



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Sistema Web Inteligente para la Gestión del ahorro de Energía en Refrigeración e Iluminación en Hoteles de México

TESIS PROFESIONAL
para obtener el título de
INGENIERO EN COMPUTACIÓN

PRESENTA
OSCAR ESTRADA VARGAS

DIRECTOR DE TESIS
Dr. Nicolas Ceferino Kemper Valverde



Ciudad Universitaria, México, 2013

ÍNDICE

I.- Introducción..... 7

II.- Problemática Energética en Hoteles de México denominados PYMES..... 9

 II.1 Descripción de las PYMES 9

 II.2 Planteamiento del Problema en las PYMES. 9

 II.3 Solución al Ahorro de Energía por medio de la Gestión Energética..... 10

III.- Eficiencia Energética..... 11

 III.1 Definición e Importancia de la Gestión Energética..... 11

 III.2 Beneficios del ahorro de energía. 12

 III.3 Gestión Energética para el ahorro de Energía en Hoteles de México..... 14

IV.- Fundamentos Teóricos y Prácticos. 17

 IV.1 Sistemas Expertos. 17

 IV.1.1 Historia y características..... 17

 IV.1.2 Tipos e Importancia..... 18

 IV.1.3 Ventajas: 19

 IV.1.4 Limitaciones 19

 IV.1.5 Aplicación al sistema web. 20

 IV.2 Programación WEB..... 21

 IV.2.1 Tecnologías para Desarrollo WEB. 21

 IV.2.1.1 Programación en la Nube..... 21

 IV.2.1.1.1 Tipos de Nube 22

 IV.2.1.1.2 Computación en la Nube 23

 IV.2.1.1.3 Ventajas 24

 IV.2.1.1.4 Arquitecturas de Programación en la Nube..... 25

 IV.2.1.1.5 Páginas WEB..... 33

 IV.2.2 Lenguajes de programación WEB..... 34

 IV.2.2.1 HTML..... 35

 IV.2.2.2 Hojas de Estilo en Cascada (CSS)..... 37

IV.2.2.3 JavaScript	39
IV.2.2.4 JQuery.....	40
IV.2.2.5 AJAX.....	40
IV.2.2.6 PHP.....	41
IV.3 WEB 2.0.....	44
IV.3.1 Historia, Evolución y Características de la WEB 2.0.....	44
IV.3.2 Frameworks para Aplicaciones WEB	47
IV.3.3 Frameworks para PHP.....	48
IV.3.3.1 Modelo Vista Controlador.....	50
IV.3.3.2 ZEND.....	52
IV.3.3.3 Symfony	53
IV.3.3.4 CodeIgniter	53
IV.3.3.5 Seagull.....	54
IV.3.4 Yii Framework	54
IV.4 Servidor WEB	60
IV.4.1 Arquitectura Cliente Servidor	61
IV.4.1.1 Aplicación del lado del Servidor.....	62
IV.4.1.2 Aplicación del lado del Cliente.....	63
IV.4.2 Envío y Recepción de Datos (POST y GET).....	65
IV.4.2.1 método GET.....	65
IV.4.2.2 Método POST.....	66
IV.4.3 Funcionamiento del Servidor WEB.....	67
IV.5 Base de Datos Orientada a WEB.....	69
IV.5.1 Bases de Datos	69
IV.5.1.1 Descripción y Características.....	69
IV.5.1.2 Ventajas	70
IV.5.1.3 Desventajas	71
IV.5.1.4 Modelos.....	71
IV.5.1.5 Modelo Relacional.....	73
IV.5.1.5.1 Ventajas	73

IV.5.2 Modelo Entidad Relación.....	74
IV.5.3 SQL.....	76
V.- Desarrollo e Implementación del Sistema web Inteligente.....	79
V.1 Descripción y Estructura Base	79
V.2 Modelos de las Áreas en la Gestión Energética.....	84
V.2.1 Refrigeración.....	87
V.2.2 Iluminación Interior	93
V.2.3 Iluminación Exterior.....	100
V.3 Código de Desarrollo	106
V.3.1 Estructura Básica General del Sistema (Template)	106
V.3.2 Módulo de Registro, Autenticación y Validación.....	110
V.3.3 Módulo de Ingreso al Sistema y Encuesta.....	125
V.3.4 Módulo de Diagnóstico Energético	132
VI.- Pruebas y Generación de Resultados	149
VI.1 Demostración del Sistema web Inteligente para el Ahorro de Energía.....	150
VII.- Conclusiones.....	175
VIII.- Bibliografía.....	177

ÍNDICE DE FIGURAS

Figura III.1 Modelo del Sistema de Gestión de la Energía ISO 50001.....	13
Figura IV.2.1 Computación en la Nube.....	23
Figura IV.2.1.1.4.1.1 Aplicaciones del SaaS.....	27
Figura IV.2.1.1.4.1.2 Kits de desarrollo de software de PaaS.....	28
Figura IV.2.1.1.4.1.3 Implementación sobre IaaS.....	31
Figura IV.2.1.1.4.1.4.1 Funcionalidad de los Modelos de Servicios.....	32
Figura IV.2.1.1.4.1.4.2 Beneficios de los Modelos de Servicios.....	32
Figura IV.2.1.1.5 Funcionalidad Básica de un Navegador Web.....	33
Figura IV.2.2.2 Formación una página HTML.....	37
Figura IV.2.2.6 Función general de PHP.....	43
Figura IV.3.3.1 Interacción entre una petición y su respuesta usando el MVC.....	52
Figura IV.3.4.1 Estructura estática de una aplicación Yii.....	56
Figura IV.3.4.2 Flujo de trabajo de una aplicación Yii.....	57
Figura V.1.1 Diagrama de bloques de Eficiencia Energética en Hoteles.....	80
Figura V.1.2 Petición del Cliente al Servidor.....	83
Figura V.2.1 Diagrama de Flujo del Módulo de Autenticación.....	84
Figura V.2.2 Diagrama de Flujo para el Diagnóstico de la Gestión Energética.....	86

ÍNDICE DE TABLAS

Tabla IV.3.3 Comparativa de los Frameworks para PHP.....	49
Tabla V.1 Normas Mexicanas que se Aplican a las Áreas.....	81

INTRODUCCIÓN

Capítulo 1

I.- Introducción.

Hoy día en México se vive una significativa encrucijada energética y medio ambiental. Por un lado, su producción y reservas petroleras decrecen aceleradamente, sin poder consolidar hasta ahora, opciones alternas. Por el otro, su importancia en el mundo y su nivel de emisiones de gases de efecto invernadero llevan a que se hace patente la necesidad de que la energía sea considerada como un factor de costos que requiere especial atención.

La mala administración de la energía se debe en gran medida a la ignorancia sobre el ahorro de la misma ya que los microempresarios no cuentan con el personal o los conocimientos necesarios y aunado a la falta de un buen presupuesto han llevado a las PYMES en categoría de hotel a no realizar las inversiones necesarias para cumplir con un funcionamiento óptimo obteniendo un uso inadecuado de la energía, generando costos elevados, pérdida de energía y daños al medio ambiente.

Con el avance tecnológico que existe actualmente se han desarrollado sistemas para la resolución de problemas, puestos en práctica para un sin fin de usos, aplicados a la educación, investigación, ciencia, desarrollo tecnológico, medicina, ingeniería, etc. por lo que es posible delegar responsabilidades a los sistemas de cómputo para optimizar y hacer que las cargas de trabajo sean más prácticas y eficientes.

En la actualidad las empresas hoteleras en categoría de PYMES han visto como los energéticos han pasado de ser un factor marginal en su estructura de costos a constituir un rubro importante en los mismos, a la vez que, la necesidad de lograr un mayor equilibrio entre economía y medio ambiente, por este motivo México necesita renovarse y redefinirse en este nuevo contexto. El reducir el consumo de energía a través de la implementación de un Sistema Web Inteligente de Gestión de la Energía es la clave para combatir la escasez de energía y el consecuente aumento del costo de la misma de igual forma puede ayudarle a reducir las emisiones de Gases del Efecto Invernadero .

PROBLEMÁTICA ENERGÉTICA EN HOTELES DE MÉXICO DENOMINADOS PYMES

Capítulo 2

II.- Problemática Energética en Hoteles de México denominados PYMES.

II.1 Descripción de las PYMES

La pequeña y mediana empresa conocida también por el acrónimo PYME es una empresa con características distintivas, y tiene dimensiones con ciertos límites ocupacionales y financieros prefijados por los Estados o regiones.

Las PYMES son agentes con lógicas, culturas, intereses y un espíritu emprendedor específicos. [1]

Las pequeñas y medianas empresas son entidades independientes, con una alta predominancia en el mercado de comercio, quedando prácticamente excluidas del mercado industrial por las grandes inversiones necesarias y por las limitaciones que impone la legislación en cuanto al volumen de negocio y de personal [2]. Por todo ello una pyme nunca podrá superar ciertas ventas anuales o una cantidad de personal.

Las pequeñas y medianas empresas cumplen un importante papel en la economía de todos los países. Las principales razones de su existencia son [3]:

- Pueden realizar productos individualizados en contraposición con las grandes empresas que se enfocan más a productos más estandarizados.
- Sirven de tejido auxiliar a las grandes empresas. La mayor parte de las grandes empresas se valen de empresas subcontratadas menores para realizar servicios u operaciones que de estar incluidas en el tejido de la gran corporación redundaría en un aumento de coste.
- Existen actividades productivas donde es más apropiado trabajar con empresas pequeñas.

II.2 Planteamiento del Problema en las PYMES.

México vive una significativa encrucijada energética y medio ambiental por lo que día a día la demanda energética que requiere la sociedad en sus diversos sectores de desarrollo va en aumento, haciendo que los requerimientos de los recursos necesarios para su producción también sea mayor.

Esto afecta no solo al medio ambiente, sino que el sector económico y social también se ve perjudicado. Además si observamos la situación actual, notaremos que los cambios se hacen cada vez más notorios y prominentes, lo que conlleva a tomar medidas ya no como una opción sino como a una necesidad donde la energía sea considerada como un factor de costos que requiere atención.

Uno de los principales problemas de las PYMES del sector turístico en categoría de hoteles es la carencia de conocimiento, el no integrar el sistema como un todo, la falta de modernización y carencia económica. Esto es lo que no permite mejorar la satisfacción de sus clientes, el desempeño de su personal, el control de negocio, la disminución de desperdicios y el incremento de la rentabilidad [3].

II.3 Solución al Ahorro de Energía por medio de la Gestión Energética.

Los individuos y las organizaciones que son consumidores directos de la energía pueden desear ahorrar energía para reducir costos energéticos y promover sostenibilidad económica, política y ambiental. Los usuarios industriales y comerciales pueden desear aumentar eficacia y maximizar así su beneficio. Entre las preocupaciones actuales está el ahorro de energía y el efecto medioambiental de la generación de energía eléctrica.

Todos estos problemas pueden ser solventados con el conocimiento necesario y la orientación adecuada sobre lo que el hotelero en categoría de PYME debe y tiene que hacer y aquí es donde nace la necesidad de homogeneizar todas estas insuficiencias para lograr programas y planes de ahorro energético donde es necesaria una gestión energética confiable y eficiente, pero sobre todo se requiere que la gente adquiera una conciencia económica, social y ambiental, que haga los cambios palpables y notorios en pro de una empresa más competitiva.

Con el avance tecnológico que existe actualmente ya no hay excusa para no tener implementado algún sistema de gestión energético y con un sistema eficiente para la gestión energética ahorramos energía, dinero y cuidaremos el medio ambiente lo que sin duda alguna traería muchísimos beneficios a nuestro país.

EFICIENCIA ENERGÉTICA

Capítulo 3

III.- Eficiencia Energética.

III.1 Definición e Importancia de la Gestión Energética.

La gestión energética puede concebirse como un esfuerzo organizado y estructurado para conseguir la máxima eficiencia en el suministro, conversión y utilización de los recursos energéticos. Esto es, lograr un uso más racional de la energía, que permita reducir el consumo de la misma sin perjuicio de la comodidad, productividad, calidad de los servicios y, de un modo general, sin reducir el nivel de vida. Puede considerarse como el mejor de los caminos para conseguir los objetivos de conservación de energía y medio ambiente.

La gestión energética implica monitoreo, registro, evaluación y acción correctiva continua sobre los equipos, áreas, procesos y personal, siendo la clave para reducir los consumos y gastos energéticos.

El monitoreo y control de la energía, la mejoría en la prácticas de operación y mantenimiento y la implementación de medidas técnico organizativas simples, de bajo costo, lograrán mucho más en el corto plazo. A menudo es más sencillo quejarse por la falta económica para invertir en medidas de ahorro de energía, que admitir que la administración exhibe puntos débiles [4].

Si no se cuenta con una administración y un mantenimiento efectivo, no se podrán alcanzar los máximos beneficios de las inversiones en tecnología eficiente.

Debido a esto podemos asegurar que el llevar una Gestión Energética inadecuada, incrementará en gran medida nuestros gastos económicos en los pagos de energía, desperdiciaremos gran cantidad de ella y estaremos afectando al medio ambiente.

En la mayoría de las empresas, y en especial en aquellas en las que el costo energético representa un porcentaje importante de los costos de producción o explotación, cabe plantearse un sistema de gestión energética, conducente a una

optimización en el uso eficiente de la energía, justificado por su rentabilidad en la reducción de los costos energéticos y la protección del medio ambiente [5].

III.2 Beneficios del ahorro de energía.

La Gestión Energética es la suma de medidas planificadas y llevadas a cabo para conseguir el objetivo de utilizar la mínima cantidad posible de energía mientras se mantienen los niveles de confort y los niveles de producción.

Para hacer un uso eficiente de la energía y, como consecuencia, para ahorrarla, las acciones se centran en la conservación de la energía, la recuperación de la energía y la sustitución de la energía. Obteniendo los siguientes beneficios:

- ✓ Aumento de la competitividad de la organización
- ✓ Uso racional y eficiente de la energía.
- ✓ Gestión de mejora continua del sistema.
- ✓ Reducción del consumo de energía.
- ✓ Asegurar el cumplimiento de normas y reglamentos técnicos.
- ✓ Establecimiento índices específicos de consumo, variables de control de los procesos y programas de mantenimiento predictivo.
- ✓ Obtener un diagnóstico de la gestión de los recursos energéticos.
- ✓ Valoración de los riesgos.

Los pasos claves de gestión energética que las empresas necesitan para manejar y reducir sus consumos de energía se centra en revisar los resultados con el objetivo de brindar un enfoque de mejoramiento continuo que pueda ser muy eficaz en el manejo energético. Además, dado que el consumo de energía es a menudo considerado por muchos dentro de las organizaciones como un gasto en lugar de ser considerado un recurso controlable.

La ISO 50001 ha implementado el concepto de “planificar-hacer-verificar-actuar” estableciendo criterios de gestión eficiente de la energía, lo que permite lograr un mejor

rendimiento energético en cualquier área de la organización, optimizando su utilización y, por consiguiente, reduciendo significativamente los costos operativos.

En la figura III.1 se muestra el Modelo del Sistema de Gestión de la Energía [6], donde se puede observar las etapas de este proceso:



Figura III.1 Modelo del Sistema de Gestión de la Energía ISO 50001.

El Modelo del Sistema de Gestión Energética ISO [6] describe cómo la gestión energética debe encajar en la jerarquía de gestión e incluir los componentes claves de un buen sistema de gestión energética como: política; planificación; establecimiento de metas y objetivos; capacitación; participación; comunicación; adquisición de equipos; y diseño.

Mediante la aplicación de los principios establecidos en la norma [7], las organizaciones ahora pondrán en práctica una sencilla estructura de trabajo para alcanzar sus metas de energía corporativas. Esto se logrará a través del mejoramiento de la competencia y efectividad de sus propios equipos de gestión y del mercado, el cual podría ofrecerles las soluciones que ellos necesitan para resolver uno de los problemas históricos en gestión energética, la sustentabilidad de los ahorros logrados.

III.3 Gestión Energética para el ahorro de Energía en Hoteles de México.

Ahora toca hablar sobre la gestión energética que existe actualmente en el sector hotelero en México, este tema se tratará basado en la experiencia personal en pláticas y prácticas que se han tenido con expertos y con los diferentes encargados de administrar, mantener y ampliar hoteles.

Actualmente en México, el problema que existe en las PYMES hoteleras es que muchos de ellos no están destinados para tal fin, es decir, son adaptados para convertirse en hoteles provocando que la administración de energía se dificulte; por ejemplo los materiales con los que está construido el edificio cambian de acuerdo a las necesidades, las adaptaciones que se hacen para proporcionar servicios de aire acondicionado, distribución del sistema de agua, instalación de calentadores para el agua y algunas más no son las más adecuadas. En general, dichas adaptaciones están implementadas sin tomar en consideración el uso del lugar y una gestión energética eficiente.

En algunos sitios adaptados como hotel, se opta por la instalación de un banco de capacitores, sin antes haber realizado un estudio y haber identificado los factores que más influyen en el consumo energético requerido, tales como el mal dimensionamiento de equipo, problemas en la instalación eléctrica, falta de mantenimiento a equipo eléctrico como; bombas, lámparas y equipos de acondicionamiento, etc. La dificultad que se puede presentar para la implementación de una gestión energética suele ser, en general, la Insuficiente especialización del personal técnico en las diferentes áreas.

En México, la gestión energética en hoteles pequeños y medianos o denominados PYMES es casi nula, solo existen algunas políticas de ahorro energético que no están difundidas para su correcto uso además de que no se lleva un plan continuo con el fin de optimizar y hacer más eficiente el consumo energético. Muchas

veces la falta de información hace que no haya interés por parte del sector hacia la implementación de un buen plan de gestión energético.

Es muy evidente que no existe una gestión energética actual, ya que la poca o nula información que se tiene sobre el tema no ayuda; por lo tanto es necesario establecer las bases que den comienzo a una conciencia social y las buenas prácticas en la gestión energética actual.

El sector hotelero en México, tiene mucho potencial de desarrollo, solo hay que iniciar con una buena implementación de la gestión energética, con esto, no solo se logrará un gran ahorro económico también se promoverán las nuevas tecnologías y la modernización del sector empresarial, se disminuirá el impacto que se tiene sobre el ambiente conservando recursos para las futuras generaciones; contribuyendo a la formación de una cultura energética y ambiental que pondrá en alto el nombre de nuestro país.

Como ya hemos analizado, hoy en día la gestión eficiente de la energía ya no es una opción, sino una necesidad. Con el avance tecnológico que existe actualmente es posible desarrollar un sistema de gestión energético basado en las tecnologías de la información (TI) con el cual se logre orientar e informar al hotelero sobre los beneficios económicos, sociales y ambientales que se pueden obtener en beneficio de una empresa más competitiva.

En este punto es donde nace la idea de desarrollar un Sistema web Inteligente para la gestión del ahorro de energía en hoteles de México con el cual se logre informar, capacitar y orientar al hotelero para que realicen los análisis de sus instalaciones y obtengan propuestas de mejora en el uso de la energía buscando el menor impacto al medio ambiente y la mejor remuneración económica.

FUNDAMENTOS TEÓRICOS Y PRÁCTICOS

Capítulo 4

IV.- Fundamentos Teóricos y Prácticos.

IV.1 Sistemas Expertos.

IV.1.1 Historia y características

Los sistemas expertos o sistemas inteligentes Incompletos son llamados así porque emulan el razonamiento de un experto en un dominio concreto y en ocasiones son usados por éstos. Con los sistemas expertos se busca una mejor calidad y rapidez en las respuestas dando así lugar a una mejora de la productividad del experto.

Un Sistema Experto (SE) es una aplicación informática capaz de solucionar un conjunto de problemas que exigen un gran conocimiento sobre un determinado tema. Un sistema experto es un conjunto de programas que, sobre una base de conocimientos, posee información de uno o más expertos en un área específica. Se puede entender como una rama de la inteligencia artificial, donde el poder de resolución de un problema en un programa de computadora viene del conocimiento de un dominio específico. Estos sistemas imitan las actividades de un humano para resolver problemas de distinta índole (no necesariamente tiene que ser de inteligencia artificial). También se dice que un “SE” se basa en el conocimiento declarativo (hechos sobre objetos, situaciones) y el conocimiento de control (información sobre el seguimiento de una acción) [8].

Para que un sistema experto sea una herramienta efectiva, los usuarios deben interactuar de una forma fácil, reuniendo dos capacidades:

1. **Explicar sus razonamientos o base del conocimiento:** los sistemas expertos se deben realizar siguiendo ciertas reglas o pasos comprensibles de manera que se pueda generar la explicación para cada una de estas reglas, que a la vez se basan en hechos.
2. **Adquisición de nuevos conocimientos o integrador del sistema:** son mecanismos de razonamiento que sirven para modificar los conocimientos anteriores. Sobre la base de lo anterior se puede decir que los sistemas expertos

son el producto de investigaciones en el campo de la inteligencia artificial ya que ésta no intenta sustituir a los expertos humanos, sino que se desea ayudarlos a realizar con más rapidez y eficacia todas las tareas que realiza.

Debido a esto en la actualidad se están mezclando diferentes técnicas o aplicaciones aprovechando las ventajas que cada una de estas ofrece para poder tener empresas más competitivas y prácticas. Un ejemplo de estas técnicas sería los agentes que tienen la capacidad de negociar y navegar a través de recursos en línea; y es por eso que en la actualidad es un papel preponderante en los sistemas expertos.

IV.1.2 Tipos e Importancia.

Un Sistema Experto está conformado por:

- **Base de conocimientos (BC):** Contiene conocimiento modelado extraído del diálogo con un experto.
- **Base de hechos (Memoria de trabajo):** contiene los hechos sobre un problema que se ha descubierto durante el análisis.
- **Motor de inferencia:** Modela el proceso de razonamiento humano.
- **Módulos de justificación:** Explica el razonamiento utilizado por el sistema para llegar a una determinada conclusión.
- **Interfaz de usuario:** es la interacción entre el SE y el usuario, y se realiza mediante el lenguaje natural.

Principalmente existen tres tipos de sistemas expertos:

- Basados en reglas previamente establecidas.
- Basados en casos o CBR (Case Based Reasoning).
- Basados en redes bayesianas.

En cada uno de ellos, la solución a un problema planteado se obtiene:

- Aplicando reglas heurísticas apoyadas generalmente en lógica difusa para su evaluación y aplicación.
- Aplicando el razonamiento basado en casos, donde la solución a un problema similar planteado con anterioridad se adapta al nuevo problema.

- Aplicando redes bayesianas, basadas en estadística y el teorema de Bayes.

Como los sistemas expertos son definidos por cierto conocimiento y reglas existen ventajas y desventajas en su utilización:

IV.1.3 Ventajas:

- ✓ **Permanencia:** A diferencia de un experto humano un SE (sistema experto) no envejece, y por tanto no sufre pérdida de facultades con el paso del tiempo.
- ✓ **Replicación:** Una vez programado un SE lo podemos replicar infinidad de veces.
- ✓ **Rapidez:** Un SE puede obtener información de una base de datos y realizar cálculos numéricos mucho más rápido que cualquier ser humano.
- ✓ **Bajo costo:** A pesar de que el costo inicial pueda ser elevado, gracias a la capacidad de duplicación el coste finalmente es bajo.
- ✓ **Entornos peligrosos:** Un SE puede trabajar en entornos peligrosos o dañinos para el ser humano.
- ✓ **Fiabilidad:** Los SE no se ven afectados por condiciones externas, un humano sí (cansancio, presión, etc.).
- ✓ **Consolidar** varios conocimientos.
- ✓ **Apoyo Académico.**

IV.1.4 Limitaciones

- × **Sentido común:** Para un Sistema Experto no hay nada obvio. Por ejemplo, un sistema experto sobre medicina podría admitir que un hombre lleva 40 meses embarazado, a no ser que se especifique que esto no es posible ya que un hombre no puede gestar hijos.
- × **Lenguaje natural:** Con un experto humano podemos mantener una conversación informal mientras que con un SE no podemos.
- × **Capacidad de aprendizaje:** Cualquier persona aprende con relativa facilidad de sus errores y de errores ajenos, que un SE haga esto es muy complicado.
- × **Perspectiva global:** Un experto humano es capaz de distinguir cuales son las cuestiones relevantes de un problema y separarlas de cuestiones secundarias.
- × **Capacidad sensorial:** Un SE carece de sentidos.
- × **Flexibilidad:** Un humano es sumamente flexible a la hora de aceptar datos para la resolución de un problema.
- × **Conocimiento no estructurado:** Un SE no es capaz de manejar conocimiento poco estructurado.

IV.1.5 Aplicación al sistema web.

La implementación de un sistema experto o sistema inteligente incompleto en el Sistema web Inteligente para la gestión del ahorro de energía en refrigeración e iluminación en hoteles de México se basará en la estructura general del SE donde:

- ✦ La **Base de conocimientos** contendrá conocimiento modelado y extraído del diálogo con un experto de cada área (en este caso iluminación y refrigeración).
- ✦ **Base de hechos** (Memoria de trabajo): contendrá los hechos sobre un problema que se ha descubierto durante el análisis y los catálogos que ayudaran a su resolución.
- ✦ **Motor de inferencia:** Modela el proceso de razonamiento humano a través de algoritmos que realicen el diagnóstico energético de cada área.
- ✦ **Módulos de justificación:** Los algoritmos explican el razonamiento utilizado por el sistema para llegar a una determinada conclusión o solución del problema de acuerdo con el conocimiento del experto.
- ✦ **Interfaz de usuario:** es la interacción entre el SE y el usuario, y se realiza mediante el lenguaje natural en este caso será por medio del Sistema web Inteligente para la gestión del ahorro de energía en refrigeración e iluminación en hoteles de México.

Con la implementación de la estructura del sistema experto no se podrá aplicar sentido común, un lenguaje informal, no se tendrá una perspectiva global ni capacidad sensorial pero si podremos replicar infinidad de veces el sistema obteniendo un bajo costo del mismo, no sufriremos pérdida de facultades con el paso del tiempo, se realizaran cálculos numéricos mucho más rápidos y precisos para garantizar el diagnóstico energético, se podrá implementar el sistema en entornos difíciles para el hombre sin afectación por condiciones externas y servirá principalmente como apoyo al diagnóstico energético.

IV.2 Programación WEB.

IV.2.1 Tecnologías para Desarrollo WEB.

IV.2.1.1 Programación en la Nube.

Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas que la componen funcionen como una red lógica única, de alcance mundial. Actualmente las personas dependen muy fuertemente del Internet, ya que puede ser usado para compartir recursos, enviar y recibir correos electrónicos (e-mail), platicar y realizar conferencias en línea y sobre todo para la búsqueda de información.

En los inicios de la Internet los servicios que se ofrecían eran limitados, pero con el paso del tiempo, éstos fueron creciendo y ahora, muchos de los desarrolladores de la TI se están enfocando para proporcionar todo tipo de servicio a través de la Internet.

Las aplicaciones comerciales tradicionales han sido siempre demasiado complicadas y caras. La cantidad y la variedad necesaria de hardware y software para ejecutarlas son abrumadoras. Se necesita todo un equipo de expertos para que las puedan instalar, configurar, probar, ejecutar, asegurar y actualizar. Sistemas completos construidos hasta la fecha poseen ciertas limitantes, como capacidad de almacenamiento, memoria y procesamiento. Actualmente se cuenta con la programación en la nube la cual reduce la cantidad de software y hardware utilizado solventando estas limitantes.

La computación en la nube o también llamada cloud computing (computación en la nube de internet) puede verse como una extensión de: redes de computadoras (grid computing), computación distribuida (distributed computing), computación en paralelo (parallel computing). En la computación en la nube, todo recurso puede ser proporcionado como servicio.

La computación en red (grid computing) provee de recursos a los usuarios solamente cuando son requeridos. La computación en paralelo (parallel computing), ejecuta instrucciones en paralelo para una rápida respuesta al realizar una tarea que el

usuario solicitó, en este tipo de cómputo solo se provee de una rápida respuesta, no se presenta el almacenamiento, ni la memoria como recurso para ser solicitado. Finalmente en la computación distribuida (distributed computing), la información o datos se encuentran distribuidos en diversos servidores, inclusive situados en diferentes áreas geográficas. La Computación en la Nube involucra las ventajas de cada uno de estos paradigmas para formar una fuente de recursos a cualquier nivel, en donde todo puede ser proporcionado como servicio: almacenamiento, capacidad de procesamiento y memoria. [10]

IV.2.1.1.1 Tipos de Nube

- ❖ Una **nube pública** es una nube computacional mantenida y gestionada por terceras personas no vinculadas con la organización. En este tipo de nubes tanto los datos como los procesos de varios clientes se mezclan en los servidores, sistemas de almacenamiento y otras infraestructuras de la nube. Los usuarios finales de la nube no conocen que trabajos de otros clientes pueden estar corriendo en el mismo servidor, red, sistemas de almacenamiento, etc. Aplicaciones, almacenamiento y otros recursos están disponibles al público a través el proveedor de servicios que es propietario de toda la infraestructura en sus centros de datos; el acceso a los servicios solo se ofrece de manera remota, normalmente a través de Internet.
- ❖ Las **nubes privadas** son una buena opción para las compañías que necesitan alta protección de datos y ediciones a nivel de servicio. Las nubes privadas están en una infraestructura bajo demanda gestionada para un solo cliente que controla qué aplicaciones debe ejecutarse y dónde. Son propietarios del servidor, red, y disco y pueden decidir qué usuarios están autorizados a utilizar la infraestructura.
- ❖ Las **nubes híbridas** combinan los modelos de nubes públicas y privadas. Usted es propietario de unas partes y comparte otras, aunque de una manera controlada. Las nubes híbridas ofrecen la promesa del escalado aprovisionada externamente, en demanda, pero añaden la complejidad de determinar cómo

distribuir las aplicaciones a través de estos ambientes diferentes. Las empresas pueden sentir cierta atracción por la promesa de una nube híbrida, pero esta opción, al menos inicialmente, estará probablemente reservada a aplicaciones simples sin condicionantes, que no requieran de ninguna sincronización o necesiten bases de datos complejas.

IV.2.1.1.2 Computación en la Nube

La programación web en la nube es uno de los servicios que se van perfeccionando continuamente y actualmente se tienen aplicaciones, plataformas y sistemas trabajando en su totalidad o parcialmente en internet, esto quiere decir que no requieren de estar instaladas físicamente en la computadora del usuario para su uso, basta con una conexión a internet y una cuenta asociada para explotar los recursos de esta tecnología por medio de diferentes dispositivos tal como lo muestra la figura IV.2.1. A esto es lo que se le conoce como computación en la nube, lo que es básicamente proporcionar servicios de cómputo a través de internet.



Figura IV.2.1 Computación en la Nube.

El cambio que ofrece la computación desde la nube es que permite aumentar el número de servicios basados en la red. Esto genera beneficios tanto para los proveedores, que pueden ofrecer, de forma más rápida y eficiente, un mayor número de servicios, como para los usuarios que tienen la posibilidad de acceder a ellos, disfrutando de la transparencia e inmediatez del sistema y de un modelo de pago por consumo.

IV.2.1.1.3 Ventajas

Algunas ventajas de la computación en la nube son las siguientes:

- ✓ **Rapidez:** Los servicios más básicos de la nube funcionan por sí solos. Para servicios de software y base de datos más complejos, la computación en nube permite saltarse la fase de adquisición de hardware y el consiguiente gasto, por lo cual es perfecta para almacenamiento.
- ✓ **Actualización:** La mayoría de los proveedores actualizan constantemente su software, agregando nuevas funciones tan pronto como están disponibles esto genera una actualización homogénea y rápida para no dejar sin servicios a los usuarios.
- ✓ **Adaptativa:** Adaptable rápidamente a negocios en crecimiento que el sistema en nube está diseñado para hacer frente a fuertes aumentos en la carga de trabajo. Esto incrementa la agilidad de respuesta, disminuye los riesgos y los costos operacionales, porque sólo escala lo que crece y paga sólo lo que usa.
- ✓ **Movilidad:** El sistema en nube está diseñado para ser utilizado a distancia, así que el personal de la empresa tendrá acceso a la mayoría de los sistemas en cualquier lugar donde se encuentre.
- ✓ **Económica:** El proveedor ofrece servicios a múltiples empresas, las cuales se benefician de compartir una moderna y compleja infraestructura, pagando solamente por lo que realmente utilizan, eliminando así gastos en infraestructura innecesaria.

La computación en la nube aún tiene todavía que enfrentar varios retos en los aspectos de seguridad, confiabilidad, interoperabilidad y disponibilidad.

Todo usuario requiere seguridad en la información, sobre todo cuando se trata de información sensible y confidencial: información sobre identidad, cuentas de correo electrónico para uso de negocios, datos bancarios, etc. En la nube, todo debe estar asegurado, previniendo ataques (hacking) y fallas técnicas tanto en software y

hardware, por la forma en cómo se escalan los recursos y servicios, lograr la seguridad requerida puede ser una tarea compleja donde la confianza de los usuarios hacia los prestadores de servicios es un papel muy importante.

La computación en la nube consigue aportar estas ventajas, apoyándose sobre una infraestructura tecnológica dinámica que se caracteriza, entre otros factores, por un alto grado de automatización, una rápida movilización de los recursos, una elevada capacidad de adaptación para atender a una demanda variable, así como virtualización avanzada y un precio flexible en función del consumo realizado, evitando además el uso fraudulento del software.

