivos de su popularidad son:

- Independencia del fabricante.
- Soporta múltiples tecnologías.
- Puede funcionar en máquinas de cualquier tamaño.
- Estándar de EEUU desde 1983.



La arquitectura de un sistema en TCP/IP tiene una serie de metas:

- La independencia de la tecnología usada en la conexión a bajo nivel y la arquitectura del ordenador.
- Conectividad Universal a través de la red.
- Reconocimientos de extremo a extremo.
- Protocolos estandarizados

Este protocolo constituye la base sobre la que se soporta Internet y es utilizado para unir computadoras que pueden tener diferentes sistemas operativos y que pueden ser tanto PC como minicomputadoras o computadoras centrales. Su característica principal es que se dividen las tareas para un mejor funcionamiento: La tarea de IP es llevar los datos por paquetes de un sitio a otro. Las computadoras que encuentran las vías para llevar los datos de una red a otra (denominadas enrutadores) utilizan IP para trasladar los datos. En resumen IP mueve los paquetes de datos a granel, mientras TCP se encarga del flujo y asegura que los datos estén correctos. Como otros tantos protocolos, este posee ventajas y desventajas que lo caracterizan: está diseñado para enrutar y tiene un grado muy elevado de fiabilidad, es adecuado para redes grandes y medianas, así como en redes empresariales. Se utiliza a nivel mundial para conectarse a Internet y a los servidores Web. Es compatible con las herramientas estándar para analizar el funcionamiento de la red. Un inconveniente de TCP/IP es que es más difícil de configurar y de mantener que NetBEUI o IPX/SPX; además es algo más lento en redes con un volumen de tráfico medio bajo. Sin embargo, puede ser más rápida en redes con un volumen de tráfico grande donde haya que enrutar un gran número de marcos.

El protocolo TCP/IP se puede utilizar en grandes redes empresariales como por ejemplo en campus universitarios o en complejos empresariales en donde utilizan muchos enrutadores y conexiones a mainframe o a ordenadores UNIX. También se puede utilizar en redes pequeñas en donde 100 ó 200 estaciones de trabajo funcionando con Windows acceden a servicios de intranet o Internet mediante un servidor NT que ofrezca servicios Web a través del servidor de información de Internet (IIS) de Microsoft. (Wikipedia.org)



1.4.5. Sistemas Gestores de Base de Datos (SGBD)

Un Sistema Gestor de Bases de Datos (SGBD) o DBMA es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos.

Algunos ejemplos de SGBD son PostgreSQL, MySQL, SQL Server, etc.

Un SGBD debe permitir:

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

Características de un Sistema Gestor de Base de Datos: (CAVSI, 2004)

Abstracción de la información. Los SGBD ahorran a los usuarios detalles acerca del almacenamiento físico de los datos. Da lo mismo si una base de datos ocupa uno o cientos de archivos, este hecho se hace transparente al usuario. Así, se definen varios niveles de abstracción.

Independencia. La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.

Redundancia mínima. Un buen diseño de una base de datos logrará evitar la aparición de información repetida o redundante. De entrada, lo ideal es lograr una redundancia nula; no obstante, en algunos casos la complejidad de los cálculos hace necesaria la aparición de redundancias.



Consistencia. En aquellos casos en los que no se ha logrado esta redundancia nula, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se actualicen de forma simultánea.

Seguridad. La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra protegida frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipularla o destruirla; o simplemente ante las torpezas de algún usuario autorizado pero despistado. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.

Integridad. Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de hardware, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada.

Respaldo y recuperación. Los SGBD deben proporcionar una forma eficiente de realizar copias de respaldo de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.

Control de la concurrencia. En la mayoría de entornos (excepto quizás el doméstico), lo más habitual es que sean muchas las personas que acceden a una base de datos, bien para recuperar información, bien para almacenarla. Y es también frecuente que dichos accesos se realicen de forma simultánea. Así pues, un SGBD debe controlar este acceso concurrente a la información, que podría derivar en inconsistencias.

PostgreSQL

Es un servidor de base de datos relacional orientada a objetos de software libre, publicado bajo la licencia BSD40.

Como muchos otros proyectos open source, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y



organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).

Principales características:

- Alta concurrencia: Mediante un sistema denominado MVCC (Acceso concurrente Multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

- Amplia variedad de tipos nativos. PostgreSQL provee nativamente soporte para:
- Números de precisión arbitraria.
- Texto de largo ilimitado.
- Figuras geométricas (con una variedad de funciones asociadas)
- Direcciones IP (IPv4 e IPv6).
- Bloques de direcciones estilo CIDR.
- Direcciones MAC.
- Adicionalmente los usuarios pueden crear sus propios tipos de datos, los que pueden ser por completo indexables gracias a la infraestructura GiST de PostgreSQL. Algunos ejemplos son los tipos de datos GIS creados por el proyecto PostGIS. (Eaprende.com, 2001)

Gracias a su licencia BSD, se permite la utilización del código para ser comercializado. Uno de los casos ejemplo es la de Enterprise DB (Postgresql Plus), la cual incluye varios agregados y una interfaz de desarrollo basada en Java. Entre otras empresas que utilizan Postgresql para comercializar se encuentra CyberTech (Alemania), con su producto CyberCluster.

MySQL



Es un sistema de gestión de bases de datos relacional, fue creada por la empresa sueca MySQL AB, la cual tiene el copyright del código fuente del servidor SQL, así como también de la marca.

MySQL es un software de código abierto, licenciado bajo la GPL de la GNU, aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de otra manera, se vulneraría la licencia GPL.

El lenguaje de programación que utiliza MySQL es Structured Query Language (SQL) que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales. (Nieves Borrero, et al., 2007)

Inicialmente, MySQL carecía de algunos elementos esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de esto, atrajo a los desarrolladores de páginas web con contenido dinámico, debido a su simplicidad, de tal manera que los elementos faltantes fueron complementados por la vía de las aplicaciones que la utilizan. Poco a poco estos elementos faltantes, están siendo incorporados tanto por desarrolladores internos, como por desarrolladores de software libre.

En las últimas versiones se pueden destacar las siguientes características principales:

- El principal objetivo de MySQL es velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.



- Flexible sistema de contraseñas (passwords) y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintas lenguas.

Ventajas: (Eaprende.com, 2001)

- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Facilidad de configuración e instalación.
- Soporta gran variedad de Sistemas Operativos
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Conectividad y seguridad

Desventajas:

- Un gran porcentaje de las utilidades de MySQL no están documentadas.
- No es intuitivo, como otros programas (ACCESS).

SQL Server

SQL Server es un conjunto de objetos eficientemente almacenados. Los objetos donde se almacena la información se denominan tablas, y éstas a su vez están compuestas de filas y columnas. En el centro de SQL Server está el motor de SQL Server, el cual procesa los comandos de la base de datos. Los procesos se ejecutan dentro del sistema operativo y entienden únicamente de conexiones y de sentencias SQL.

SQL Server incluye herramientas para la administración de los recursos que el ordenador proporciona y los gestiona para un mejor rendimiento de la base de datos.

Transact-SQL es el lenguaje que utiliza SQL Server para poder enviar peticiones tanto de consultas, inserciones, modificaciones, y de borrado a las tablas, así como otras



peticiones que el usuario necesite sobre los datos. En definitiva, es un lenguaje que utiliza SQL Server para poder gestionar los datos que contienen las tablas.

El lenguaje estándar SQL (Structured Query Language) se emplea para los sistemas de bases de datos relacionales RDBMS (Relational Database Management System), es el estándar ANSI (American National Standards Institute). (FormaSelect)

Características de Microsoft SQL Server:

- Soporte de transacciones.
- Escalabilidad, estabilidad y seguridad.
- Soporta procedimientos almacenados.
- Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- Además permite administrar información de otros servidores de datos.

Este sistema incluye una versión reducida, llamada MSDE con el mismo motor de base de datos pero orientado a proyectos más pequeños, que en sus versiones 2005 y 2008 pasa a ser el SQL Express Edition, que se distribuye en forma gratuita. (Islasoft.com, 2006)

1.4.6. AjaxCRUD

Es un conjunto de funciones API de PHP de código abierto que permiten conectarse a una base de datos de MySQL y de manera sencilla ejecutar las operaciones convencionales de consulta a la base de datos (crear, insertar, actualizar, borrar filas), además posee un conjunto de funciones de mucha utilidad a la hora de crear una aplicación Web. (Ajaxcrud.com, 2009)

1.5. Arquitectura

1.5.1. Patrones Arquitectónicos

Para el diseño de aplicaciones con sofisticadas interfaces se emplea el patrón de diseño MCV. La lógica de una interfaz de usuario cambia con más frecuencia que los almacenes de datos y la lógica de negocio. Si se realiza un diseño ofuscado, es decir, una forma de mezclar los componentes de interfaz y de negocio, entonces, la consecuencia será que, cuando se necesite cambiar la interfaz, tendrá que modificarse trabajosamente los componentes de negocio, por lo que propiciará mayor trabajo y más riesgo de error. Se trata de realizar un diseño que desacople la vista del modelo, con el fin de perfeccionar la reusabilidad. De este modo las modificaciones son las que impactan en menor medida en la lógica de negocio o de datos. (Jiménez, 2009)

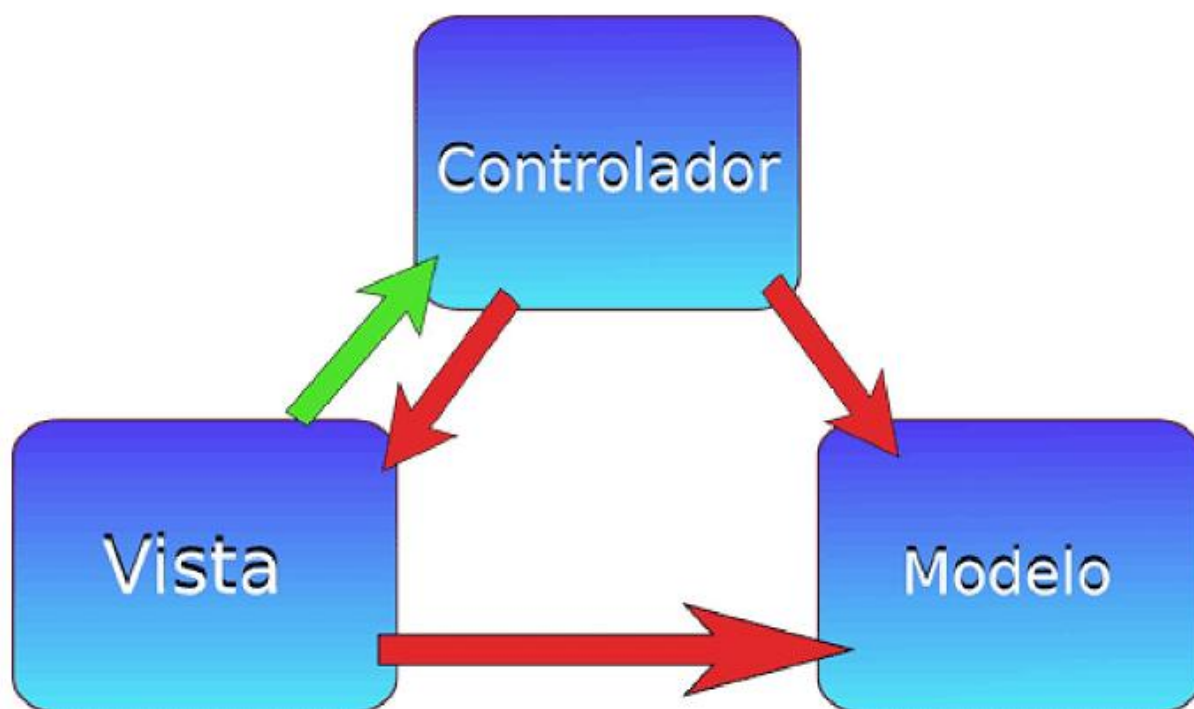


Figura 1.1. Modelo vista controlador

Elementos del patrón:

- Modelo: datos y reglas de negocio.
- Vista: muestra la información del modelo al usuario.
- Controlador: gestiona las entradas del usuario.

1.5.2. Arquitectura en capas



Los sistemas o arquitecturas en capas constituyen uno de los estilos que aparecen con mayor frecuencia mencionados como categorías mayores del catálogo o por el contrario, como una de las posibles imágenes de algún estilo envolvente. Definen el estilo en capas como una organización jerárquica tal, que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones de la inmediatamente inferior. (REYNOSO, y otros, 2007)

La arquitectura por capas es un estilo de arquitectura en la que el objetivo primordial es la separación de la lógica de negocio de la lógica de diseño, un ejemplo básico es separar la capa de datos, de la capa de presentación al usuario. La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de los niveles, simplemente es necesario conocer las API que existen entre niveles.

El diseño de sistemas informáticos suele usar las arquitecturas multinivel o programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables, (que pueden ampliarse con facilidad en caso de que las necesidades aumenten). El diseño más en boga actualmente es el diseño en tres capas.