IV.2.1.1.4 Arquitecturas de Programación en la Nube.

La computación en nube ha evolucionado en una amplia variedad de servicios que incluyen recursos compartidos, software y plataformas "a demanda", el crecimiento ha sido dispar, sin embargo existe un consenso en las funciones ejecutadas por aplicaciones, sistemas operativos y el hardware físico.

El Instituto Nacional para Estándares y Tecnología de los Estados Unidos de América (U.S. National Institute for Standards and Technology – NIST) ha establecido cinco características esenciales que deben estar presentes en un servicio para que se puedan considerar en la nube [12]:

1. **Bajo demanda y automatizado.** El cliente puede aprovisionarse de forma unilateral recursos de cómputo cuando lo requiera tales como, tiempos de servidor y almacenamiento, sin la necesidad de interacción con otra persona por parte del proveedor del servicio, es decir, todo el proceso debe ser automatizado.
2. **Acceso vía red amplia.** Las capacidades y el control de un servicio en la nube deben estar disponibles vía Internet (u otras redes, como las privadas) utilizando protocolos estándar, y bajo una amplia gama de plataformas, incluyendo dispositivos móviles.

3. **Fuente de recursos.** Recursos físicos y virtuales son asignados y reasignados de forma dinámica dependiendo de la demanda, resultando en ahorro de costos para el cliente.
4. **Flexibilidad eficiente.** Aprovechamiento de recursos de cómputo casi inmediatos, para un rápido escalamiento hacia arriba, hacia abajo, y acorde a la demanda.
5. **Servicio medido.** Los recursos de cómputo utilizados por el cliente son monitoreados, controlados, reportados y cargados en completa transparencia, habilitando un convenio de “paga lo que consumes”.

IV.2.1.1.4.1 Servicios de la Computación en la Nube.

Basado en lo anterior existen tres tipos de servicios los cuales son denominados capas de la computación en la nube:

IV.2.1.1.4.1.1 Software como Servicio (Software as a Service, SaaS)

Software como servicio (Software as a Service) se encuentra en la capa más alta y caracteriza una aplicación completa ofrecida como un servicio de una sola instancia del software que corre en la infraestructura del proveedor y sirve a múltiples organizaciones de clientes.

Software como servicio es un término utilizado para describir el software desplegado en Internet y se caracteriza por que el proveedor licencia la aplicación al suscriptor en un modelo de "servicio por demanda". Usualmente el servicio se vende como "pagar por lo que se consume", donde los pagos se realizan mensualmente dependiendo del número de usuarios y servicios usados. Los principales segmentos de mercado del modelo SaaS se encuentran en tópicos como: administración de contenido, colaboración y administrador de las relaciones del consumidor (Customer Relationship Management, CRM) tal como muestra la figura IV.2.1.1.4.1.1.

Algunos ejemplos prominentes de SaaS son: Gmail, Google Docs y Salesforce.com.



Figura IV.2.1.1.4.1.1 Aplicaciones del SaaS.

El usuario final no está involucrado en el diseño de la aplicación (salvo algunas personalizaciones y serie de configuraciones), servidores, redes, o estructura de almacenamiento. Sin embargo, es posible recuperar cierta retroalimentación sobre el uso de las aplicaciones para introducir mejoras, cambios de imagen, automatización, etc. El Software como Servicio (SaaS) está revolucionando el mercado de las aplicaciones con las siguientes ventajas:

- ✓ Sus datos siempre estarán seguros al ser tratados por empresas absolutamente dedicadas a ello y siempre cumpliendo la normativa vigente.
- ✓ No se necesita de instalación, mantenimiento, ni personal calificado trabajando en la empresa. Puede controlar su aplicación sin demasiados conocimientos.
- ✓ Inversión inicial mucho menor: el costo de la aplicación se reparte en el tiempo, siendo mucho más accesible que un desembolso inicial de golpe. El costo, además es casi siempre menor que la compra de la aplicación.
- ✓ Fácil de usar: las aplicaciones se vuelven más sencillas al utilizar la interfaz de los sitios web. En pocos días el usuario estará capacitado y trabajando.
- ✓ Cualquier dispositivo con conexión a Internet permite que sus empleados trabajen a pleno rendimiento y en el lugar del mundo donde se encuentren. Solo con abrir un navegador web se accede a la información.
- ✓ Totalmente configurables, se ofrece como un software estándar, pero permite una personalización absoluta a la empresa cliente.
- ✓ Totalmente escalable: solo pagas por las necesidades del momento.
- ✓ Evita la renovación de equipos a corto plazo. Todos los equipos soportan una conexión a internet, y al menos un navegador web, por lo que puede aguantar con sus equipos funcionando sin necesidad de aumentar la capacidad [11].

IV.2.1.1.4.1.2 Plataforma como Servicio (Platform as a Service , PaaS)

Plataforma como Servicio (Platform as a Service) se refiere a un modelo que no sólo ofrece la plataforma de despliegue y adicionalmente una plataforma de desarrollo de aplicaciones completa. Mientras que en el modelo SaaS se ofrecen aplicaciones listas para utilizarse, el modelo PaaS brinda la opción de construir una aplicación personalizada utilizando la plataforma de desarrollo ofrecida. Los proveedores PaaS ofrecen por medio de sus plataformas soporte para los lenguajes de programación más comunes como Java o .NET reduciendo la dependencia de plataformas SaaS, que usualmente casan los usuarios y organizaciones con su plataforma.

Bajo este esquema se ofrece un Sistema Operativo (SO) junto con colecciones de diversos lenguajes de programación y kits de desarrollo de software para que el usuario final pueda desarrollar sus propias aplicaciones como lo muestra la figura Figura IV.2.1.1.4.1.2.



Figura IV.2.1.1.4.1.2 Kits de desarrollo de software de PaaS.

Los proveedores de un servicio PaaS normalmente ofrecen diversas herramientas que permiten al usuario u organización controlar a diferentes niveles el uso que se le da al servicio PaaS, se pueden monitorear factores como el número de veces que es usada cierta aplicación, el tiempo que se pasa con cierto programa, la actividad que se desarrolla con él, la eficiencia del software y los diversos problemas que se puedan experimentar. Los principales representantes del mercado en el ámbito PaaS son: Amazon Elastic Beanstalk, Microsoft Azure, Google App Engine.

Los defensores del servicio PaaS, argumentan los siguientes beneficios: incremento en la productividad, permite a las organizaciones poner a la venta sus aplicaciones más rápido y reducir costos en licencias de software. También se elimina la necesidad de que el desarrollador tenga que configurar su entorno de trabajo (servidor, red, almacenamiento, configuración, escalamiento, etc.), así como el trabajo de mantenimiento, seguridad y actualizaciones al servidor y sistema operativo. Todo esto es gestionado por el proveedor de servicio PaaS.

Las plataformas PaaS pueden tomar diversos enfoques, desde el relativamente simple, en donde el desarrollador básicamente solo ensambla bloques de código generado por el proveedor; hasta un enfoque más complejo en donde se involucra el diseño y generación de código por parte del desarrollador.

Si el sistema PaaS se cae, si se carece de una conexión a internet cuando el proveedor no permite el trabajo local o más aún si el proveedor deja de proporcionar el servicio a causa de cierre del negocio, la organización se verá afectada, y particularmente si se trabajan con aplicaciones que son de vital importancia para las actividades cotidianas y de misión crítica.

Además, los desarrolladores están limitados a trabajar únicamente con la plataforma que el proveedor les proporciona. La aplicación una vez terminada, estará configurada para ser ejecutada únicamente por la plataforma elegida. Por tanto, si la organización desea migrar de plataforma, será una tarea compleja y tediosa, es por ello que antes de escoger una plataforma sea conveniente hace un estudio detallado y minucioso sobre las que existen en el mercado.

IV.2.1.1.4.1.3 Infraestructura como servicio (Infrastructure as a Service, IaaS)

Infraestructura como servicio (Infrastructure as a Service) es el tercer modelo de implementación de computación en la nube y hace referencia a plataformas que ofrecen infraestructura de cómputo y usualmente se encuentran desplegadas sobre un entorno de virtualización.

La plataforma brinda la alternativa de escalar la infraestructura de manera vertical, es decir, subir y bajar los recursos de cómputo a demanda y se paga por los recursos consumidos.

Los servicios IaaS son típicamente provistos por centros de datos (data centers), que rentan hardware moderno a los usuarios, liberando la preocupación que se acarrea por la depreciación, constante actualización y configuración. Una de las principales características de IaaS es la virtualización, que da la facilidad al usuario de ejecutar su propio sistema operativo huésped por encima del software virtualizado que proporciona el proveedor. La virtualización en IaaS es un paso clave en la distribución, automatización y de una escalable distribución, instalación, y mantenimiento del software.

Algunas ventajas de adoptar un modelo IaaS son las siguientes:

- ✓ Disminución de costos en general
 - Los proveedores de servicio IaaS generalmente cuentan con un mayor poder adquisitivo que las pequeñas y medianas empresas.
 - Ancho de banda con redistribución y mejor eficiencia en su utilización de la fuente de recursos de cómputo.
- ✓ Evitar tecnología obsoleta
 - Con servicios de TI de alta eficiencia como base del negocio, los proveedores de servicio IaaS tienden a tener equipo de última generación y las últimas tecnologías del mercado.
- ✓ Personal cualificado y capacitado
 - La escala de un proveedor de servicio IaaS le permite reclutar profesionales especialistas según requiera el negocio.

Este modelo ofrece el más alto grado de flexibilidad, así como el menor grado de dependencia con la plataforma permitiendo a los usuarios migrar las aplicaciones de un proveedor a otro. Por otro lado una implementación sobre IaaS requiere instalación, configuración y mantenimiento adicionales tal como se muestra en la figura

IV.2.1.1.4.1.3. Amazon Web Services y Rackspace son dos proveedores destacados de IaaS.

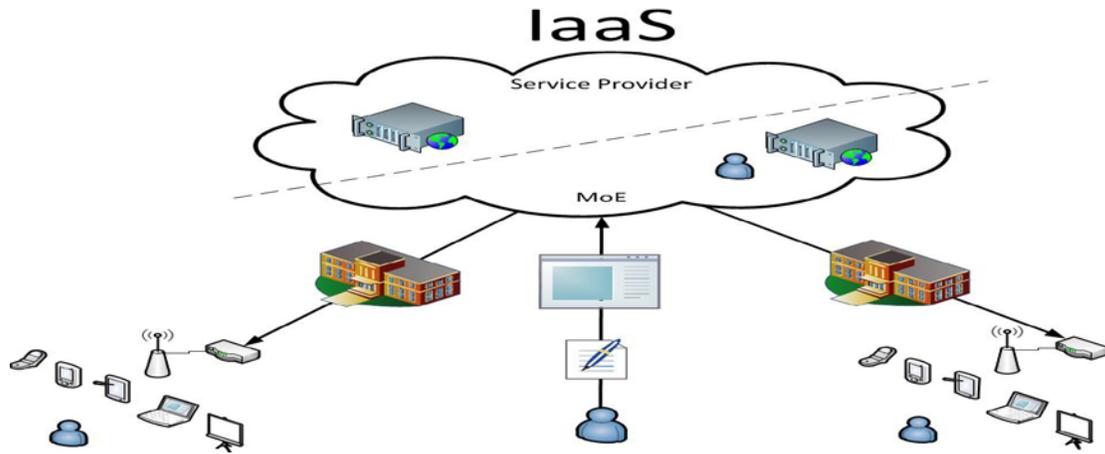


Figura IV.2.1.1.4.1.3 Implementación sobre IaaS.

IV.2.1.1.4.1.4 Diferencia entre capas.

PaaS es similar a IaaS, con la excepción de que en una PaaS se provee de un conjunto de lenguajes de programación y herramientas denominadas como plataformas. Generalmente un servicio PaaS está enfocado a la comunidad de desarrolladores, PaaS se puede ver como una analogía a los servidores de aplicaciones, sólo que ofreciendo características de la nube. Debido a que el software de la plataforma depende del proveedor del servicio, PaaS limita el tipo de aplicaciones que se pueden construir.

SaaS es esencialmente la entrega de aplicaciones para la IT y los usuarios a través de Internet. Un servicio SaaS es análogo a un modelo cliente/servidor, con la excepción de que el servidor es sustituido por el centro de datos del proveedor, los clientes son simplemente los navegadores web en las pc de escritorio o dispositivos móviles, y el servicio ofrece los beneficios de la nube, como la elasticidad y un plan de paga de lo que consumes.

La figura IV.2.1.1.4.1.4.1 muestra los diferentes modelos de servicios describiendo su funcionalidad y un ejemplo de ella.

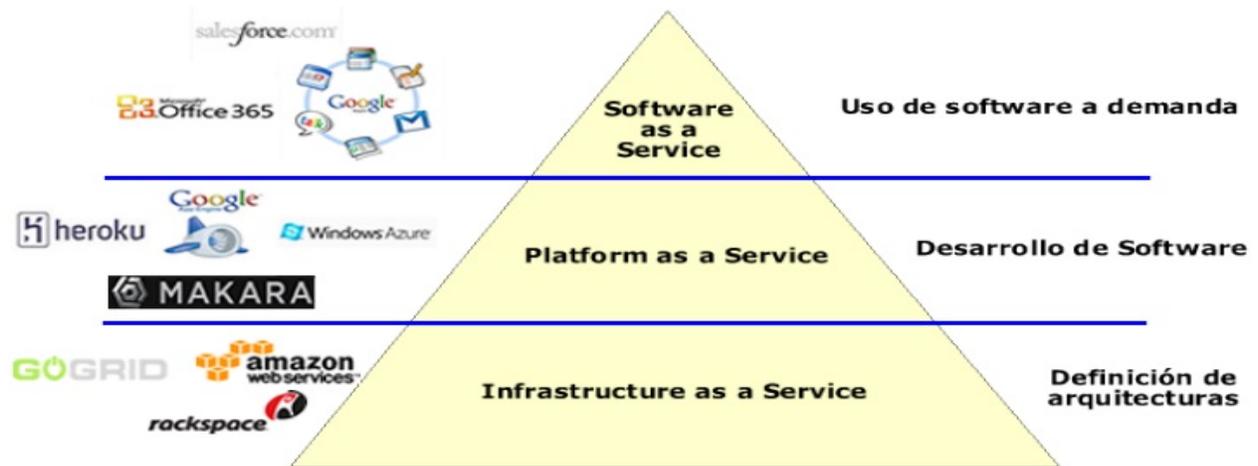


Figura IV.2.1.1.4.1.4.1 Funcionalidad de los Modelos de Servicios.

Para una organización la decisión de qué tipo de implementación utilizar depende de diferentes criterios a tomar en cuenta: control, seguridad, libertad de implementación, mantenimiento, configuración, etc; pero en cualquier caso la organización se vería beneficiada debido a que solamente se factura lo que se consume.

La figura IV.2.1.1.4.1.4.2 muestra los beneficios de cada modelo descrito con anterioridad.

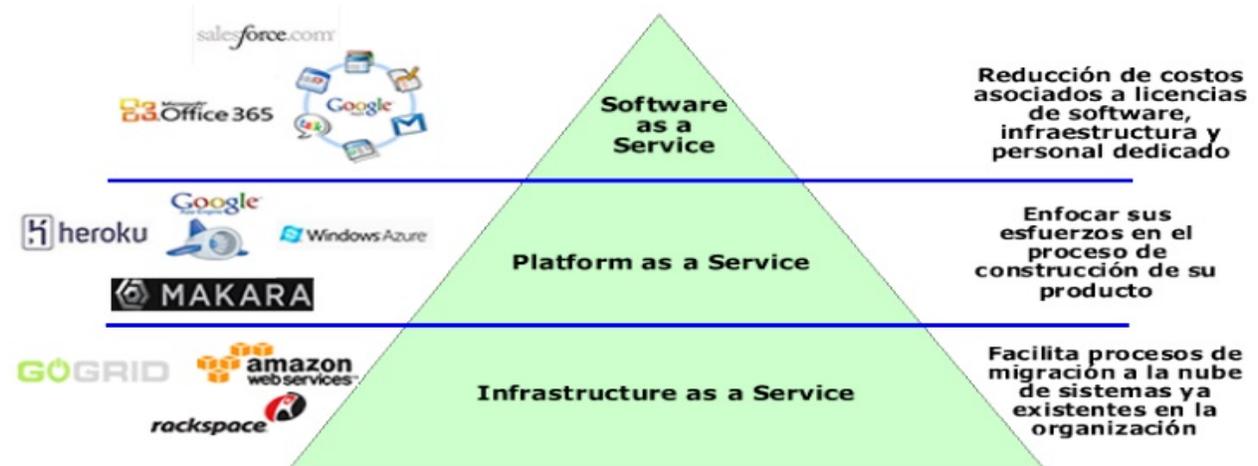


Figura IV.2.1.1.4.1.4.2 Beneficios de los Modelos de Servicios.

IV.2.1.1.5 Páginas WEB.

Una página web o página electrónica es el nombre de un documento o información electrónica adaptada para la *World Wide Web* y que puede ser accedida mediante un navegador. Esta información se encuentra generalmente en formato HTML o XHTML, y puede proporcionar navegación a otras páginas web mediante enlaces de hipertexto. Las páginas web frecuentemente incluyen otros recursos como hojas de estilo en cascada, guiones (*scripts*) e imágenes digitales, entre otros.

Las páginas web pueden estar almacenadas en un equipo local o un servidor web remoto. El servidor web puede restringir el acceso únicamente para redes privadas, p. ej., en una intranet corporativa, o puede publicar las páginas en la World Wide Web. El acceso a las páginas web es realizado mediante su transferencia desde servidores utilizando el protocolo de transferencia de hipertexto (HTTP).

Un navegador o navegador web es una aplicación que opera a través de Internet, interpretando la información de archivos y sitios web para que éstos puedan ser leídos. La funcionalidad básica de un navegador web es permitir la visualización de documentos de texto, posiblemente con recursos multimedia incrustados tal como se muestra en la figura IV.2.1.1.5 donde se toma como ejemplo el navegador “Firefox”.



Figura IV.2.1.1.5 Funcionalidad Básica de un Navegador Web.

Los documentos pueden estar ubicados en la computadora en donde está el usuario, pero también pueden estar en cualquier otro dispositivo que esté conectado en la computadora del usuario o a través de Internet, y que tenga los recursos necesarios para la transmisión de los documentos como por ejemplo un software servidor web.

Como ya se ha mencionado con anterioridad para poder ver un archivo HTML es necesario un navegador web y por ende este último se aloja en un equipo el cual puede estar en intranet o en internet, al estar en la nube de internet se necesita tener uno o varios de los modelos de servicios antes mencionados dependiendo de los requerimientos del sistema ya que ellos proporcionan la infraestructura y dependiendo del modelo de servicio el software que se utilizará para el sistema.

Para fines del sistema web Inteligente para la gestión del ahorro de energía en refrigeración e iluminación en hoteles de México se utilizara una infraestructura, software y aplicación propia; consistiendo en un servidor y una IP fija que serán descritos más adelante a detalle.

IV.2.2 Lenguajes de programación WEB.

Un lenguaje de programación es un lenguaje formal diseñado para expresar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana.

Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila (de ser necesario) y se mantiene el código fuente de un programa informático se le llama programación.

También la palabra programación se define como el proceso de creación de un programa de computadora, mediante la aplicación de procedimientos lógicos, a través de los siguientes pasos:

- ❖ El desarrollo lógico del programa para resolver un problema en particular.
- ❖ Escritura de la lógica del programa empleando un lenguaje de programación específico (codificación del programa).

- ❖ Ensamblaje o compilación del programa hasta convertirlo en lenguaje de máquina.
- ❖ Prueba y depuración del programa.
- ❖ Desarrollo de la documentación.

Desde los inicios de Internet, fueron surgiendo diferentes demandas por los usuarios y se dieron soluciones mediante lenguajes estáticos. A medida que paso el tiempo, las tecnologías fueron desarrollándose y surgieron nuevos problemas a dar solución. Esto dio lugar a desarrollar lenguajes de programación dinámicos para la web, que permitieran interactuar con los usuarios y utilizar sistemas de Bases de Datos.

Actualmente existen diferentes lenguajes de programación para desarrollar en la web, estos han ido surgiendo debido a las tendencias y necesidades de las plataformas.

IV.2.2.1 HTML.

En 1989 existían dos técnicas que permitían vincular documentos electrónicos, por un lado los hipervínculos (*enlaces*) y por otro lado un poderoso lenguaje de etiquetas denominado SGML. Por entonces, Tim Berners-Lee, quien trabajaba en el Centro Europeo de Investigaciones Nucleares CERN da a conocer a la prensa que estaba trabajando en un sistema que permitirá acceder a ficheros en línea que funcionaba sobre redes de computadoras o máquinas electrónicas basadas en el protocolo TCP/IP. Inicialmente fue desarrollado para que se pueda compartir fácilmente información entre científicos de distintas universidades e institutos de investigación de todo el mundo.

A principios de 1990, Tim Berners-Lee define por fin el HTML como un subconjunto del conocido SGML y crea algo más valioso aún, el World Wide Web. En 1991, Tim Berners-Lee crea el primer navegador web, ViolaWWW, que funcionaría en modo texto y sobre un sistema operativo UNIX.

Los trabajos para crear un sucesor del HTML, denominado HTML +, comenzaron a finales de 1993. HTML+ se diseñó originalmente para ser un superconjunto del HTML que permitiera evolucionar gradualmente desde el formato HTML anterior, llegando hasta ahora a la versión HTML 5 la cual aún no está estandarizada del todo por lo que la versión HTML4 sigue en uso.

HTML por las siglas en inglés de HyperText Markup Language (lenguaje de marcado hipertextual), hace referencia al lenguaje de marcado predominante para la elaboración de páginas web que se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes.

El HTML se escribe en forma de “etiquetas”, rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir o hacer referencia a un tipo de programa llamado *script*, el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

El lenguaje HTML es un estándar reconocido en todo el mundo y cuyas normas define un organismo sin ánimo de lucro llamado World Wide Web Consortium, más conocido como W3C. Como se trata de un estándar reconocido por todas las empresas relacionadas con el mundo de Internet, una misma página HTML se visualiza de forma muy similar en cualquier navegador de cualquier sistema operativo.

La flexibilidad de HTML puede parecer un aspecto positivo, pero el resultado final son páginas con un código HTML desordenado, difícil de mantener y muy poco profesional. Afortunadamente, XHTML soluciona estos problemas añadiendo ciertas normas en la forma de escribir las etiquetas y atributos. A continuación se muestran las cinco restricciones básicas que introduce XHTML respecto a HTML en la sintaxis de sus etiquetas: [13]

1. Las etiquetas se tiene que cerrar de acuerdo a como se abren.
2. Los nombres de las etiquetas y atributos siempre se escriben en minúsculas.
3. El valor de los atributos siempre se encierra con comillas.

4. Los atributos no se pueden comprimir.
5. Todas las etiquetas deben cerrarse siempre.

Actualmente, entre HTML 4.01 y XHTML 1.0, la mayoría de diseñadores escogen XHTML. En un futuro cercano, si los diseñadores deben elegir entre HTML 5 y XHTML 1.1 o XHTML 2.0, quizás la elección sea diferente.

IV.2.2.2 Hojas de Estilo en Cascada (CSS).

Originalmente, las páginas HTML sólo incluían información sobre sus contenidos de texto e imágenes. Con el desarrollo del estándar HTML, las páginas empezaron a incluir también información sobre el aspecto de sus contenidos: tipos de letra, colores y márgenes.

Incluir en una misma página HTML los contenidos, el diseño y la programación complica en exceso su mantenimiento. Normalmente, los contenidos y el diseño de la página web son responsabilidad de diferentes personas, por lo que es conveniente separarlos.

Las hojas de estilo en cascada o “Cascading Style Sheets” por sus siglas en ingles CSS hacen referencia a un lenguaje de hojas de estilos usado para describir la presentación semántica, es decir, el aspecto y formato de un documento escrito en lenguaje de marcas. Su aplicación más común es dar estilo a páginas webs escritas en lenguaje HTML y XHTML, pero también puede ser aplicado a cualquier tipo de documentos XML, incluyendo SVG y XUL.

CSS es el mecanismo que permite separar los contenidos definidos mediante XHTML y el aspecto que deben presentar esos contenidos para formar una página HTML tal como se muestra en la figura IV.2.2.2 [13].



Figura IV.2.2.2 Formación una página HTML.

Para dar formato a un documento HTML, puede emplearse CSS de tres formas distintas:

1. Mediante CSS introducido por el autor del HTML :

✂ **Un estilo en línea** (online) es un método para insertar el lenguaje de estilo de página directamente dentro de una etiqueta HTML. Esta manera de proceder no es totalmente adecuada. El incrustar la descripción del formateo dentro del documento de la página Web, a nivel de código, se convierte en una manera larga, tediosa y poco elegante de resolver el problema de la programación de la página.

✂ **Una hoja de estilo interna**, que es una hoja de estilo que está incrustada dentro de un documento HTML, dentro del elemento <head>, marcada por la etiqueta <style>. De esta manera se obtiene el beneficio de separar la información del estilo del código HTML propiamente dicho.

✂ **Una hoja de estilo externa**, es una hoja de estilo que está almacenada en un archivo diferente al archivo donde se almacena el código HTML de la página Web. Esta es la manera de programar más potente, porque separa completamente las reglas de formateo para la página HTML de la estructura básica de la página.

2. Estilos CSS introducidos por el usuario que ve el documento, mediante un archivo CSS especificado mediante las configuraciones del navegador, y que sobrescribe los estilos definidos por el autor en una o varias páginas web.

3. Los estilos marcados "por defecto" por los user agent, para diferentes elementos de un documento HTML, como por ejemplo, los enlaces.

Algunas ventajas de utilizar CSS son:

- ✓ Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.

- ✓ Separación del contenido de la presentación, lo que facilita al creador, diseñador, usuario o dispositivo electrónico que muestre la página, la modificación de la visualización del documento sin alterar el contenido del mismo, sólo modificando algunos parámetros del CSS.
- ✓ Optimización del ancho de banda de la conexión, pues pueden definirse los mismos estilos para muchos elementos con un sólo selector; o porque un mismo archivo CSS puede servir para una multitud de documentos.
- ✓ Mejora en la accesibilidad del documento, pues con el uso del CSS se evitan antiguas prácticas necesarias para el control del diseño.

IV.2.2.3 JavaScript.

JavaScript fue desarrollado originalmente por Brendan Eich de Netscape con el nombre de *Mocha*, el cual fue renombrado posteriormente a *LiveScript*, para finalmente quedar como JavaScript. El cambio de nombre coincidió aproximadamente con el momento en que Netscape agregó soporte para la tecnología Java en su navegador web Netscape Navigator en la versión 2.002 en diciembre de 1995. La denominación produjo confusión, dando la impresión de que el lenguaje es una prolongación de Java, y se ha caracterizado por muchos como una estrategia de mercadotecnia de Netscape para obtener prestigio e innovar en lo que eran los nuevos lenguajes de programación web.

JavaScript es un lenguaje de programación interpretado que se define como orientado a objetos, basado en prototipos, imperativo y dinámico. Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio mayoritariamente en widgets es también significativo.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

IV.2.2.4 JQuery.

jQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

jQuery es software libre y de código abierto permitiendo su uso en proyectos libres y privativos. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio. Se considera que jQuery está diseñado para cambiar la forma en que se escribe Javascript.

IV.2.2.5 AJAX

AJAX, acrónimo de “Asynchronous JavaScript And XML” (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o “RIA” (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dado que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

IV.2.2.6 PHP.

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; sin embargo la implementación principal de PHP es producida ahora por “The PHP Group” y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la “PHP License”, la “Free Software Foundation” considera esta licencia como software libre.

PHP es un lenguaje "Open Source" interpretado de alto nivel, especialmente pensado para desarrollos web y el cual puede ser incrustado en páginas HTML. La mayoría de su sintaxis es similar a C, Java y Perl y es fácil de aprender. La meta de este lenguaje es permitir escribir a los creadores de páginas web, páginas dinámicas de una manera rápida y fácil, aunque se pueda hacer mucho más con PHP.

Todo el código PHP embebido en una página o documento HTML será ejecutado en el servidor (servidor web) antes de que se envíe la página al usuario que la requiera, de esta manera, se pueden generar estructuras de programación de alto nivel que requieran de una lógica más compleja.

Algunas compañías y organizaciones hacen uso de PHP a través de un “framework” de desarrollo, que integra un modelo o patrón de abstracción para el desarrollo de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de negocio en tres componentes distintos.

La mayoría de los “frameworks” de desarrollo para PHP incorporan el modelo MVC (modelo-vista-controlador) el cual será descrito más adelante a detalle.

Lo que distingue a PHP de algo como Javascript del lado del cliente es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente

recibirá el resultado de ejecutar el script, aunque no se sabría el código subyacente que era. El servidor web puede ser incluso configurado para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene.

Debido a las siguientes características es que se consideró el lenguaje PHP para ser utilizado en el sistema:

- ✓ Está orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- ✓ Es considerado un lenguaje fácil de aprender, ya que en su desarrollo se simplificaron distintas especificaciones, como es el caso de la definición de las variables primitivas, ejemplo que se hace evidente en el uso de php arrays.
- ✓ El código fuente escrito en PHP es invisible al navegador web y al cliente, ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- ✓ Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- ✓ Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones).
- ✓ Posee una amplia documentación en su sitio web 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 aplicar técnicas de programación orientada a objetos. Incluso aplicaciones como Zend framework, empresa que desarrolla PHP, están totalmente desarrolladas mediante esta metodología.
- ✓ No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- ✓ Tiene manejo de excepciones desde PHP5.

- ✓ Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar, aun haciéndolo, el programador puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador (MVC), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes.
- ✓ Debido a su flexibilidad ha tenido una gran acogida como lenguaje base para las aplicaciones WEB de manejo de contenido, y es su uso principal.

PHP Funciona de la siguiente manera: El usuario escribe una URL en su navegador y éste envía una petición HTTP al servidor web. El servidor lee la página PHP la interpreta y envía al navegador del usuario un documento compilado en formato HTML tal como lo muestra la siguiente figura:



Figura IV.2.2.6 Función general de PHP.

PHP es utilizado en millones de sitios web, entre los más destacados se encuentran Wikipedia.org, Facebook.com y Wordpress.com.

Como ya se ha mencionado anteriormente PHP tiene la capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad donde destaca su conectividad con MySQL y PostgreSQL, lo que nos servirá para conectarnos a una base de datos para realizar las operaciones necesarias.

IV.3 WEB 2.0.

En los últimos años la Web ha crecido y ha pasado de ser un grupo de herramientas de trabajo para los científicos del CERN a convertirse en un espacio global de información. En la actualidad está tanto volviendo a sus raíces como herramienta de lectura y/o escritura como entrando en una fase más social y participativa. Estas tendencias han generado la sensación de que la Web está entrando en una “segunda fase”, una versión nueva y “mejorada”: la Web 2.0.

IV.3.1 Historia, Evolución y Características de la WEB 2.0.

Tim Berners-Lee y Robert Cailliau crearon la web alrededor de 1990, durante estas dos últimas décadas ha sufrido una evolución extraordinaria y asombrosa, apareciendo en 2004 el concepto de Web 2.0 fruto de esta evolución de la tecnología.

Antes de la llegada de las tecnologías de la Web 2.0 se utilizaban páginas estáticas programadas en HTML que no eran actualizadas frecuentemente. El éxito de las “.com” dependía de webs más dinámicas donde los sistemas de gestión de contenidos servían páginas HTML dinámicas creadas al vuelo desde una base de datos actualizada. En ambos sentidos, el conseguir visitas y la estética visual eran considerados como factores.

El término Web 2.0 comprende aquellos sitios web que facilitan el compartir información, la interoperabilidad, el diseño centrado en el usuario y la colaboración en la World Wide Web. Un sitio Web 2.0 permite a los usuarios interactuar y colaborar entre sí como creadores de contenido generado por usuarios en una comunidad virtual, a diferencia de sitios web estáticos donde los usuarios se limitan a la observación pasiva de los contenidos que se han creado para ellos.

Un número de nuevos servicios y aplicaciones basados en la Red, que en alguna medida se están utilizando en educación, demuestran su fundamentación en el concepto de Web 2.0. No son realmente programas como tales, sino servicios o

- URLs sencillas con significado semántico.
- Soporte para postear en un blog.
- JCC y APIs REST o XML.
- JSON.
- Algunos aspectos de redes sociales.
- Mashup (aplicación web híbrida).

✧ **Generales:**

- El sitio debe estar listo para la entrada de cualquier persona.
- La información debe poderse introducir y extraer fácilmente
- Los usuarios deberían controlar su propia información.
- Basada exclusivamente en la Web: los sitios Web 2.0 con más éxito pueden ser utilizados enteramente desde un navegador.
- La existencia de links es requisito imprescindible.

La Web 2.0 ha originado la democratización de los medios haciendo que cualquiera tenga las mismas posibilidades de publicar noticias que un periódico tradicional. Grupos de personas crean blogs que al día de hoy reciben más visitas que las versiones online de muchos periódicos. La Web 2.0 ha reducido considerablemente los costes de difusión de la información. Al día de hoy podemos tener gratuitamente nuestra propia emisora de radio online, nuestro periódico online, nuestro canal de vídeos, etc. Al aumentar la producción de información aumenta la segmentación de la misma, lo que equivale a que los usuarios puedan acceder a contenidos que tradicionalmente no se publican en los medios convencionales.

Debido a las grandes necesidades de programación que requiere la web 2.0 se desarrollaron “frameworks” o marcos de aplicación o conjunto de bibliotecas orientadas a la reutilización a muy gran escala de componentes software para el desarrollo rápido de aplicaciones facilitando el desarrollo de sistemas, sitios y paginas en la plataforma web 2.0.

IV.3.2 Frameworks para Aplicaciones WEB

La palabra inglesa "framework" o marco de trabajo define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

En el desarrollo de *software*, un framework o infraestructura digital, es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de *software* concretos, que puede servir de base para la organización y desarrollo de *software*. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Son diseñados con la intención de facilitar el desarrollo de *software*, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de *software* que tratando con los tediosos detalles de bajo nivel de proveer un sistema funcional. Por ejemplo, un equipo que usa Apache Struts para desarrollar un sitio web de un banco, puede enfocarse en cómo los retiros de ahorros van a funcionar en lugar de preocuparse de cómo se controla la navegación entre las páginas en una forma libre de errores. Sin embargo, hay quejas comunes acerca de que el uso de frameworks añade código innecesario y que la preponderancia de frameworks competitivos y complementarios significa que el tiempo que se pasaba programando y diseñando ahora se gasta en aprender a usar los frameworks aunque el aprendizaje solo ocurre una sola vez.

Un framework para aplicaciones web es un framework diseñado para apoyar el desarrollo de sitios web dinámicos, aplicaciones web y servicios web. Este tipo de frameworks intenta aliviar el exceso de carga asociado con actividades comunes usadas en desarrollos web. Por ejemplo, muchos *framework* proporcionan bibliotecas

para acceder a bases de datos, estructuras para plantillas y gestión de sesiones, y con frecuencia facilitan la reutilización de código.

En general, con el término framework, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.

Ya que consideramos con anterioridad el lenguaje de programación PHP como base para la generación del sistema; a continuación se describen las características de algunos de los Frameworks más completos para PHP.

IV.3.3 Frameworks para PHP.

Cuando tenemos que desarrollar un proyecto en PHP, muchas veces empezamos escribiendo todo el código desde cero, lo que no es muy práctico. Para ello tenemos disponibles los llamados frameworks para PHP, que nos hará más óptimo el trabajo.

Los llamados frameworks para PHP, son un conjunto de archivos PHP que vienen preparados con toda la estructura necesaria para desarrollar varios tipos de proyectos. En la tabla IV.3.3 se muestran los frameworks que existen para el lenguaje PHP donde [23]:

- **MVC:** Indica si el marco viene con soporte incorporado para una configuración de Modelo-Vista-Controlador.
- **Multiple DB:** Indica si el marco es compatible con múltiples bases de datos sin tener que cambiar nada.
- **ORM:** Indica si el marco apoya un mapeador objeto-record, por lo general una implementación de ActiveRecord.
- **DB Objects:** Indica si el marco incluye otros objetos de base de datos, como un TableGateway.
- **Plantillas:** Indica si el marco tiene un motor de plantillas incorporado.

- **Caching:** Indica si el marco incluye un objeto de almacenamiento en caché o alguna manera otra forma de almacenamiento en caché.
- **Validación:** Indica si el marco tiene una validación incorporado o componente de filtrado.
- **Ajax:** Indica si el marco viene con soporte incorporado para Ajax.
- **Módulo de autenticación:** Indica si el marco tiene un módulo incorporado para el manejo de la autenticación de usuario.
- **Módulos:** Indica si el marco tiene otros módulos, como un feed RSS, módulo analizador PDF o cualquier otra cosa (útil).
- **EDP:** Indica si el marco contiene programación de manejo de eventos

Tabla IV.3.3 Comparativa de los Frameworks para PHP.

PHP Framework	PHP4	PHP5	MVC	Multiple DB's	ORM	DB Objects	Templates	Caching	Validation	Ajax	Auth Module	Modules	EDP
Akelos 	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
ash.MVC 	-	✓	✓	-	-	✓	✓	-	✓	-	✓	✓	-
CakePHP 	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	-
CodeIgniter 	✓	✓	✓	✓	-	✓	✓	✓	✓	-	-	-	-
DIY 	-	✓	✓	-	✓	✓	✓	✓	-	✓	-	-	-
eZ Components 	-	✓	-	✓	-	✓	✓	✓	✓	-	-	-	-
Fusebox 	✓	✓	✓	✓	-	-	-	✓	-	✓	-	✓	-
PHP on TRAX 	-	✓	✓	✓	✓	✓	-	-	✓	✓	-	✓	-
PHPDevShell 	-	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	-
PhpOpenbiz 	-	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	-	-
Prado 	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
QPHP 	✓	✓	✓	✓	-	✓	✓	-	✓	✓	✓	✓	✓
Seagull 	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
Symfony 	-	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	-
WACT 	✓	✓	✓	✓	-	✓	✓	-	✓	-	-	✓	-
WASP 	-	✓	✓	-	-	✓	✓	-	✓	✓	✓	✓	-
Yii 	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Zend 	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
ZooP 	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	-	-

Una vez que conocemos todas las características de los diferentes framework que se pueden utilizar con PHP podemos evaluarlos y optar por seleccionar uno de acuerdo a las siguientes consideraciones:

- ❖ **Soporte de bases de datos:** Este es un aspecto muy importante que no debe ser dejado de lado. Por ejemplo, si usas Oracle o SQLite puedes elegir el framework Kohana. En caso que estés buscando un framework con soporte para MySQL, puedes elegir CodeIgniter. También asegúrate de que el servidor de bases de datos soporte este tipo de bases de datos.
- ❖ **Soporte de la comunidad:** la comunidad es una de las cosas más importantes que debe ser tomada en cuenta. El soporte que proporciona la comunidad te ayuda en esos momentos difíciles cuando surgen problemas con el framework. Visita el sitio web de la comunidad para asegurarte que te darán consejos si lo requieres.
- ❖ **Documentación:** debes preocuparte si ves un framework sin instrucciones o guías. El framework debe tener documentación actualizada que responda a los temas problemáticos.
- ❖ **Arquitectura Modelo – Vista – Controlador** – asegúrate que el framework tiene una arquitectura MVC (Modelo Vista Controlador). Los frameworks de calidad entregan a sus usuarios plug-ins, bibliotecas, extensiones y mucho más.

IV.3.3.1 Modelo Vista Controlador

Una de las consideraciones más relevantes es el Modelo Vista Controlador (MVC) ya que es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello el MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario.

Este patrón de diseño se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento.

De manera genérica, los componentes de MVC se podrían definir como:

- ✂ El **Modelo**: Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación. Envía a la vista aquella parte de la información que en cada momento se le solicita para que sea mostrada. Las peticiones de acceso o manipulación de información llegan al modelo a través del controlador.
- ✂ El **Controlador**: Responde a eventos usualmente acciones del usuario e invoca peticiones al modelo cuando se hace alguna solicitud sobre la información por ejemplo, editar un documento o un registro en una base de datos. También puede enviar comandos a su vista asociada si se solicita un cambio en la forma en que se presenta de modelo por ejemplo, desplazamiento o scroll por un documento o por los diferentes registros de una base de datos, por tanto se podría decir que el controlador hace de intermediario entre la vista y el modelo.
- ✂ La **Vista**: Presenta el modelo (información y *lógica de negocio*) en un formato adecuado para interactuar usualmente la interfaz de usuario por tanto requiere de dicho modelo la información que debe representar como salida.

Los primeros frameworks basados en MVC para desarrollo web planteaban un enfoque de cliente ligero en el que casi todas las funciones, tanto de la vista, el modelo y el controlador recaían en el servidor. En este enfoque, el cliente manda la petición de cualquier hipervínculo o formulario al controlador y después recibe de la vista una página completa y actualizada usando el modelo correspondiente tal como se muestra en la figura IV.3.3.1; tanto el modelo como el controlador (y buena parte de la vista) están completamente alojados en el servidor.

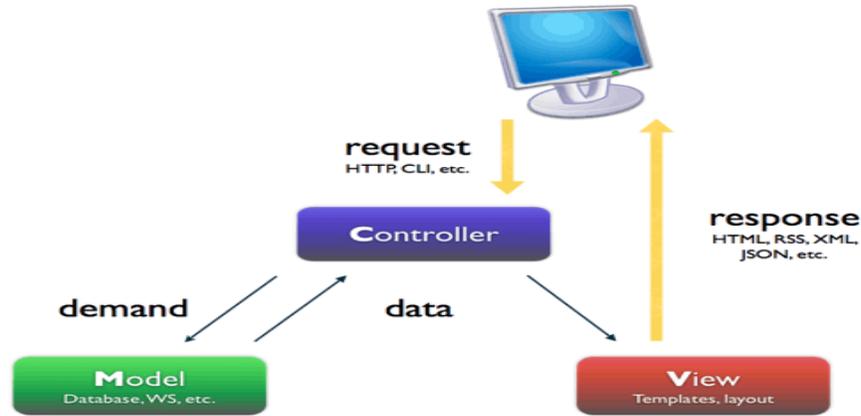


Figura IV.3.3.1 Interacción entre una petición y su respuesta usando el MVC.

Se han desarrollado multitud de frameworks que implementan este patrón; estos frameworks se diferencian básicamente en la interpretación de como las funciones MVC se dividen entre cliente y servidor. Las nuevas tecnologías web contienen herramientas como JavaScriptMVC, jQuery y AJAX que permiten que ciertos componentes MVC se ejecuten parcial o totalmente en el cliente.

De acuerdo con la tabla y las consideraciones antes mencionadas podemos decir que el mejor framework a utilizar para el sistema está entre los siguientes:

IV.3.3.2 ZEND

Zend Framework es un framework PHP orientado a objetos con una fuerte implementación del patrón MVC, una abstracción de datos muy simple de usar y un rendimiento muy estable, siendo uno de los frameworks más robustos de la actualidad. Además, al ser un framework de código abierto, evitaremos posibles problemas con licencias como nos podría ocurrir con algún framework restrictivo o de pago.

Zend Framework se basa en la simplicidad, las mejores prácticas orientadas a objetos, las licencias amigables corporativas y un código base ágil rigurosamente. Zend Framework se centra en la construcción más segura, confiable y moderno aplicaciones Web 2.0 y Web servicios y consumo API ampliamente disponible de proveedores

líderes como Google, Amazon, Yahoo!, Flickr, así como los proveedores de API y catalogadores como Strikelron y ProgrammableWeb. La configuración requiere de muchos requerimientos y configuraciones lo que lo hace confuso y tedioso.

IV.3.3.3 Symfony

Un framework PHP construido enteramente para PHP 5, por lo que no es compatible, pero al mismo tiempo ofrece un rendimiento excelente en la última versión de PHP. Con Symfony el programador se podrá centrar solo realmente en lo que es la programación, obviando tareas comunes y centrándose en el proyecto entre manos. Su objetivo es acelerar la creación y el mantenimiento de las aplicaciones web, y para reemplazar las tareas de codificación repetitivas. Symfony viene con adicionales herramientas que ayudan a probar, depurar y documentar su proyecto.

El pequeño número de requisitos previos hacen Symfony fácil de instalar en cualquier configuración; sólo tienes Unix o Windows con un servidor web y PHP 5 instalado es compatible con casi todos los sistemas de bases de datos por lo que los beneficios del marco no vienen a costa de un aumento de los costos de alojamiento, dependiendo solamente de la versión 5 de PHP.

IV.3.3.4 CodeIgniter

Framework increíblemente ligero que usa el diseño MVC y contiene una muy buena estructuración de sus librerías. Usarlo es tan fácil como descomprimir su contenido en una carpeta en el servidor, sólo con eso ya funcionará. Es uno de los frameworks más indicados para las personas que quieran iniciarse en el uso de los mismos lo que lo hacen bueno para aprender y no para aplicar.

CodeIgniter se basa libremente en el popular Modelo-Vista-Controlador patrón de desarrollo. Aunque las clases de vista y el controlador son una parte necesaria de desarrollarse en CodeIgniter, los modelos son opcionales.

IV.3.3.5 Seagull

Es un marco de programación orientada a objetos madura para la construcción de red, línea de comandos y aplicaciones GUI. Licenciado bajo BSD, el proyecto permite a PHP desarrolladores integrar fácilmente y gestionar los recursos de código y crear aplicaciones complejas de forma rápida. Seagull es un Framework compatible con PHP 4 y PHP 5 con el que podremos realizar aplicaciones y páginas web de una forma muy sencilla y visual, ya que al mismo tiempo es un CMS que cuenta con interfaz gráfica, lo cual acelerará bastante el trabajo en algunas ocasiones.

Muchas aplicaciones PHP ya están perfectamente integradas en el proyecto, así como los diversos motores de plantillas, herramientas de prueba y el código de biblioteca administrada. Si usted es un principiante, el marco ofrece una serie de aplicaciones de ejemplo que puede ser personalizado y ampliado para adaptarse a sus necesidades.

IV.3.4 Yii Framework

Yii es un framework PHP de aplicaciones web de código abierto, orientado a objetos, basado en componentes MVC. El proyecto Yii comenzó en enero de 2008, con el fin de solucionar algunos inconvenientes del framework PRADO . Por ejemplo, en sus primeras versiones PRADO fue lento al manejar páginas complejas, tenía una curva de aprendizaje y muchos controles eran difíciles de personalizar, mientras Yii es mucho más eficiente en ese momento.

Yii tiene un repositorio de extensiones que consta de componentes que el usuario ha contribuido. También había una biblioteca de extensión oficial llamada *Zi*, que después se fusionó con el marco central. Está incluido en cada lanzamiento desde Yii versión 1.1.0, e incluye comportamientos y widgets adicionales, tales como las redes y la interfaz de usuario jQuery [24].

Yii viene con una colección de documentos oficiales, como un tutorial para desarrollar una aplicación de blog sencillo, una guía que da la descripción de cada función y una referencia de clase que le da todos los detalles de propiedades, métodos y eventos. También hay una documentación que es aportada por los usuarios, la mayoría de los cuales está disponible en el wiki en la página oficial.

Yii es liberado bajo la licencia BSD New (licencia para 3 cláusulas). Esto significa que se pueden utilizar de forma gratuita para desarrollar ya sea de código abierto o propietario aplicaciones web.

El texto que figura en la documentación oficial de Yii tiene licencia para el público bajo la licencia de documentación libre GNU (GFDL). En general, el contenido de la documentación Yii puede ser copiado, modificado y redistribuido siempre y cuando la nueva versión conceda las mismas libertades a otros y reconoce los autores del artículo de la documentación Yii utilizados [25].

Yii implementa el patrón de diseño MVC el cuál es ideal para la programación Web. MVC apunta a separar la lógica de negocio de la interfaz de usuario, así los programadores pueden fácilmente cambiar cada parte sin afectar unas a otras. Además de MVC, Yii también introduce un controlador frontal, llamado aplicación, el cual representa el contexto de ejecución de las peticiones en proceso. La aplicación resuelve las peticiones de los usuarios y las despacha al controlador apropiado para futuros manejos.

La figura IV.3.4.1 muestra la estructura estática de una aplicación Yii [26] la cual inicia con una petición al “index.php” de la aplicación en caso de ser necesario usa algunos componentes extras, la aplicación utiliza el controlador para interactuar con el modelo y la vista, en caso de ser necesario usa uno o varios “widget” para generar la vista y si no la genera directamente con las instrucciones del controlador.

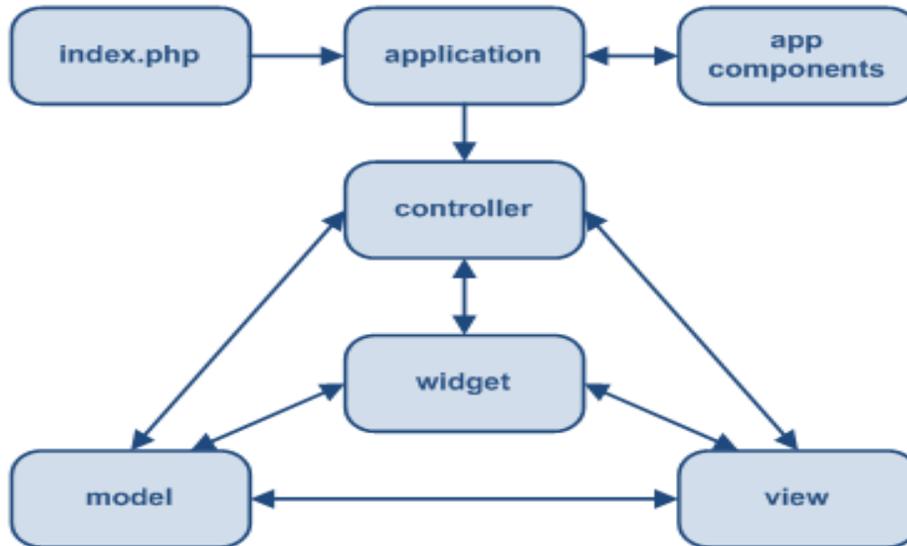


Figura IV.3.4.1 Estructura estática de una aplicación Yii.

Los siguientes pasos muestran el típico flujo de trabajo de una aplicación Yii de la figura IV.3.4.1 cuando se está manejando la petición de un usuario [27]:

1. Un usuario hace una petición a una URL y el servidor web maneja la petición ejecutando el bootstrap script index.php.
2. El bootstrap script crea una instancia de la aplicación y la ejecuta.
3. La aplicación obtiene la información detallada de la petición de los usuarios desde un componente de la aplicación llamado petición (request).
4. La aplicación determina la petición del controlador y actúa con la ayuda de un componente de la aplicación llamado manejador de url's (urlManager). Para este ejemplo, el controlador es Post que hace referencia a la clase PostController; y la acción es mostrar (show) el cuál su significado es determinado por el controlador.
5. La aplicación crea una instancia de la petición del controlador para más adelante manejar la petición del usuario. El controlador determina que la acción mostrar (show) se refiere a un método llamado actionShow en la clase del controlador.

Entonces crea y ejecuta filtros (por ej: controles de acceso) asociados a esta acción. La acción es ejecutada si los filtros lo permiten.

6. La acción lee un modelo Post el cuál si ID es 1 en la base de datos.
7. La acción asocia una vista llamada mostrar (show) con el modelo Post.
8. La vista lee y muestra los atributos del modelo Post.
9. La vista ejecuta algunos artilugios.
10. Los resultados de la vista son embebidos en un diseño (layout).
11. La acción completa la prestación de la vista y muestra el resultado al usuario.

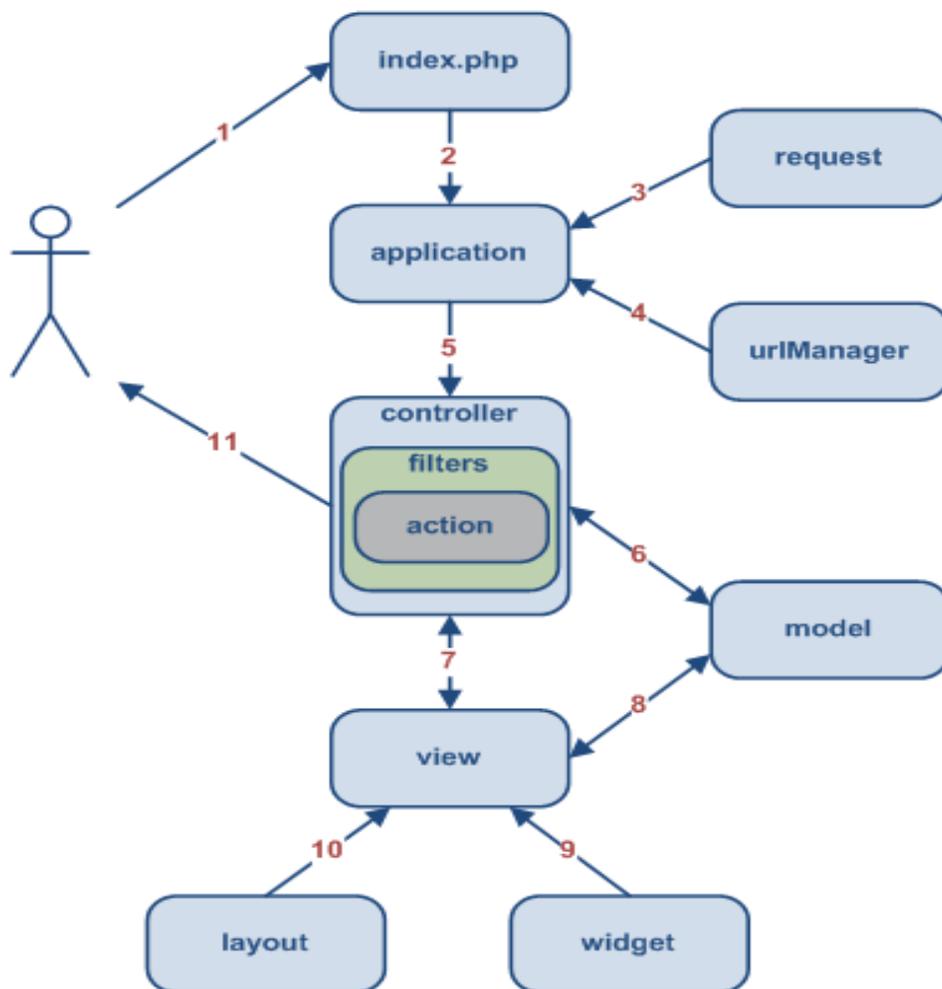


Figura IV.3.4.1 Flujo de trabajo de una aplicación Yii.

Las características principales del framework Yii incluyen:

- ✓ Patrón de diseño Modelo Vista Controlador (MVC).
- ✓ Database Access Objects (DAO), query builder, Active Record y migración. de base de datos.
- ✓ Integración con jQuery.
- ✓ Entradas de Formulario y validación.
- ✓ Widgets de Ajax como autocompletado de campos de texto y demás.
- ✓ Soporte de Autenticación incorporado. Además soporta autorización vía role-based access control (RBAC) jerárquico.
- ✓ Personalización de aspectos y temas.
- ✓ Generación compleja automática de WSDL, especificaciones y administración de peticiones Web service.
- ✓ Internacionalización y localización (I18N and L10N) y que soporta traducciones, formato de fecha y hora, formato de números, y localización de la vista.
- ✓ Esquema de “caching” por capas. Soporta el cache de datos, cache de páginas, cache por fragmentos y contenido dinámico. El medio de almacenamiento del cache puede ser cambiado.
- ✓ El manejo de errores y login: Los errores son manejados y personalizados, y los log de mensajes pueden ser categorizados, filtrados y movidos a diferentes destinos.
- ✓ Las medidas de seguridad incluyen la prevención cross-site scripting (XSS), prevención cross-site request forgery (CSRF), prevención de la manipulación de cookies, etc.
- ✓ Herramientas para pruebas unitarias y funcionales basados en PHPUnit y Selenium.
- ✓ Generación automática de código para el esqueleto de la aplicación, aplicaciones CRUD, etc.
- ✓ Generación de código por componentes de Yii y la herramienta por línea de comandos cumple con los estándares de XHTML.

- ✓ Cuidadosamente diseñado para trabajar bien con código de terceros. Por ejemplo, es posible usar el código de PHP o Zend Framework en una aplicación Yii.

Yii es un framework PHP de alto rendimiento basado en componentes para el desarrollo de aplicaciones web a gran escala. Yii está escrito en estricta programación orientada a objetos y viene con referencia de clase completa y completos tutoriales. Desde MVC, DAO / ActiveRecord, widgets, almacenamiento en caché, RBAC jerárquica, Web servicios, tematización, localización (I18N y L10N), Yii ofrece casi todas las características que necesita el desarrollo de aplicaciones Web 2.0 de hoy y todos estos vienen sin incurrir en muchos gastos.

Como cuestión de hecho y relevancia por sus muy buenas características Yii es uno de los frameworks PHP más eficientes de todos con una configuración y utilización sencilla, por tal motivo, consideraremos al framework YII como el más idóneo para facilitar la programación del Sistema web Inteligente para la gestión del ahorro de energía en refrigeración e iluminación en hoteles de México.

IV.4 Servidor WEB

Iniciaremos por definir lo que es un servidor, como su nombre lo indica, son los dispositivos de red que brindan un servicio a otros dispositivos, a los cuales se les conoce como clientes. En general quien realiza esta tarea es un software especializado, pero comúnmente se conoce como servidor al equipo físico donde se ejecuta, el cual es el centro de la infraestructura de la red.

Los servidores se pueden clasificar en distintos tipos de los cuales podemos mencionar:

- ✦ **Los Servidores de archivos:** su función es la de centralizar la información de todos los usuarios de la red almacenando archivos de usuario, los cuales acceden remotamente a los mismos.
- ✦ **Los servidores de aplicaciones o servidor WEB** en donde el servidor permite el procesamiento de datos de una aplicación de cliente.
- ✦ **Los servidores de correo** en donde se mueven y almacenan el correo electrónico a través de las redes corporativas (vía LANs y WANs) y a través de Internet.
- ✦ **Los servidores de base de datos** los cuales poseen un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

Para nuestros propósitos usaremos un servidor de aplicaciones, servidor web o servidor HTTP ya que al ser un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales con el cliente generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente resulta ser el servidor apto para el sistema.

El código recibido por el cliente suele ser compilado y ejecutado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se utiliza el protocolo HTTP para estas comunicaciones,

perteneciente a la capa de aplicación del modelo OSI. Básicamente un servidor WEB consta de un intérprete HTTP el cual se mantiene a la espera de peticiones de clientes y le responde con el contenido según sea solicitado. El cliente, una vez recibido el código, lo interpreta y lo exhibe en pantalla.

Además los servidores pueden disponer de un intérprete de otros lenguajes de programación que ejecutan código embebido dentro del código HTML de las páginas que contiene el sitio antes de enviar el resultado al cliente como Ajax, JQuery, etc.

IV.4.1 Arquitectura Cliente Servidor

La arquitectura cliente-servidor (C/S) es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa.

En la arquitectura C/S el remitente de una solicitud es conocido como cliente y sus características son:

- Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación.
- Espera y recibe las respuestas del servidor.

- Por lo general, puede conectarse a varios servidores a la vez.
- Normalmente interactúa directamente con los usuarios finales mediante una interfaz gráfica de usuario.

Al receptor de la solicitud enviada por el cliente se conoce como servidor y sus características son:

- Al iniciarse esperan a que lleguen las solicitudes de los clientes, desempeñan entonces un papel pasivo en la comunicación.
- Tras la recepción de una solicitud, la procesan y luego envían la respuesta al cliente.
- Por lo general, aceptan conexiones desde un gran número de clientes (en ciertos casos el número máximo de peticiones puede estar limitado).
- No es frecuente que interactúen directamente con los usuarios finales.

La red cliente-servidor es aquella red de comunicaciones en la que todos los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes cada vez que estos son solicitados. Esto significa que todas las gestiones que se realizan se concentran en el servidor, de manera que en él se disponen los requerimientos provenientes de los clientes que tienen prioridad, los archivos que son de uso público y los que son de uso restringido, los archivos que son de sólo lectura y los que, por el contrario, pueden ser modificados, etc.

IV.4.1.1 Aplicación del lado del Servidor.

Una aplicación del lado del servidor es cualquier programa o conjunto de instrucciones diseñadas con la finalidad de que un Servidor Web las procese para realizar alguna acción. Las aplicaciones del lado del servidor están escritas mediante algún lenguaje de programación, entre los que destacan PHP, ASP.NET, Perl, Python y Ruby donde el 75% de las aplicaciones del lado del servidor están escritas en PHP, seguido de ASP y las demás opciones usadas de forma alternativa y muy casual.

La programación del lado del servidor se utiliza generalmente para proporcionar una interfaz y para limitar el acceso del cliente a la base de datos u otras fuentes de datos. Estos scripts pueden reunir las características del cliente para su uso en la personalización de la respuesta en base a las características, los requisitos de los usuarios, los derechos de acceso, etc.

Cuando el servidor de datos sirve de una manera comúnmente utilizada, por ejemplo de acuerdo con el protocolo HTTP o FTP, los usuarios pueden tener su elección de un número de programas de cliente, actualmente, la mayoría de los navegadores web modernos pueden solicitar y recibir datos a través de los protocolos. En el caso de aplicaciones más especializadas, los programadores pueden escribir su propio servidor, cliente, y el protocolo de comunicaciones, que sólo se puede utilizar uno con el otro.

Un servidor web tiene la función de procesar los scripts del lado del servidor para dar una salida en HTML y otros lenguajes del lado del cliente al Navegador Web del cliente. La información a procesar podrá ser cedida por el cliente al script mediante cualquier aplicación en el entorno del Navegador. Para ello pueden utilizarse formularios web, enlaces con los valores implícitos en la cadena o cualquier otro método.

Las aplicaciones del lado del cliente reducen la cantidad de errores o problemas de compatibilidad ya que el código se ejecuta en un servidor usando un lenguaje y software. La programación en el servidor también se puede cifrar cuando los usuarios envían variables de formulario, la protección de los usuarios frente a cualquier intento de intrusión.

IV.4.1.2 Aplicación del lado del Cliente.

En una aplicación informática lado del cliente o “client-side” en inglés es el ámbito en el cual un cliente ejecuta sus operaciones en una relación cliente-servidor dentro de una red informática. Un cliente se entiende como una aplicación por ejemplo

un navegador web, que se ejecuta en la computadora local del usuario y se conecta a un servidor cuando es necesario.

Las operaciones principales se realizan mayoritariamente en el lado del cliente, ya que la aplicación necesita de información o funcionalidades que únicamente están disponible en este ámbito y no en el del servidor, ya sea porque el usuario necesita observar y proporcionar nuevos datos, o porque el servidor carece de capacidad de procesamiento para ejecutar todas operaciones relativas a los clientes a los que sirve. Además, si se maximiza el número de operaciones que ocurren en el lado del cliente, sin interactuar con el servidor, tomarán menos tiempo, se usará menos ancho de banda y se aumentará la seguridad de la aplicación.

Cuando el servidor envía información en un protocolo habitual, por ejemplo HTTP o FTP, los usuarios pueden escoger entre varios programas “cliente” diferentes para procesarla, ya sea un navegador web moderno que sea capaz de enviar y recibir información usando los dos protocolos, o una aplicación específica, como un cliente FTP.

Los siguientes lenguajes de programación implementados al sistema se ejecutan del lado del cliente:

- **JavaScript:** Esencialmente el lenguaje de script del lado del cliente. El resto son variantes del mismo.
- **Ajax:** Utilizado para la implementación de JavaScript y XML sin afectar a la propia página web.
- **Jquery:** utilizado como una biblioteca de JavaScript en el que se puede utilizar fácilmente secuencias de comandos comúnmente utilizados sin especificaciones.
- **ActionScript:** Utilizado para crear aplicaciones web interactivas animadas de Adobe Flash Player utilizando Adobe Flash Pro.

Los programas que se ejecutan en el ordenador del usuario sin enviar o recibir información en una red no se consideran clientes, y las operaciones realizadas en estas aplicaciones no se consideran operaciones en el lado del cliente.

IV.4.2 Envío y Recepción de Datos (POST y GET).

IV.4.2.1 método GET

El concepto GET es obtener información del servidor. Traer datos que están en el servidor, ya sea en un archivo o base de datos, al cliente. Independientemente de que para eso tengamos que enviar (request) algún dato que será procesado para luego devolver la respuesta (response) que esperamos, como por ejemplo un identificador para obtener una noticia de la base de datos.

Las llamadas GET pueden ser cacheadas por el historial del navegador, indexadas por buscadores, agregar los enlaces a nuestros favoritos o hasta pasar una url completa a otra persona para que directamente ingrese a esa página. Usualmente para los métodos get se utilizan hipervínculos, enlaces o links, formularios web, barra de direcciones y scripts.

Al realizar una petición GET se realizan los siguientes pasos:

1. Se produce un socket con un servidor dado en dirección IP mediante TCP. Por lo general las direcciones que el navegador posee inicialmente son direcciones DNS (Domain Name System) o direcciones alfanuméricas que deberá convertir a direcciones numéricas.
2. Si la dirección dada es DNS y no existe una regla en la base de datos DNS, el Host Resolver Request solicita al servidor DNS la o las direcciones IPs correspondientes. El navegador crea una nueva regla y almacena la dirección IP junto a la dirección DNS en su base de datos de reglas DNS.
3. Una vez almacenada la regla se realiza una petición a la base de datos DNS para recuperar los valores de la regla.

4. Se produce un socket con la dirección IP mediante TCP. La dirección IP puede haberse recuperado.
5. Se crea la petición GET estableciendo la “url”, un “flag”, la “priority” de la petición y el “method” que es implícitamente GET.
6. Se abre y/o se crea una entrada en el http cache
7. Se realiza la petición GET. Se leen las cabeceras HTTP de la http “transaction” y más tarde el cuerpo de la http “transaction”.
8. Se consulta en el caché de disco si existe una entrada en el caché asociada al recurso que se ha solicitado. Los valores son “created” (*true* o *false*) y “key” (la url del recurso).
9. Si la entrada no existe (si el valor de created es *false*) se escriben los datos en el caché.
10. Se concluye la operación y se muestra en pantalla la información.