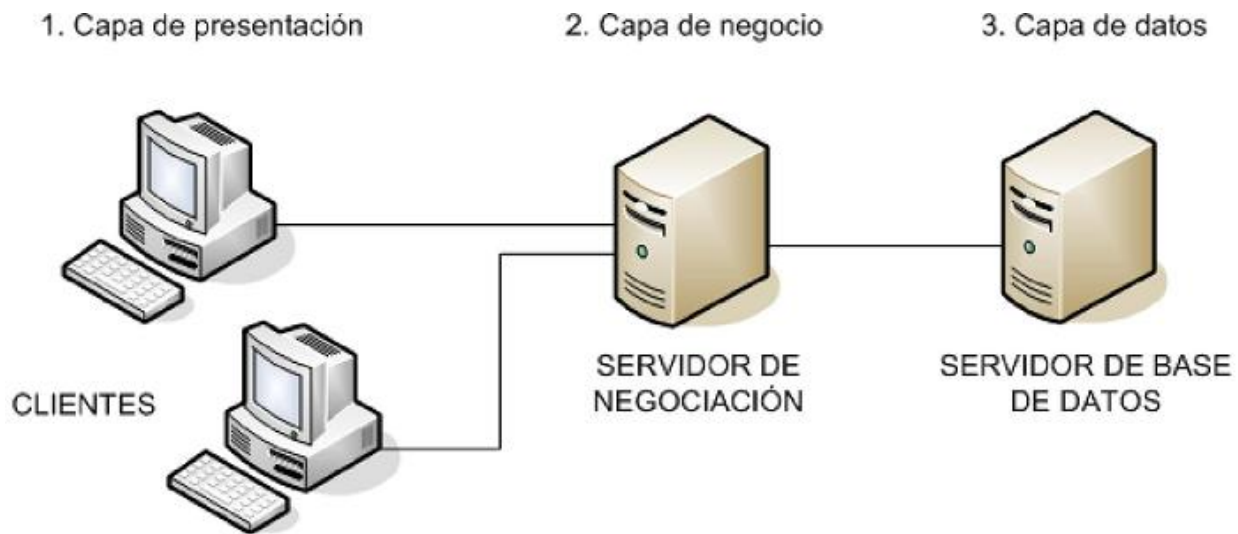


Figura 1.2. Arquitectura en tres capas.

Capas o niveles:

- Capa de presentación o interface: es la capa de que le permite al usuario interactuar con el sistema, captura y le comunica la información al mismo, dando un mínimo de proceso, (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la del negocio.
- Capa de lógica o de negocio: es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio e incluso lógica del negocio, pues es aquí donde se establecen las reglas que deben cumplirse. Esta capa se comunica con la de presentación para recibir las solicitudes y presentar los resultados, y con la capa de datos para solicitar al gestor de bases de datos para almacenar o recuperar datos de él.
- Capa de datos: es donde se ubican los datos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de los mismos, reciben solicitudes de almacenamiento o de recuperación de información desde la lógica del negocio.



Todas estas capas pueden residir en un único ordenador, esto no sería lo normal, lo más usual es que haya una multitud de ordenadores donde reside la capa de interface (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en un mismo ordenador, y si el crecimiento de las necesidades lo aconseja, pueden dividirse en dos o más ordenadores. Así, si el tamaño o complejidad de la base de datos aumenta, pueden separarse en varios ordenadores los cuáles recibirán las peticiones del ordenador en que resida la capa de negocio. Si por el contrario, la complejidad fuese en la capa de negocio lo que obligase a la separación, esta lógica del negocio podría residir en uno o más ordenadores que realizarían las solicitudes a una única base de datos.

1.5.3. Modelo Cliente-Servidor

El esquema cliente-servidor “es un modelo de computación en el que el procesamiento requerido para ejecutar una aplicación o conjunto de aplicaciones relacionadas se divide entre dos o más procesos que cooperan entre sí”.

Los principales componentes del esquema cliente-servidor son entonces los Clientes, los Servidores y la infraestructura de comunicaciones. (Monografias.com, 2007)

Modelo cliente-servidor de 2 capas

Las aplicaciones cliente-servidor clásicas o de 2 capas, como su nombre lo indica, agrupan la lógica de presentación (interfaz) y la lógica de aplicación en la máquina cliente y acceden a fuentes de datos compartidos a través de una conexión de red que se encuentran en el servidor de datos.

La ventaja que presenta este tipo de aplicaciones es que los datos están centralizados. Esta centralización beneficia a la empresa pues es más fácil compartir los datos, se simplifica la generación de reportes y se proporciona consistencia en el acceso a los datos.

A continuación se enumeran las principales desventajas que presentan este tipo de aplicaciones:



- **Difíciles de mantener:** Esto viene dado por el hecho de que son difíciles de mantener las reglas de negocio de la lógica de aplicación ya que estas están programadas en cada cliente y esto implica que cualquier cambio tiene que ser redistribuido en todos los clientes.
- **Se compromete la confidencialidad:** Al tener programada la lógica de aplicación en el cliente este tiene a su disposición todas las reglas de negocio de la empresa.
- **Están estrechamente limitadas a una fuente de datos:** Los clientes casi siempre están configurados para acceder a una base de datos en particular por lo que mover los datos a una base de datos diferente se hace realmente complicado. (Janium.com, 2009)

Modelo cliente-servidor de 3 capas

En la arquitectura de 3 capas la presentación, la lógica de aplicación y los elementos de datos están conceptualmente separados. Los componentes de la capa de presentación manejan la interacción con el usuario y realizan las peticiones del cliente a los componentes de la capa intermedia. Los componentes de la capa intermedia, manipulan la lógica de negocio y hacen las peticiones a la base de datos.

A continuación se enumeran algunas ventajas de las aplicaciones de 3 capas:

- Los componentes de la aplicación pueden ser desarrollados en cualquier lenguaje general lo que posibilita que el grupo de desarrolladores no se centre en el uso de un solo lenguaje.
- Los componentes están centralizados lo que posibilita su fácil desarrollo, mantenimiento y uso.
- Los componentes de la aplicación pueden estar esparcidos en múltiples servidores permitiendo una mayor escalabilidad.
- Los problemas de limitación para las conexiones a las bases de datos se minimizan ya que la base de datos solo es vista desde la capa intermedia y no desde todos los

clientes. Además que las conexiones y los drivers de las bases de datos no tienen que estar en los clientes.

- Los componentes de aplicación de la capa intermedia pueden ser asegurados centralmente usando una infraestructura común. Se pueden conceder o denegar los permisos componente a componente simplificando la administración. (Ronda Amador, y otros, 2002)

1.5.4. Dispositivos de Hardware de Control Automático

Nport Server Lite

El Nport Server Lite es una fortaleza industrial debido a que es un dispositivo de comunicación serie basado en red, con capacidad para unir hasta cuatro dispositivos RS-232 ó RS-422/485, tales como CNCs, PLCs, Scanners, directamente a una red Ethernet a una velocidad de transferencia de 10/100 Mbps corriendo sobre el protocolo TCP/IP.

1.5.5. Arquitectura de los Sistemas Informáticos de Control utilizando NPort Server Lite.

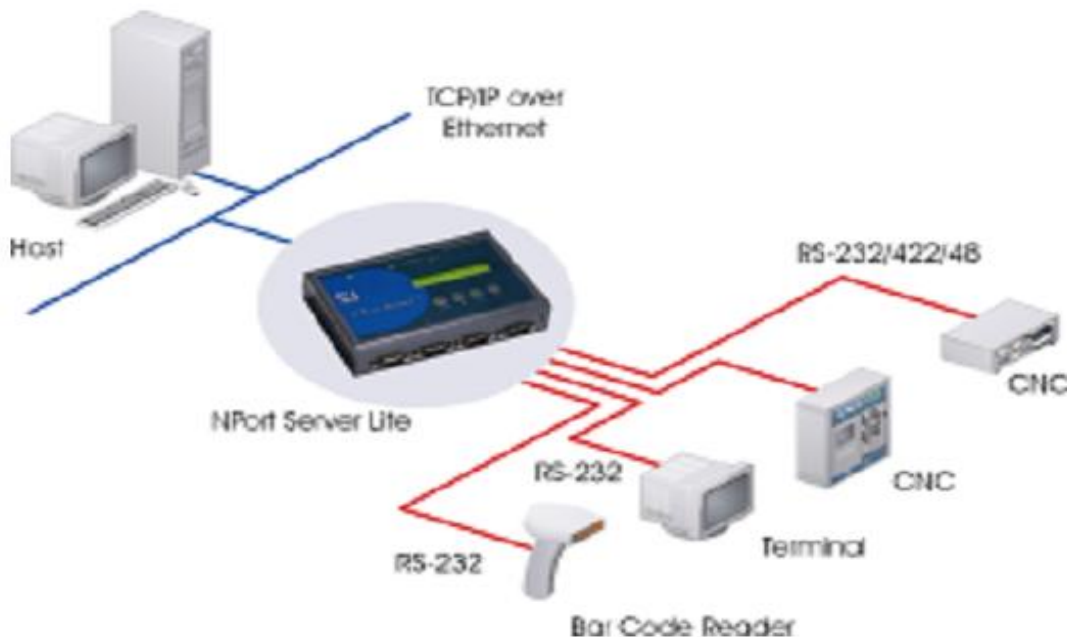


Figura 1.3 Arquitectura de los Sistemas Informáticos de Control utilizando NPort Server Lite



El modo de conexión directa o cruda permite que el Nport actúe como un servidor pasivo, que espera por las peticiones de servicio de un anfitrión cliente mediante un puerto de enchufe (el puerto predeterminado de escucha del Nport es el 4001). Primeramente el anfitrión se conecta al servidor Nport mediante el protocolo TCP y el puerto asociado a dicho servidor, una vez establecida la conexión, el programa cliente puede leer o escribir los datos seriales.

1.6. Metodología de Desarrollo de Software

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Pueden ser comparadas con un plan de contingencias en el que se va indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además quienes deben participar en el desarrollo de las actividades y qué papel deben de tener. Detallan además la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

Metodologías Ágiles

En una reunión celebrada en febrero de 2001 en Utah-EEUU, nace el término "ágil" aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas. Varias de las denominadas metodologías ágiles ya estaban siendo utilizadas con éxito en proyectos reales, pero les faltaba una mayor difusión y reconocimiento. Tras esta reunión se creó The Agile Alliance, una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida fue el Manifiesto Ágil, un documento que resume la filosofía "ágil". (Crystal Methodologies, 2008)



Actualmente, existe una amplia variedad de metodologías ágiles. Las más difundidas se enumeran a continuación.

- Extreme Programming
- Crystal Clear
- SCRUM
- DSDM (Dynamic System Development Method)
- FDD

1.7. Metodología XP.

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. (BECK, 1999)

1.7.1. Valores de la XP.

Cuatro valores que promueven la metodología XP:

Simplicidad: XP propone el principio de hacer las cosas más simple que pueda funcionar, en relación al proceso y la codificación. Es mejor hacer hoy algo simple, que hacerlo complicado y probablemente nunca usarlo mañana.

Comunicación: Algunos problemas en los proyectos tienen su origen en que alguien no dijo algo importante en algún momento. XP hace imposible la falta de comunicación.

Retroalimentación: Retroalimentación concreta y frecuente del cliente, del equipo y de los usuarios finales da una mayor oportunidad de dirigir el esfuerzo eficientemente.

Coraje: El coraje (valor) existe en el contexto de los otros 3 valores

1.7.2. Proceso XP.



El ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos:

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

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración.

El ciclo de vida ideal de XP consiste de seis fases: Exploración, Planificación de la Entrega, Iteraciones, Producción, Mantenimiento y Muerte del proyecto. (BECK, 1999)

1.7.3. Prácticas XP

La mayoría de estas prácticas no son nuevas, han sido reconocidas por la industria como mejores prácticas durante años. En la XP, dichas prácticas son llevadas al extremo para obtener más que la suma de las partes.

La principal suposición que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione. Esto se consigue gracias a las tecnologías disponibles para ayudar en el desarrollo de software y a la aplicación disciplinada de las siguientes prácticas. (Beck, 1999)

El juego de la planificación. Hay una comunicación frecuente el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración.



Entregas pequeñas. Producir rápidamente versiones del sistema que sean operativas, aunque no cuenten con toda la funcionalidad del sistema. Esta versión ya constituye un resultado de valor para el negocio. Una entrega no debería tardar más 3 meses.

Metáfora. El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema (conjunto de nombres que actúen como vocabulario para hablar sobre el dominio del problema, ayudando a la nomenclatura de clases y métodos del sistema).

Diseño simple. Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.

Pruebas. La producción de código está dirigida por las pruebas unitarias. Éstas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema

Refactorización (*Refactoring*). Es el proceso de modificar el código de un sistema de software de modo que no se altere su comportamiento externo pero se mejore su estructura interna. Es una técnica disciplinada de reestructuración de código. Parte del proceso puede automatizarse y de hecho existen herramientas que facilitan la tarea.

Programación en parejas. Toda la producción de código debe realizarse con trabajo en parejas de programadores. Esto conlleva ventajas implícitas (menor tasa de errores, mejor diseño, mayor satisfacción de los programadores).

Propiedad colectiva del código. Cualquier programador puede cambiar cualquier parte del código en cualquier momento.

Integración continua. Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.



Cuarenta horas por semana. Se debe trabajar un máximo de cuarenta horas por semana. No se trabajan horas extras en dos semanas seguidas. Si esto ocurre, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo.

Cliente in-situ. El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Éste es uno de los principales factores de éxito del proyecto XP. El cliente conduce constantemente el trabajo hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita.

Estándares de programación. XP enfatiza que la comunicación de los programadores es a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación para mantener el código legible. (BECK, 1999). Las seis prácticas esenciales por donde uno de empezar:

- Pruebas.
- Refactorización.
- Programación en pares.
- Juego de planificación.
- Liberaciones pequeñas.
- Integración continua

1.7.4. ¿Por qué elegir XP?

Actualmente, XP es el método ágil más documentado (hay una colección de libros “XP Series” de Addison Wesley) y extendido. Existe una gran comunidad de desarrolladores XP.

Otra de las ventajas de XP es que no es necesario adoptarlo en forma completa, sino que pueden utilizarse varias de sus prácticas en forma independiente. Esto hace que el costo de su implementación sea mucho más accesible que el de otras metodologías. Un estudio



a la bibliografía, muestra las ventajas y desventajas que tiene XP y que exponemos en los dos puntos siguientes.

Ventajas de XP

- Puede ser implementado en forma parcial (elegir sólo algunas de las prácticas)
- Puede ser implementado en forma gradual
- Puede adaptarse a las necesidades de cualquier equipo de desarrollo. De hecho, Kent Beck recomienda a los equipos que lo adapten a sus necesidades (BECK, 1999).
- Exige que se establezca una comunicación más fluida con el cliente y que este tenga mayor participación en el proceso de desarrollo. La consecuencia de esto es que el cliente se involucre más en el desarrollo del producto.
- Actualmente es la metodología ágil más extendida y documentada
- Se realizan pruebas constantemente del sistema.

Desventajas de XP

- XP no es escalable a equipos de muchos desarrolladores (a lo sumo 15)
- Es una metodología nueva y no está ampliamente probada
- Requiere un equipo de programadores altamente especializados y/o con experiencia considerable
- Requiera alto compromiso del equipo de desarrollo, lo cual a veces es difícil debido a la precaria situación contractual de la gente que trabaja en consultoras.

1.7.5. Fases de la metodología XP

Fase I: Planificación

- Se escriben historias de usuario, cuya idea principal es describir un caso de uso en dos o tres líneas con terminología del cliente (de hecho, se supone que deben ser escritos por el mismo), de tal manera que se creen *test* de aceptación para historias



de usuarios (*user storie*) y permita hacer una estimación de tiempo de desarrollo del mismo.

- Se crea un plan de lanzamiento (*release planning*), que debe servir para crear un calendario que todos puedan cumplir y en cuyo desarrollo hayan participado todas las personas involucradas en el proyecto. Se usa como base las historias de usuario, participando el cliente en la elección de las que se desarrollarán, y según las estimaciones de tiempo de los mismos se crearán las iteraciones del proyecto.
- El desarrollo se divide en iteraciones, cada una de las cuales comienzan con un plan de iteración, para el que se eligen las historias de usuario a desarrollar y las tareas de desarrollo.
- Se cambia el proceso cuanto sea necesario, para adaptarlo al proyecto.

Fase II: Diseño

- Se eligen los diseños funcionales más simples.
- Se elige una metáfora del sistema para que el nombrado de clases, siga una misma línea, facilitando la reutilización y la comprensión del código.
- Se escriben tarjetas de clase-responsabilidades-colaboración (CRC) para cada objeto, que permitan abstraerse al pensamiento estructurado y que el equipo de desarrollo completo participe en el diseño.

Fase III: Codificación

- El cliente está siempre disponible, de ser posible, cara a cara. La idea es que forme parte del equipo de desarrollo, y esté presente en todas las fases de XP. La idea es usar el tiempo del cliente para estas tareas en lugar de crear una detallada especificación de requisitos, y evitar la entrega de un producto insuficiente, que le hará perder tiempo.
- El código se ajustará a unos estándares de codificación, asegurando la consistencia y facilitando la comprensión y refactorización del código.



- Las pruebas unitarias se codifican antes que el código en sí, haciendo que la codificación de este último sea más rápida, y que cuando se afronte la misma se tenga más claro, qué objetivos tiene que cumplir lo que se va a codificar.
- La programación del código se realiza en parejas, para aumentar la calidad del mismo. En cada momento, sólo habrá una pareja de programadores que integre código.
- Se integra código y se lanza dicha integración de manera frecuente, evitando divergencias en el desarrollo y permitiendo que todo el mundo trabaje con la última versión del desarrollo. De esta manera, se evitará pasar grandes períodos de tiempo integrando el código al final del desarrollo, ya que las incompatibilidades serán detectadas enseguida.
- Se usa la propiedad colectiva del código, lo que se traduce en que cualquier programador puede cambiar cualquier parte del código. El objetivo es fomentar la contribución de ideas por parte de todo el equipo de desarrollo.
- Se deja la optimización para el final.
- No se hacen horas extra de trabajo.

Fase IV: Pruebas

- Todo el código debe tener pruebas unitarias, y debe pasarlas antes de ser lanzado.
- Cuando se encuentra un error de codificación o bug, se desarrollan pruebas para evitar volver a caer en el mismo.
- Se realizan pruebas de aceptación frecuentemente, publicando los resultados de las mismas. Estas pruebas son generadas a partir de las historias de usuarios elegidas para la iteración, y son "pruebas de caja negra", en las que el cliente verifica el correcto funcionamiento de lo que se está probando. Cuando se pasa la prueba de aceptación, se considera que la correspondiente historia de usuario se ha completado.

1.8. Herramientas CASE (Computer – Aided Software Engineering)



Las herramientas CASE (Ingeniería de Software Asistida por Computadoras), son aplicaciones informáticas que tienen como objetivo fundamental solucionar y afrontar los problemas de mala calidad de software y documentación inadecuada. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo de un determinado proyecto o software, ya que brindan la posibilidad de realizar cálculos de costos, generan código fuente automáticamente de un diseño previamente dado, poseen compilación automática, ayudan con la documentación y juegan un papel importante en la detección de errores.

Por tales ventajas algunas personas han dado su criterio personal sobre el concepto CASE, que se define como:

“Herramientas individuales para ayudar al desarrollador de software o administrador de proyecto durante una o más fases del desarrollo de software o mantenimiento del mismo”.

Embarcadero ER/Studio: Es una herramienta de modelado de datos, se usa para el diseño y la construcción lógica y física de bases de datos. Su ambiente es de gran alcance y multinivel. Simple y fácil al usuario, ayuda a las organizaciones para tomar decisiones en cómo resolver embotellamientos de los datos, elimina redundancia y alcanza en última instancia usos de más alta calidad que entreguen datos más eficientes y exactos a la empresa. (Bureaudeprensa.com)

1.9. Lenguaje Unificado de Modelado (UML)

Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Se usa para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir. Capta la información sobre la estructura estática y el comportamiento dinámico de un sistema, no es un lenguaje de programación.

Las herramientas CASE pueden ofrecer generadores de código de UML para una gran variedad de lenguaje de programación, así como construir modelos por ingeniería inversa a partir de programas existentes. UML es un lenguaje de propósito general para el



modelado orientado a objetos, es también un lenguaje de modelamiento visual que permite una abstracción del sistema y sus componentes. (Corp, 2006)

Objetivos del UML: (Vico.org, 2002)

- Es un lenguaje de modelado de propósito general que pueden usar todos los modeladores. No tiene propietario y está basado en el común acuerdo de gran parte de la comunidad informática.
- No pretende ser un método de desarrollo completo. No incluye un proceso de desarrollo paso a paso, incluye todos los conceptos que se consideran necesarios para utilizar un proceso moderno iterativo, basado en construir una sólida arquitectura para resolver requisitos dirigidos por casos de uso.
- Ser tan simple como sea posible pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir. Necesita ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de software, como son la encapsulación y componentes.
- Ser un lenguaje universal, como cualquier lenguaje de propósito general.
- Imponer un estándar mundial.

1.10. Arquitectura a utilizar

Una arquitectura es el conjunto de decisiones significativas sobre la organización del sistema software, la selección de los elementos estructurales y sus interfaces, con los que se compone el sistema, junto con su comportamiento tal como se especifica en las colaboraciones entre esos elementos, la composición de esos elementos estructurales y de comportamiento en subsistemas progresivamente más amplios, y el estilo de arquitectura que guía esta organización, estos elementos y sus interfaces, sus colaboraciones, y su composición. Ejemplo: los patrones de diseño relacionados con el diseño de los objetos y frameworks de pequeña y mediana escala, que son aplicables al diseño de una solución para conectar los elementos de gran escala que se definen mediante los patrones de arquitectura, y durante el trabajo de diseño detallado para



cualquier aspecto del diseño local. También se conocen como patrones de micro-arquitectura. El patrón fachada, que se puede utilizar para proporcionar la interfaz de una capa a la siguiente. La arquitectura en capas o arquitectura n capas es la que se ha seleccionado para darle solución a la problemática planteada anteriormente, debido a las facilidades presentadas anteriormente.

1.11. Tecnologías y Herramientas a utilizar

Luego del estudio realizado se arriba a la conclusión de que el Sistema estará guiado por la metodología de desarrollo XP por sus facilidades, documentación y flexibilidad, además es la que mejor se adapta a las condiciones de desarrollo del sistema propuesto.

En la implementación del sistema, se utilizaron los dos entornos de programación antes descritos, Dreamweaver por las facilidades que brinda para el trabajo de diseño Web y PhpDesigner 2008 v6.0.0 por las ventajas que proporciona a la hora de escribir algoritmos complejos en los cuales es necesario correr paso a paso el programa, además ambos entornos de desarrollo soportan el lenguaje de programación PHP , el cual fue seleccionado para el desarrollo de la aplicación por la características ventajosas que brinda en comparación con otros lenguajes de programación. Como servidor Web: Apache 2.0, por ser multiplataforma y garantizar que la aplicación a su vez lo sea.

Como Sistema Gestor de Base de Datos se determina el uso de MySQL 5.0, por su rapidez cuando se trabaja con una base de datos pequeña o mediana, además de estar totalmente integrado con los entornos de desarrollo seleccionados y el lenguaje de programación escogido. Se utilizó algunas de las funcionalidades que proporciona AjaxCRUD para disminuir el tiempo de desarrollo. La aplicación se sustentará en una Arquitectura Cliente – Servidor de tres capas y se utilizaron algunos principios del Modelo-Vista-Controlador.



Como resultado del análisis realizado durante el presente capítulo se concluye que: las herramientas estudiadas relacionadas con el proceso de control del consumo energético no responden a las necesidades del centro. Debido a esto se necesita implementar un módulo para el control del consumo energético. Para ello se realizó un estudio de las tecnologías actuales y se seleccionaron las más adecuadas para la implementación del sistema.



CAPÍTULO 2: PLANEACIÓN Y DISEÑO

En este capítulo, se introduce la fase de planeación y diseño, donde se detallan las necesidades del cliente, se describen las funcionalidades que serán objeto de automatización mediante el empleo de las historias de usuarios (HU), se realiza una estimación del esfuerzo necesario para las mismas y se establece un plan de iteraciones necesarias sobre el sistema, para su terminación.

2.1. Personal relacionado con el Sistema

<i>Personas relacionadas con el sistema</i>	Justificación
<i>Especialista</i>	Esta es la persona que tiene conocimiento en la materia de control de consumo energético, y está encargada de la gestión de la información del mismo.
<i>Administrador</i>	Es la persona encargada de asesorar y dar seguimiento del estado del proceso de desarrollo.
<i>Desarrollador</i>	Es la persona responsable de llevar a cabo la implementación del sistema.

Tabla 1 Personal relacionado con el Sistema

2.2. Lista de Reserva

Después de conocer el personal relacionado, se procede a realizar el análisis de las funcionalidades que debe cumplir la aplicación para dar respuesta a los mismos. Para ello se enumerarán mediante una lista de reserva, las funcionalidades que el sistema debe ser capaz de cumplir.

De acuerdo a lo antes expuesto, el sistema debe ser capaz de:

- Insertar registro de control del local



- Calcular automáticamente el consumo aproximado del local
- Mostrar registro de control del local
- Modificar registro de control del local
- Eliminar registro de control del local
- Insertar contadores
- Mostrar contadores
- Modificar contadores
- Eliminar contadores
- Insertar mediciones de contadores
- Mostrar mediciones de contadores
- Modificar mediciones de contadores
- Eliminar mediciones de contadores
- Actualización del consumo mensual de electricidad
- Insertar control de actividades de transportación
- Mostrar control de actividades de transportación
- Modificar control de actividades de transportación
- Eliminar control de actividades de transportación
- Actualización del consumo mensual de combustible
- Generar reporte de consumo mensual por locales
- Generar reporte de consumo mensual de transportación
- Generar reporte de consumo mensual por áreas administrativas
- Conexión del sistema con MULTILIN
- Saber estado de conexión con MULTILIN
- Lista de locales por áreas administrativas
- Lista de locales por subáreas administrativas
- Lista de locales por tipos de locales
- Lista de locales por áreas de consumo final

2.3. Historias de Usuario



Las HU, son la técnica utilizada en XP para detallar los requisitos del software. Son el resultado directo del intercambio entre los usuarios y desarrolladores a través de reuniones donde las conocidas tormenta de ideas (*brain storm*) arrojan no solo los requerimientos, sino también las posibles soluciones; representan una forma rápida de administrar las necesidades de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para gestionarlos, debido a que un requerimiento de software es descrito de forma concreta y sencilla utilizando el lenguaje común del usuario.