Generalmente usamos links para ejecutar llamadas GET ya que la idea del link es simplemente “solicitar” una información (página web) al servidor y que sea devuelta como una respuesta.

IV.4.2.2 Método POST

El concepto POST es enviar información desde el cliente para que sea procesada y actualice o agregue información en el servidor, como sería la carga o actualización en sí de una noticia. Cuando enviamos (request) datos a través de un formulario, estos son procesados y luego a través de una redirección por ejemplo devolvemos (response) alguna página con información.

Los datos a enviar al servidor se incluyen en el cuerpo de la misma petición con las cabeceras HTTP asignadas correspondientemente respecto al tipo de petición. Generalmente se asocia con los formularios web en el que los datos suelen ser cifrados para enviarlos de manera segura al servidor.

Los datos que se envían en el cuerpo de la petición POST deben tener algún formato que permita manipularlos en un futuro procesamiento. Por ello la petición debe tener asignada la cabecera Content-Type cuyo valor será la codificación de los datos. De este modo el sistema podrá diferenciar entre variables aisladas, datos binarios, texto plano, o cualquier otro tipo de formato. El formato de una cadena de datos se denomina MIME y es el valor que deberá ser incluido en esta cabecera.

POST te sirve para enviar información que será procesada por el script php para almacenarla en la base de datos y GET sirve más para pasar parámetros entre páginas tales como indicadores de las acciones a realizar, ids de registros, valores de variables sencillos, etc.

IV.4.3 Funcionamiento del Servidor WEB.

El servidor que utilizamos es el servidor HTTP Apache ya que es un servidor web HTTP de código abierto, para plataformas Unix, Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP y la noción de sitio virtual. El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation [29]. Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache es usado principalmente para enviar páginas web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web. Apache es el componente de servidor web en la popular plataforma de aplicaciones LAMP, junto a MySQL y los lenguajes de programación PHP, Perl, Python y Ruby.

Este servidor web es redistribuido como parte de varios paquetes propietarios de software, incluyendo la base de datos Oracle y el IBM WebSphere application server.

Mac OS X integra apache como parte de su propio servidor web y como soporte de su servidor de aplicaciones WebObjects.

Apache es usado para muchas otras tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable. Un ejemplo es al momento de compartir archivos desde una computadora personal hacia Internet.

De forma general podemos decir que el Servidor web se ejecuta en un ordenador manteniéndose a la espera de peticiones por parte de un cliente en este caso un navegador web y que responde a estas peticiones adecuadamente, mediante una página web que se exhibirá en el navegador o mostrando el respectivo mensaje si se detectó algún error.

A modo de ejemplo, al teclear una url en nuestro navegador, éste realiza una petición HTTP al servidor de dicha dirección. El servidor responde al cliente enviando el código HTML de la página; el cliente, una vez recibido el código, lo interpreta y lo exhibe en pantalla. Como vemos con este ejemplo, el cliente es el encargado de interpretar el código HTML, es decir, de mostrar las fuentes, los colores y la disposición de los textos y objetos de la página; el servidor tan sólo se limita a transferir el código de la página sin llevar a cabo ninguna interpretación de la misma.

Además de la transferencia de código HTML, los Servidores web pueden entregar aplicaciones web. Éstas son porciones de código que se ejecutan cuando se realizan ciertas peticiones o respuestas HTTP.

Las aplicaciones de servidor muchas veces suelen ser la mejor opción para realizar aplicaciones web. La razón es que, al ejecutarse ésta en el servidor y no en la máquina del cliente, éste no necesita ninguna capacidad añadida, como sí ocurre en el caso de querer ejecutar aplicaciones javascript o java. Así pues, cualquier cliente dotado de un navegador web básico puede utilizar este tipo de aplicaciones.

IV.5 Base de Datos Orientada a WEB.

IV.5.1 Bases de Datos

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital o electrónico, y por ende se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos.

Las aplicaciones más usuales son para la gestión de empresas e instituciones públicas. También son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental.

IV.5.1.1 Descripción y Características.

Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular. Entre las principales características de los sistemas de base de datos podemos mencionar:

- ❖ Independencia lógica y física de los datos.
- ❖ Redundancia mínima.
- ❖ Acceso concurrente por parte de múltiples usuarios.
- ❖ Integridad de los datos.
- ❖ Consultas complejas optimizadas.
- ❖ Seguridad de acceso y auditoría.
- ❖ Respaldo y recuperación.
- ❖ Acceso a través de lenguajes de programación estándar.

Los Sistemas de Gestión de Base de Datos (en inglés DataBase Management System) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

IV.5.1.2 Ventajas

- ✓ **Control sobre la redundancia de datos:** Los sistemas de ficheros almacenan varias copias de los mismos datos en ficheros distintos.
- ✓ **Consistencia de datos:** Eliminando o controlando las redundancias de datos se reduce en gran medida el riesgo de que haya inconsistencias.
- ✓ **Compartición de datos:** En los sistemas de bases de datos, la base de datos pertenece a la empresa y puede ser compartida por todos los usuarios que estén autorizados.
- ✓ **Mantenimiento de estándares:** Gracias a la integración es más fácil respetar los estándares necesarios, tanto los establecidos a nivel de la empresa como los nacionales e internacionales.
- ✓ **Mejora en la integridad de datos:** La integridad de la base de datos se refiere a la validez y la consistencia de los datos almacenados. Normalmente, la integridad se expresa mediante restricciones o reglas que no se pueden violar.
- ✓ **Mejora en la seguridad:** La seguridad de la base de datos es la protección de la base de datos frente a usuarios no autorizados.
- ✓ **Mejora en la accesibilidad a los datos:** Muchos SGBD proporcionan lenguajes de consultas o generadores de informes que permiten al usuario hacer cualquier tipo de consulta sobre los datos.
- ✓ **Mejora en la productividad:** El SGBD proporciona muchas de las funciones estándar que el programador necesita escribir en un sistema de ficheros
- ✓ **Mejora en el mantenimiento:** En los sistemas de archivos, las descripciones de los datos se encuentran inmersas en los programas de aplicación que los manejan, en caso de mantenimiento, la información no se comprometerá.
- ✓ **Aumento de la concurrencia:** Los SGBD gestionan el acceso concurrente a la base de datos y garantizan que no ocurran problemas de conexión.
- ✓ **Mejora en los servicios de copias de seguridad:** Los SGBD actuales funcionan de modo que se minimiza la cantidad de trabajo perdido cuando se produce un fallo gracias a las copias de seguridad.

IV.5.1.3 Desventajas

- × **Complejidad:** Los SGBD son conjuntos de programas que pueden llegar a ser complejos con una gran funcionalidad. Es preciso comprender muy bien esta funcionalidad para poder realizar un buen uso de ellos.
- × **Coste del equipamiento adicional:** Tanto el SGBD, como la propia base de datos, pueden hacer que sea necesario adquirir más espacio de almacenamiento. Además, para alcanzar las prestaciones deseadas, es posible que sea necesario adquirir una máquina más grande o una máquina que se dedique solamente al SGBD. Todo esto hará que la implantación de un sistema de bases de datos sea más cara.
- × **Vulnerable a los fallos:** El hecho de que todo esté centralizado en el SGBD hace que el sistema sea más vulnerable ante los fallos que puedan producirse. Es por ello que deben tenerse copias de seguridad (Backup).

IV.5.1.4 Modelos

Un modelo de datos es básicamente una "descripción" de algo conocido como contenedor de datos, así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de base de datos; por lo general se refieren a algoritmos, y conceptos matemáticos.

Algunos modelos con frecuencia utilizados en las bases de datos son:

- ❖ **Bases de datos jerárquicas:** En este modelo los datos se organizan en forma de árbol invertido, en donde un *nodo padre* de información puede tener varios *hijos*. El nodo que no tiene padres es llamado *raíz*, y a los nodos que no tienen hijos se los conoce como *hojas*.
- ❖ **Base de datos de red:** Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de *nodo*: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

- ❖ **Bases de datos transaccionales:** Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible.
- ❖ **Bases de datos relacionales:** Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos por Edgar Frank Codd, de los laboratorios IBM, no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas".
En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia, esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.
- ❖ **Bases de datos multidimensionales:** Son bases de datos ideadas para desarrollar aplicaciones muy concretas. Básicamente no se diferencian demasiado de las bases de datos relacionales, la diferencia está más bien a nivel conceptual; en las bases de datos multidimensionales los campos o atributos de una tabla pueden ser de dos tipos, o bien representan dimensiones de la tabla, o bien representan métricas que se desean estudiar.
- ❖ **Bases de datos orientadas a objetos:** Este modelo, bastante reciente, y propio de los modelos informáticos orientados a objetos, trata de almacenar en la base de datos los *objetos* completos como estado y comportamiento.
En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos.
- ❖ **Bases de datos documentales:** Permiten la indexación a texto completo, y en líneas generales realizar búsquedas más potentes. "Tesauro" es un sistema de índices optimizado para este tipo de bases de datos.

- ❖ **Bases de datos deductivas:** Un sistema de base de datos deductiva, es un sistema de base de datos pero con la diferencia de que permite hacer deducciones a través de inferencias. Se basa principalmente en reglas y hechos que son almacenados en la base de datos.

IV.5.1.5 Modelo Relacional.

El modelo relacional para la gestión de una base de datos es un modelo de datos basado en la lógica de predicados y en la teoría de conjuntos. Es el modelo más utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente.

En este modelo todos los datos son almacenados en relaciones, y como cada relación es un conjunto de datos, el orden en el que éstos se almacenen no tiene relevancia. Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar por un usuario no experto. La información puede ser recuperada o almacenada por medio de consultas que ofrecen una amplia flexibilidad y poder para administrar la información.

Este modelo considera la base de datos como una colección de relaciones. De manera simple, una relación representa una tabla que no es más que un conjunto de filas, cada fila es un conjunto de campos y cada campo representa un valor que interpretado describe el mundo real. Cada fila también se puede denominar tupla o registro y a cada columna también se le puede llamar campo o atributo, como por ejemplo, el nombre de campo, tipo y longitud; a este campo generalmente se le denomina ID, identificador o clave

IV.5.1.5.1 Ventajas

- ✓ Garantiza herramientas para evitar la duplicidad de registros, a través de campos claves o llaves.
- ✓ Garantiza la integridad referencial: Así al eliminar un registro elimina todos los registros relacionados dependientes.
- ✓ Favorece la normalización por ser más comprensible y aplicable.

Existe software exclusivamente dedicado a tratar con bases de datos relacionales. Este software se conoce como SGBD (Sistema de Gestión de Base de Datos relacional) o RDBMS (del inglés *Relational Database Management System*).

Entre los gestores o manejadores actuales más populares encontramos MySQL, PostgreSQL, Oracle, DB2, INFORMIX, Interbase, FireBird, Sybase y Microsoft SQL Server.

IV.5.2 Modelo Entidad Relación

Para poder utilizar de una forma adecuada el modelo relacional en una base de datos se utiliza un diagrama o modelo entidad-relación (ER) debido a que es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades.

Formalmente, los diagramas ER son un lenguaje gráfico para describir conceptos. Informalmente, son simples dibujos o gráficos que describen información que trata un sistema de información y el software que lo automatiza.

El modelo de datos entidad-relación está basado en una percepción del mundo real que consta de tres elementos básicos:

1. **Entidad:** Representa una “cosa” u "objeto" del mundo real con existencia independiente, es decir, se diferencia unívocamente de otro objeto o cosa, incluso siendo del mismo tipo, o una misma entidad.

Una entidad puede ser un objeto con existencia física como: una persona, un animal, una casa, etc. denominada entidad concreta; o un objeto con existencia conceptual como: un puesto de trabajo, una asignatura de clases, un nombre, etc. denominada entidad abstracta.

Una entidad está descrita y se representa por sus características o atributos. Por ejemplo, la entidad Persona tiene las características: Nombre, Apellido, Género,

Estatura, Peso, Fecha de nacimiento. Se representan por medio de un rectángulo y pueden ser de tipo: maestras, transaccionales, históricas y temporales.

2. **Atributos:** Los atributos son las características que definen o identifican a una entidad. Estas pueden ser muchas, y el diseñador solo utiliza o implementa las que considere más relevantes. Los atributos son las propiedades que describen a cada entidad en un conjunto de entidades.

En un conjunto de entidades del mismo tipo, cada entidad tiene valores específicos asignados para cada uno de sus atributos, de esta forma, es posible su identificación unívoca. Nótese que dos o más entidades diferentes pueden tener los mismos valores para algunos de sus atributos, pero nunca para todos.

Para cada atributo, existe un dominio del mismo, este hace referencia al tipo de datos que será almacenado o a restricciones en los valores que el atributo puede tomar. Cuando algún atributo correspondiente a una entidad no tiene un valor determinado, recibe el valor nulo, bien sea porque no se conoce, porque no existe o porque no se sabe nada al respecto del mismo.

Se representan mediante un círculo o elipse etiquetado mediante un nombre en su interior. Cuando un atributo es identificativo de la entidad se suele subrayar dicha etiqueta. Por motivos de legibilidad, los atributos suelen no aparecer representados en el diagrama entidad-relación, sino descritos textualmente en otros documentos adjuntos.

3. **Relación:** Describe cierta dependencia entre entidades o permite la asociación de las mismas. Una relación tiene sentido al expresar las entidades que relaciona. Se representan mediante un rombo etiquetado en su interior con un verbo. Este rombo se debe unir mediante líneas con las entidades (rectángulos) que relaciona, para así saber cuál es la relación que lleva cada uno.

Dado un conjunto de relaciones en el que participan dos o más conjuntos de entidades, la correspondencia de cardinalidad indica el número de entidades con las que puede estar relacionada una entidad dada.

Dado un conjunto de relaciones binarias y los conjuntos de entidades A y B, la correspondencia de cardinalidades puede ser:

- **Uno a Uno:** Una entidad de A se relaciona únicamente con una entidad en B y viceversa.
- **Uno a varios:** Una entidad en A se relaciona con cero o muchas entidades en B. Pero una entidad en B se relaciona con una única entidad en A.
- **Varios a Uno:** Una entidad en A se relaciona exclusivamente con una entidad en B. Pero una entidad en B se puede relacionar con 0 o muchas entidades en A.
- **Varios a Varios:** Una entidad en A se puede relacionar con 0 o muchas entidades en B y viceversa.

IV.5.3 SQL

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, *Structured Query Language* o *Lenguaje Estructurado de Consultas*, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales y que resulta muy amigable con PHP [31].

SQL por sus siglas en inglés “structured query language” es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en ellas.

Una de sus características es el manejo del álgebra y el cálculo relacional que permiten efectuar consultas con el fin de recuperar de forma sencilla información de interés de bases de datos, así como hacer cambios en ella.

Es un lenguaje declarativo de "alto nivel" o "de no procedimiento" que, gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros, y no a registros individuales, permite una alta productividad en codificación y la orientación a

objetos. De esta forma, una sola sentencia puede equivaler a una o más sentencias que se utilizarían en un lenguaje convencional de alto nivel para efectuar la misma operación.

En SQL existen dos clasificaciones para las sentencias que manipulan la base de datos [30]:

- **Lenguaje de Definición de Datos (DDL):** Es el que se encarga de la modificación de la estructura de los objetos de la base de datos. Incluye órdenes para modificar, borrar o definir las tablas en las que se almacenan los datos de la base de datos. Existen cuatro operaciones básicas: CREATE, ALTER, DROP y TRUNCATE.
- **Lenguaje de Manipulación de Datos (DML):** Permite a los usuarios llevar a cabo las tareas de consulta o manipulación de los datos, organizados por el modelo de datos adecuado. Incluye las operaciones: SELECT, UPDATE, DELETE, INSERT al igual que algunas cláusulas de condicionamiento: WHERE, JOIN, ON, LIKE, IN.

Generalmente los programadores y desarrolladores de software trabajan más con el DML, mientras que los administradores de bases de datos (DBAs) utilizan ambos, además de funciones avanzadas del lenguaje SQL como disparadores (TRIGGERS), procedimientos almacenados (STORED PROCEDURES) y funciones (FUNCTIONS) las cuales hacen de SQL un lenguaje indicado para nuestro sistema.

DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA WEB INTELIGENTE

Capítulo 5

V.- Desarrollo e Implementación del Sistema web Inteligente.

V.1 Descripción y Estructura Base

Con el desarrollo del sistema, los hoteles de México tendrán una herramienta para ser utilizada por su personal que les permitirá determinar problemas de gestión en la eficiencia energética, facilitar el diagnóstico, monitoreo y evaluación de las medidas de ahorro de energía, buscando el menor impacto al medio ambiente y una mejor administración de la energía.

El Sistema Web Inteligente para la gestión del ahorro de energía está enfocado en las áreas de refrigeración e iluminación este adquirirá información por parte del usuario para efectuar un diagnóstico en el ahorro de energía y apoyado de la información de la base de datos formulará un informe correspondiente, se generaran propuestas para el ahorro de energía, buscando una mejor administración energética orientado a el sector de servicios turísticos particularmente en hoteles denominadas PyMES.

El sistema adapta el conocimiento de expertos que permitirá proporcionar y evaluar medidas de ahorro acorde a los datos proporcionados por los hoteleros para las diferentes áreas de análisis, con esto, se pretende obtener un uso adecuado de la energía generando una reducción de costos.

El Sistema web Inteligente para la gestión del ahorro de energía en refrigeración e iluminación en hoteles de México se desarrollará con base en la teoría sobre TI mencionada en el capítulo anterior y toma como estructura la figura V.1 la cual consta de cinco módulos que describen el proceso de eficiencia energética en hoteles:

1. **Módulo1 - Encuesta:** El diagnóstico energético inicia con una encuesta la cual sirve para recabar información general sobre las diferentes áreas para que posteriormente se pida información particular del área a tratar.

2. **Módulo2 – Diagnóstico Energético:** Se establecen los métodos de diagnóstico por medio de un algoritmo con el que se obtendrán los índices o indicadores de la eficiencia energética por cada área.
3. **Módulo3 – Determinación de las Medidas de Ahorro:** Con base en los índices del diagnóstico se determinaran las medidas de ahorro de energía para cada área por medio de métodos de diagnósticos elaborados por expertos.
4. **Módulo4 – Implementación de las Medidas de Ahorro:** Se aplican las recomendaciones del diagnóstico para cada área.
5. **Módulo5 – Monitoreo de las Medidas Implantadas:** Una vez implementadas las recomendaciones se realiza un nuevo diagnóstico para corroborar que las recomendaciones fueron las adecuadas o ver si existen algunas nuevas mejoras por medio de una retroalimentación que permite la mejora continua de la gestión energética.

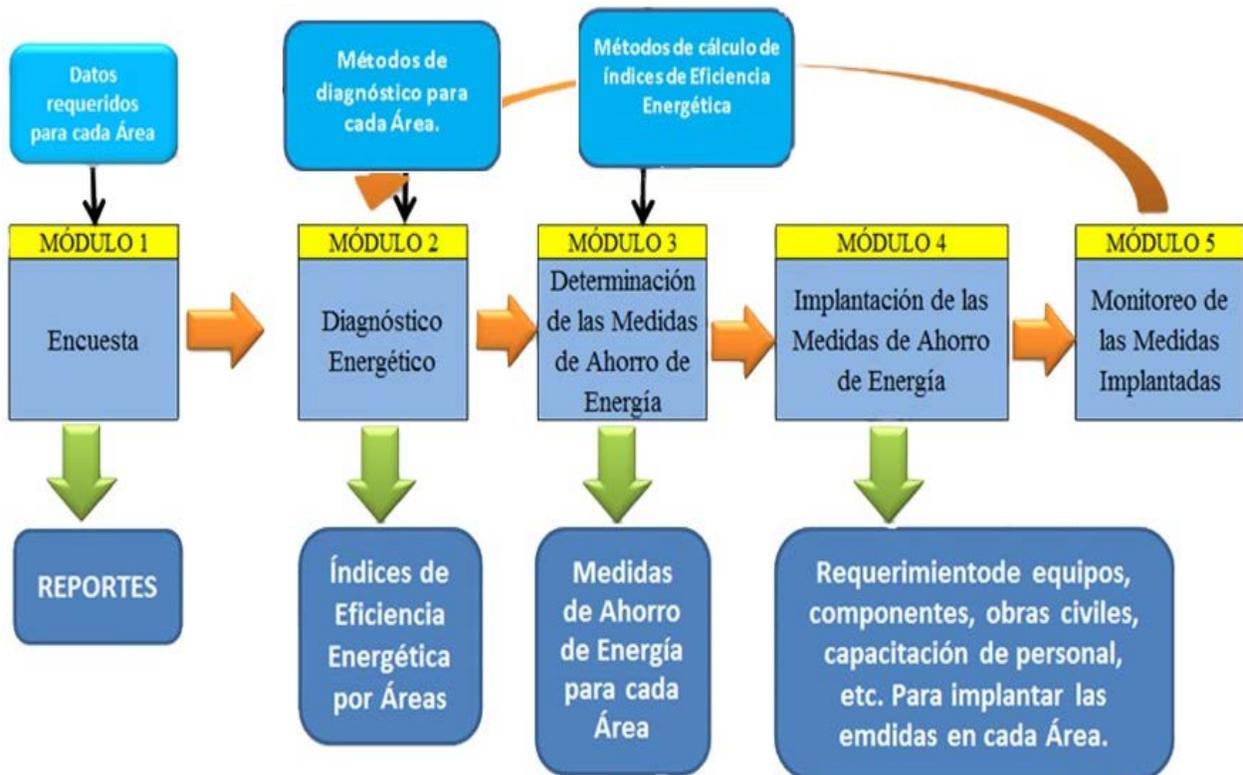


Figura V.1.1 Diagrama de bloques de Eficiencia Energética en Hoteles.

El sistema contempla las áreas más significativas dentro de la gestión energética las cuales son:

- ❖ Iluminación
 - Interior
 - Exterior
- ❖ Aire Acondicionado
- ❖ Calentamiento de Agua
- ❖ Refrigeración
- ❖ Elevadores
- ❖ Bombeo
- ❖ Sistemas Pasivos

Ya que este trabajo se enfoca sólo en las áreas de Iluminación interior, Iluminación exterior y refrigeración, serán explicadas y descritas a detalle, sin embargo el proceso es el mismo para todas las demás áreas dentro del sistema.

La tabla V.1 muestra las normas mexicanas que se aplican a las áreas tratadas en este documento.

Tabla V.1 Normas Mexicanas que se Aplican a las Áreas.

Área	Norma Aplicada	Descripción
Iluminación Interior	NOM 025 ENER	Condiciones de iluminación en los centros de trabajo.
	NOM 013 ENER	Eficiencia energética para sistemas de alumbrado en vialidades.
	NOM 007 ENER	Eficiencia energética en sistemas de alumbrado en edificios no residenciales.
Iluminación Exterior	NOM 025 ENER	Condiciones de iluminación en los centros de trabajo.
	NOM 013 ENER	Eficiencia energética para sistemas de alumbrado en vialidades.
	NOM 007 ENER	Eficiencia energética en sistemas de alumbrado en edificios no residenciales.
Refrigeración	NOM-015-ENER	Eficiencia energética de refrigeradores y congeladores electrodomésticos. Límites, métodos de prueba y etiquetado.

El equipo de cómputo utilizado como servidor cuenta con el siguiente Hardware:

- Servidor WEB (Equipo)
 - Dell PowerEdge T710
 - Procesador: Intel Xeon serie 5500.
 - Memoria: 12GB DDR3.
 - Disco Duro: Unidades de estado sólido SAS y SATA de 1 TB.
 - 2 NIC Ethernet Gigabit Broadcom® NetXtreme II™ 5709c de doble puerto integradas con conmutación por error y equilibrio de carga.
 - Tarjeta de video: Matrox G200 integrado.

El Software que se empleó en el desarrollo del sistema fue el siguiente:

- Base de Datos
 - Server Version: 5.6.13 MySQL Community Server (GPL).
- Servidor Web
 - Apache 2.2.25.
- Lenguajes de programación y librerías
 - YI Framework 1.1.14
 - XHTML 1.0 Transitional.
 - Javascript / jQuery 1.6.1
 - CSS 2.1
 - PHP 5.5.3

La figura V.1.2 muestra una petición del cliente al servidor ejemplificando como es la interacción entre estos, cuando se ingresa al sistema o se realiza un diagnóstico se hace por medio de una nube hibrida donde la “IaaS” proporciona el hardware y el SaaS proporciona el software, los siguientes pasos describen este proceso:

1. **Petición:** Una petición inicia cuando el cliente quiere ingresar al sistema con lo que dispara un “get” que puede ser una petición a la página principal del sistema o alguna otra dentro del proceso de diagnóstico.

2. **Búsqueda en el Repositorio:** Al recibir el servidor la petición de que un usuario quiere ingresar a cierta página, el servidor busca en su repositorio o banco de páginas cual es la que se solicita.
3. **Obtención:** Una vez encontrada la página solicitada el servidor la toma para ser mostrada.
4. **Creación de Código:** El archivo solicitado es analizado para ver que recursos necesita para su correcta visualización generando la página con sus respectivos elementos.
5. **Consulta a la Base de Datos:** En caso de ser necesario se realizan algunas consultas a la base de datos para aportar elementos a la página a mostrar.
6. **Respuesta:** Se devuelve le código HTML correspondiente con todos los elementos que contiene la página solicitada generando una página web con cual el usuario puede interactuar con el sistema.

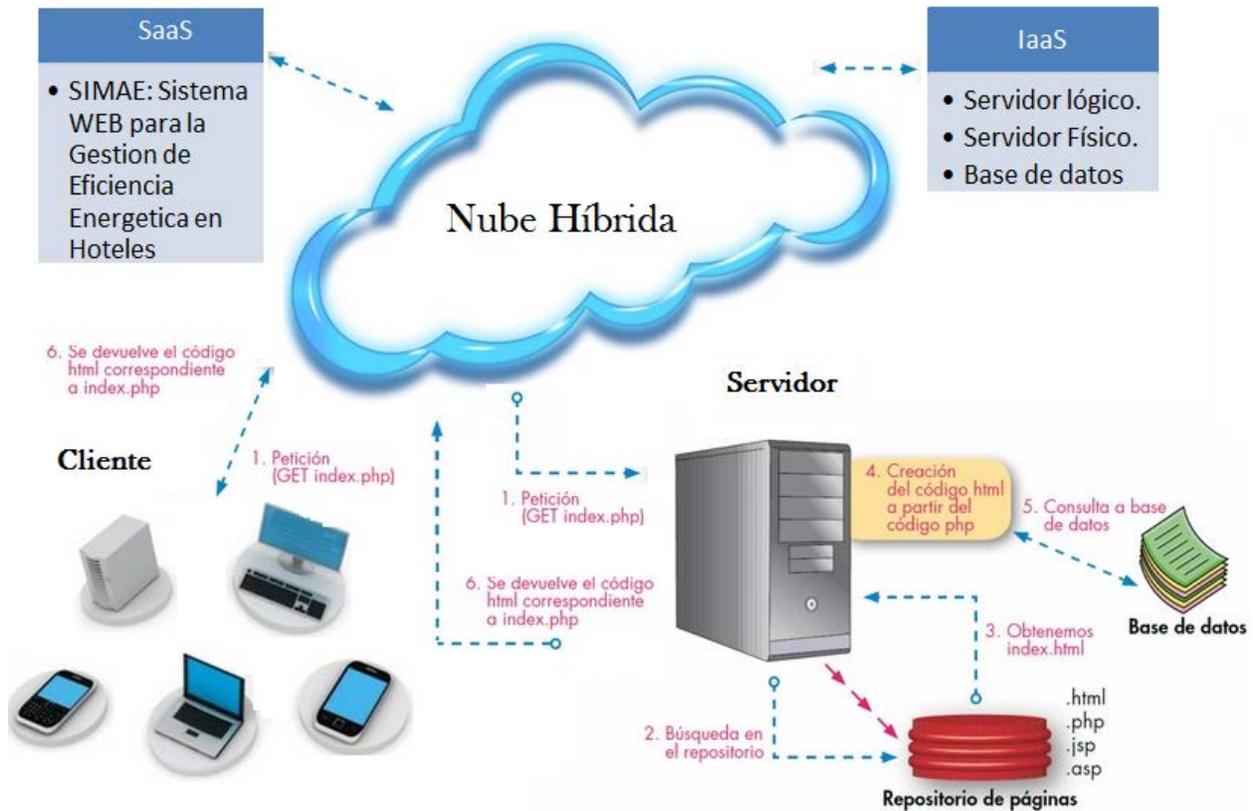


Figura V.1.2 Petición del Cliente al Servidor.

V.2 Modelos de las Áreas en la Gestión Energética.

Una forma de entender cómo trabaja el sistema es por medio de un diagrama de flujo; el cual ilustra y describe el proceso que lleva el diagnóstico de cierta área y un diagrama relacional; el cual describe la base de datos que se utilizara para el algoritmo del área en cuestión.

Como inicio es necesario establecer un módulo de autenticación y de registro de usuario con lo que se garantiza el acceso al sistema mediante un “usuario” y su “clave” de este modo ligaré un usuario a un hotel en particular y este estará sujeto al diagnóstico energético.

En la figura V.2.1 se realiza una solicitud al servidor el cual muestra la página principal de inicio “index.php”, para tener acceso al sistema se necesita registrarse o si ya se está registrado se necesita autenticarse obteniendo el acceso por medio de la base de datos, si la autenticación fue verdadera se ingresa al sistema y en caso contrario se regresa a la página principal del sistema.

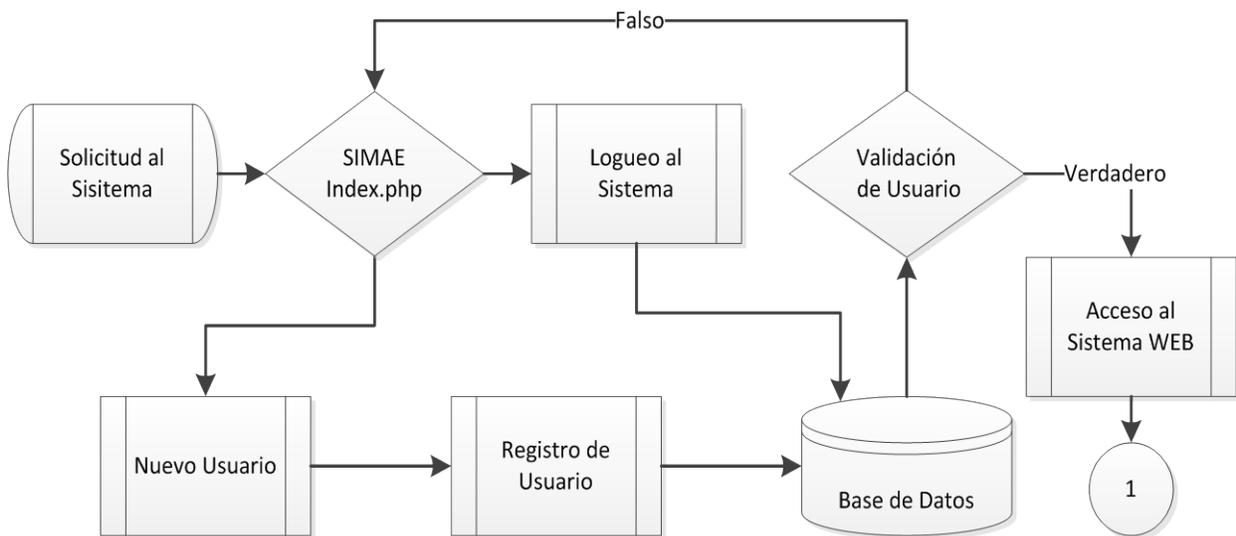
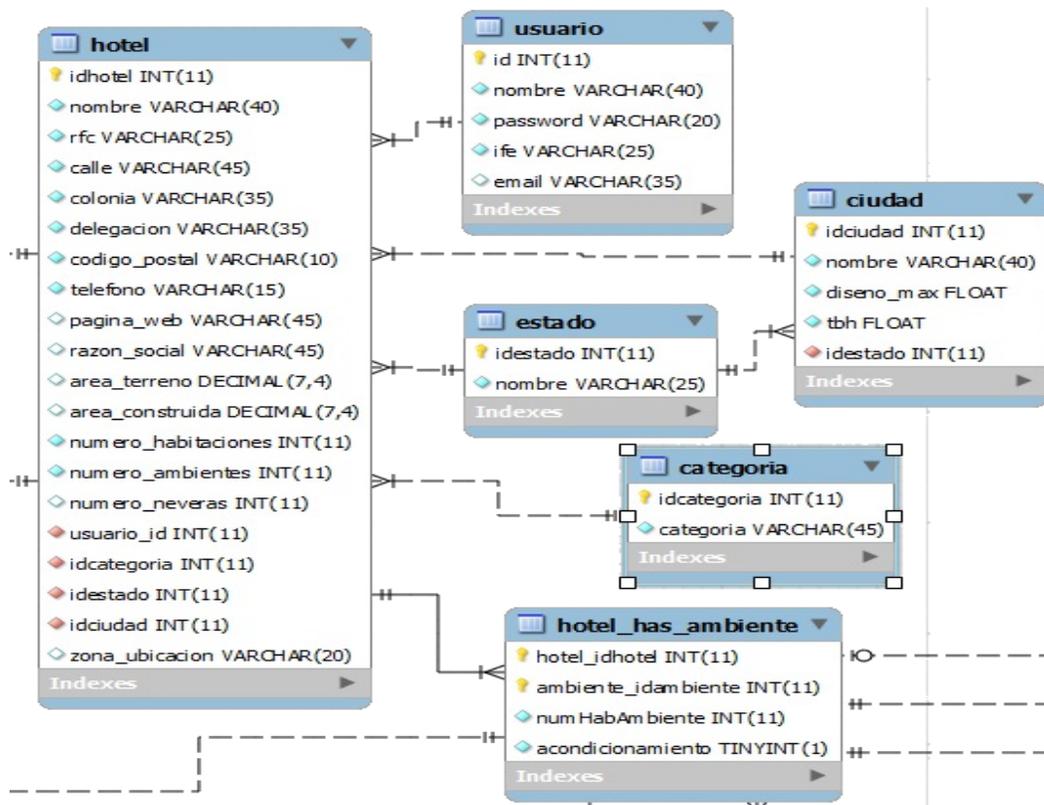


Figura V.2.1 Diagrama de Flujo del Módulo de Autenticación.