Las HU permiten responder ágilmente a los requerimientos cambiantes y aunque se redactan desde las perspectivas de los clientes, también los desarrolladores pueden brindar ayuda en la identificación de las mismas. Para definir las se emplea la siguiente plantilla.

Modelo de planilla de historia de usuario

Historia de usuario	
Número: No. Historia de usuario	Usuario: Usuario entrevistado para obtener la función requerida a automatizar.
Nombre: nombre de la historia de usuario que sirve para identificarla mejor entre los desarrolladores y el cliente.	
Prioridad en el negocio: Importancia: Alta / Media / Baja	Riesgo en desarrollo: Dificultad: Alta / Media / Baja
Puntos estimados: Estimación: de 1 a 3 puntos	Iteración asignada: Iteración a la que corresponde
Programador responsable: Nombre de encargado de programación.	



Descripción: Se especifican las operaciones por parte del usuario y las respuestas del sistema.
Observaciones: Algunas observaciones de interés, como glosario, información sobre usuario etc.

Tabla 2 Planilla de historia de usuario.

HU No.1: Gestionar el registro de control por Locales.

Historia de usuario	
Número: 1	Usuario: Controlador del Local
Nombre: Gestión de la información del control de consumo energético.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Alexei Cala Hinojosa.	
Descripción: El usuario debe insertar los datos de control correspondientes al intervalo de tiempo controlado. En el caso de los locales donde no existe control, el consumo se calcula aplicando proporciones. Una vez insertados estos datos la información se registrará en el historial de control del local para poder ser mostrada y actualizada.	
Observaciones: Confirmado con el cliente.	

Tabla. 3 HU No.1: Gestionar el registro de control por Locales.

Para ver las historias de usuario ver [ANEXO 1 HISTORIAS DE USUARIO](#).

2.4. Planificación de Entregas



En esta fase se establece la prioridad de cada HU, y a continuación, se realiza una estimación del esfuerzo necesario de cada una de ellas por parte de los programadores. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debe obtenerse en no más de dos a tres meses.

Las estimaciones asociadas a la implementación de las historias se establecen empleando como medida el punto de estimación. Un punto de estimación equivale a una semana ideal de programación, donde los miembros de los equipos de desarrollo, trabajan el tiempo planeado sin ningún tipo de interrupción, este punto de estimación que se utiliza para representar la semana ideal, es de 5 días. Las historias generalmente tienen un valor de 1 a 3 puntos. Además, se mantiene un registro de la velocidad de desarrollo, establecida por puntos de iteración, basado fundamentalmente en la suma de los puntos de estimación correspondientes a las HU, que fueron terminadas en la última iteración.

2.4.1. Estimación de Esfuerzo por Historias de Usuario

Para el buen desarrollo del sistema propuesto, se realizó una estimación para cada una de las HU identificadas, y se obtienen los resultados que se muestran a continuación:

Estimación de Esfuerzo por Historias de Usuario

Historias de usuario	Puntos de estimación
Gestionar el registro de control por Locales	3 semana
Gestión de control de actividades de transportación	2 semana
Gestión de mediciones	2 semanas
Gestión de contadores	1 semana
Conexión con MULTILIN	2 semanas
Reporte de consumo mensual	2 semana
Filtro de locales	1 semana

Tabla 4 Estimación de esfuerzo por historias de usuario



2.4.2. Planificación de Iteraciones

A partir de las HU antes expuestas y la estimación del esfuerzo propuesto para la realización de las mismas, se procede a realizar la planificación de la etapa de implementación del sistema, apoyándose en el tiempo e intentando concentrar las funcionalidades relacionadas en una misma iteración. En este plan se establece cuántas iteraciones serán necesarias realizar sobre el sistema para su terminación. El plan de iteraciones puede contener indicaciones sobre cuáles HU se incluirán en un release, lo cual debe ser consistente con el contenido de una o dos iteraciones.

En relación con lo antes tratado se decide realizar el sistema en 3 iteraciones, las cuales se explican de forma detalla a continuación:

Primera iteración:

Esta iteración tiene como objetivo darle cumplimiento a las HU que se consideraron de mayor importancia para el desarrollo de la aplicación. Al concluir dicha iteración se contará con todas las funcionalidades descritas en las HU 1, 3, 7 las cuales hacen alusión a la inserción, visualización, etc. de la información del control del consumo energético por locales así como la realización de las lecturas de los contadores, también en esta iteración se crea un filtro el cual facilita el trabajo de control de los locales. Se tendrá la primera versión de prueba, que contará con un modelo de desarrollo que incorporan todas las funcionalidades antes vistas, este modelo se presentarán al cliente con el objetivo de obtener una retroalimentación del mismo para posteriores iteraciones del producto.

Segunda iteración:

Esta iteración tiene como finalidad desarrollar las HU 2, 4. Las mismas son las que brindan las funcionalidades de obtener y visualizar la información relacionada con el control de las actividades de transportación, y la gestión de los contadores energéticos. La versión que se obtenga de esta iteración en unión con la entregada en la iteración anterior se le facilitará al cliente para comprobar si cumple con las necesidades antes acordadas con él.

Tercera iteración:



Esta última iteración del módulo tiene como propósito llevar a cabo el desarrollo de la HU 5 y 6. Las cuales brindan las funcionalidades de emitir reportes de consumo energético por locales, por actividades de transportación, por áreas administrativas y conectar el sistema con MULTILIN. Estas HU serán integradas con el resultado de las iteraciones anteriores, y como fruto de esta integración se obtendrá la versión 1.0 del producto final. A partir de este momento el software será puesto a un proceso de prueba para evaluar el desempeño del mismo.

2.4.3. Plan de duración de las iteraciones

Como parte del ciclo de vida de un proyecto guiado por la metodología de desarrollo de software XP, se crea el plan de duración de cada una de las iteraciones que se llevarán a cabo durante el desarrollo del mismo. Este plan tiene como finalidad mostrar la duración de cada iteración, así como el orden en que serán implementadas las HU en cada una de las mismas.

Plan de duración de las iteraciones

Iteración	Historias de usuario	Duración total
Iteración 1	Gestionar el registro de control por Locales Gestión de mediciones Filtro de locales	6 semanas
Iteración 2	Gestión de control de actividades de transportación Gestión de contadores	3 semanas
Iteración 3	Conexión con MULTILIN Reporte de consumo mensual	4 semanas

Tabla 5 Plan de duración de las iteraciones



En este capítulo se abordó la fase de planeación y diseño donde se presentaron las distintas HU con la participación del cliente, se llevó a efecto la planificación de iteraciones de cada HU a partir de la estimación del esfuerzo necesario de las mismas, culminando así esta fase y se determina que el equipo de trabajo está listo para pasar a la siguiente etapa de desarrollo.

CAPÍTULO 3: DESARROLLO Y PRUEBAS

En este capítulo se inicia la fase de desarrollo y pruebas conforme a la metodología XP. Se presenta el modelo de datos empleado para la aplicación concluyente, se presentan las tarjetas, clases, responsabilidades y colaboradores, que permitirán trabajar con una metodología basada en objetos, se realiza el desarrollo de las iteraciones a partir del desglose de las historias de usuario en tareas. Se muestran las interfaces gráficas de usuario diseñadas para la aplicación final. Se describen además las pruebas realizadas y se indican las respuestas de la aplicación en el empleo de las diferentes funcionalidades, así como los posibles mensajes de error, información o aceptación que emite la misma cuando se utiliza una de estas funcionalidades.

3.1 Modelo de Datos

En esta parte se muestra el modelo de datos empleado para la aplicación. Para la gestión del control de los locales, es empleada la tabla `Control_Local`, donde se almacenan los datos básicos del control de los locales. Cuando se realiza una actividad de transportación, los datos a controlar en la misma son guardados en la tabla `Control_Activ_Transp`. Una vez insertada una nueva actividad de transportación se debe modificar el registro de consumo mensual de transportación del mes, el cual es almacenado en la tabla `Consumo_Control_Transp_Mes`. Las mediciones de los contadores energéticos son guardadas en la tabla `mediciones` y una vez insertada una nueva medición se modifica el registro de control del consumo de energía eléctrica el cual es almacenado en la tabla `consumo_control_energia`. En la tabla `contadores` se guardan los datos de los mismos.

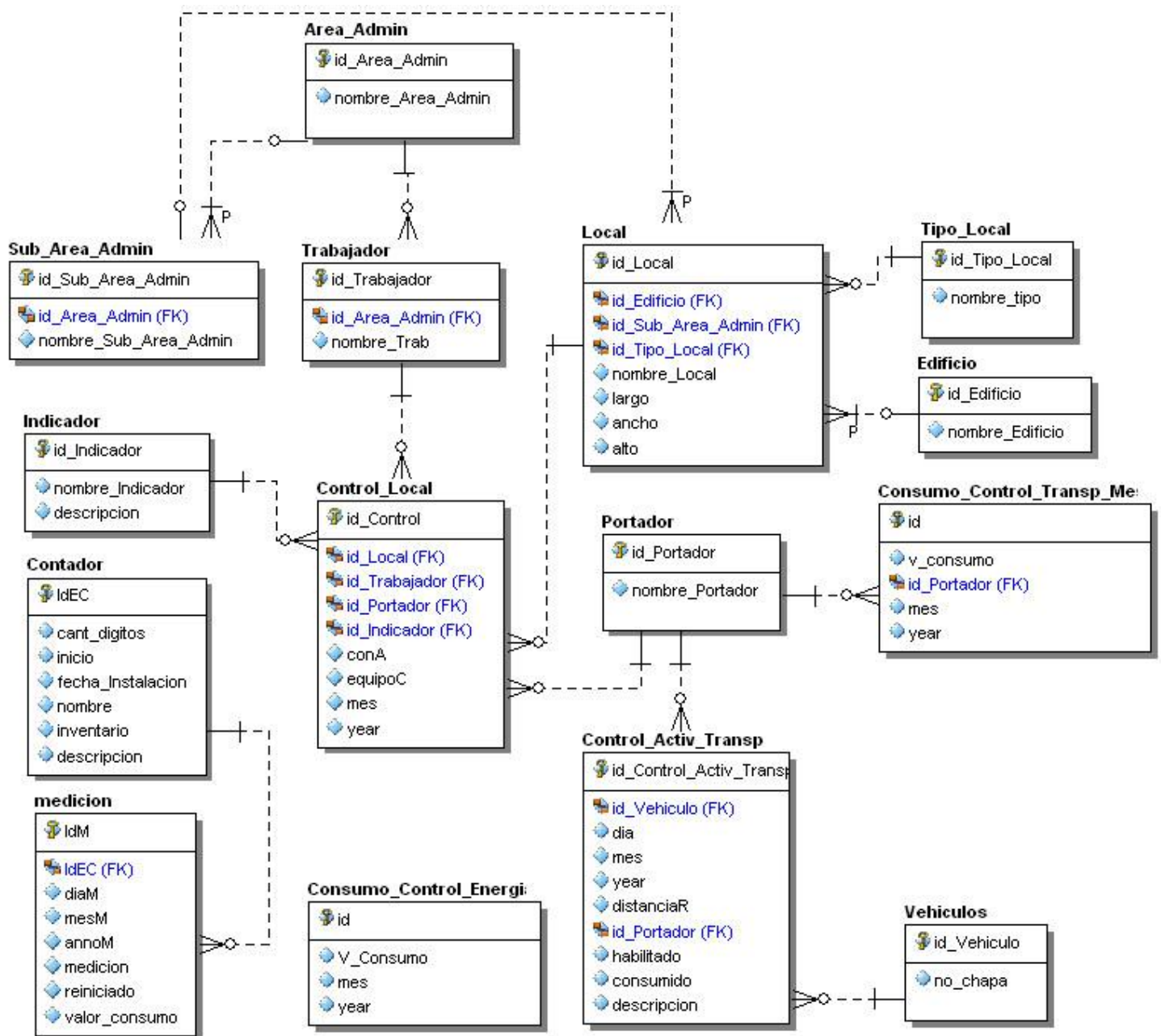


Figura. 3.1. Modelo de datos del módulo de control del consumo energético

3.2 Clases, Responsabilidades y Colaboradores

En este epígrafe tiene lugar la realización de las tarjetas de clases, responsabilidades y colaboración, conocidas tradicionalmente como *tarjetas CRC*, las cuáles se realizan con el



objetivo de facilitar la comunicación y documentar los resultados. Además, las mismas permiten la total participación y contribución del equipo de desarrollo en la tarea de diseño. Una tarjeta CRC representa un objeto, por tanto es una clase, cuyo nombre se ubica en forma de título en la parte superior de la tarjeta, los atributos y las responsabilidades más significativas se colocan a la izquierda y las clases implicadas con cada responsabilidad a la derecha, en la misma línea de su requerimiento correspondiente. Para mejor comprensión de las mismas, se determina agruparlas por HU.

Tarjeta CRC para la clase Mediciones

Nombre de la clase: Medicion	
Tipo de la clase: Lógica de negocio	
Responsabilidades:	Colaboradores:
Equipo de control	
día	
mes	
año	
medición	
reiniciado	
Valor de consumo	
Set	
•	
•	
•	
Get	
•	
•	
•	

Tabla 6 Tarjeta CRC para la clase Mediciones

**Tarjeta CRC para la clase MedicionDAO**

Nombre de la clase: MedicionDAO	
Tipo de la clase: Acceso a datos	
Responsabilidades:	Colaboradores:
Salvar Medicion	Medicion
Modificar	Medicion
Eliminar	Medicion
Obtener todas las mediciones	Medicion
Obtener una medicion	Medicion
Existe	Medicion

Tabla 7 Tarjeta CRC para la clase MedicionDAO

Tarjeta CRC para la clase MedicionServicio

Nombre de la clase: MedicionServicio	
Tipo de la clase: Servicio	
Responsabilidades:	Colaboradores:
Salvar medición	MedicionDAO, CalcularC
Modificar	MedicionDAO
Eliminar	MedicionDAO
Obtener todas las mediciones	MedicionDAO
Obtener una medición	MedicionDAO
Existe	MedicionDAO

Tabla 8 Tarjeta CRC para la clase MedicionServicio

Para ver las tarjetas CRC remitirse a [ANEXO 3: TARJETAS CRC](#)



3.3 Desarrollo de las Iteraciones

Durante la fase planificación y diseño fueron detalladas las historias de usuario correspondientes a cada una de las iteraciones a desarrollar, teniendo en cuenta las prioridades y restricciones de tiempo, previstas por el cliente.

3.3.1 Tareas por Historias de Usuario

Dentro del contenido de este plan, las HU se descomponen en tareas de programación o ingeniería, y a su vez, estas son asignadas al equipo de desarrollo para su implementación. Las tareas no tienen que ser entendidas necesariamente por el cliente, pues las mismas, sólo son utilizadas por los miembros del equipo de desarrollo, por lo que pueden ser escritas en lenguaje técnico. Las mismas se representan mediante las tarjetas de tareas.

Tareas por Historias de Usuario

Historias de usuario	Tareas
Gestionar el registro de control por locales	● Insertar Registro de control del local ● Mostrar registro de control del local ● Modificar registro de control del local ● Eliminar registro de control del local
Gestión de control de actividades de transportación	● Insertar control de actividades de transportación ● Mostrar control de actividades de transportación ● Modificar control de actividades de transportación ● Eliminar control de actividades de transportación ● Actualización del consumo mensual de combustible
Gestión de mediciones	● Insertar mediciones de contadores ● Mostrar mediciones de contadores ● Modificar mediciones de contadores ● Eliminar mediciones de contadores ● Calcular el consumo del día
Gestión de contadores	● Insertar contadores ● Mostrar contadores



	• Modificar contadores • Eliminar contadores
Conexión del sistema con MULTILIN	• Conectar el sistema con MULTILIN • Saber estado de la conexión con MULTILIN
Reporte de consumo mensual	• Generar reporte de consumo mensual por locales • Generar reporte de consumo mensual de combustible en transportación • Generar reporte de consumo por áreas
Filtro de locales	• Lista de locales por Áreas Administrativas • Lista de locales por Subáreas Administrativas • Lista de locales por edificios • Lista de locales por tipos de locales • Lista de locales por Áreas de Consumo Final

Tabla 9 Tareas por Historias de Usuario

Historias de usuario abordadas en la primera iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Gestionar el registro de control por Locales	3	2.5
Gestión de mediciones	2	1.5
Filtro	1	0.8

Tabla 10 Historias abordadas en la primera iteración

Historias de usuario abordadas en la segunda iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación inicial	Real
Gestión de control de actividades de transportación	2	1.4
Gestión de contadores	1	0.5

Tabla 11 Historias abordadas en la segunda iteración

Historias de usuario abordadas en la tercera iteración

Historias de usuario	Tiempo de estimación (semanas)	
	Estimación	Real



	inicial	
Conexión con MULTILIN	2	1.5
Reportes de consumo mensual	2	1.8

Tabla 12 Historias abordadas en la tercera iteración

Para ver las tarjetas de ingeniería remitirse al [ANEXO 2: TAREAS DE INGENIERÍA](#).

3.4 Pruebas

En la Programación Extrema es esencial el desarrollo de las pruebas, permitiendo probar continuamente el código. Cada vez que se desea implementar las funcionalidades que tendrá el software, XP propone una redacción sencilla de prueba, para ser pasada por el código posteriormente. El proceso constante de las pruebas permite la obtención un producto con mayor calidad ofreciendo a los programadores una mayor certeza en el trabajo que desempeñan. En la metodología XP hay dos tipos de pruebas; las unitarias o desarrollo dirigido por pruebas (TDD *test driven development*), desarrolladas por los programadores verificando su código de forma automática, y las pruebas de aceptación, las cuáles son evaluadas luego de culminar una iteración verificando así que se cumplió la funcionalidad requerida por el cliente. Con estas normas se obtiene un código simple y funcional de manera bastante rápida y eficiente. Por esto es importante pasar las pruebas al 100%.

3.4.1 Desarrollo dirigido por Pruebas

El desarrollo dirigido por pruebas, se enfoca en la implementación orientada a pruebas. El código debe ser probado paso a paso para lograr un resultado, aunque no con lógica para el negocio, pero si funcional. Algunas personas confunden este término con las llamadas “pruebas de caja blanca” las cuáles se les practican a los métodos u operaciones para medir la funcionalidad del mismo, desde el punto de vista de validez del cliente. Sin embargo, el TDD se aplica antes de comenzar a implementar cada paso de la tarea en desarrollo, asumiendo que la prueba es insatisfactoria desde un inicio. Sólo una vez que se haya cumplido de la forma más sencilla posible la lógica del código a probar se asume como cumplida. Luego se realiza un proceso conocido como “refactorización” de código



perteneciente a una de las doce prácticas planteadas por la metodología XP, el cual consiste en mantener el código en buen estado, modificándolo activamente para que conserve claridad y sencillez. Es esencia el TDD, se enfoca en la lógica del negocio y las pruebas de caja blanca en la lógica del negocio.

3.4.2 Pruebas de Aceptación

Las pruebas de aceptación en XP, se pueden asociar con las pruebas de caja negra que se aplican en otras metodologías de desarrollo, sólo que se crean a partir de las historias de usuario y no por un listado de requerimientos. Durante las iteraciones, las HU se traducen a pruebas de aceptación. En ellas se especifican desde la perspectiva del cliente, los escenarios para probar que una historia de usuario ha sido implementada correctamente. La misma puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo que persiguen estas pruebas, es garantizar que las funcionalidades solicitadas por el cliente han sido realizadas. Una HU no se considera completa hasta que no ha transitado por sus pruebas de aceptación. Luego de ver los paradigmas anteriores empleados para la realización de las pruebas y reunirse con el cliente para su análisis, el mismo decidió que se lleve a cabo el proceso mediante las pruebas de aceptación.

La planilla utilizada para plasmar el contenido de las pruebas de aceptación se muestra a continuación.

Planilla de prueba de aceptación

Prueba de aceptación
HU: Nombre de la historia de usuario que va a comprobar su funcionamiento.
Nombre: Nombre del caso de prueba.
Descripción: Descripción del propósito de la prueba.
Condiciones de ejecución: Precondiciones para que la prueba se realice.
Entrada/Pasos ejecución: Pasos para probar la funcionalidad.
Resultado: Resultado que se desea de la prueba.
Evaluación de la prueba: Aceptada o denegada.

Tabla 13 Planilla de prueba de aceptación.



Para ver las pruebas de aceptación remitirse al [ANEXO 4 PRUEBAS DE ACEPTACIÓN](#)

En este capítulo se llevó a cabo la fase de desarrollo y diseño donde se presentó el modelo de datos de la aplicación a construir, logrando una visión detallada de sus atributos y las relaciones entre sus clases, además, las principales clases mediante el empleo de las tarjetas CRC. Se realizó el desarrollo de las iteraciones a partir de la distribución de tareas por HU, y se les practicó las pruebas de aceptación a las funcionalidades de mayor importancia.



CAPÍTULO 4: ESTUDIO DE FACTIBILIDAD

En los momentos actuales es muy importante tener en cuenta que a la hora de desarrollar un proyecto, el mismo cuente con eficacia y eficiencia a la hora de su implementación, y por supuesto se hace preciso efectuar y evaluar la factibilidad antes de su elaboración. En general los productos informáticos no están exentos de posibles riesgos en la concepción del proyecto, por lo que es válido minimizar de forma razonable recursos humanos, materiales y financieros, de ahí que es de vital importancia estimar la relación costo-beneficio, así como el esfuerzo, capital humano y el tiempo de desarrollo que se demanda en la ejecución de los mismos.

En este capítulo se expone el estudio de factibilidad del proyecto, centrado en estimaciones de esfuerzo humano, tiempo de desarrollo para su ejecución y costo. Se estiman los beneficios tangibles e intangibles que representan para el sistema propuesto, un análisis de costos y beneficios.

4.1. Evaluación Costo-Beneficio

La mayoría, por no decir todos los proyectos de informática, son evaluados según el criterio de Costo-Beneficio. Esta Metodología, plantea que la conveniencia de la ejecución de un proyecto se determina por la observación conjunta de dos factores:

- El costo, que involucra la implementación de la solución informática, adquisición y puesta en marcha del sistema hardware/software y los costos de operación asociados.
- La efectividad, que se entiende como la capacidad del proyecto para satisfacerla necesidad, solucionar el problema o lograr el objetivo para el cual se ideó, es decir, un proyecto será más o menos efectivo con relación al mayor o menor cumplimiento que alcance en la finalidad para la cual fue ideado (costo por unidad de cumplimiento del objetivo). El desarrollo de un producto informático, siempre tiene un costo.



Este puede estar justificado por los beneficios tanto tangibles como intangibles que origina el mismo. En este proceso, se necesita de una selección adecuada de los elementos más convenientes para su evaluación.

4.2.Efectos Económicos.

- Efectos directos.
- Efectos indirectos.
- Efectos externos.
- Intangibles.

Efectos directos:

POSITIVOS:

- Los usuarios con acceso al Sistema tendrán la posibilidad de crear registros confiables información de consumo energético.
- Se mejora la eficiencia y calidad del proceso de control del consumo energético
- A pesar de no contar con un instrumento de control para cada local se obtendrá una medida del consumo aproximado de los mismos para futuras tomas de decisiones.
- Se obtendrá un resumen con los consumos de cada área Administrativa.
- Se facilitará el proceso de llevar todas las mediciones realizadas.
- La información mostrada podrá ser guardada en formato pdf.

NEGATIVOS:

- Para el uso de esta aplicación implementada en plataforma Web se necesitará que la misma sea ejecutada con el navegador Mozilla Firefox, porque es con el que se trabajó en la elaboración del producto por lo que el diseño está adaptado a este tipo de navegador.



- El sistema necesita hardware y software de control adicional para responder a algunas funcionalidades.

Efectos indirectos:

- Los efectos económicos observados que pudiera repercutir sobre otros mercados no son perceptibles, aunque este proyecto no está construido con la finalidad de venta.

Efectos externos:

- Se obtuvo un producto disponible que facilita gran parte del trabajo de control de consumo energético en el ISMMM.

Intangibles:

- En la valoración económica siempre hay elementos como perjuicio o beneficio, pero al momento de ponderar en unidades monetarias esto resulta difícil o prácticamente imposible.

4.3. Elementos para identificar los Costos y Beneficios del Proyecto

Para la identificación de los costos y beneficios del proyecto que son pertinentes para su evaluación, es necesario definir una situación base o situación sin proyecto; la comparación de lo que sucede con proyecto, versus lo que hubiera sucedido sin proyecto, definirá los costos y beneficios pertinentes del mismo (María, 2009). Estos escenarios, resultan ser una herramienta determinante, puesto que ayudan en gran medida en la definición de los elementos necesarios para la evaluación.

- **Situación Sin Proyecto**

Para llevar a cabo la gestión del control del consumo energético se realizan los siguientes pasos:



1. Realizar las mediciones de los contadores y determinar el consumo total de ese portador.
2. Determinar el consumo de electricidad del día para cada medición
3. Registrar las actividades de transportación.
4. Obtenerse reportes de consumo real según corresponda.

Nota: En este proceso se pueden cometer errores debido al gran volumen de información que se maneja.

- **Situación Con Proyecto**

Para llevar a cabo la gestión del control del consumo energético se realizan los siguientes pasos:

1. Realizar las lecturas de los contadores.
2. Calculo automático del consumo de electricidad en el día
3. Realizar el control individual de cada local.
4. Gestionar los datos de las actividades de transportación.
5. Conocer estado de la conexión con Nport.
6. Establecer conexión con el servidor de control automático Nport.
7. Obtener de forma inmediata un resumen del consumo en formato pdf.

Expuestas ambas situaciones, se procede a continuación con la identificación de los costos y beneficios del proyecto.

4.4. Factibilidad Económica

El análisis de factibilidad económica identifica los costos y beneficios asociados con el proyecto. El mismo incluye cuatro categorías (Torres, 2005):

- Costo de desarrollo
- Costos operacionales
- Beneficios tangibles



- Beneficios intangibles

Esta tesis, por motivos de comprensión, englobará los costos de desarrollo y los costos operacionales en Evaluación Económica.

4.5. Evaluación Económica

Los principales factores a considerar para el cálculo de los costos son los relacionados al personal, hardware y software, los que se pueden calcular de diversas maneras, que muchas veces se limitan al buen criterio y a la experiencia.

Para determinar el costo económico del proyecto, se desglosaron los costos en moneda libremente convertible y moneda nacional.