El siguiente fragmento del DER describe la estructura básica de la base de datos para el módulo de autenticación, mostrando la relación que hay entre el hotel, los usuarios, la ciudad, el estado y la categoría del hotel, estas relaciones se rigen con las siguientes reglas:

- Un usuario solo podrá tener registrado un hotel, a la vez que un hotel solo podrá estar asociado con un solo usuario. Relación 1-1 entre Usuario y Hotel.
- El hotel puede estar conformado por varios ambientes (habitaciones simples, habitaciones dobles, suites, cocinas, patios, etc.), y un determinado ambiente se puede asociar con varios hoteles. Relación N-N entre Hotel y Ambiente.
- Para cada ambiente que tiene el hotel, pueden existir varios espacios, (por ejemplo, hay 15 habitaciones dobles y 10 habitaciones simples), y un espacio pertenece a solo un ambiente. Relación 1-N entre Ambiente y Espacio.
- El hotel solo puede pertenecer a un estado y a una ciudad. Relación 1-1 entre el hotel y el estado.



La figura V.2.2 muestra de forma general el proceso que sigue el sistema para realizar la gestión energética para cualquier área, inicia con la encuesta en donde se realiza el registro de ambientes y de tarifas, una vez registrados los ambientes se puede proceder a seleccionar un área a la cual se le aplicará un algoritmo generado por expertos para realizar el diagnóstico energético con el que se obtendrán indicadores que servirán para dar las recomendaciones más adecuadas para que se realice su implementación, una vez implementadas las recomendaciones se realizara un nuevo diagnóstico para garantizar que las recomendaciones fueron las idóneas y así mostrar el ahorro energético generado.

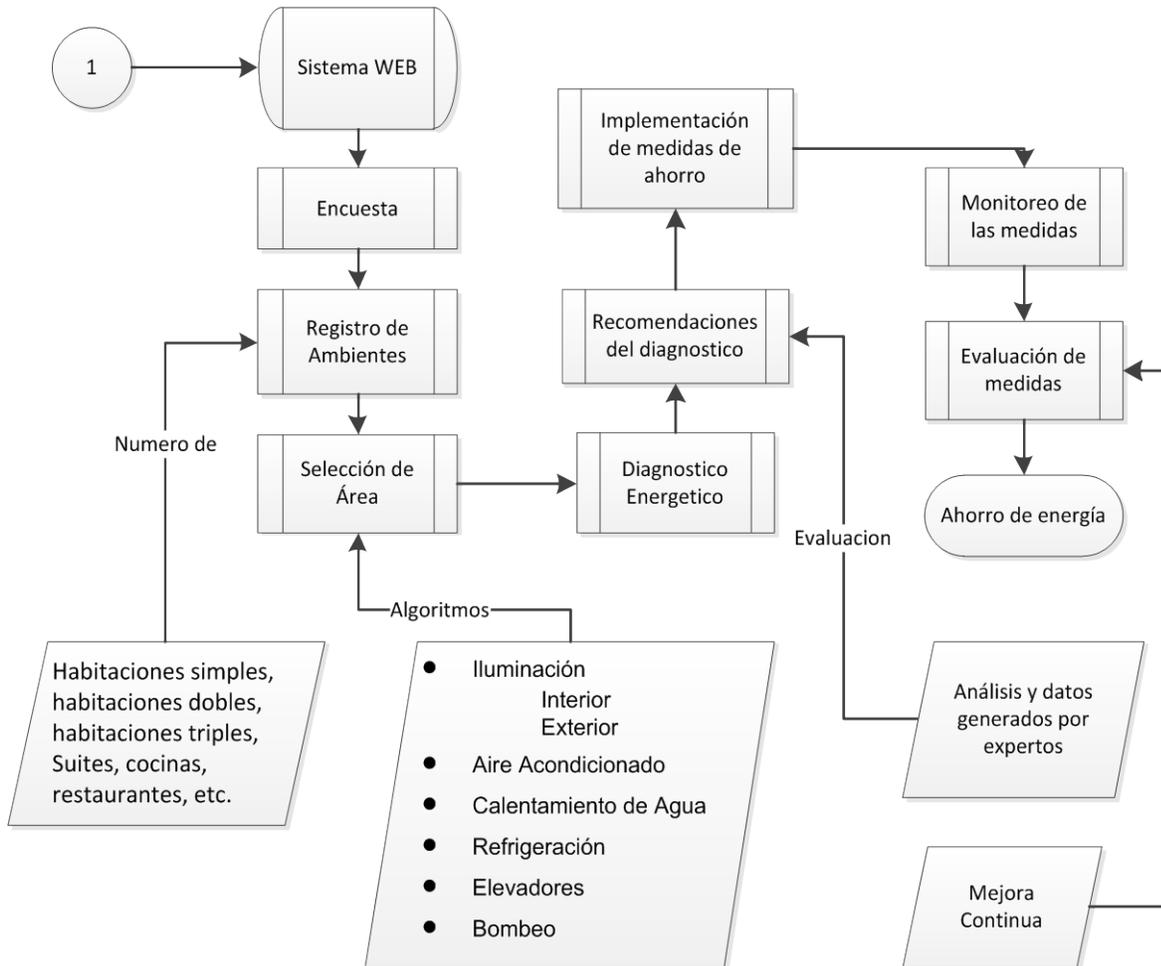
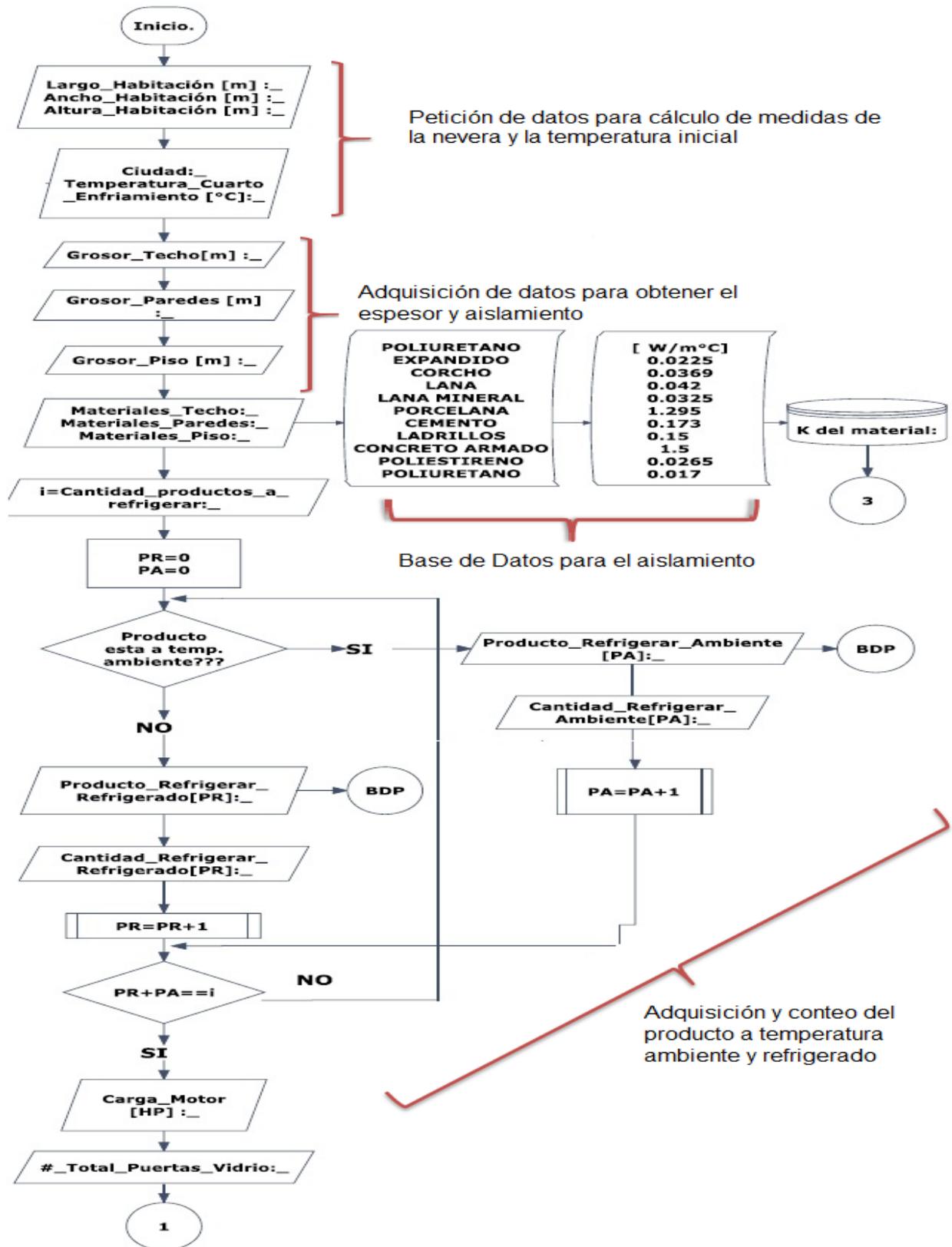


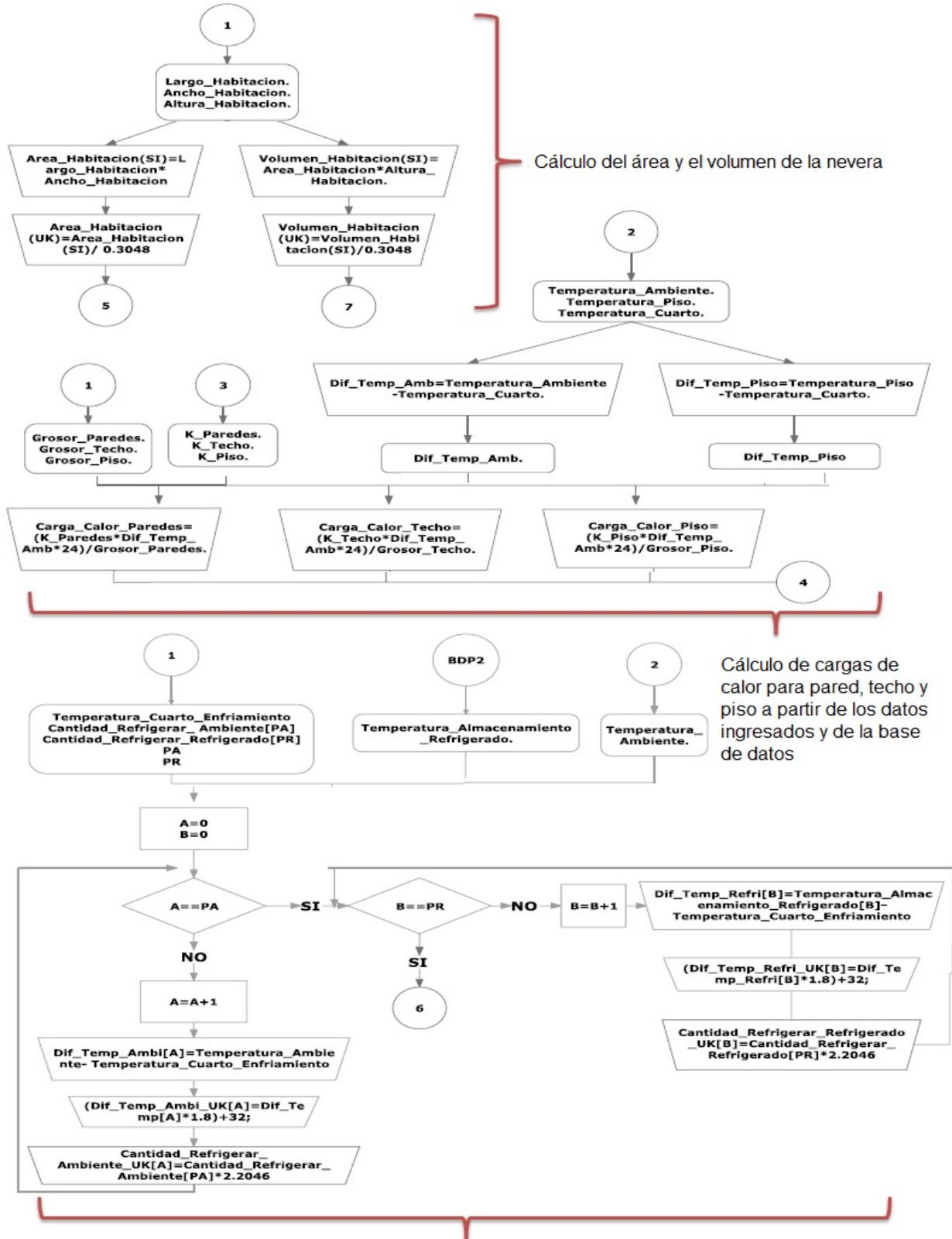
Figura V.2.2 Diagrama de Flujo para el Diagnóstico de la Gestión Energética.

A continuación demostraré el proceso anterior y para esto emplearé las áreas de refrigeración e iluminación, primero mostraré su diagrama de flujo el cual describirá los elementos que se necesitaran para el cálculo del diagnóstico energético, después el diagrama entidad relación para representar la base de datos que se usara para cada área.

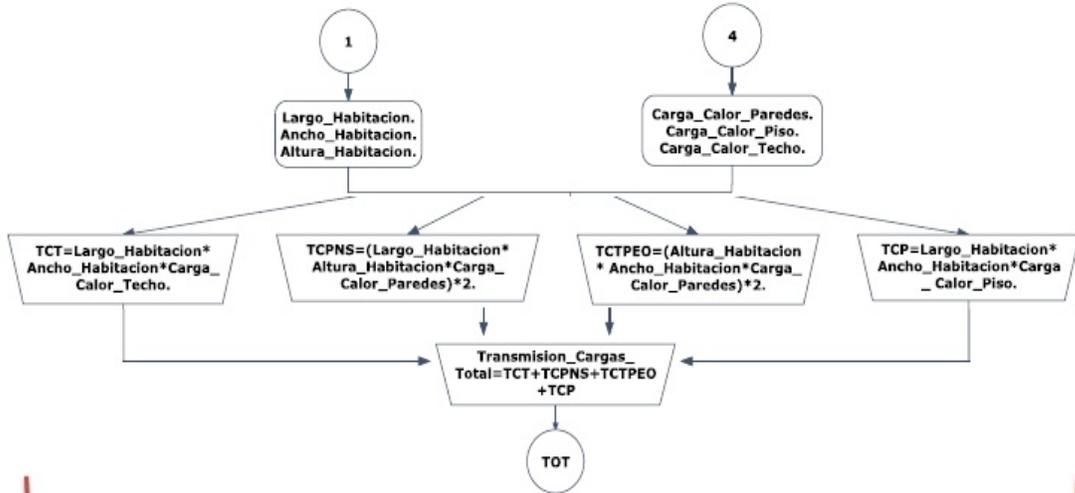
V.2.1 Refrigeración

- **Diagrama de flujo:** El siguiente diagrama de flujo muestra el proceso que sigue el diagnóstico de refrigeración para calcular las cargas de energía de una nevera dentro de los cuales destacan las siguientes peticiones y cálculos:
 - ❖ Petición de datos para cálculo de medidas de la nevera y la temperatura inicial.
 - ❖ Adquisición de datos para obtener el espesor y aislamiento.
 - ❖ Adquisición y conteo del producto a temperatura ambiente y refrigerada.
 - ❖ Cálculo del área y el volumen de la nevera.
 - ❖ Cálculo de cargas de calor para pared, techo y piso a partir de los datos ingresados y de la base de datos.
 - ❖ Cálculo para obtener la carga del producto a refrigerar a partir de los productos y la temperatura ingresados.
 - ❖ Cálculos de cargas por transmisión de las paredes, techo y piso de la nevera.
 - ❖ Cálculos de cargas adicionales por puertas de vidrio y motor.
 - ❖ Cálculos de cargas de todo el contenido refrigerado contemplando la carga por cambio de aire.
 - ❖ Cálculos para la obtención de la carga total general en toneladas de refrigeración.

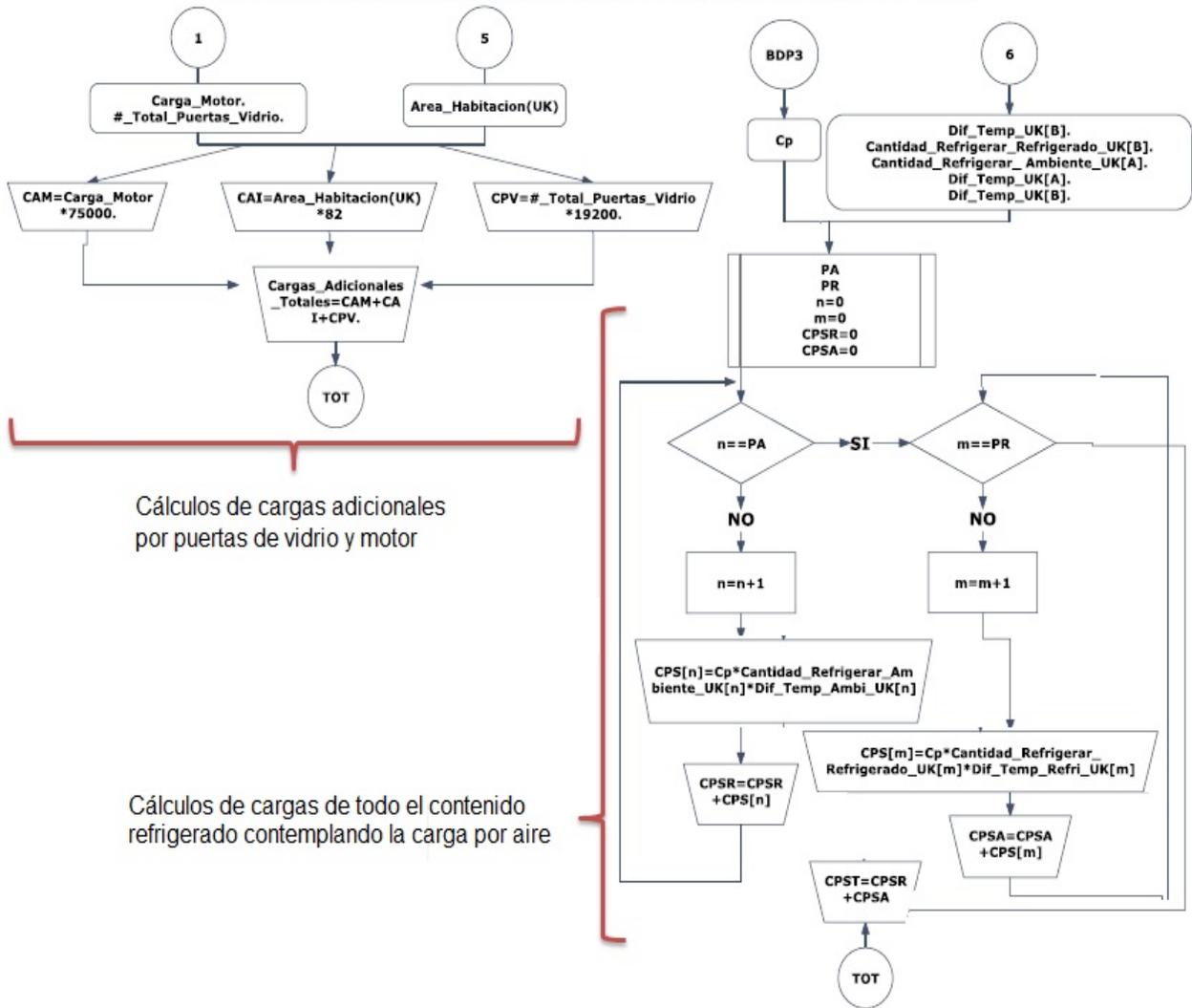




Cálculo para obtener la carga del producto a refrigerar a partir de los productos y la temperatura ingresados.

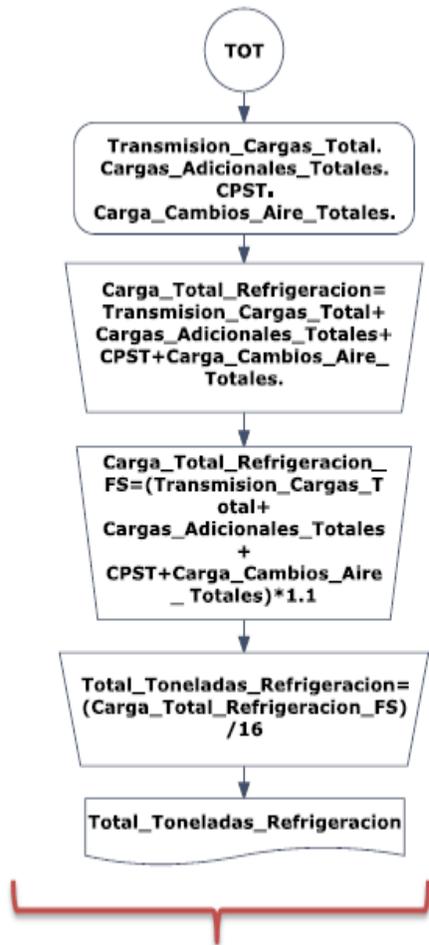


Cálculos de cargas por transmisión de las paredes, techo y piso de la nevera

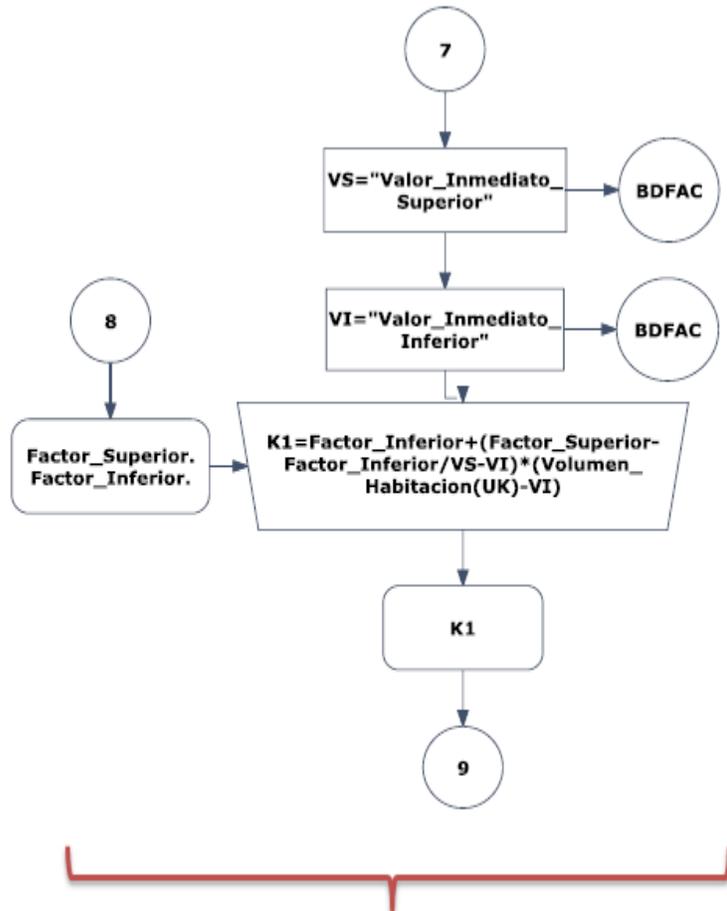


Cálculos de cargas adicionales por puertas de vidrio y motor

Cálculos de cargas de todo el contenido refrigerado contemplando la carga por aire

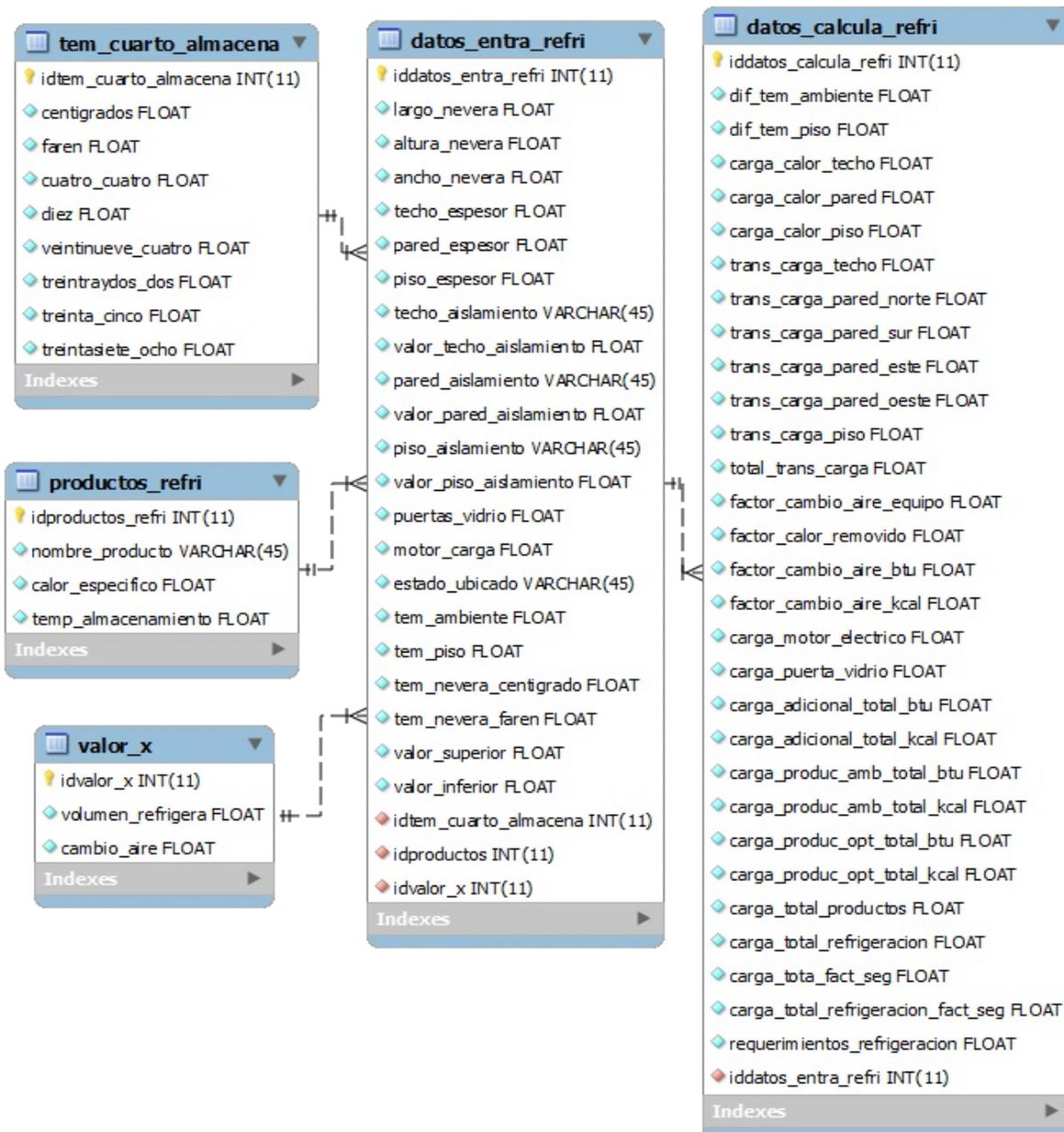


Cálculos para la obtención de la carga total general en toneladas de refrigeración



Cálculos para obtener las cargas de factor inferior y superior por aire

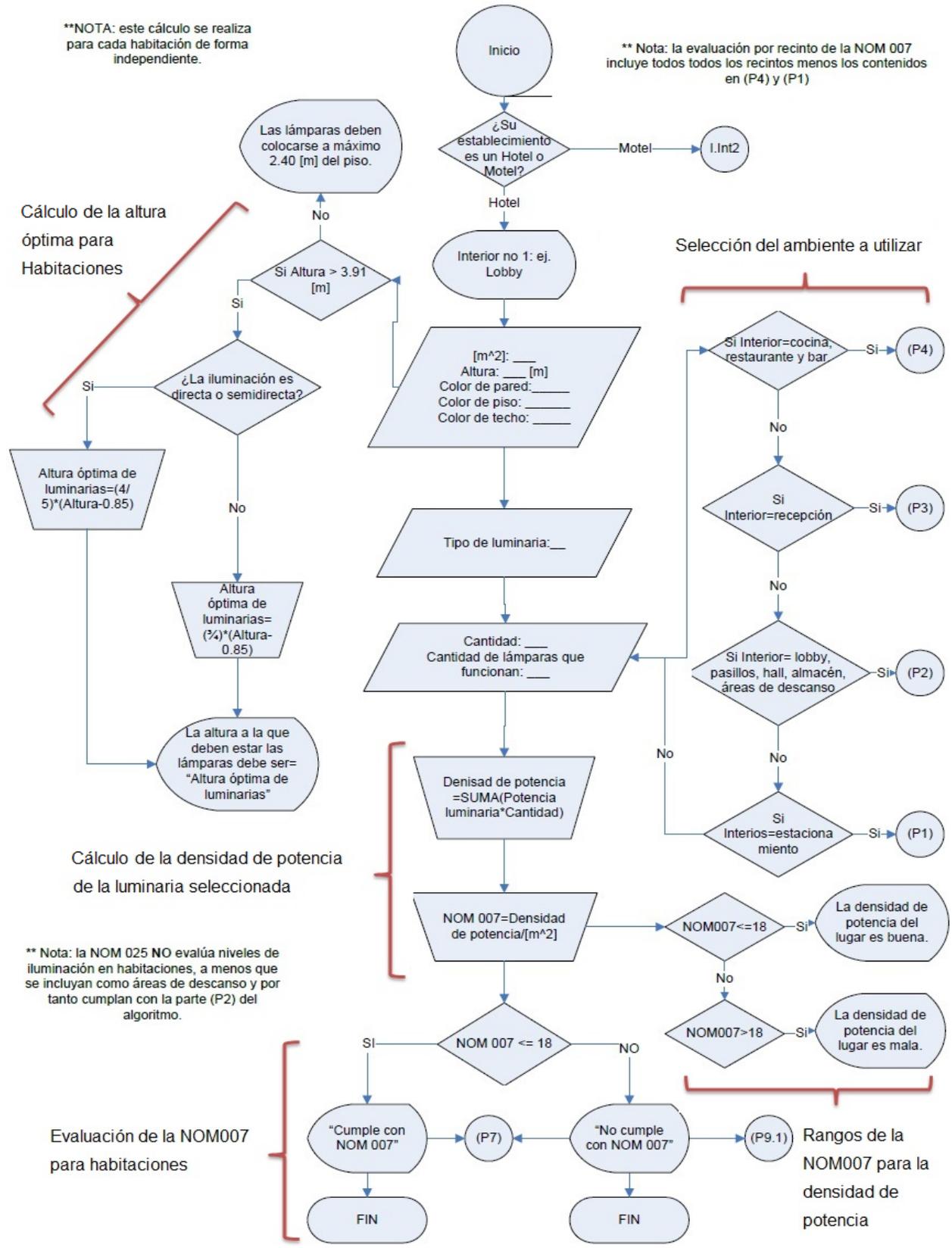
- **Diagrama relacional:** Es la representación de la Base de Datos que se utiliza para realizar el diagnóstico energético para neveras, la siguiente porción del DER general muestra las relaciones entre los datos de entrada para la nevera con los cuales se realizan los cálculos con ayuda de catálogos de productos y tablas de temperatura para que posteriormente sean guardados para su futura utilización.