Costos en moneda libremente convertible

Ficha de Costo		
		Precios (\$)
Costos Moneda Libremente Convertible		
Costos Directos		
Compra de equipos de cómputo		0.00
Alquiler de equipos de cómputo		0.00
Compra de licencia de Software		0.00
Depreciación de equipos		25.00
Materiales directos		7.50
Subtotal		32.50
Costos Indirectos		
Formación del personal que elabora el proyecto		0.00
Gastos en llamadas telefónicas		0.00
Gastos para el mantenimiento del centro		0.00
Know How		0.00
Gastos en representación		0.00
Subtotal		0.00
Gastos de Distribución y Venta		
Participación en ferias o exposiciones		0.00



Gastos en transportación		0.00
Compra de materiales de propagandas		0.00
Subtotal		0.00
	Total	32.50

Tabla 14 Costos en moneda libremente convertible

Costos en Moneda Nacional

Ficha de Costo	
	Precios (\$)
Costos Moneda Nacional	
Costos Directos	
Salario del personal que laborará en el proyecto	100.00
12,5% del total de gastos por salarios se dedica a la seguridad social	0.00
9.09% de salario total, por concepto de vacaciones a acumular	0.00
Gasto por consumo de energía eléctrica	15.00
Gastos en llamadas telefónicas	0.00
Gastos administrativos	0.00
Subtotal	115.00
Costos Indirectos	
Know How	0.00
Subtotal	0.00
Gastos de Distribución y Venta	
Total	115.00

Tabla 15 Costos en Moneda Nacional

La evaluación económica se efectúa conjuntamente con la que se puede llamar evaluación técnica del proyecto, que consiste en cerciorarse de la factibilidad técnica del mismo. En el análisis de la Factibilidad Técnica del proyecto, se pudo apreciar que:

- Se cuenta con hardware mayor a los requisitos mínimos



- Se cuenta con la disponibilidad de software

Por lo que se puede inferir que el proyecto es factible técnicamente.

Normalmente no se encuentran problemas en relación con el mercado o la tecnología disponible que se empleará en la fabricación del producto; por tanto la decisión de inversión casi siempre recae en la evaluación económica.

Los costos de los proyectos de informática son relativamente simples de cuantificar, no así los beneficios, que se presentan como ahorro de costos con respecto a la situación base. (Rebaza, 2007)

4.5.1. Beneficios tangibles

Son los ingresos adicionales y/o reducción de costos que el nuevo sistema proveerá. Debido a que este proyecto no es construido con la finalidad de comercializarse, no posee ingresos monetarios perceptibles, no así con la reducción de costos, pues el sistema proveerá, mejoras en la gestión y la toma de decisiones.

Estos elementos son de muy difícil cuantificación, por lo que en esta tesis se consideran sólo como intangibles.

4.5.2. Beneficios y Costos Intangibles en el proyecto

Costos

- Resistencia al cambio.

Beneficios

- Mejor comodidad para los usuarios.
- Mejora la calidad del control del consumo.
- Menor tiempo empleado en el proceso de control del consumo.
- Conexión con un servidor de control automático.



Como se hizo referencia anteriormente, la técnica seleccionada para evaluar la factibilidad del proyecto es la Metodología Costo- Efectividad. Dentro de la misma la técnica de punto de equilibrio aplicable a proyectos donde los beneficios tangibles no son evidentes, el análisis se basa exclusivamente en los costos. Para esta técnica es imprescindible definir una variable discreta que haga variar los costos. Teniendo en cuenta que el costo para este proyecto es despreciable, tomaremos como costo el tiempo en minutos empleado para resolver la gestión del control del consumo energético y la variable sería la complejidad de las pruebas que se realizan durante este proceso.

Valores de la variable (Solución manual):

- Gestionar la lectura diaria de cada contador. Cuatro variables (10 min.).
- Buscar los locales según su ubicación y efectuar el control. Tres variables(15 min)
- Elaborar un resumen del consumo de combustible del mes por cada portador energético. Dos variables (10 min).
- Elaborar un resumen de consumo aproximado de cada área administrativa. una variable (45 min.).

Valores de la variable (Solución con el software):

- Cargar cada formulario con los datos necesarios para la gestión de las lecturas diarias. Cuatro variables (3 min.).
- Cargar los formularios con los datos necesarios para filtrar los locales según su ubicación y efectuar el control de los mismos. Dos variable (0.8 min.).
- Obtener resumen de consumo de combustible en transportación. Una variable (0.16 min.).
- Obtener un resumen de consumo aproximado de cada Área Administrativa. Dos variables (1 min.).

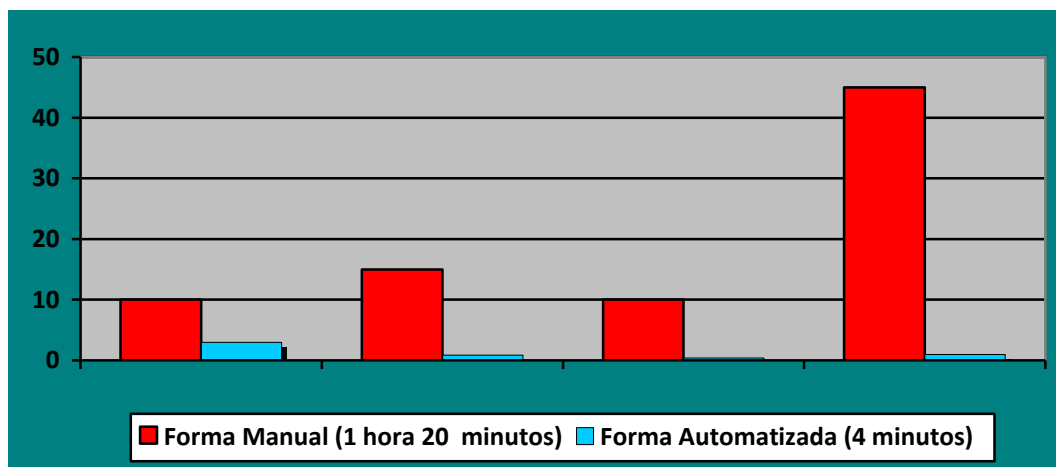


Figura 4.1 Punto de equilibrio.

En este capítulo se mostró el Estudio de Factibilidad Costo-Beneficio con el cual se analizaron los efectos económicos, los beneficios y costos intangibles del proyecto. Al mismo tiempo, se calculó el costo de su ejecución, el cual arrojó como resultado la utilización de \$ 32.50 en CUC y \$ 115.00 en MN que demostraron la factibilidad del mismo.



CONCLUSIONES

Con el desarrollo de la aplicación Web para el control del consumo energético del ISMMM se dio cumplimiento a los objetivos propuestos en este trabajo, pues se obtuvo como resultado un producto informático en el que se aplican los resultados de la investigación realizada arribándose a las siguientes conclusiones:

- Se elaboró el marco teórico metodológico que fundamenta la investigación, permitiendo analizar la estructura del proceso de control del consumo energético en el ISMMM.
- Se realizó el análisis para determinar las principales herramientas usadas en el desarrollo de la aplicación.
- Se creó un módulo para el Sistema GEA que permite realizar el control del consumo energético en el ISMMM.
- Se realizó un estudio de factibilidad donde se obtuvo como resultado el tiempo de estimación para el desarrollo del sistema, así como el costo.
- Se obtuvo una gráfica donde se comparan los tiempos empleados para la realización del control del consumo energético de forma manual y de forma automática, donde se evidencia la disminución de este tiempo al realizar la tarea de forma automática con la ayuda del software.



RECOMENDACIONES

1. Se recomienda la explotación de las funcionalidades que brinda el software en la institución.
2. Graficar el comportamiento del consumo de los portadores en cada año.
3. Instalar un servidor NPort e incorporarle al módulo un sistema especializado para el control automático.
4. Generar reportes en tiempo real del consumo eléctrico con ayuda del componente especializado que se recomendó anteriormente



BIBLIOGRAFÍA

Achour Mehdi, y otros Manual de PHP [En línea]. - 2005. - enero de 2011. - <http://es.php.net/manual/es/>.

Ajaxcrud.com Ajaxcrud.com [En línea]. - 2009. - febrero de 2011. - <http://www.ajaxcrud.com/index.php>.

BECK K. Extreme Programming Explained. Embrace Change [Libro]. - [s.l.] : Pearson Education, 1999.

BLANCO CRIADO A. [En línea] // XAMPP. - 2008. - diciembre de 2010. - <http://www.adictosaltrabajo.com/tutoriales>.

Bureaudeprensa.com [En línea] // Embarcadero ER/Studio. - enero de 2011. - http://bureaudeprensa.com/es/view.php?bn=bureaudeprensa_software&key=115375595.

CAVSI [En línea] // ¿Qué es un Sistema Gestor de Bases de Datos o SGBD? . - 2004. - diciembre de 2010. - <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.

Ciberaula.com Ciberaula.com [En línea] // Una Introducción a APACHE. - 2005. - diciembre de 2010. - http://linux.ciberaula.com/articulo/linux_apache_intro..

CIIN Blog del CIIN (Centro de Innovación en Integración de Cantabria) [En línea] // Software As a Service (SaaS): ¿Qué es?. - 2005. - noviembre de 2010. - <http://geeks.ms/blogs/ciin/archive/2007/10/05/software-as-a-service-sas-191-qu-233-es.aspx>.

Crystal Methodologies [En línea] // Crystal Methodologies. - 2008. - enero de 2011. - <http://crystalmethodologies.blogspot.com>.

DESOFT Ficha Técnica Celador S2C [Libro]. - 2009.



Dondo Agustín PHP en castellano [En línea] // ¿Por qué elegir PHP?. - 2005. - diciembre de 2010. - <http://www.programacion.com/php/articulo/porquephp/>.

Eaprende.com Aprende.com. [En línea] // Gestor de Base de Datos: MySQL, PostgreSQL, SQLite. - 2001. - enero de 2011. - <http://www.eaprende.com/gestor-de-basededatos-mysql-postgresql-sqlite.html..>

Graham Paul The Other Road Ahead [Libro]. - 2001.

Hervás Gómez Carlos Las nuevas tecnologías en la Educación Primaria. Internet: los procesos de enseñanza-aprendizaje con la world wide web [Libro]. - [s.l.]: Grupo de Tecnología Educativa, Universidad de Sevilla, 2006.

Htmlpoint.com Htmlpoint.com [En línea] // ¿Qué es Apache?. - 2008. - diciembre de 2010. - <http://www.htmlpoint.com/faq/apache/01.htm>.

Intercambiosvirtuales.org [En línea] // PHP Designer 2008 v6.0.2.0 Professional. - 2009. - enero de 2011. - <http://www.intercambiosvirtuales.org/software/php-designer-2008-v6020-professional>.

Islasoft.com Islasoft.com [En línea] // Diferencias entre Microsoft Access y Microsoft SQL Server/MSDE.. - 2006. - enero de 2011. - http://www.islasoft.com/Producto/Diferencias_SQL_Server_Office.pdf..

Janium.com Janium.com [En línea] // Aplicaciones basadas en Web. - 2009. - noviembre de 2010. - <http://www.janium.com/page2/page1/page6/page7/page7.html..>

JIMÉNEZ IGLESIAS E.R Sistema automatizado para la gestión del producto final de los servicios de la UEB Organización empresarial de la Empresa Empleadora del Níquel. (SGPFS) [Libro]. - [s.l.] : Instituto Superior Minero Metalúrgico , 2009.

María F. S. Rincondelvago.com [En línea] // "Evaluación de proyectos.". - 2009. - noviembre de 2010. - <http://html.rincondelvago.com/evaluacion-de-proyectos.html..>



Masternewmedia.org Masternewmedia.org [En línea] // Beneficios De Las Aplicaciones Basadas En Web Y El Anuncio De Microsoft De La Era “En Vivo”. - 2005. - diciembre de 2010. -

http://www.masternewmedia.org/es/aplicaciones_web/temas_de_aplicaciones_web/Beneficios_De_Las_Aplicaciones_Basadas_En%20Web_Y_El_Anuncio_De_Microsoft_De_La_Era_En_Vivo.htm.

Millán Tejedor Ramón Jesús [En línea] // Desarrollo de Sitios web dinámicos. - 2005. - enero de 2011. - <http://www.ramonmillan.com/documentos/dhtml.pdf>..

Monografias.com Monografias.com [En línea]. - 2008. - enero de 2011. - <http://www.monografias.com/trabajos14/modelodebase/modelodebase.shtml> ..

Monografias.com Monografias.com [En línea] // Definición arquitectura cliente servidor. - 2007. - enero de 2011. - <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml>..

Moxa.com www.moxa.com [En línea]. - 2009. - febrero de 2011.

Neuron Corp. [En línea] // ¿Qué es UML?. - 2006. - enero de 2011. - http://www.neuronsrl.com.ar/training/uml/uml_intro.html..

Nieves Borrero Martha y Góngora Rodríguez Asnier Herramienta Informática para automatizar los procesos en el Laboratorio de Calidad de Software: Módulo Gestión de las No Conformidades [Libro]. - Ciudad de La Habana: Universidad de las Ciencias Informáticas, 2007.

Rebaza J. C. V. Sociedad de Estudiantes de Ciencia de la Computación SECC [En línea] // Metodología para la Priorización de Proyectos Informáticos. - 2007. - enero de 2011. - <http://www.seccperu.org/node/302>..

REYNOSO C y KICCILLOF N. [En línea] // Estilos y Patrones en la Estrategia de Arquitectura de Microsoft. - 2007. - <http://www.willydev.net/descargas/prev/Estiloypatron.pdf>.



Ronda Amador Yoelys, Cobas Santos Kadir y Marrero Viñas Daymel El modelo cliente-servidor de tres capas para el trabajo en redes de información [Libro]. - 2002.

Torres D. [En línea] // Proyectos Informáticos. - 2005. - enero de 2011. - <http://notasprisma.tripod.com/Proyectos.htm..>

Universidad de Valparaíso Educnet-Universidad de Valparaíso [En línea] // Evaluación del lenguaje.. - 2006. - enero de 2011. - <http://educnet.decom-uv.cl/educnet/uploads/Implementacion.pdf?nombre=p379/Implementacion.pdf..>

Universidad Nacional de Lomas de Zamora [En línea] // ¿Qué es Internet?. - 2008. - diciembre de 2010. - <http://www.unlz.edu.ar/biblioteca/tutores/histoweb/datosgral.html>.

Vegas Jesús El Servidor Web [Libro]. - [s.l.] : Universidad de Valladolid, 2002.

Vico.org Vico.org [En línea] // Unified Modeling Language. - 2002. - enero de 2011. - <http://www.vico.org/FormMentorOutsourcingUML.pdf..>

Wikipedia.org Wikipedia, la enciclopedia libre [En línea]. - enero de 2011. - <http://www.eswikipedia.org>.



GLOSARIO DE TÉRMINOS

Web 2.0: El término Web 2.0 fue acuñado por Tim O'Reilly en 2004 para referirse a una segunda generación en la historia de la Web basada en comunidades de usuarios y una gama especial de servicios, como las redes sociales, los blogs y los wikis, que fomentan la colaboración y el intercambio ágil de información entre los usuarios

MULTILIN: Es un dispositivo para el control automático de consumo energético

NPort: Un servidor de dispositivo de serie industrial que mejora la habilidad de un anfitrión de Windows (i.e., la pc) para controlar múltiples dispositivos de serie sobre una red de Ethernet basado en TCP / IP.

RAM: Memoria de Acceso Aleatorio (Random Access Memory)

Bugs: Errores presentes en la aplicación que atentan contra su correcto funcionamiento

HTTP: Protocolo de Transferencia de Hipertexto (HyperText Transfer Protocol)

JSP: Java Server Pages (Ver Epígrafe 1.2.4)

Logs: Término para definir la creación de un fichero donde se registran las acciones de los usuarios

GTK: Conjunto multiplataforma de bibliotecas para el desarrollo de interfaces gráficas de usuarios

DBMA: Por sus siglas en inglés: Database Management System

BSD: Licencia de software libre permisiva (Berkeley Software Distribution)

CIDR: Encaminamiento Inter-Dominios sin Clases (Classless Inter-Domain Router)

IBM: International Business Machines (conocida también como el Gigante Azul)

DDL: Lenguaje de Definición de Datos (Data Definition Language)

DML: Lenguaje de Manipulación de Datos (Data Manipulation Language)

**ANEXOS****ANEXO 1: HISTORIAS DE USUARIOS****HU No.2: Gestión de control de actividades de transportación**

Historia de usuario	
Número: 1	Usuario: Controlador del consumo
Nombre: Gestión de control de actividades de transportación	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 3
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite el registro y control del consumo de combustible real correspondiente a una actividad de transportación.	
Observaciones: Confirmado con el cliente.	

Tabla16. HU No.16: Gestión de control de actividades de transportación

HU No.3: Gestión de mediciones

Historia de usuario	
Número: 1	Usuario: Controlador del consumo
Nombre: Gestión de mediciones.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Alexei Cala Hinojosa.	



Descripción: Permite registrar la lectura diaria de los contadores y mostrar el consumo correspondiente a los distintos días del mes, así como el consumo total de energía eléctrica registrada por el contador.
Observaciones: Confirmado con el cliente.

Tabla 17. HU No.3: Gestión de mediciones

HU No.4: Gestión de contadores.

Historia de usuario	
Número: 1	Usuario: Controlador del consumo
Nombre: Gestión de contadores	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Alexei Cala Hinojosa.	
Descripción: El usuario debe insertar los datos correspondientes a los contadores energéticos, estos datos podrán ser mostrados, eliminados y actualizados.	
Observaciones: Confirmado con el cliente.	

Tabla 18. HU No.4: Gestión de contadores.

HU No.5: Conexión con MULTILIN.

Historia de usuario	
Número: 1	Usuario: Controlador del consumo
Nombre: Registro de control de MULTILIN	



Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 3
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite que el sistema se conecte con el controlador automático.	
Observaciones: Confirmado con el cliente.	

Tabla 19. HU No.5: Conexión con MULTILIN.

HU No.6: Reportes de consumo mensual

Historia de usuario	
Número: 1	Usuario: Controlador del consumo
Nombre: Gestión de mediciones.	
Prioridad en el negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite mostrar distintos reportes de consumo en formato PDF	
Observaciones: Confirmado con el cliente.	

Tabla 20. HU No.6: Reportes de consumo mensual

HU No.7: Crear Filtro de Locales.

Historia de usuario



Número: 1	Usuario: Controlador del consumo		
Nombre: Crear Filtro de Locales			
Prioridad en el negocio: Alta		Riesgo en desarrollo: Alto	
Puntos estimados: 3		Iteración asignada: 1	
Programador responsable: Alexei Cala Hinojosa.			
Descripción: El usuario debe seleccionar las opciones o restricciones que desee para mostrar la información.			
Observaciones: Confirmado con el cliente.			

Tabla 21. HU No.7: Crear Filtro de Locales.



ANEXO 2: TAREAS DE INGENIERÍA

Tarea de Ingeniería No.1: Insertar registro de control del local

Tarea ingeniería	
Número tarea: 1	Número historia: 1
Nombre tarea: Insertar Registro de control del local.	
Tipo de tarea : Desarrollo	Puntos estimados: 2
Fecha inicio: 1/2/2011	Fecha fin: 9/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Esta tarea posibilita insertar un registro de control de consumo energético para cada local calculando automáticamente el consumo aproximado en proporción con el consumo planificado en caso de ser necesario.	

Tabla 22. Tarea de Ingeniería No.1: Insertar Registro de control del local

Tarea de Ingeniería No.2: Mostrar registro de control del local

Tarea ingeniería	
Número tarea: 2	Número historia: 1
Nombre tarea: Mostrar registro de control del local	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 3/2/2011	Fecha fin: 4/2/2011
Programador responsable: Alexei Cala Hinojosa.	



Descripción: Esta tarea muestra un registro de todos los controles que se han realizado durante el mes, es realizada por el controlador del local.

Tabla 23. Tarea de Ingeniería No.2: Mostrar registro de control del local

Tarea de Ingeniería No.3: Modificar registro de control del local

Tarea ingeniería	
Número tarea: 3	Número historia: 1
Nombre tarea: Modificar registro de control del local	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 4/2/2011	Fecha fin: 6/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Esta tarea permite modificar los registros de consumo de cada local para actualizarlos, es realizada por el controlador del local.	

Tabla 24. Tarea de Ingeniería No.3: Modificar registro de control del local

Tarea de Ingeniería No.4: Eliminar registro de control del local

Tarea ingeniería	
Número tarea: 4	Número historia: 1
Nombre tarea: Eliminar registro de control del local	
Tipo de tarea : Desarrollo	Puntos estimados: 1



Fecha inicio: 5/2/2011	Fecha fin: 6/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite eliminar un registro de control determinado, es realizada por el controlador del local.	

Tabla 25. Tarea de Ingeniería No.4: Eliminar registro de control del local

Tarea de Ingeniería No.5: Insertar control de actividades de transportación

Tarea ingeniería	
Número tarea: 4	Número historia: 2
Nombre tarea: Insertar control de actividades de transportación	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 8/2/2011	Fecha fin: 12/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite insertar un registro de control de da transportación.	

Tabla 26 Tarea de Ingeniería No.5: Insertar control de actividades de transportación

Tarea de Ingeniería No.6: Mostrar control de actividades de transportación

Tarea ingeniería	
Número tarea: 1	Número historia: 2
Nombre tarea: Mostrar control de actividades de transportación	



Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 8/2/2011	Fecha fin: 12/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra todas las actividades de transportación realizadas en un período de tiempo al usuario.	

Tabla 27. Tarea de Ingeniería No.6: Mostrar control de actividades de transportación

Tarea de Ingeniería No.7: Modificar control de actividades de transportación

Tarea ingeniería	
Número tarea: 2	Número historia: 2
Nombre tarea: Modificar control de actividades de transportación	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 8/2/2011	Fecha fin: 12/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite modificar los datos de consumo de las actividades de transportación.	

Tabla 28. Tarea de Ingeniería No.7: Modificar control de actividades de transportación

Tarea de Ingeniería No.8: Eliminar control de actividades de transportación

Tarea ingeniería	
Número tarea: 3	Número historia: 2
Nombre tarea: Eliminar control de actividades de transportación	



Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 8/2/2011	Fecha fin: 12/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Elimina de la base de datos una actividad de transportación	

Tabla 29. Tarea de Ingeniería No.8: Eliminar control de actividades de transportación

Tarea de Ingeniería No.9: Actualización del consumo mensual de combustible

Tarea ingeniería	
Número tarea: 4	Número historia: 2
Nombre tarea: Actualización del consumo mensual de combustible	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 13/2/2011	Fecha fin: 14/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Al insertarse una nueva actividad de transportación, esta tarea permite modificar el registro de consumo mensual de combustible	

Tabla 30. Tarea de Ingeniería No.9: Actualización del consumo mensual de combustible

Tarea de Ingeniería No.10: Insertar mediciones de contadores

Tarea ingeniería	
Número tarea: 1	Número historia: 3



Nombre tarea: Insertar mediciones de contadores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 15/3/2011	Fecha fin: 17/3/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite registrar la lectura diaria de un contador y calcular el consumo de electricidad del contador hasta la fecha de lectura.	

Tabla 31. Tarea de Ingeniería No.10: Insertar mediciones de contadores

Tarea de Ingeniería No.11: Mostrar mediciones de contadores

Tarea ingeniería	
Número tarea: 2	Número historia: 3
Nombre tarea: Mostrar mediciones de contadores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 17/2/2011	Fecha fin: 18/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra todas las lecturas hechas durante el mes para cada contador.	

Tabla 32. Tarea de Ingeniería No.11: Mostrar mediciones de contadores

**Tarea de Ingeniería No.12: Modificar mediciones de contadores**

Tarea ingeniería	
Número tarea: 3	Número historia: 3
Nombre tarea: Modificar mediciones de contadores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 19/2/2011	Fecha fin: 20/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite modificar los datos de las mediciones	

Tabla 33. Tarea de Ingeniería No.12: Modificar mediciones de contadores

Tarea de Ingeniería No.13: Eliminar mediciones de contadores

Tarea ingeniería	
Número tarea: 4	Número historia: 3
Nombre tarea: Eliminar mediciones de contadores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 20/2/2011	Fecha fin: 20/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Elimina una medición, es realizada por el energético del ISMM	

Tabla 34. Tarea de Ingeniería No.13: Eliminar mediciones de contadores

**Tarea de Ingeniería No.14: Calcular el consumo del día**

Tarea ingeniería	
Número tarea: 4	Número historia: 3
Nombre tarea: Calcular el consumo del día	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 21/2/2011	Fecha fin: 23/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Calcula el consumo el consumo diario de electricidad cuando se inserta o se modifica una nueva medición.	

Tabla 35. Tarea de Ingeniería No.14: Calcular el consumo del día

Tarea de Ingeniería No.15: Insertar contadores

Tarea ingeniería	
Número tarea: 1	Número historia: 4
Nombre tarea: Insertar contadores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 30/2/2011	Fecha fin: 31/2/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Inserta un nuevo contador energético, es realizada por el energético del ISMM.	

Tabla 36. Tarea de Ingeniería No.15: Insertar contadores

**Tarea de Ingeniería No.16: Mostrar contadores**

Tarea ingeniería	
Número tarea: 2	Número historia: 4
Nombre tarea: Mostrar contadores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 30/3/2011	Fecha fin: 31/3/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: muestra la información correspondiente a los contadores energéticos, es realizada por el energético.	

Tabla 37. Tarea de Ingeniería No.16: Mostrar contadores

Tarea de Ingeniería No.17: Modificar contadores

Tarea ingeniería	
Número tarea: 3	Número historia: 4
Nombre tarea: Modificar contadores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 30/3/2011	Fecha fin: 31/3/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite modificar los datos de los contadores energéticos	

Tabla 38. Tarea de Ingeniería No.17: Modificar contadores

Tarea de Ingeniería No.18: Eliminar contadores



Tarea ingeniería	
Número tarea: 4	Número historia: 4
Nombre tarea: Eliminar contadores	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 30/3/2011	Fecha fin: 31/3/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Elimina un contador de la base de datos.	

Tabla 39. Tarea de Ingeniería No.18: Eliminar contadores

Tarea de Ingeniería No.19: Conectar el sistema con MULTILIN

Tarea ingeniería	
Número tarea: 1	Número historia: 5
Nombre tarea: Conectar el sistema con MULTILIN	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 3/6/2011	Fecha fin: 8/6/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite conectar el sistema con MULTILIN mediante NPort.	