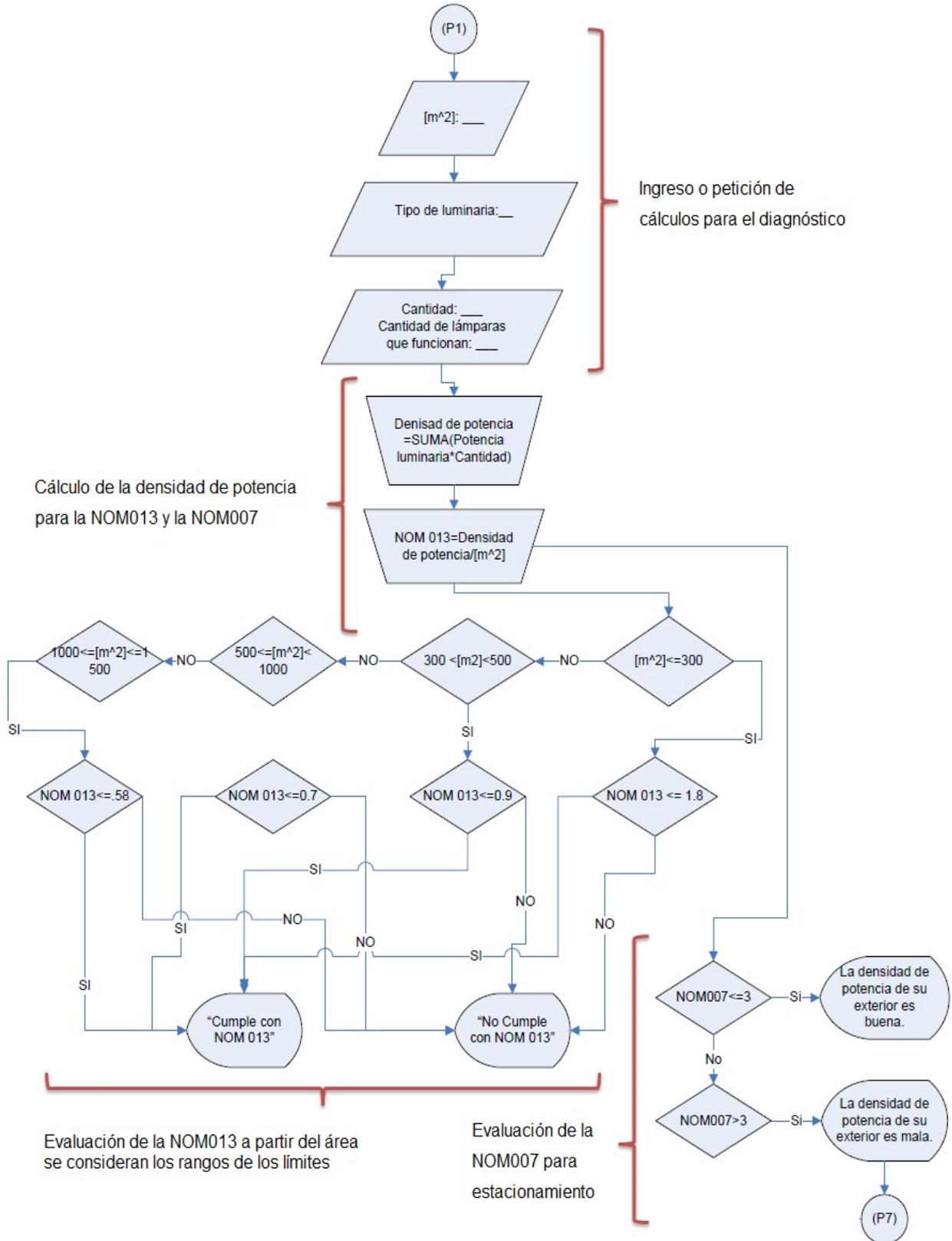


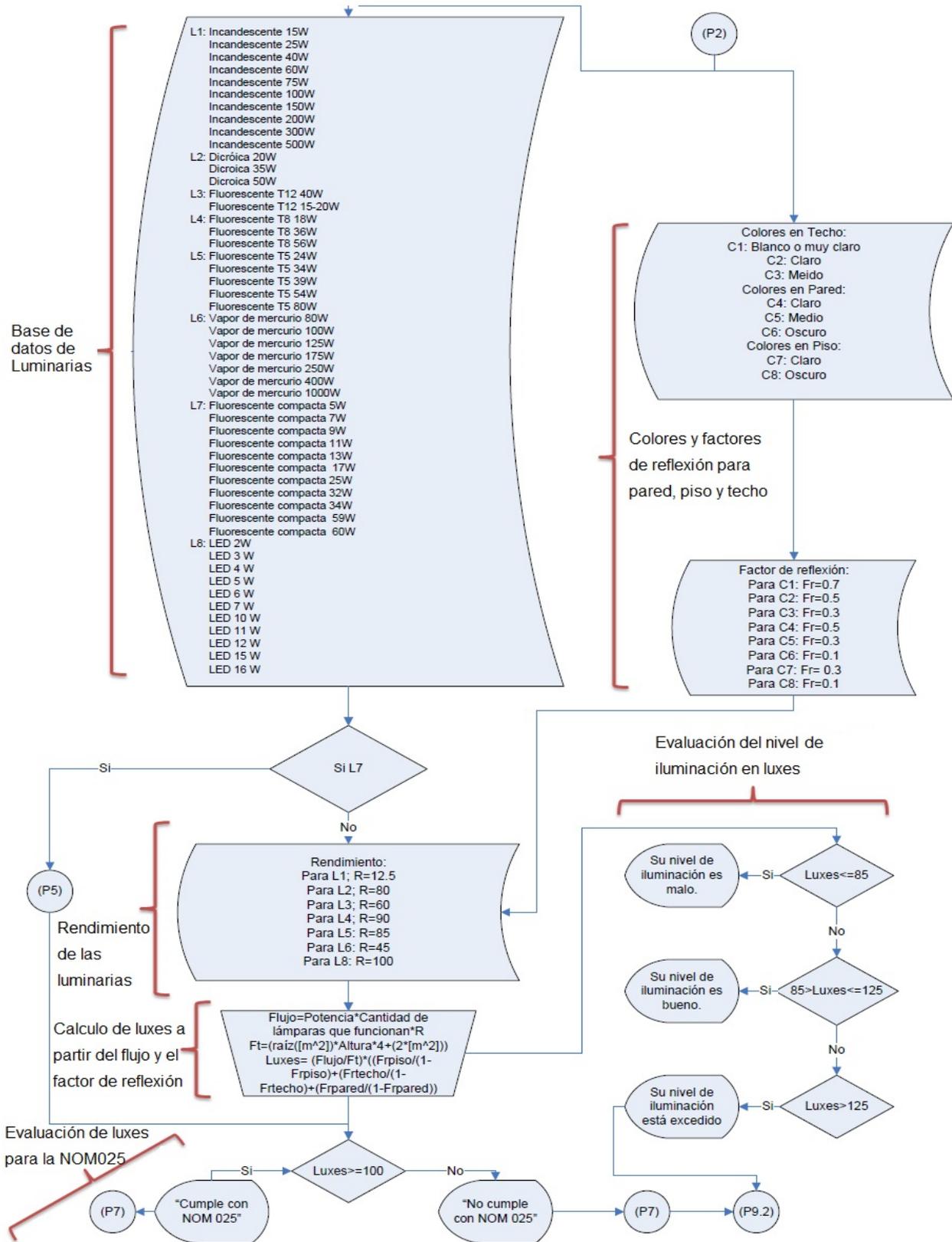
V.2.2 Iluminación Interior

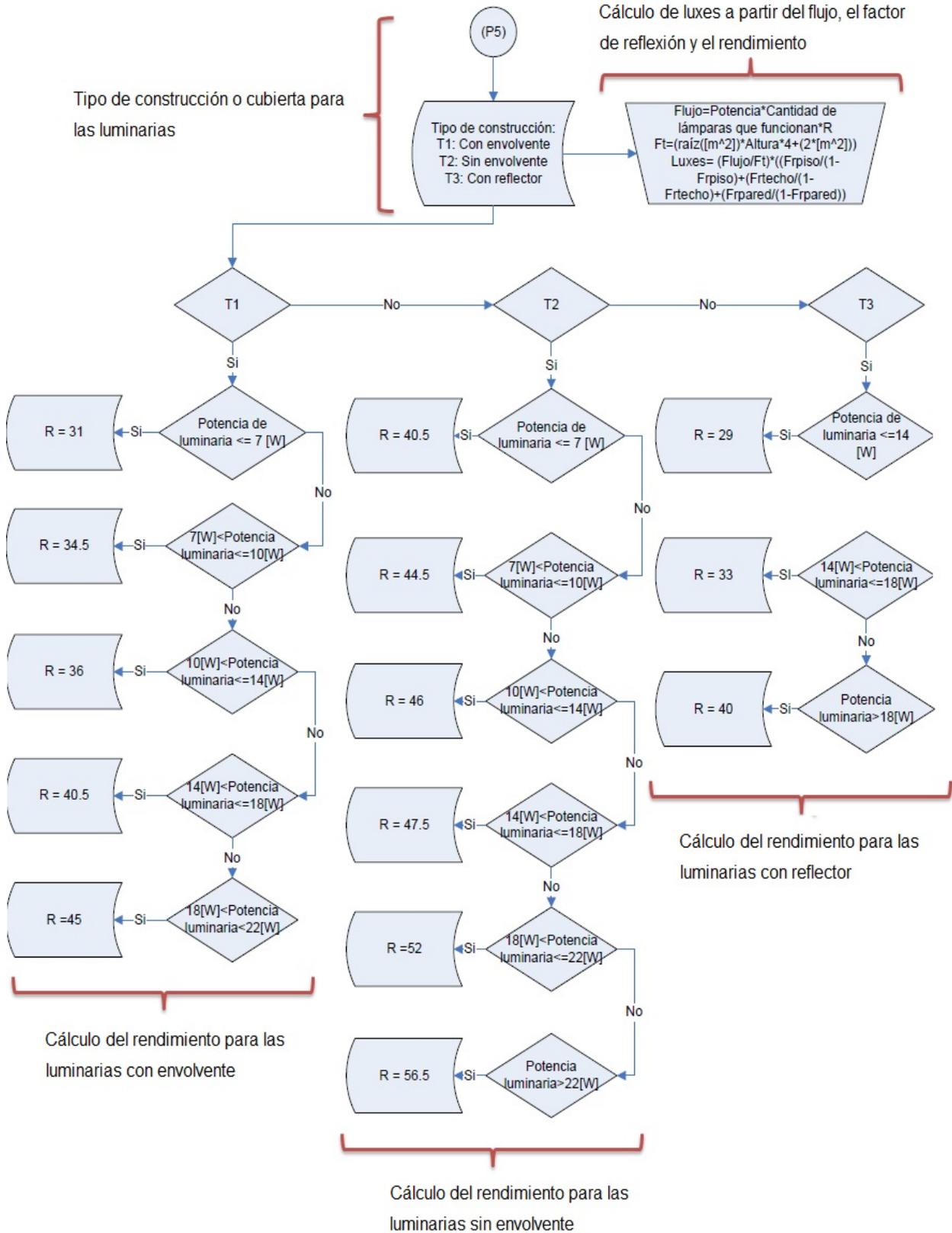
- **Diagrama de flujo:** Representación del proceso por el que pasa el diagnóstico de Iluminación Interior para el cálculo del consumo eléctrico y emisiones de CO₂. Las evaluaciones energéticas en el área de iluminación contemplan las normas mexicanas descritas en la tabla V.1 con las cuales se realizan los siguientes cálculos y evaluaciones para interiores:
 - ❖ Cálculo de la altura óptima para habitaciones.
 - ❖ Selección del ambiente a utilizar.
 - ❖ Cálculo de la densidad de potencia de la luminaria seleccionada.
 - ❖ Evaluación de la NOM007 para habitaciones.
 - ❖ Rangos de la NOM007 para la densidad de potencia.
 - ❖ Ingreso o petición de cálculos para el diagnóstico.
 - ❖ Cálculo de la densidad de potencia para la NOM013 y la NOM007.
 - ❖ Evaluación de la NOM013 a partir del área se consideran los rangos de los límites.
 - ❖ Colores y factores de reflexión para pared, piso y techo.
 - ❖ Evaluación del nivel del nivel de iluminación en luxes.
 - ❖ Rendimiento de las luminarias.
 - ❖ Cálculo de luxes a partir del flujo, el factor de reflexión y el rendimiento.
 - ❖ Evaluación de luxes para la NOM025.
 - ❖ Cálculo del rendimiento para las luminarias con envoltente.
 - ❖ Tipo de construcción o cubierta para las luminarias.
 - ❖ Cálculo del rendimiento para las luminarias con envoltente.
 - ❖ Cálculo del rendimiento para las luminarias sin envoltente.
 - ❖ Cálculo del rendimiento para las luminarias con reflector.
 - ❖ Cálculo para determinar el ahorro de la luminaria recomendada.
 - ❖ Cálculo para las emisiones de CO₂.
 - ❖ Suma de todas las cargas generadas en las diferentes áreas para obtener la carga total eléctrica del hotel.

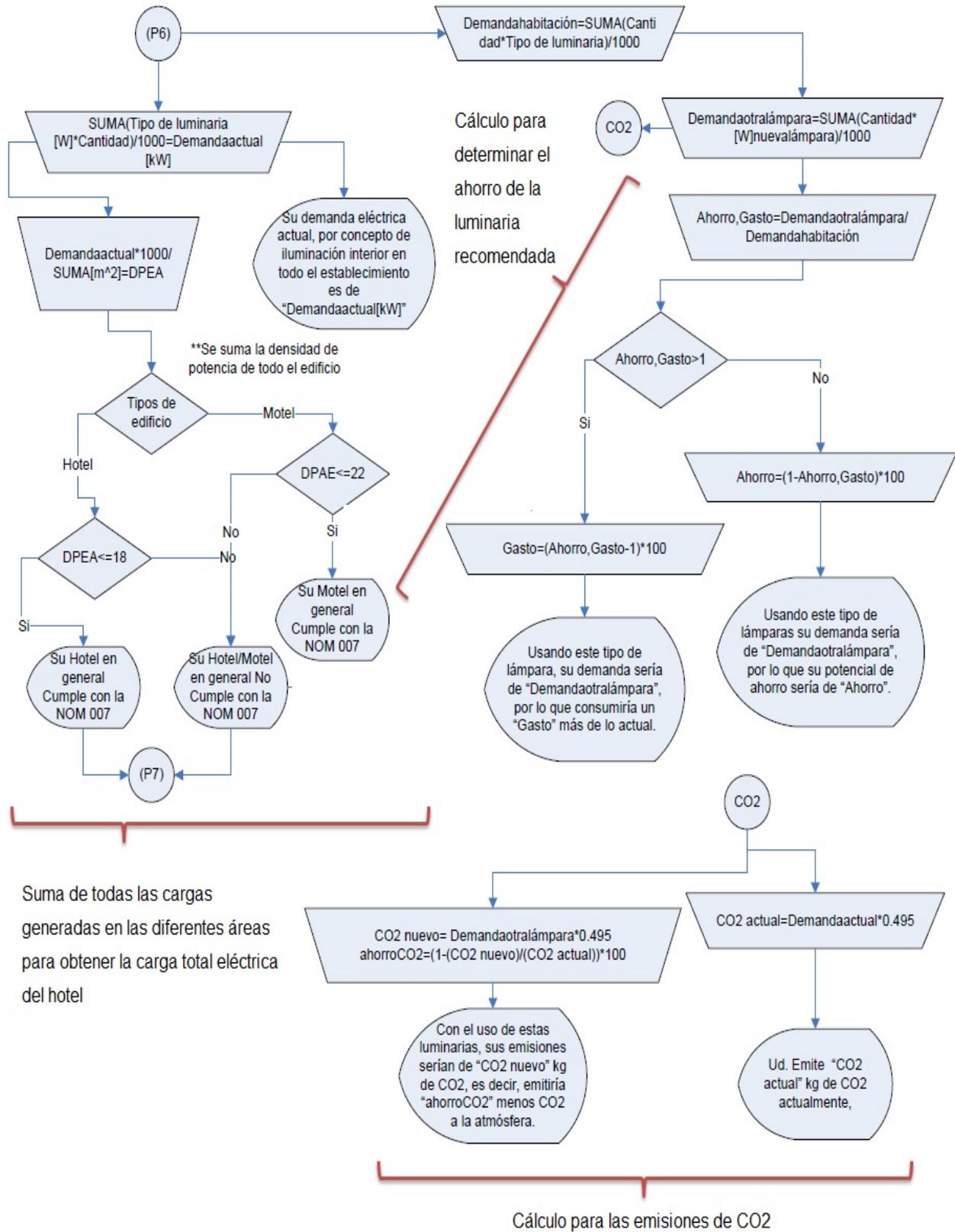
V.- DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA WEB INTELIGENTE



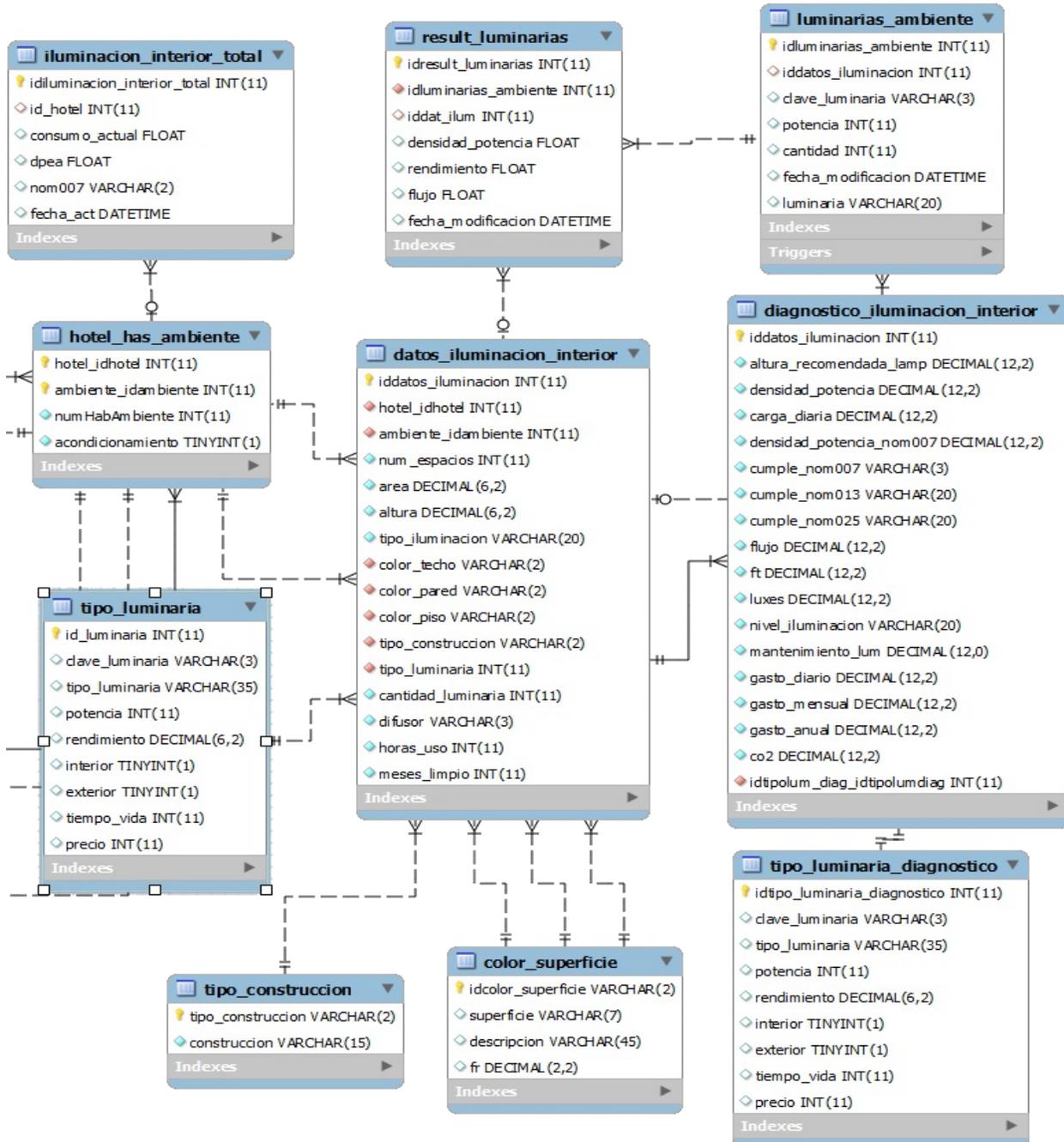






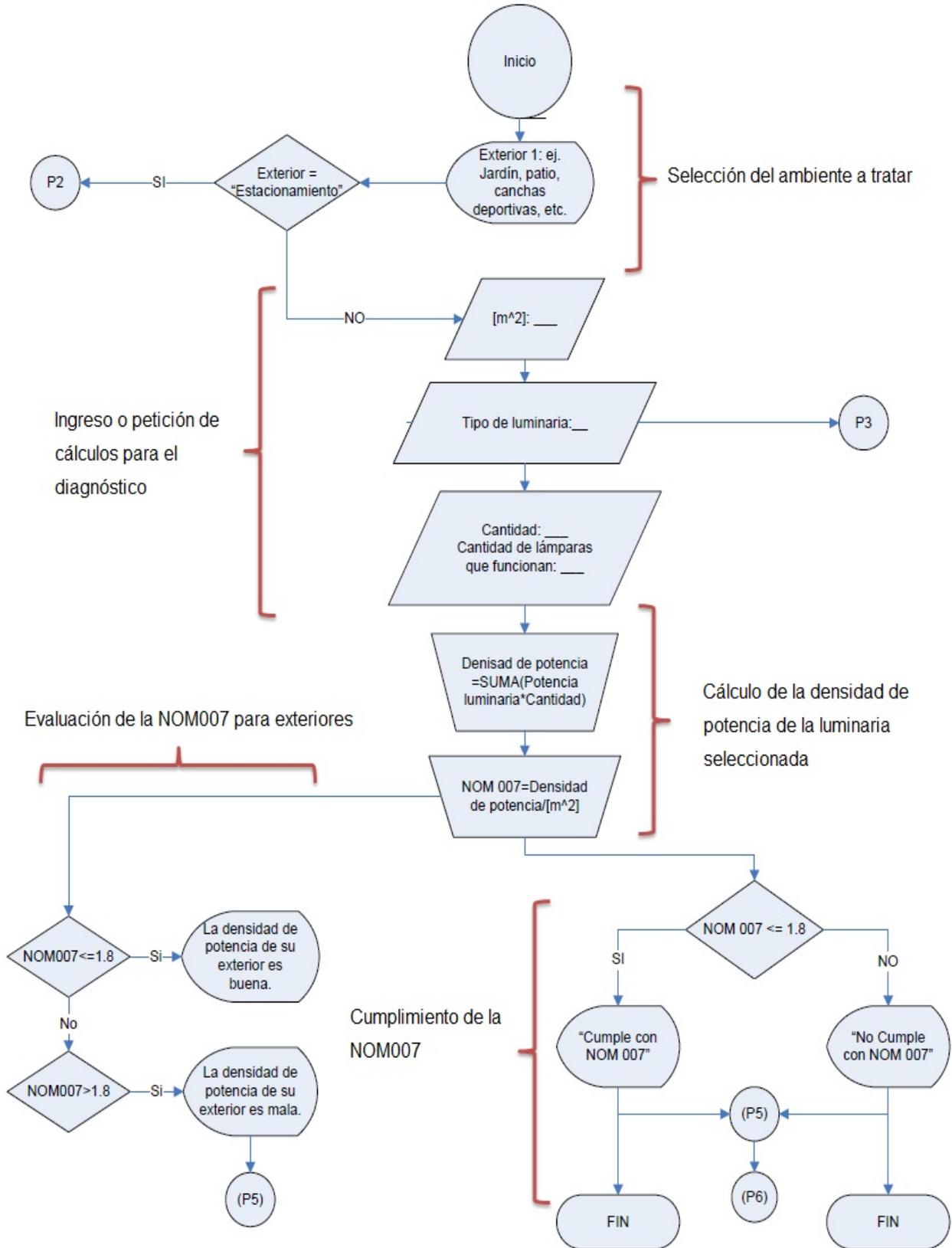


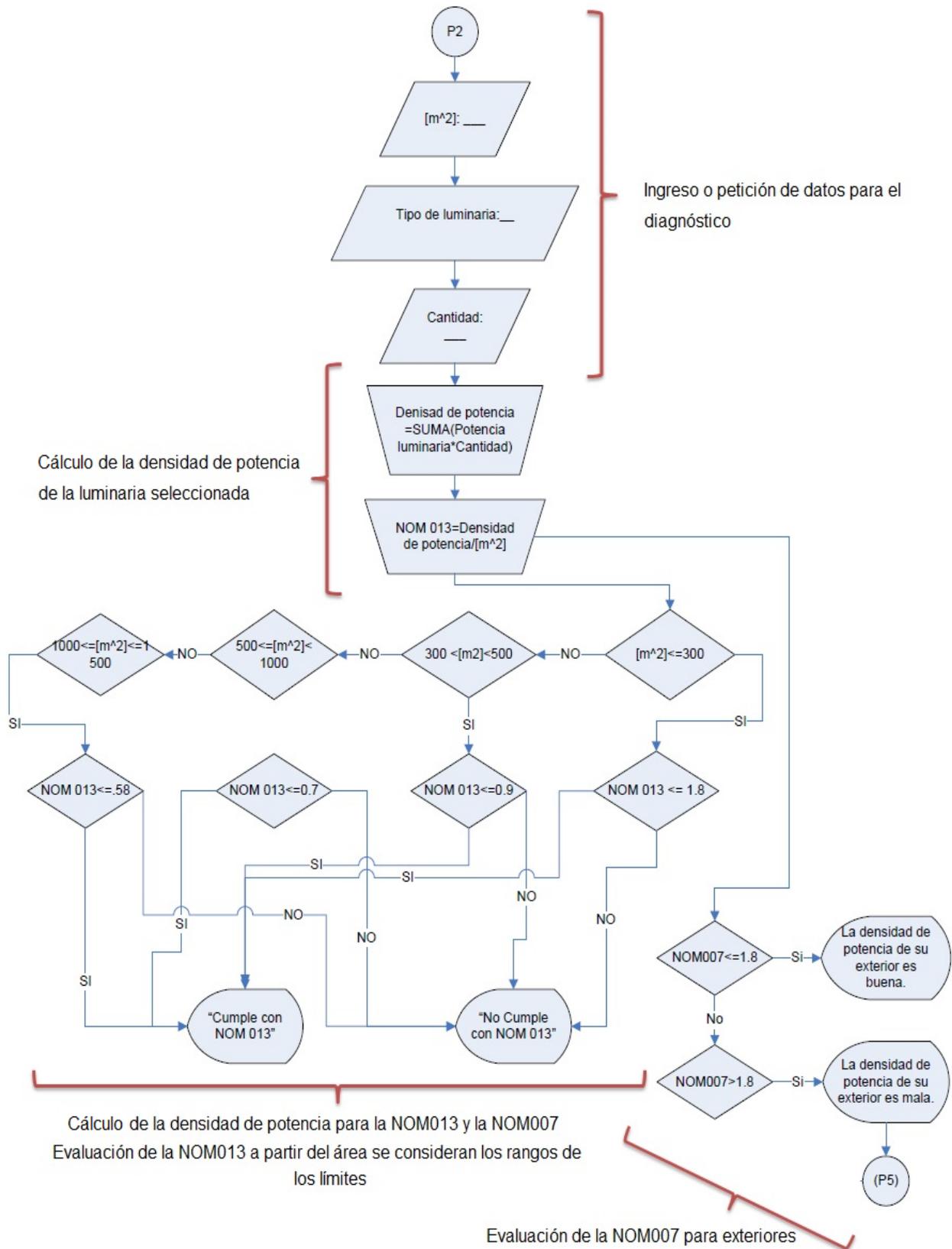
- **Diagrama Entidad Relación:** El siguiente fragmento del DER representa la base de datos para el área de iluminación Interior y las relaciones entre los datos ingresados, los catálogos utilizados para los cálculos o su representación y los datos generados y almacenados para su evaluación y futura utilización en las recomendaciones.

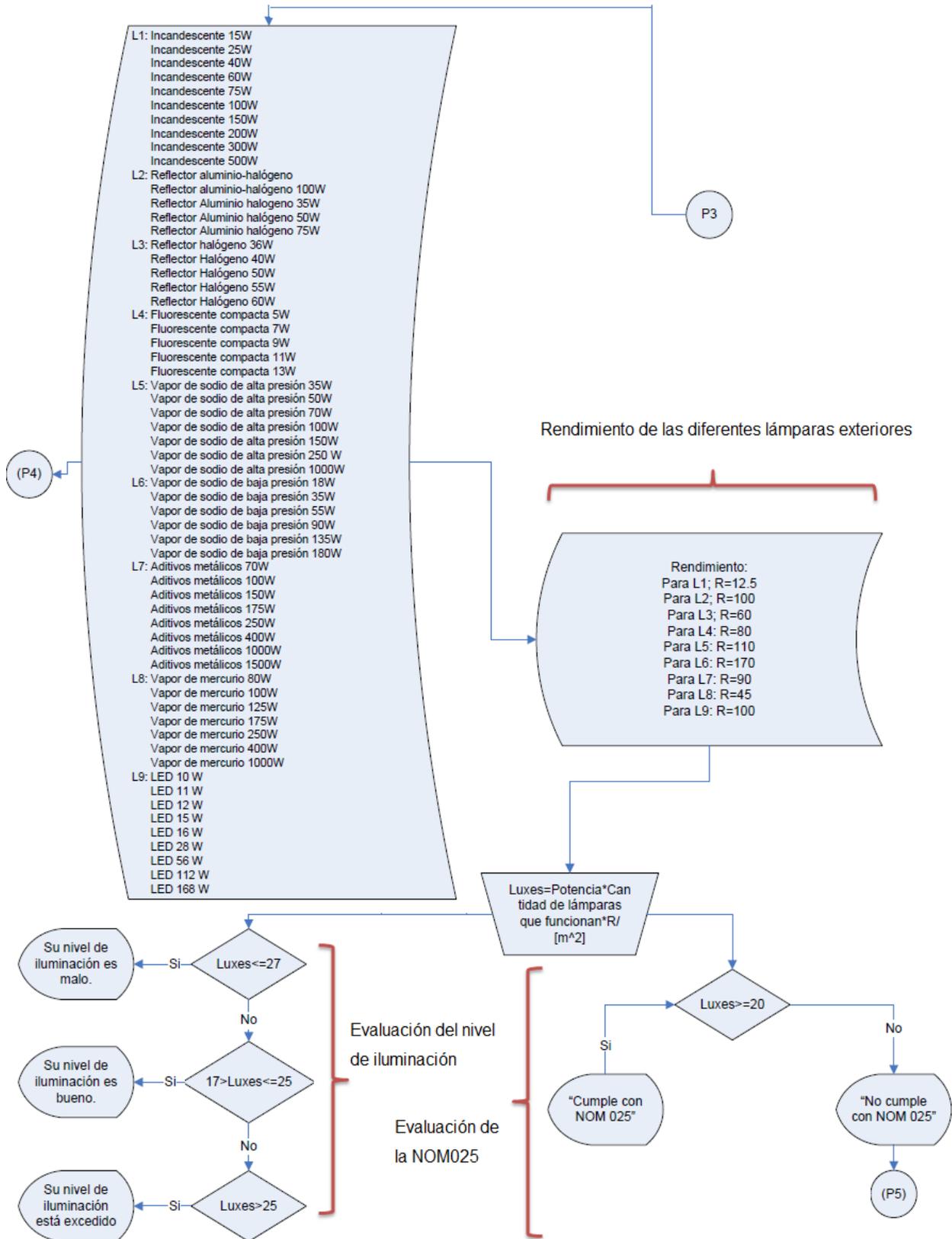


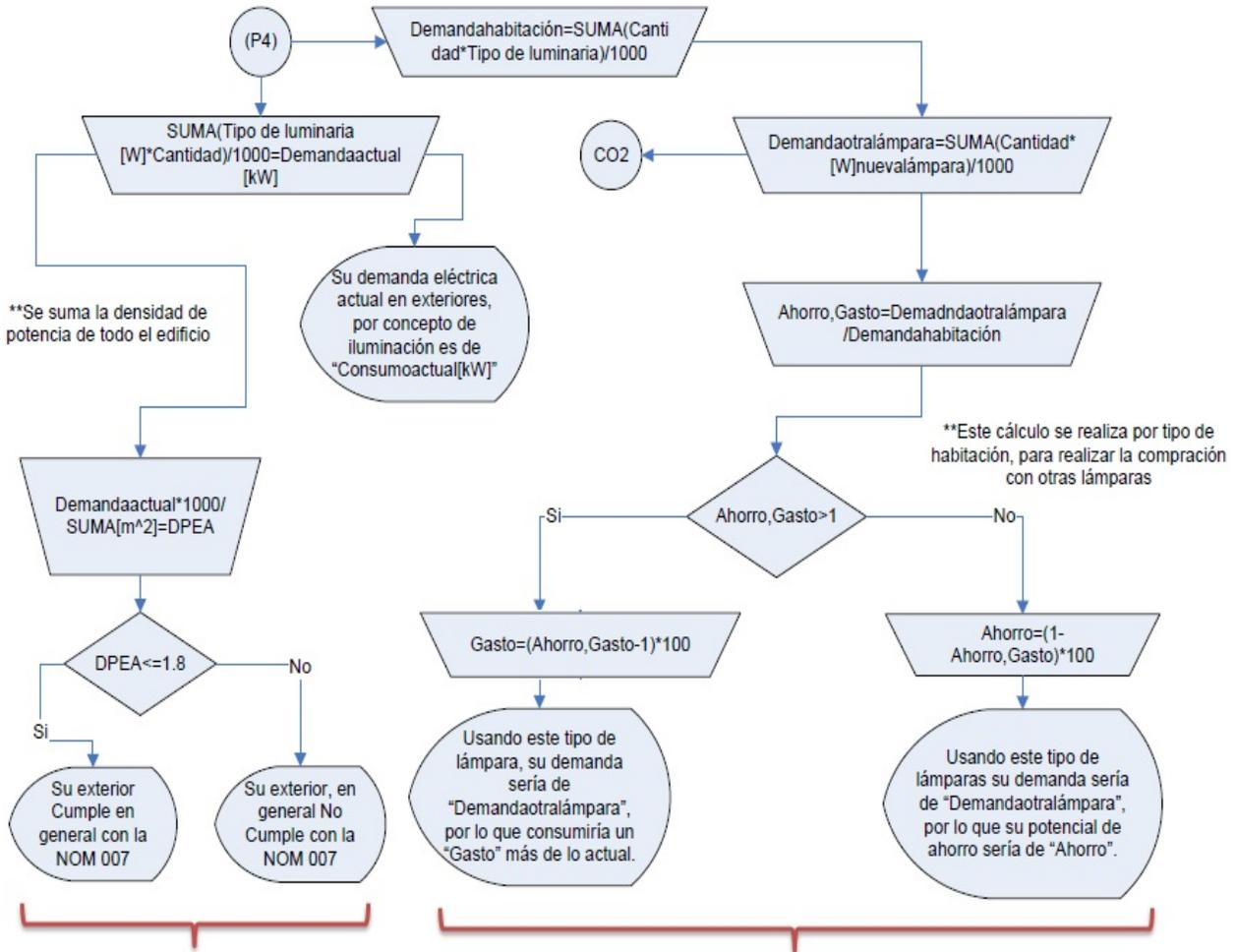
V.2.3 Iluminación Exterior

- **Diagrama de flujo:** Representación del proceso por el que pasa el diagnóstico de Iluminación exterior para el cálculo del consumo eléctrico y emisiones de CO₂. Las evaluaciones energéticas en el área de iluminación contemplan las normas mexicanas descritas en la tabla V.1 con las cuales se realizan los siguientes cálculos y evaluaciones para las áreas exteriores:
 - ❖ Selección del ambiente a tratar.
 - ❖ Ingreso o petición de datos para el diagnóstico.
 - ❖ Cálculo de la densidad de potencia de la luminaria seleccionada.
 - ❖ Evaluación de la NOM007 para exteriores.
 - ❖ Cumplimiento de la NOM007.
 - ❖ Cálculo de la densidad de potencia para la NOM013 y la NOM007.
 - ❖ Evaluación de la NOM013 a partir del área se consideran los rangos de los límites.
 - ❖ Rendimiento de las diferentes lámparas exteriores.
 - ❖ Evaluación del nivel de iluminación.
 - ❖ Evaluación de la NOM025.
 - ❖ Suma de la carga energética generada por las luminarias para exteriores y evaluación de la NOM007.
 - ❖ Cálculos para generar el ahorro de consumo sobre lo demandado al cambiar la luminaria.
 - ❖ Cálculo para las emisiones de CO₂.
 - ❖ Suma de cargas eléctricas exteriores para el hotel.