Tarea 40. de Ingeniería No.19: Conectar el sistema con MULTILIN

Tarea de Ingeniería No.20: Saber estado de la conexión con MULTILIN

Tarea ingeniería



Número tarea: 1	Número historia: 5
Nombre tarea: Saber estado de la conexión con MULTILIN	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 10/5/2011	Fecha fin: 13/5/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Permite realizar consulta del estado de la conexión con MULTILIN.	

Tabla 41. Tarea de Ingeniería No.20: Saber estado de la conexión con MULTILIN

Tarea de Ingeniería No.21: Lista de locales por áreas administrativas

Tarea ingeniería	
Número tarea: 2	Número historia: 7
Nombre tarea: Lista de locales por áreas administrativas	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 14/5/2011	Fecha fin: 18/5/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra los locales de un área seleccionada	

Tabla 42. Tarea de Ingeniería No.21: Lista de locales por áreas administrativas

Tarea de Ingeniería No.22: Lista de locales por subáreas administrativas

Tarea ingeniería



Número tarea: 3	Número historia: 7
Nombre tarea: Lista de locales por subáreas administrativas	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 19/5/2011	Fecha fin: 20/5/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra los locales de una Subárea seleccionada, es realizada por el energético	

Tabla 43. Tarea de Ingeniería No.22: Lista de locales por subáreas administrativas

Tarea de Ingeniería No.23: Lista de locales por edificios

Tarea ingeniería	
Número tarea: 4	Número historia: 7
Nombre tarea: Lista de locales por edificios	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 21/5/2011	Fecha fin: 22/5/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra todos los locales de un edificio determinado	

Tabla 44. Tarea de Ingeniería No.23: Lista de locales por edificios

Tarea de Ingeniería No.24: Lista de locales por tipos de locales

Tarea ingeniería



Número tarea: 5	Número historia: 7
Nombre tarea: Lista de locales por tipos de locales	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 23/5/2011	Fecha fin: 24/5/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra todos los locales según el tipo de local seleccionado (aulas, laboratorios, talleres, etc.)	

Tabla 45. Tarea de Ingeniería No.24: Lista de locales por tipos de locales

Tarea de Ingeniería No.25: Lista de locales por áreas de consumo final

Tarea ingeniería	
Número tarea: 6	Número historia: 7
Nombre tarea: Lista de locales por áreas de consumo final	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 25/5/2011	Fecha fin: 26/5/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra todos los locales atendiendo al área de consumo final a la cual pertenece.	

Tabla 46. Tarea de Ingeniería No.25: Lista de locales por áreas de consumo final

Tarea de Ingeniería No.26: Reporte de consumo mensual por locales

Tarea ingeniería



Número tarea: 6	Número historia: 6
Nombre tarea: Reporte de consumo mensual por locales	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 1/6/2011	Fecha fin: 2/6/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra un reporte del control del consumo energético de cada Local.	

Tabla 47. Tarea de Ingeniería No.26: Reporte de consumo mensual por locales

Tarea de Ingeniería No.27: Generar reporte de consumo mensual de combustible en
transportación

Tarea ingeniería	
Número tarea: 6	Número historia: 6
Nombre tarea: Generar reporte de consumo mensual de combustible en transportación	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 3/6/2011	Fecha fin: 6/6/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra un reporte del consumo mensual de combustible consumido en las actividades de transportación.	

Tabla 48. Tarea de Ingeniería No.27: Generar reporte de consumo mensual de
combustible en transportación

**Tarea de Ingeniería No.28:** Generar reporte de consumo por Áreas

Tarea ingeniería	
Número tarea: 6	Número historia: 6
Nombre tarea: Generar reporte de consumo por Áreas	
Tipo de tarea : Desarrollo	Puntos estimados: 1
Fecha inicio: 3/6/2011	Fecha fin: 5/6/2011
Programador responsable: Alexei Cala Hinojosa.	
Descripción: Muestra un reporte del consumo mensual de de cada Area.	

Tabla 49. Tarea de Ingeniería No.28: Generar reporte de consumo por Áreas

ANEXO 3: TARGETAS CRC
Tarjeta CRC No.1: Control Local

Nombre de la clase: ControlLocal	
Tipo de la clase: Lógica de negocio	
Responsabilidades:	Colaboradores:
Local	
Equipo de control	
Controlador	
Mes	
Año	
Constructor de la clase	
Sets . . .	
Gets . . .	

Tabla 50. Tarjeta CRC No.1: Control Local

Tarjeta CRC No.2: Control Local Acceso a Datos

Nombre de la clase: ControllocalDAO	
Tipo de la clase: Acceso a datos	
Responsabilidades:	Colaboradores:
Salvar	Controllocal,Connect
Modificar	Controllocal,Connect
Eliminar	Controllocal,Connect



Obtener Todos	Controllocal,Connect
Otener un Objeto	Controllocal,Connect
Saber si Existe el Objeto	Controllocal,Connect

Tabla 51. Tarjeta CRC No.2: Control Local Acceso a Datos

Tarjeta CRC No.3: Servicio ControlLocal

Nombre de la clase: ControllocalService	
Tipo de la clase: Servicio	
Responsabilidades:	Colaboradores:
Salvar	ControllocalDAO
Actualizar	ControllocalDAO
Eliminar	ControllocalDAO
Obtener Todos	ControllocalDAO
Otener un Objeto	ControllocalDAO

Tabla 52. Tarjeta CRC No.3: Servicio ControlLocal

Tarjeta CRC No.4: Contadores

Nombre de la clase: Contador	
Tipo de la clase: Lógica de negocio	
Responsabilidades:	Colaboradores:
Gestionar Contadores	AjaxCrud

Tabla 53. Tarjeta CRC No.4: Contadores

Tarjeta CRC No.5: Transportación

Nombre de la clase: Transportacion
Tipo de la clase: Lógica de negocio



Responsabilidades:	Colaboradores:
Gestionar Control de transportación	AjaxCrud

Tabla 54. Tarjeta CRC No.5: Tarjeta CRC No.5: Transportación

Tarjeta CRC No.6: MULTILIN

Nombre de la clase: Nport	
Tipo de la clase: Lógica de negocio	
Responsabilidades:	Colaboradores:
NPort	NPort
Ping_NPort	ComprobarC

Tabla 55. Tarjeta CRC No.5: MULTILIN

Tarjeta CRC No.6: Filtro

Nombre de la clase: Filtro	
Tipo de la clase: Lógica de negocio	
Responsabilidades:	Colaboradores:
Lista de locales por Áreas Lista de locales por Subáreas Lista de locales por edificios Lista de locales por tipos de locales Lista de locales por Áreas de consumo	Funciones

Tabla 56. Tarjeta CRC No.6: Filtro

Tarjeta CRC No.7: Medición

Nombre de la clase: Medición	
Tipo de la clase: Lógica de negocio	



Responsabilidades:	Colaboradores:
contador lectura mes año consumo	

Tabla 57. Tarjeta CRC No.7: Medición

Tarjeta CRC No.8: MedicionDAO

Nombre de la clase: MedicionDAO	
Tipo de la clase: Acceso a datos	
Responsabilidades:	Colaboradores:
Salvar Modificar Eliminar	Medicion, Connect Medicion, Connect Medicion, Connect

Tabla 58. Tarjeta CRC No.8: MediciónDAO

Tarjeta CRC No.9: Connect

Nombre de la clase: Connect	
Tipo de la clase: Lógica de negocio	
Responsabilidades:	Colaboradores:
servidor usuario contraseña puerto Constructor	

Tabla 59. Tarjeta CRC No.9: Connect

ANEXO 4: PRUEBAS DE ACEPTACIÓN

Prueba de aceptación No 1. Gestión de mediciones

Prueba de aceptación: Gestión de mediciones
HU: Gestión de mediciones.
Nombre: Prueba para comprobar la gestión de las mediciones.
Descripción: Muestra las mediciones con sus consumos.
Condiciones de ejecución: el usuario debe seleccionar la opción medición
Entrada/Pasos ejecución: • El usuario selecciona la opción medición. • El usuario selecciona el mes y el año de la medición
Resultado: • Son visualizados correctamente todas las mediciones del mes en una tabla. • En esa tabla las mediciones pueden ser actualizadas • Es posible eliminar mediciones • Cuando se inserta una nueva medición se calcula automáticamente el consumo del día y se actualiza el consumo mensual de electricidad.
Evaluación de la prueba: Aceptada.

Tabla 60. Prueba de aceptación No 1. Gestión de mediciones

Prueba de aceptación No 2. Filtro

Prueba de aceptación
HU: Filtro de locales.
Nombre: Prueba para comprobar la visualización de los locales.
Descripción: muestra la opción que seleccione el usuario.
Condiciones de ejecución: el usuario debe seleccionar las opciones en el filtro y como mínimo debe seleccionar el área
Entrada/Pasos ejecución: • El usuario selecciona la opción control local. • Luego selecciona las restricciones que brinda el filtro
Resultado: • Es mostrado un error en caso de no seleccionar el área • Son visualizados correctamente todos los locales según las restricciones seleccionadas por el usuario.
Evaluación de la prueba: Aceptada.

Tabla 61. Prueba de aceptación No 2. Filtro

Prueba de aceptación No 3. Actividades de transportación

Prueba de aceptación
HU: Gestión de actividades de transportación.
Nombre: Prueba para comprobar el trabajo de la clase con AjaxCrud.
Descripción: muestra las actividades en una tabla de gestión generada por AjaxCrud.
Condiciones de ejecución: El usuario debe seleccionar la opción control de transportación
Entrada/Pasos ejecución: • El usuario selecciona la opción.
Resultado: • Es mostrada una tabla con todos los atributos de las actividades, las cuales pueden ser modificadas, eliminadas, además se permite insertar nuevas actividades.
Evaluación de la prueba: Aceptada.

Tabla 62. Prueba de aceptación No 3. Actividades de transportación

Prueba de aceptación No 4. Contadores

Prueba de aceptación
HU: Gestión de contadores.
Nombre: Prueba para comprobar el trabajo de la clase con AjaxCrud.
Descripción: muestra los contadores en una tabla de gestión generada por AjaxCrud.
Condiciones de ejecución: El usuario debe seleccionar la opción contadores
Entrada/Pasos ejecución: • El usuario selecciona la opción.
Resultado: • Es mostrada una tabla con todos los atributos de los contadores, los cuales pueden ser modificados, eliminados, además se permite insertar nuevos contadores.
Evaluación de la prueba: Aceptada.

Tabla 63. Prueba de aceptación No 4. Contadores

Prueba de aceptación No 5. Control Local

Prueba de aceptación
HU: Gestionar el registro de control por Locales
Nombre: Prueba para comprobar la Gestión del control de los locales
Descripción: permite realizar el control del consumo de un portador en un local
Condiciones de ejecución: El usuario debe seleccionar la opción controlar cuando son visualizados los locales en el filtro
Entrada/Pasos ejecución: • El usuario selecciona la opción. • El usuario inserta los datos correspondientes
Resultado: • Es mostrado un mensaje indicándole al usuario la realización correcta de la tarea. • Si el usuario selecciona la opción control colectivo es mostrado un mensaje indicándole al usuario que el sistema calculó un consumo estimado para el local en base a lo que éste se había planificado. • En caso de ya existir el control es mostrado un mensaje indicándole al usuario que ese control ya se realizó.
Evaluación de la prueba: Aceptada.

Tabla 64. Prueba de aceptación No 5. Control Local

Prueba de aceptación No 6. Conexión con MULTILIN

Prueba de aceptación
HU: Conexión con MULTILIN
Nombre: Prueba para comprobar la conexión del sistema con NPport
Descripción: permite conectar el sistema con NPort
Condiciones de ejecución: El usuario debe seleccionar la opción MULTILIN
Entrada/Pasos ejecución: • El usuario selecciona la opción. • El usuario inserta la dirección IP del servidor y el puerto de escucha del mismo
Resultado: • En caso de no existir el servidor el sistema brinda un mensaje al usuario indicándole el problema ocurrido. • El usuario puede saber el estado de la conexión con el servidor mediante la realización del comando ping desde el sistema.



Evaluación de la prueba: Aceptada.

Tabla 65. Prueba de aceptación No 6. Conexión con MULTILIN

Prueba de aceptación No 7. Reportes de consumo mensual

Prueba de aceptación
HU: Reportes de consumo mensual
Nombre: Prueba para comprobar la generación de reportes
Descripción: permite obtener diferentes reportes de consumo
Condiciones de ejecución: el usuario debe seleccionar la opción reportes de consumo
Entrada/Pasos ejecución: • El usuario selecciona la opción. • El usuario selecciona el tipo de reporte • Introduce los datos para generar el reporte
Resultado: • Se visualiza la ventana de diálogo donde el cliente decide si va a guardar el reporte o verlo en ese momento. • En caso de seleccionar la opción guardar el documento es guardado correctamente. • En caso de seleccionar la opción abrir, es mostrado un documento en formato pdf con los datos del reporte. • Se muestra un mensaje de error en caso de introducir la fecha incorrectamente.
Evaluación de la prueba: Aceptada.

Tabla 66. Prueba de aceptación No 7. Reportes de consumo mensual