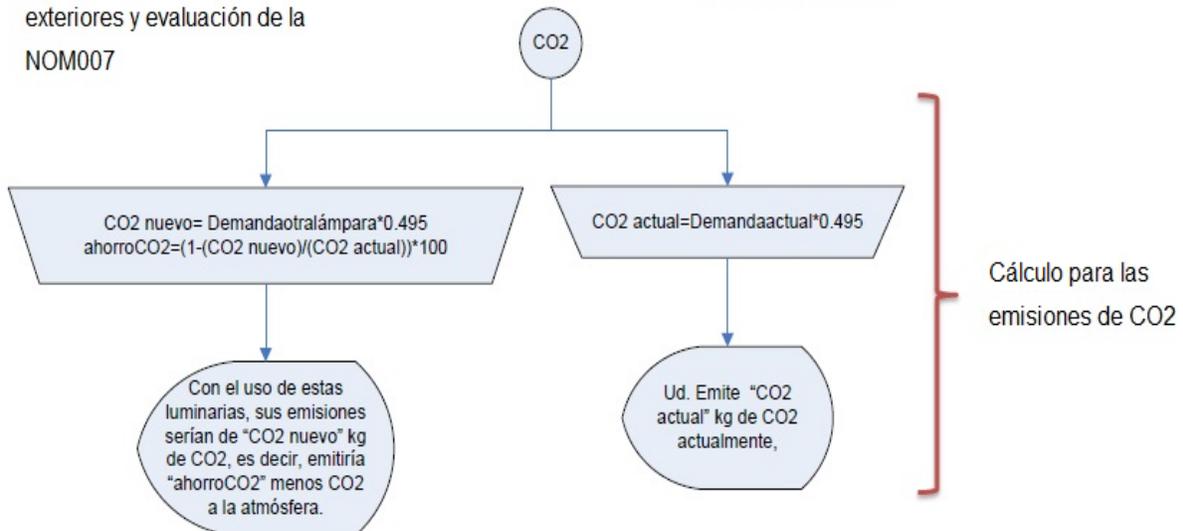






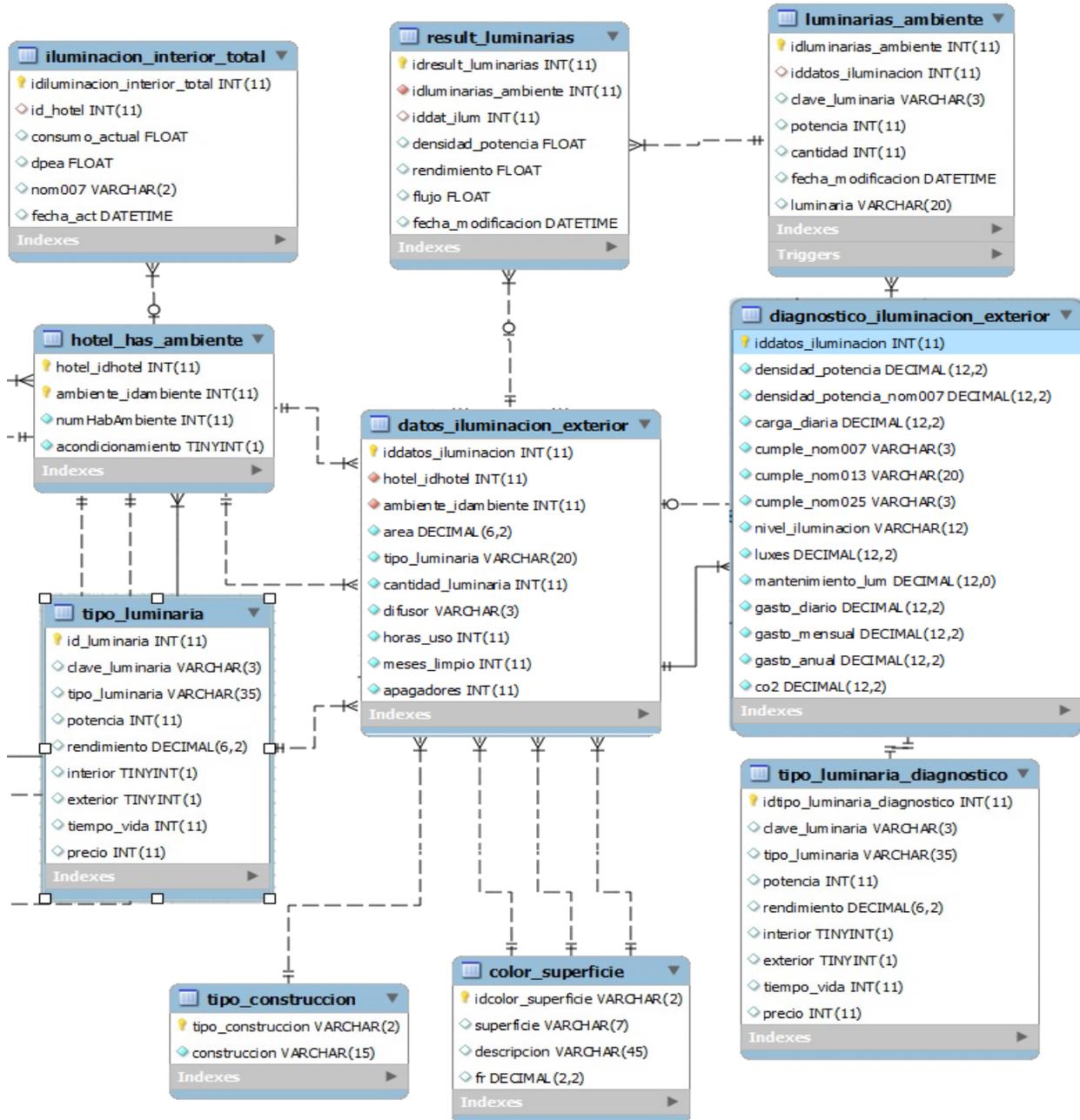
Suma de la carga energética generada por las luminarias para exteriores y evaluación de la NOM007

Cálculos para generar el ahorro de consumo sobre lo demandado al cambiar la luminaria



Cálculo para las emisiones de CO2

- **Diagrama entidad relación:** Fragmento del DER de la base de datos para el área de iluminación exterior en el que se apoya para almacenar los datos ingresados y con ayuda de los catálogos generar los cálculos para su representación, almacenamiento y futura utilización en las recomendaciones del diagnóstico.



V.3 Código de Desarrollo

Una vez mostrado el proceso por el cual se rige el sistema y los algoritmos de iluminación interior, iluminación exterior y refrigeración podemos iniciar con el código de desarrollo del cual solo se mostraran las partes más importantes para facilitar el entendimiento y comprensión del mismo.

El funcionamiento del ambiente de trabajo dentro de la nube son cuestiones técnicas que se refiere a la configuración del servidor web y las extensiones para Apache, Yii, PHP y MySQL que por fines prácticos no se analizará el código necesario, ya que, para su configuración existe documentación adecuada en las respectivas páginas oficiales. Tampoco se mostrara la creación y configuración de la base de datos ya que es un proceso trivial y sencillo de implementar teniendo como referencia los modelos relacionales antes mostrados.

Como ya habíamos mencionado con anterioridad el framework Yii se basa en el Modelo Vista Controlador (MVC) por lo tanto para cada área dentro del sistema se tiene un modelo, una vista y un controlador donde un modelo; es representado por una clase o clases que hacen referencia a objetos con los que se accede a la base de datos, una vista; es representada por la interfaz que se le presenta al usuario por medio de los elementos tales como etiquetas de texto y entradas de formularios y de un controlador; que es una clase que gestiona las peticiones de usuario y es encargado de crear las instancias de los modelos y generar las vistas.

A continuación se muestran las partes más significativa de los modelos, vistas y controladores de cada área:

V.3.1 Estructura Básica General del Sistema (Template)

Para que el usuario pueda interactuar con el sistema se necesita una interface que permita esta comunicación la cual está controlada por “layouts” que son esquemas de distribución de los elementos dentro un diseño, en este caso, el esquema o diseño general está regido por un encabezado, una columna, una barra horizontal, un pie de página y un área de trabajo donde se realizaran los procesos. El siguiente código muestra la generación de este primer “template” antes de ingresar al sistema:

Barra horizontal: Muestra los enlaces principales del sistema y el estado del usuario es decir si esta autenticado o no.

```
<div id="mainmenu">
  <?php $this->widget('zii.widgets.CMenu',array(
    'items'=>array(
      array('label'=>'Inicio', 'url'=>array('/site/index')),
      array('label'=>'SIMAE', 'url'=>array('/site/acceso'), 'visible'=>Yii::app()->user->getState(
'activeMenuElements')),
      array('label'=>'Noticias', 'url'=>array('/site/construccion', 'view'=>'construccion'), 'active'=>false,
        'visible'=>Yii::app()->user->getState('activeMenuElements')),
      array('label'=>'Salir ('.Yii::app()->user->name.'), 'url'=>array('/site/logout'), 'visible'=>!Yii::app
()->user->isGuest)
    ),
  )); ?>
</div><!-- mainmenu -->
```

Área de trabajo: Es un área que se actualiza dependiendo del proceso en el que se encuentre el usuario.

```
<?php if(isset($this->breadcrumbs)):?>
  <?php $this->widget('zii.widgets.CBreadcrumbs', array(
    'links'=>$this->breadcrumbs,
  )); ?><!-- breadcrumbs -->
<?php endif?>

<?php echo $content; ?>

<div class="clear"></div>
```

Pie de página: Muestra los derechos reservados y el año de gestión.

```
<div id="footer">
  Copyright &copy; <?php echo date('Y'); ?> por CCADET - UNAM.<br/>
  Todos los derechos reservados.<br/>
</div><!-- footer -->
```

Encabezado: Muestra logos importantes y una presentación en Flash como parte general de la presentación.

```
<div id="header">
<div class="span-14" id="logos">
  
  <map name="logos_map" id="logos_map">
    <area shape="rect" coords="2,18,80,84" href="http://www.conacyt.gob.mx" target="new"
      alt="Consejo Nacional de Ciencia y Tecnología" />
    <area shape="rect" coords="98,19,237,80" href="http://www.sener.gob.mx" target="new"
      alt="Secretaría de Energía" />
    <area shape="rect" coords="263,13,330,85" href="http://www.unam.mx" target="new"
      alt="Universidad Nacional Autónoma de México" />
  </map>
</div>
```

```

        <area shape="rect" coords="367,13,426,82" href="http://www.uacam.mx" target="new"
            alt="Universidad Autónoma de Campeche" />
        <area shape="rect" coords="467,13,526,83" href="http://www.uv.mx" target="new"
            alt="Universidad Veracruzana" />
    </map>
</div>
<div class="span-10 last" id="flash-logo">
    <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" width="100%" height="100" id="FlashID"
        title="simae-logo">
        <param name="movie" value="<?php echo Yii::app()->request->baseUri; ?>/images/titulo_small.swf" />
        <param name="quality" value="high" />
        <param name="wmode" value="opaque" />
        <param name="swfversion" value="6.0.65.0" />
        <param name="scale" value="exactfit" />
        <!-- This param tag prompts users with Flash Player 6.0 r65 and higher to download the latest version of
Flash Player. Delete it if you don't want users to see the prompt. -->
        <param name="expressinstall" value="/Scripts/expressInstall.swf" />
        <!-- Next object tag is for non-IE browsers. So hide it from IE using IECC. -->
        <!--[if !IE]>-->
        <object type="application/x-shockwave-flash"
            data="<?php echo Yii::app()->request->baseUri; ?>/images/titulo_small.swf" width="100%" height="100">
            <!--<![endif]-->
            <param name="quality" value="high" />
            <param name="wmode" value="opaque" />
            <param name="swfversion" value="6.0.65.0" />
            <param name="expressinstall" value="/Scripts/expressInstall.swf" />
            <param name="scale" value="exactfit" />
            <!-- The browser displays the following alternative content for users with Flash Player 6.0 and older. -->
            <div>
                <h4>Content on this page requires a newer version of Adobe Flash Player.</h4>
                <p><a href="http://www.adobe.com/go/getflashplayer"><img src=
"http://www.adobe.com/images/shared/download_buttons/get_flash_player.gif" alt="Get Adobe Flash player" width="112"
height="33" /></a></p>
            </div>
            <!--[if !IE]>-->
            </object>
        <!--<![endif]-->
        </object>
    </div><!-- flash-logo -->
</div><div class="clear"></div>
</div><!-- header -->

```

El área de trabajo de este "template" se utilizara para mostrar la información principal como presentación e información para el usuario y con esto se tiene la página principal del sistema antes de autenticarse:

Información principal (Index.php)

```

<?php
    $this->pageTitle=Yii::app()->name.' - Sistema Multiagente para la Gestión de la Eficiencia Energética
en Hoteles';
?>

<div class="span-18 prepend-1" id="content-container">
    <div class="span-8 colborder">
        <div>
            <h4 class="rojo-sener">OBJETIVO</h4>
            <p>Sistema Web para la Gestión de la Eficiencia Energética en Hoteles.
                Que busca mejorar el consumo y eficiencia energética en el sector hotelero.
            </p>
        </div>
        <hr />
        <div>
            <h4 class="rojo-sener">ORGANISMOS</h4>
            <ul>
                <li>UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO</li>
                <li>UNIVERSIDAD AUTÓNOMA DE CAMPECHE</li>
                <li>UNIVERSIDAD VERACRUZANA</li>
                <li>FONDO SECTORIAL</li>
            </ul>
        </div>
        <hr />
        <div>
            <h4 class="rojo-sener">CONTACTO</h4>
            <p>Sección que incluirá información de contacto, para el envío de comentarios y sugerencias sobre
el uso del sistema, así como el reporte de
posibles errores que puedan existir.
            </p>
        </div>
    </div>

    <div class="span-9 last">
        <h4 class="rojo-sener">NOTICIAS SOBRE AHORRO DE ENERGÍA EN HOTELES</h4>
        <div>
            <p>Sección dedicada a la publicación de experiencias e implementaciones que han tenido diversos
hoteles con el sistema.
                Se presentan casos de éxito, en donde se resumen las medidas de ahorro más significativas que
el sistema ha
                propuesto para las diversas áreas con las que cuenta el hotel.
            </p>
        </div>
        <div>
            <p>Apartados adicionales para la presentación de los reportes más significativos efectuados en
otros hoteles,
                que de igual manera han sido
                beneficiados, gracias a la utilización de los diversos módulos para la gestión de la
eficiencia energética
                que implementa el sistema.
            </p>
        </div>
    </div>
</div>

```

V.3.2 Módulo de Registro, Autenticación y Validación

Para ingresar al sistema se necesita registrarse y así autenticarse por lo que la autenticación de usuarios requiere de las funciones principales como acceder, salir y autenticar las cuales están contenidas en el controlador del sitio llamado “siteController”; este controlador se encarga de gestionar el sitio de forma general para mostrar en el área de trabajo del “template” los procesos o peticiones realizadas, el código es el siguiente:

```
<?php

class SiteController extends Controller
{
    public function filters() {
        return array(
            'checkLogin + acceso login',
        );
    }
    public function accessRules()
    {
        return array(
            array('allow',
                'actions'=>array('index', 'construccion', 'error'),
                'users'=>array('*'),
            ),
            array('allow',
                'actions'=>array('logout'),
                'users'=>array('@'),
            ),
            array('allow',
                'actions'=>array('acceso', 'login'),
                'users'=>array('?'),
            ),
            array('deny',
                'users'=>array('*'),
            ),
        );
    }
    public function actions()
    {
        return array(
            'captcha'=>array(
                'class'=>'CCaptchaAction',
                'backColor'=>0xFFFF,
            ),
            'construccion'=>array(
                'class'=>'CViewAction',
            ),
        );
    }
}
```

Reglas de acceso para el controlador de usuario

Acciones para llamar a la vista, obtener la información y el color de fondo

```

public function actionContact()
{
    $model=new ContactForm;
    if(isset($_POST['ContactForm']))
    {
        $model->attributes=$_POST['ContactForm'];
        if($model->validate())
        {
            $headers="From: {$model->email}\r\nReply-To: {$model->email}";
            mail(Yii::app()->params['adminEmail'],$model->subject,$model->body,$headers);
            Yii::app()->user->setFlash('contact','Thank you for contacting us. We will respond to
you as soon as possible.');
```

}
 }
 \$this->render('contact',array('model'=>\$model));
}
public function actionLogin()
{
 \$model=new LoginForm;

 // if it is ajax validation request
 if(isset(\$_POST['ajax']) && \$_POST['ajax']==='login-form')
 {
 echo CActiveForm::validate(\$model);
 Yii::app()->end();
 }

 // collect user input data
 if(isset(\$_POST['LoginForm']))
 {
 \$model->attributes=\$_POST['LoginForm'];

 if(\$model->validate() && \$model->login())
 \$this->redirect(array('entry/index'));
 }
 // display the login form
 \$this->render('login',array('model'=>\$model));
}
public function actionLogout()
{
 Yii::app()->user->logout();
 \$this->redirect(Yii::app()->homeUrl);
}
public function actionAcceso() {
 \$this->render('acceso');
}
public function filterCheckLogin(\$filterChain) {
 if (!Yii::app()->user->isGuest) {
 \$this->redirect(array('entry/index'));
 }
 \$filterChain->run();
}
}
}

Validación al contactar al administrador por medio del correo electrónico

Autenticación al sistema donde se valida atributos y se redirecciona a la página correspondiente

Salida del sistema donde se valida el cierre de sesión

Al ingresar al sistema se tiene la opción de registrarse o autenticarse lo cual se logra por medio de la siguiente vista:

```

<?php
    $this->pageTitle=Yii::app()->name.' - Acceso!;
?>

<div class="span-23 prepend-1 last">
    <h4 class="rojo-sener">ACCESO AL SISTEMA</h4>
    <p>
        Para usar el Sistema Multiagente para la Gestión de la Eficiencia Energética en Hoteles
        (SIMAE), debes acceder al sistema, hazlo ahora:
    </p>
    <div class="span-10" style="text-align: center;">
        <?php
            $baseUrl = Yii::app()->request->baseUrl;
            $imageIdNewUsr = 'nuevo_usr';
            $imgHover = "{$baseUrl}/images/nuevoUsr_hover_btn.gif";
            $imgNormal = "{$baseUrl}/images/nuevoUsr_btn.gif";
            $img = '';

            echo CHtml::link($img, array('/usuario/create'), array(
                'id' => 'go_newUsr',
            ));

            Yii::app()->clientScript->registerScript('change_newUsr',
                '$("#go_newUsr").mouseenter(function() {
                    $("#'. $imageIdNewUsr. ').attr("src", "'. $imgHover. '");
                });
                $("#go_newUsr").mouseleave(function() {
                    $("#'. $imageIdNewUsr. ').attr("src", "'. $imgNormal. '");
                });'
            );
        ?>
    </div>
    <div class="span-10" style="text-align: center;">
        <?php
            $imageIdExstUsr = 'existente_usr';
            $imgHover_exstUsr = "{$baseUrl}/images/existenteUsr_hover_btn.gif";
            $imgNormal_exstUsr = "{$baseUrl}/images/existenteUsr_btn.gif";
            $img_exstUsr = '';

            echo CHtml::link($img_exstUsr, array('/site/login'), array(
                'id' => 'go_exstUsr',
            ));

            Yii::app()->clientScript->registerScript('change_exstUsr',
                '$("#go_exstUsr").mouseenter(function() {
                    $("#'. $imageIdExstUsr. ').attr("src", "'. $imgHover_exstUsr. '");
                });
                $("#go_exstUsr").mouseleave(function() {
                    $("#'. $imageIdExstUsr. ').attr("src", "'. $imgNormal_exstUsr. '");
                });'
            );
        ?>
    </div>
</div>

```

Registros

Al registrar un usuario se pide información general acerca de él como nombre, teléfono, alias, etc. y para procesar la información se usan los siguientes controladores, modelos y vistas:

Controlador: “usuarioController”; Por medio de este se pueden crear, actualizar y eliminar los usuarios dentro del sistema.

```
<?php
class UsuarioController extends Controller
{
    public function filters()
    {
        return array(
            'accessControl',
        );
    }
    public function accessRules()
    {
        return array(
            array('allow',
                'actions'=>array('index','view'),
                'users'=>array('*'),
            ),
            array('allow',
                'actions'=>array('create'),
                'users'=>array('?'),
            ),
            array('allow',
                'actions'=>array('admin','delete'),
                'users'=>array('admin'),
            ),
            array('deny',
                'users'=>array('*'),
            ),
        );
    }
    public function actionView($id)
    {
        $this->render('view',array(
            'model'=>$this->loadModel($id),
        ));
    }
    public function actionCreate()
    {
        $model=new Usuario;
        if(isset($_POST['Usuario']))
        {
            $model->attributes=$_POST['Usuario'];
            if($model->save()) {
                $this->redirect(array('hotel/create'));
            }
        }
        $this->render('create',array(
            'model'=>$model,
        ));
    }
}
```

Reglas de acceso para el control de usuario

Creación de un usuario

```

public function actionUpdate($id)
{
    $model=$this->loadModel($id);
    if(isset($_POST['Usuario']))
    {
        $model->attributes=$_POST['Usuario'];
        if($model->save())
            $this->redirect(array('view','id'=>$model->id));
    }

    $this->render('update',array(
        'model'=>$model,
    ));
}

public function actionDelete($id)
{
    if(Yii::app()->request->isPostRequest)
    {
        // we only allow deletion via POST request
        $this->loadModel($id)->delete();

        if(!isset($_GET['ajax']))
            $this->redirect(isset($_POST['returnUrl']) ? $_POST['returnUrl'] : array('admin'));
    }
    else
        throw new CHttpException(400,'Petición inválida. Favor de no efectuar la petición nuevamente.');
```

Actualización de un usuario

Borrado de un Usuario

```

}

public function actionIndex()
{
    $dataProvider=new CActiveDataProvider('Usuario');
    $this->render('index',array(
        'dataProvider'=>$dataProvider,
    ));
}

public function actionAdmin()
{
    $model=new Usuario('search');
    $model->unsetAttributes(); // clear any default values

    if(isset($_GET['Usuario']))
        $model->attributes=$_GET['Usuario'];

    $this->render('admin',array(
        'model'=>$model,
    ));
}

public function loadModel($id)
{
    $model=Usuario::model()->findByPk($id);
    if($model===null)
        throw new CHttpException(404,'La página solicitada no existe.');
```

Funciones de administrador, búsqueda y redireccionamiento de los usuarios

```

    return $model;
}

```

Modelo: “Usuario”; en este modelo se realizan las peticiones a la base de datos y el cifrado de las contraseñas además de relacionar el usuario con el hotel.

```
class Usuario extends CActiveRecord
{
    const NOTHING_STARTED = 0;
    const ENCUESTA_AMBIENTES_PROCESADO = 1;
    const ENCUESTA_TARIFAS_PROCESADO = 2;
    const DIAGNOSTICO_ILUM_INT_PROCESADO = 1;
    const DIAGNOSTICO_ILUM_EXT_PROCESADO = 2;
    const DIAGNOSTICO_AA_PROCESADO = 3;
    const DIAGNOSTICO_CALENTAMIENTO_A_PROCESADO = 4;
    const DIAGNOSTICO_REFRIGERACION_PROCESADO = 5;
    const DIAGNOSTICO_ELEVADORES_PROCESADO = 6;
    const DIAGNOSTICO_BOMBEO_PROCESADO = 7;
    const DIAGNOSTICO_SUBESTACIONES_PROCESADO = 8;
    const DIAGNOSTICO_SIST_PASIVOS_PROCESADO = 9;

    public function rules()
    {
        return array(
            array('nombre, id_usuario, password, password_repeat, ife', 'required'),
            array('nombre', 'length', 'max'=>50),
            array('id_usuario', 'length', 'max'=>20),
            array('id_usuario', 'email', 'unique'),
            array('password', 'length', 'max'=>20),
            array('password', 'compare'),
            array('ife', 'length', 'max'=>25),
            array('email', 'length', 'max'=>35),
            array('email', 'email'),
            array('hashed_password, progress_encuesta, progress_diagnostico', 'safe'),
            // The following rule is used by search().
            // Please remove those attributes that should not be searched.
            array('id, nombre, password, ife, email', 'safe', 'on'=>'search'),
        );
    }

    public function relations()
    {
        return array(
            'hotel' => array(self::HAS_ONE, 'Hotel', 'idhotel'),
        );
    }

    public function attributeLabels()
    {
        return array(
            'nombre' => 'Nombre',
            'id_usuario' => 'Id de Usuario',
            'password' => 'Contraseña',
            'password_repeat' => 'Vuelva a escribir su contraseña',
            'ife' => 'Ife',
            'email' => 'E-mail',
        );
    }
}
```

Variables constantes para guardar un estatus de acceso

Reglas y características del modelo del usuario

Relaciones entre el usuario y el hotel además del etiquetado de los tributos

```

public function search()
{
    $criteria=new CDbCriteria;

    $criteria->compare('id',$this->id);
    $criteria->compare('nombre',$this->nombre,true);
    $criteria->compare('password',$this->password,true);
    $criteria->compare('ife',$this->ife,true);
    $criteria->compare('email',$this->email,true);

    return new CActiveDataProvider($this, array(
        'criteria'=>$criteria,
    ));
}

protected function beforeSave() {
    parent::beforeSave();
    if ($this->password && $this->isNewRecord) {
        $this->id_usuario = strtolower($this->id_usuario);
        $this->hashed_password = $this->hash_password
            ($this->password);

        return true;
    }
    return true;
}

private function hash_password($password) {
    $salt = bin2hex(mcrypt_create_iv(32, MCRYPT_DEV_URANDOM));
    $hash = hash('sha256', $salt . $password);

    return $salt . $hash;
}

public function afterSave() {
    parent::afterSave();
    if ($this->isNewRecord) {
        $loginObj = new LoginForm;
        $loginObj->username = $this->id_usuario;
        $loginObj->password = $this->password;

        $loginObj->login();
    }
}

public function validate_password($password) {
    $salt = substr($this->hashed_password, 0, 64);
    $hash = substr($this->hashed_password, 64, 64);

    $password_hash = hash('sha256', $salt . $password);

    return $password_hash == $hash;
}

```

Funcion con los criterios de busqueda de un usuario:

Verificación de la contraseña, el id de usuario y el cifrado de la contraseña antes de guardar la información.

Cifrado de la contraseña con un hash y un salt

Despues de guardar los datos se le asigna un id al usuario y su contraseña

Validación de la contraseña por medio de la salt y el hash

En criptografía, la “Salt” comprende bits aleatorios que son usados como una de las entradas en una función derivadora de claves. La otra entrada es habitualmente una contraseña. La salida de la función derivadora de claves se almacena como la versión cifrada de la contraseña. La “Salt” puede también ser usada como parte de una clave en un cifrado u otro algoritmo criptográfico. La función de derivación de claves generalmente usa una función hash. Esto se describe en el código anterior.

Un “Salt” criptográfico es un dato que se utiliza durante el proceso de hash para eliminar la posibilidad de que el resultado pueda buscarse a partir de una lista de pares precalculados de hash y sus entradas originales, conocidas como tablas “rainbow”.

Es decir, un “Salt” es un pequeño dato añadido que hace que los hash sean significativamente más difíciles de “crackear”. Existe un gran número de servicios online que ofrecen grandes listas de códigos hash precalculados, junto con sus datos de entrada originales. El uso de “salt” hace muy difícil o imposible encontrar el hash resultante en cualquiera de estas listas.

Para mayor seguridad, el valor de “Salt” se guarda en secreto, separado de la base de datos de contraseñas. Esto aporta una gran ventaja cuando la base de datos es robada, pero la “Salt” no. El beneficio aportado por usar una contraseña con “Salt” es que un ataque simple de diccionario contra los valores cifrados es impracticable si la “Salt” es lo suficientemente larga. En este modelo, la contraseña es cifrada y almacenada en la base de datos con el algoritmo de Hash seguro de 256 bits.

El nivel de seguridad y control es gestionado por un componente del framework Yii denominado ‘authManager’, que nos permitirá aplicar un algoritmo de cifrado para validar los usuarios. El modelo de validación se implementa a través de la clase ‘CUserIdentity’ como sigue:

```
class UserIdentity extends CUserIdentity
{
    private $_id;

    public function authenticate()
    {
        $username = strtolower($this->username);
        $pass = $this->password;
        $user = Usuario::model()->findByAttributes(array('id_usuario'=>$username));

        if ($user == null) {
            $this->errorCode = self::ERROR_UNKNOWN_IDENTITY;
        } else if ($user->validate_password($pass)) {
            $this->_id = $user->id;
            $this->username = $user->id_usuario;
            $this->errorCode = self::ERROR_NONE;
        }

        return $this->errorCode === self::ERROR_NONE;
    }

    public function getId() {
        return $this->_id;
    }
}
```

Vista: “form”; Esta vista muestra la interfaz para el llenado de datos del usuario donde se presenta un formulario donde pide los datos generales del usuario.

```

<div class="form">

<?php $form=$this->beginWidget('CActiveForm', array(
    'id'=>'usuario-form',
    'enableAjaxValidation'=>false,
)); ?>

<p class="note">Los campos marcados con <span class="required">*</span> son obligatorios.</p>

<?php echo $form->errorSummary($model); ?>

<div class="row">
    <?php echo $form->labelEx($model,'nombre'); ?>
    <?php echo $form->textField($model,'nombre',array('size'=>40,'maxlength'=>40)); ?>
    <?php echo $form->error($model,'nombre'); ?>
</div>

<div class="row">
    <?php echo $form->labelEx($model,'id_usuario'); ?>
    <?php echo $form->textField($model,'id_usuario',array('size'=>20,'maxlength'=>20)); ?>
    <?php echo $form->error($model,'id_usuario'); ?>
</div>

<div class="row">
    <?php echo $form->labelEx($model,'password'); ?>
    <?php echo $form->passwordField($model,'password',array('size'=>20,'maxlength'=>20)); ?>
    <?php echo $form->error($model,'password'); ?>
</div>

<div class="row">
    <?php echo $form->labelEx($model,'password_repeat'); ?>
    <?php echo $form->passwordField($model,'password_repeat',array('size'=>20,'maxlength'=>20)); ?>
    <?php echo $form->error($model,'password_repeat'); ?>
</div>

<div class="row">
    <?php echo $form->labelEx($model,'ife'); ?>
    <?php echo $form->textField($model,'ife',array('size'=>25,'maxlength'=>25)); ?>
    <?php echo $form->error($model,'ife'); ?>
</div>

<div class="row">
    <?php echo $form->labelEx($model,'email'); ?>
    <?php echo $form->textField($model,'email',array('size'=>35,'maxlength'=>35)); ?>
    <?php echo $form->error($model,'email'); ?>
</div>

<div class="row buttons">
    <?php echo CHtml::submitButton($model->isNewRecord ? 'Registrar' : 'Guardar'); ?>
</div>
<?php $this->endWidget(); ?>

</div><!-- form -->

```

Al pulsar el botón de envío de formulario “Guardar” se ejecuten las instrucciones de validación, las cuales, siendo validadas satisfactoriamente, serán pasadas al modelo que se encargará de introducir los datos en la base de datos. Si la petición fue realizada satisfactoriamente, es decir, si los datos fueron introducidos en la base de datos, y no se encuentra un usuario registrado con el mismo nombre, se procede al registro del hotel por medio del siguiente controlador, modelo y vista:

Controlador: “hotelController”; este controlador se encarga de crear, borrar o actualizar la información correspondiente a un hotel el cual está ligado a un usuario en particular y llama un Ajax para validar al usuario y el hotel.

```

<?php
class HotelController extends Controller
{
    public function filters()
    {
        return array(
            'accessControl',
            array(
                'application.filters.UserHasHotel + create, update',
                'context' => 'hotel',
            ),
        );
    }
    public function accessRules()
    {
        return array(
            array('allow',
                'actions'=>array('index', 'view'),
                'users'=>array('*'),
            ),
            array('allow',
                'actions'=>array('create', 'update', 'getCiudades'),
                'users'=>array('@'),
            ),
            array('allow',
                'actions'=>array('admin', 'delete'),
                'users'=>array('admin'),
            ),
            array('deny',
                'users'=>array('*'),
            ),
        );
    }
    public function actionCreate()
    {
        $model=new Hotel;
        if(isset($_POST['Hotel']))
        {
            $model->attributes=$_POST['Hotel'];
            $model->area_terreno = (float)$_POST['Hotel']['area_terreno'];
            $model->area_construida = (float)$_POST['Hotel']['area_construida'];
            if($model->save())
                $this->redirect(array('entry/index'));
        }

        $this->render('create',array(
            'model'=>$model,
        ));
    }
}

```

Acciones de control del hotel; creación y actualización

Reglas de acceso y control de la creación, actualización, borrado y obtención de ciudades del hotel

Creación de un hotel con los atributos de área y terreno

```

public function actionUpdate($id)
{
    $model=$this->loadModel($id);
    if(isset($_POST['Hotel']))
    {
        $model->attributes=$_POST['Hotel'];
        if($model->save())
            $this->redirect(array('view','id'=>$model->idhotel));
    }

    $this->render('update',array(
        'model'=>$model,
    ));
}
public function actionDelete($id)
{
    if(Yii::app()->request->isPostRequest)
    {
        // we only allow deletion via POST request
        $this->loadModel($id)->delete();

        // if AJAX request (triggered by deletion via admin grid view), we should not redirect the browser
        if(!isset($_GET['ajax']))
            $this->redirect(isset($_POST['returnUrl']) ? $_POST['returnUrl'] : array('admin'));
    }
    else
        throw new CHttpException(400,'Invalid request. Please do not repeat this request again.');
```

Funciones de actualización y borrado de un hotel

```

}
public function actionIndex()
{
    $dataProvider=new CActiveDataProvider('Hotel');
    $this->render('index',array(
        'dataProvider'=>$dataProvider,
    ));
}
public function actionAdmin()
{
    $model=new Hotel('search');
    $model->unsetAttributes(); // clear any default values
    if(isset($_GET['Hotel']))
        $model->attributes=$_GET['Hotel'];

    $this->render('admin',array(
        'model'=>$model,
    ));
}
protected function performAjaxValidation($model)
{
    if(isset($_POST['ajax']) && $_POST['ajax']==='hotel-form')
    {
        echo CActiveForm::validate($model);
        Yii::app()->end();
    }
}
public function actionGetCiudades() {
    $ciudades = Ciudad::model()->findAll('idestado=:idestado',
        array(':idestado'=>(int) $_POST['Hotel']['idestado']));
    $ciudades = CHtml::listData($ciudades, 'idciudad', 'nombre');
    if (!$ciudades) {
        echo CHtml::tag('option', array('value'=>0), CHtml::encode('--Sin ciudades asociadas--'), true);
        return;
    }
    foreach ($ciudades as $value=>$name) {
        echo CHtml::tag('option', array('value'=>$value), CHtml::encode($name), true);
    }
}
}

```

Función de búsqueda de un hotel por sus atributos

Validación del hotel cuando se registra

Modelo: “Hotel”; este modelo procesa toda la información obtenida y realiza la interacción con la base de datos para sus futuras peticiones por medio del cual se obtiene el estado y dependiendo del estado se obtiene la ciudad y la categoría del hotel gracias a las funciones de jQuery usadas.

```

class Hotel extends CActiveRecord
{
    public static function model($className=__CLASS__)
    {
        return parent::model($className);
    }

    public function rules()
    {
        // NOTE: you should only define rules for those attributes that
        // will receive user inputs.
        return array(
            array('nombre, rfc, calle, colonia, delegacion, codigo_postal, telefono, idcategoria, idestado, idciudad',
                'required'),
            array('codigo_postal, telefono, numero_neveras, idcategoria, idestado, idciudad', 'numerical',
                'integerOnly'=>true),
            array('nombre', 'length', 'max'=>40),
            array('rfc', 'length', 'max'=>25),
            array('calle, pagina_web', 'length', 'max'=>45),
            array('colonia, delegacion', 'length', 'max'=>35),
            array('area_terreno, area_construida', 'length', 'max'=>7),
            array('zona_ubicacion', 'length', 'max'=>20),
            array('idhotel, numero_habitaciones, numero_ambientes', 'safe'),
            // The following rule is used by search().
            // Please remove those attributes that should not be searched.
            array('idhotel, nombre, rfc, calle, colonia, delegacion, codigo_postal, telefono, pagina_web,
                area_terreno, area_construida, numero_habitaciones, numero_ambientes, numero_neveras, idcategoria, idestado, idciudad,
                zona_ubicacion', 'safe', 'on'=>'search'),
        );
    }

    public function relations()
    {
        return array(
            'owner' => array(self::BELONGS_TO, 'Usuario', 'idhotel'),
            'categoria' => array(self::BELONGS_TO, 'Categoria', 'idcategoria'),
            'ciudad' => array(self::BELONGS_TO, 'Ciudad', 'idciudad'),
            'estado' => array(self::BELONGS_TO, 'Estado', 'idestado'),
            'ambientes' => array(self::MANY_MANY, 'Ambiente', '{{hotel_has_ambiente}}(hotel_idhotel,
                ambiente_idambiente)'),
            'puertas_ventanas_exteriores' => array(self::HAS_MANY, 'PuertaVentanaExterior', 'idhotel'),
            'superficies_exteriores' => array(self::HAS_MANY, 'SuperficieExterior', 'idhotel'),
        );
    }
}

```

```

public function search()
{
    $criteria=new CDbCriteria;
    $criteria->compare('idhotel',$this->idhotel);
    $criteria->compare('nombre',$this->nombre,true);
    $criteria->compare('rfc',$this->rfc,true);
    $criteria->compare('calle',$this->calle,true);
    $criteria->compare('colonia',$this->colonia,true);
    $criteria->compare('delegacion',$this->delegacion,true);
    $criteria->compare('codigo_postal',$this->codigo_postal);
    $criteria->compare('telefono',$this->telefono);
    $criteria->compare('pagina_web',$this->pagina_web,true);
    $criteria->compare('area_terreno',$this->area_terreno,true);
    $criteria->compare('area_construida',$this->area_construida,true);
    $criteria->compare('numero_habitaciones',$this->numero_habitaciones);
    $criteria->compare('numero_ambientes',$this->numero_ambientes);
    $criteria->compare('numero_neveras',$this->numero_neveras);
    $criteria->compare('idcategoria',$this->idcategoria);
    $criteria->compare('idestado',$this->idestado);
    $criteria->compare('idciudad',$this->idciudad);
    $criteria->compare('zona_ubicacion',$this->zona_ubicacion,true);
    return new CActiveDataProvider($this, array(
        'criteria'=>$criteria,
    ));
}

public function getCategorias() {
    $categorias = Categoria::model()->findAll();
    $categoriasArray = CHtml::listData($categorias, 'idcategoria',
    'categoria');
    return $categoriasArray;
}

public function getEstados() {
    $estados = Estado::model()->findAll();
    $estadosArray = CHtml::listData($estados, 'idestado', 'nombre')
    return $estadosArray;
}

public function getCiudades() {
    if (!$this->isNewRecord) {
        $ciudades = Ciudad::model()->findAllByAttributes(array('idestado'=>$this->idestado));
        $ciudades = CHtml::listData($ciudades, 'idciudad', 'nombre');
        return $ciudades;
    } else {
        return array();
    }
}

public function beforeSave() {
    parent::beforeSave();
    if ($this->isNewRecord) {
        $idUserario = Yii::app()->user->id;

        $this->idhotel = $idUserario;

        return true;
    }
    return true;
}

```

Atributos del hotel

Obtención de la categoría del hotel

Obtención del estado y ciudad donde se localiza el hotel

Antes de guardar asocia la información con el usuario

Vista: “form”; por medio de esta interface el usuario ingresa la información básica de un hotel como nombre, ubicación, área, etc.

```
<div class="form">

<?php $form=$this->beginWidget('CActiveForm', array(
    'id'=>'hotel-form',
    'enableAjaxValidation'=>false,
)); ?>

<?php if (Yii::app()->user->hasFlash('hotelRequired')): ?>
    <p>
        <?php echo Yii::app()->user->getFlash('hotelRequired'); ?>
    </p>
<?php endif; ?>

<p class="note">Los campos marcados con <span class="required">*</span> son obligatorios.</p>

<?php echo $form->errorSummary($model); ?>
<div class="row">
    <?php echo $form->labelEx($model, 'nombre'); ?>
    <?php echo $form->textField($model, 'nombre', array('size'=>40, 'maxlength'=>40)); ?>
    <?php echo $form->error($model, 'nombre'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model, 'rfc'); ?>
    <?php echo $form->textField($model, 'rfc', array('size'=>25, 'maxlength'=>25)); ?>
    <?php echo $form->error($model, 'rfc'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model, 'calle'); ?>
    <?php echo $form->textField($model, 'calle', array('size'=>45, 'maxlength'=>45)); ?>
    <?php echo $form->error($model, 'calle'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model, 'colonia'); ?>
    <?php echo $form->textField($model, 'colonia', array('size'=>35, 'maxlength'=>35)); ?>
    <?php echo $form->error($model, 'colonia'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model, 'delegacion'); ?>
    <?php echo $form->textField($model, 'delegacion', array('size'=>35, 'maxlength'=>35)); ?>
    <?php echo $form->error($model, 'delegacion'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model, 'codigo_postal'); ?>
    <?php echo $form->textField($model, 'codigo_postal'); ?>
    <?php echo $form->error($model, 'codigo_postal'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model, 'telefono'); ?>
    <?php echo $form->textField($model, 'telefono'); ?>
    <?php echo $form->error($model, 'telefono'); ?>
</div>
.. ..
```

```

<div class="row">
    <?php echo $form->labelEx($model,'pagina_web'); ?>
    <?php echo $form->urlField($model,'pagina_web',array('size'=>45,'maxlength'=>45)); ?>
    <?php echo $form->error($model,'pagina_web'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model,'area_terreno'); ?>
    <?php echo $form->textField($model,'area_terreno',array('size'=>7,'maxlength'=>7, 'value'=>'')); ?>
    <?php echo $form->error($model,'area_terreno'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model,'area_construida'); ?>
    <?php echo $form->textField($model,'area_construida',array('size'=>7,'maxlength'=>7,'value'=>'')); ?>
    <?php echo $form->error($model,'area_construida'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model,'numero_neveras'); ?>
    <?php echo $form->dropDownList($model,'numero_neveras', array(0=>'Ninguna',1=>1, 2=>2, 3=>3, 4=>4, 5=>5)); ?
    <?php echo $form->error($model,'numero_neveras'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model,'idcategoria'); ?>
    <?php echo $form->dropDownList($model,'idcategoria', $model->getCategorias()); ?>
    <?php echo $form->error($model,'idcategoria'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model,'idestado'); ?>
    <?php echo $form->dropDownList($model,'idestado', $model->getEstados(), array('ajax'=>array(
        'type' => 'POST',
        'url' => CController::createUrl('Hotel/getCiudades'),
        'update' => '#Hotel_idciudad',),
        'prompt' => '--Seleccione un estado--')); ?>
    <?php echo $form->error($model,'idestado'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model,'idciudad'); ?>
    <?php echo $form->dropDownList($model,'idciudad', $model->getCiudades(), array('prompt'=>'--Ciudades
asociadas al estado--')); ?>
    <?php echo $form->error($model,'idciudad'); ?>
</div>
<div class="row">
    <?php echo $form->labelEx($model,'zona_ubicacion'); ?>
    <?php echo $form->textField($model,'zona_ubicacion',array('size'=>20,'maxlength'=>20)); ?>
    <?php echo $form->error($model,'zona_ubicacion'); ?>
</div>
<div class="row buttons">
    <?php echo CHtml::submitButton($model->isNewRecord ? 'Create' : 'Save'); ?>
</div>
<?php $this->endWidget(); ?>

</div><!-- form -->

```

V.3.3 Módulo de Ingreso al Sistema y Encuesta

Una vez registrado el usuario y el hotel se ingresa al sistema por medio del usuario y contraseña creados anteriormente a través de la siguiente vista la cual contiene un widget que llama a la validación y uno para poder recordar al usuario y su contraseña y estar autenticado por un mes:

```

<?php
    $this->pageTitle = Yii::app()->name . ' - Acceso al Sistema';
?>

<div class="span-20 prepend-1 append-3 last">
    <h4 class="rojo-sener">CREDENCIALES DE ACCESO</h4>

    <p>Por favor ingrese los datos requeridos para acceder:</p>

    <div class="form">
        <?php $form=$this->beginWidget('CActiveForm', array(
            'id'=>'login-form',
            'enableClientValidation'=>false,
            'clientOptions'=>array(
                'validateOnSubmit'=>true,
            ),
        )); ?>

        <p class="note">Los campos marcados con <span class="required">*</span> son obligatorios.</p>

        <div class="row">
            <?php echo $form->labelEx($model,'username'); ?>
            <?php echo $form->textField($model,'username'); ?>
            <?php echo $form->error($model,'username'); ?>
        </div>

        <div class="row">
            <?php echo $form->labelEx($model,'password'); ?>
            <?php echo $form->passwordField($model,'password'); ?>
            <?php echo $form->error($model,'password'); ?>
        </div>

        <div class="row rememberMe">
            <?php echo $form->checkBox($model,'rememberMe'); ?>
            <?php echo $form->label($model,'rememberMe'); ?>
            <?php echo $form->error($model,'rememberMe'); ?>
        </div>

        <div class="row buttons">
            <?php echo CHtml::submitButton('Entrar'); ?>
        </div>

        <?php $this->endWidget(); ?>
    </div><!-- form -->
</div>

```

Widget de validación de acceso al sistema

Atributos obligatorios al registrar un usuario

Una vez dentro del sistema procedemos al llenado de la encuesta donde se recaba información general y particular de todas las áreas para realizar el diagnóstico energético de una forma rápida sin redundar en la petición de información en las diferentes áreas. El siguiente modelo, vista y controlador muestra este proceso:

Controlador: “encuestaController”; Gestiona los ambientes dentro del sistema para su utilización en las diferentes áreas como la creación y actualización de ambientes y su asociación con algún área dentro del sistema además crea las reglas de acceso y asocia el hotel, la encuesta y el usuario para así establecer los ambientes particulares de cada área.

```
<?php
class EncuestaController extends Controller
{
    public $layout = '/layouts/column2_buttons';

    public function filters()
    {
        // return the filter configuration for this controller, e.g.:
        return array(
            'accessControl',
            array(
                'application.filters.UserHasHotel',
                'context'=>'usuario',
            ),
            'checkAmbientes + createAmbientes',
        );
    }
    public function accessRules()
    {
        return array(
            array('allow', // allow authenticated users to perform 'index' action
                'actions'=>array('index', 'createAmbientes'),
                'users'=>array('@'),
            ),
            array('deny', // deny all users
                'users'=>array('*'),
            ),
        );
    }
}
```

```

public function actionIndex()
{
    $this->render('index');
}

public function actionCreateAmbientes() {
    $ambientes = Ambiente::model()->findAll();
    $ambientes = CHtml::listData($ambientes, 'idambiente', 'tipo');

    $selNumHab = $this->numberMenuSelection();
    $numHabTotales = 0;
    $numAmbientes = 0;

    if (isset($_POST['HotelHasAmbiente'])) {
        foreach ($ambientes as $idambiente=>$tipo) {
            if (isset($_POST['HotelHasAmbiente'][$idambiente]['ambiente_idambiente'])) {
                $numAmbientes++;
                $model = new HotelHasAmbiente;
                $model->attributes = $_POST['HotelHasAmbiente'][$idambiente];
                $numHabTotales = $model->numHabAmbiente;
                $model->save();
            }
        }

        if ($numAmbientes > 0) {
            $userId = Yii::app()->user->id;
            Hotel::model()->updateByPk($userId, array(
                'numero_ambientes'=>$numAmbientes,
                'numero_habitaciones'=>$numHabTotales,
            ));
            Usuario::model()->updateByPk($userId, array(
                'progress_encuesta'=>Usuario::ENCUESTA_AMBIENTES_PROCESADO,
            ));
            $this->redirect(array('encuesta/index'));
        }
    }

    $this->render('createAmbientes', array(
        'model'=>HotelHasAmbiente::model(),
        'ambientes'=>$ambientes,
        'selNumHab'=>$selNumHab,
    ));
}

public function numberMenuSelection() {
    $value_display = array(1 => 1);

    for ($i=2; $i <= 15; $i++) {
        $value_display[$i] = $i;
    }

    return $value_display;
}

public function filterCheckAmbientes($filterChain) {
    $user_id = Yii::app()->user->id;
    $hasAmbientes = HotelHasAmbiente::model()->findByAttributes(array('hotel_idhotel'=>$user_id));

    if ($hasAmbientes)
        $this->redirect(array('encuesta/index'));

    $filterChain->run();
}
}

```

Creación de ambientes con los atributos id y tipo

Asignación de números de ambientes y habitaciones

Muestra de la página para registrar ambientes

Registro del número de ambientes

Modelo: “HotelHasAmbiente”; este modelo asocia los ambientes y numero de espacios a un hotel y un usuario, donde hay dos funciones que clasifican cada ambiente para las áreas de iluminación interior y exterior con su respectiva información para así ser procesador por al área de iluminación además de agregar sí o no cuentan con acondicionamiento en al ambiente.

```

<?php
class HotelHasAmbiente extends CActiveRecord
{
    public static function model($className=__CLASS__)
    {
        return parent::model($className);
    }
    public function rules()
    {
        return array(
            array('ambiente_idambiente, numHabAmbiente, acondicionamiento', 'required'),
            array('ambiente_idambiente, numHabAmbiente, acondicionamiento', 'numerical', 'integerOnly'=>true),
            array('hotel_idhotel', 'safe'),
            array('hotel_idhotel, ambiente_idambiente, numHabAmbiente, acondicionamiento', 'safe', 'on'=>'search'),
        );
    }
    public function relations()
    {
        return array(
            'datosIluminaciones' => array(self::HAS_MANY, 'DatosIluminacion', 'ambiente_idambiente'),
            'datosIluminaciones1' => array(self::HAS_MANY, 'DatosIluminacion', 'hotel_idhotel'),
            'espacioHotels' => array(self::HAS_MANY, 'EspacioHotel', 'idhotel'),
            'espacioHotels1' => array(self::HAS_MANY, 'EspacioHotel', 'idambiente'),
            'iluminacionInteriorTotals' => array(self::HAS_MANY, 'IluminacionInteriorTotal', 'id_hotel'),
            'ambiente' => array(self::BELONGS_TO, 'Ambiente', 'ambiente_idambiente'),
        );
    }
    public function attributeLabels()
    {
        return array(
            'ambiente_idambiente' => 'Ambiente Idambiente',
            'numHabAmbiente' => 'Numero de Espacios',
            'acondicionamiento' => 'Acondicionamiento',
        );
    }
}

```

```

public function search()
{
    // Warning: Please modify the following code to remove attributes that
    // should not be searched.

    $criteria=new CDbCriteria;

    $criteria->compare('hotel_idhotel',$this->hotel_idhotel);
    $criteria->compare('ambiente_idambiente',$this->ambiente_idambiente);
    $criteria->compare('numHabAmbiente',$this->numHabAmbiente);
    $criteria->compare('acondicionamiento',$this->acondicionamiento);

    return new CActiveDataProvider($this, array(
        'criteria'=>$criteria,
    ));
}

public function beforeSave() {
    parent::beforeSave();
    if ($this->isNewRecord) {
        $this->hotel_idhotel = Yii::app()->user->id;
    }
    return true;
}

public function getAmbientesInteriores() {
    $conn = Yii::app()->db;
    $idhotel = Yii::app()->user->id;
    $sql = "SELECT a.tipo, hha.numHabAmbiente, hha.ambiente_idambiente, a.clasificacion FROM
{{hotel_has_ambiente}} hha
        JOIN {{ambiente}} a ON(hha.ambiente_idambiente = a.idambiente)
        WHERE hha.hotel_idhotel = $idhotel AND a.clasificacion IN ('I1', 'I2', 'I3', 'I4')";
    $command = $conn->createCommand($sql);
    $datosIllumInt = $command->queryAll();
    return $datosIllumInt;
}

public function getAmbientesExteriores() {
    $conn = Yii::app()->db;
    $idhotel = Yii::app()->user->id;
    $sql = "SELECT a.tipo, hha.numHabAmbiente, hha.ambiente_idambiente, a.clasificacion FROM
{{hotel_has_ambiente}} hha
        JOIN {{ambiente}} a ON(hha.ambiente_idambiente = a.idambiente)
        WHERE hha.hotel_idhotel = $idhotel AND a.clasificacion IN ('E1', 'E2')";
    $command = $conn->createCommand($sql);
    $datosIllumExt = $command->queryAll();
    return $datosIllumExt;
}
}

```

Criterios de comparación para la validación

Obtención de los ambientes interiores seleccionados para el hotel

Obtención de los ambientes exteriores seleccionados para el hotel

Vista: “Encuesta”; Esta vista muestra un Ajax que sirve para visualizar en cada ambiente seleccionado el número de espacios y si cuenta o no con acondicionamiento.

```
<?php
    $this->pageTitle=Yii::app()->name . ' - Registro de Ambientes';

    Yii::app()->clientScript->registerScript('toggle-elements-reg-ambientes',
    "$('.check-ambiente').click(function() {
        if ($(this).is(':checked')) {
            $(this).next('div').slideDown();
        } else {
            $(this).next('div').slideUp();
        }
    });"
    );
?>

<h3 class="rojo-sener">REGISTRO DE AMBIENTES</h3>

<p>
    Seleccione los ambientes con los que cuenta el hotel, así como el número de espacios para cada uno y si cuenta o
    desea acondicionamiento.
</p>

<?php
    $i=0;
    $j=2;
?>

<div class="form">
    <?php $form=$this->beginWidget('CActiveForm', array(
        'id'=>'registra-ambientes-form',
        'enableAjaxValidation'=>false,
    )); ?>

    <?php echo $form->errorSummary($model); ?>
    <?php foreach ($ambientes as $idambiente=>$tipo): ?>
        <?php
            $numHab = array();
            switch (strtolower($tipo)) {
                case 'lobby':
                case 'sala de recepcion':
                case 'bar':
                case 'spa':
                case 'gimnasio':
                case 'sala multiuso':
                case 'sala de internet':
                case 'jardin':
                case 'cancha deportiva':
                    $numHab = array(1=>1, 2, 3);
                    break;
```


V.3.4 Módulo de Diagnóstico Energético

Ya creado el usuario y asociado al hotel se continuó con el llenado de la encuesta donde se agregaron al menos un tipo de ambiente para el hotel, así como el número de espacios para cada uno y su acondicionamiento, así que, tenemos como nuevo punto de partida el módulo de diagnóstico energético que recopilará la mayoría de la información para cada área energética, con la que se realizará el análisis y generación de recomendaciones, evaluación e implementación de medidas de ahorro energético.

Debido a la gran extensión de los algoritmos y para lograr un muy buen entendimiento del diagnóstico solo se mostrara a detalle el área de Iluminación interior ya que las demás áreas siguen la misma estructura, es decir, cada área está basadas en su diagrama de flujo el cual se relaciona a su diagrama entidad relación para así asociarlo con la base de datos y generar su respectivo código, el cual, estará distribuido en un controlador, modelo y vista para su representación en el sistema.

Iluminación Interior

El módulo de Iluminación se divide en interior; el cuál se encarga de calcular las cargas eléctricas de habitaciones, oficinas, pasillo, estacionamiento cubierto, etc. y exterior; el cuál calcula las cargas eléctricas de alberca, estacionamientos, patios, etc. Una vez calculadas las cargas se genera un análisis para así poder determinar las recomendaciones por medio de los criterios establecidos por los expertos.

El siguiente modelo, vista y controlador generan el módulo de Iluminación interior:

Controlador: “**iluminacionInteriorController**”; éste controlador se encarga de mostrar y realizar los procesos que se llevan a cabo durante el diagnóstico energético como la clasificación de ambientes interiores, la adquisición de datos y el ciclo que se realiza para cada ambiente, el procesamiento de los datos y el análisis para obtener la

mejor recomendación la cual consiste en poca inversión y un ahorro económico significativo.

El siguiente código muestra la selección de los ambientes interiores ingresados en el hotel:

```
$conn = Yii::app()->db;
$sql = "SELECT a.tipo, hha.numHabAmbiente, hha.ambiente_idambiente, a.clasificacion FROM
{{hotel_has_ambiente}} hha
        JOIN {{ambiente}} a ON(hha.ambiente_idambiente = a.idambiente)
        WHERE hha.hotel_idhotel = $idhotel AND a.clasificacion IN ('I1', 'I2', 'I3', 'I4')";
$command = $conn->createCommand($sql);
$modelHasAmb = $command->queryAll();
```

Código para obtener el cálculo de la altura óptima de las luminarias:

```
if (($altura > 3.91) && $iluminacion == 'Directa')
    { $alturaOptima = round((4/5) * ($altura - 0.85), 2);}
elseif (($altura > 3.91) && $iluminacion == 'Semidirecta')
    { $alturaOptima = round((3/4) * ($altura - 0.85), 2);}
else
    { $alturaOptima = 2.40;}
```

Código para el cálculo del rendimiento de las luminarias con y sin difusor:

```
if ( $difusor == 'Si')
    {
        if (($mesesLimpio >= 0) && ($mesesLimpio < 1))
            { $renLumMantenimiento = 100;}
        elseif (($mesesLimpio >= 1) && ($mesesLimpio < 2))
            { $renLumMantenimiento = 95;}
        elseif (($mesesLimpio >= 2) && ($mesesLimpio < 5))
            { $renLumMantenimiento = 90;}
        elseif (($mesesLimpio >= 5) && ($mesesLimpio < 9))
            { $renLumMantenimiento = 85;}
        elseif (($mesesLimpio >= 9) && ($mesesLimpio < 16))
            { $renLumMantenimiento = 80;}
        elseif (($mesesLimpio >= 16) && ($mesesLimpio < 25))
            { $renLumMantenimiento = 75;}
        elseif ($mesesLimpio > 25)
            { $renLumMantenimiento = 70;}
    }
elseif ( $difusor == 'No')
    {
        if (($mesesLimpio >= 0) && ($mesesLimpio < 1))
            { $renLumMantenimiento = 100;}
        elseif (($mesesLimpio >= 1) && ($mesesLimpio < 4))
            { $renLumMantenimiento = 95;}
        elseif (($mesesLimpio >= 4) && ($mesesLimpio < 10))
            { $renLumMantenimiento = 90;}
        elseif (($mesesLimpio >= 10) && ($mesesLimpio < 19))
            { $renLumMantenimiento = 85;}
        elseif (($mesesLimpio >= 19) && ($mesesLimpio < 34))
            { $renLumMantenimiento = 80;}
        elseif ($mesesLimpio > 34)
            { $renLumMantenimiento = 75;}
    }
}
```

Código para la obtener la potencia, rendimiento y clave de la luminaria:

```
$conn = Yii::app()->db;
$sql = "SELECT potencia, rendimiento, clave_luminaria FROM {{tipo_luminaria_diagnostico}}
      WHERE idtipo_luminaria_diagnostico = $tipoLuminaria";
$command = $conn->createCommand($sql);
$lum = $command->queryAll();

foreach ($lum as $fila){
    $potenciaLum = $fila['potencia'];
    $rendimientoLum = $fila['rendimiento'];
    $claveLuminaria = $fila['clave_luminaria'];
}

```

Código para obtener le factor de reflexión de la pared, piso y techo:

```
$conn = Yii::app()->db;
$sql = "SELECT fr FROM {{color_superficie}} WHERE idcolor_superficie = '$colorTecho'";
$command = $conn->createCommand($sql);
$techo = $command->queryAll();

foreach ($techo as $fila)
{
    $frTecho = $fila['fr'];
}

```

Código para calcular la densidad de potencia de la luminaria:

```
$densidadDePotencia = round(($potenciaLum * $cantidad), 2);
$densidadDePotenciaNom007 = $densidadDePotencia / $area;
```

Código para obtener las cargas de la luminaria en periodos de tiempo:

```
$cargaDiaria = ($horasUso * $densidadDePotencia) / 1000; // En Kw
$cargaMensual = $cargaDiaria * 30; // En KW
$cargaAnual = $cargaMensual * 12; // En KW

$gastoDiario = $cargaDiaria * $precio; //En Kw/hora
$gastoMensual = $gastoDiario * 30; //En Kw/hora
$gastoAnual = $gastoMensual * 12; //En Kw/hora
```

Código para calcular el CO2 generado por las luminarias:

```
$CO2 = (($densidadDePotencia / 1000) * 0.495) * 360 * $horasUso; // En Kg anual
```

Código que muestra la valoración de las normas NOM013 y NOM007 además del cálculo de luxes generados por la luminaria a partir del tipo de construcción obteniendo la evaluación del nivel de iluminación del ambiente analizado:

```

switch ($clasificacionAmbiente) {
  case "I1":
    $scumpleNOM013 = 'No Aplica';
    if ($densidadDePotenciaNom007 <= 20)
      { $scumpleNOM007 = 'Si'; }
    else
      { $scumpleNOM007 = 'No';}

    if ($claveLuminaria == 'L7') {
      switch ($tipoConstruccion) {
        case "T1":
          if ($potenciaLum <= 7)
            { $rendimientoLum = 31;}
          elseif (($potenciaLum > 7) && ($potenciaLum <= 10))
            { $rendimientoLum = 34.5;}
          elseif (($potenciaLum > 10) && ($potenciaLum <= 14))
            { $rendimientoLum = 36;}
          elseif (($potenciaLum > 14) && ($potenciaLum <= 18))
            { $rendimientoLum = 40.5;}
          elseif ($potenciaLum > 18)
            { $rendimientoLum = 45;}
          break;
        case "T2":
          if ($potenciaLum <= 7)
            { $rendimientoLum = 40.5;}
          elseif (($potenciaLum > 7) && ($potenciaLum <= 10))
            { $rendimientoLum = 44.5;}
          elseif (($potenciaLum > 10) && ($potenciaLum <= 14))
            { $rendimientoLum = 46;}
          elseif (($potenciaLum > 14) && ($potenciaLum <= 18))
            { $rendimientoLum = 47.5;}
          elseif (($potenciaLum > 18) && ($potenciaLum <= 22))
            { $rendimientoLum = 52;}
          elseif ($potenciaLum > 22)
            { $rendimientoLum = 56.5;}
          break;
        case "T3":
          if ($potenciaLum <= 14)
            { $rendimientoLum = 29;}
          elseif (($potenciaLum > 14) && ($potenciaLum <= 18))
            { $rendimientoLum = 33;}
          elseif ($potenciaLum > 18)
            { $rendimientoLum = 40;}
          break;
      }
      $flujo = round($potenciaLum * $cantidad * $rendimientoLum, 2);
      $ft = round(sqrt($area) * $altura * 4 + (2 * $area), 2);
      $luxes = round(($flujo/$ft) * (($frPiso/(1-$frPiso)) + ($frTecho/(1-$frTecho)) + ($frPared/(1-$frPared))), 2);
    } else {
      $flujo = round($potenciaLum * $cantidad * $rendimientoLum, 2);
      $ft = round(sqrt($area) * $altura * 4 + (2 * $area), 2);
      $luxes = round(($flujo/$ft) * (($frPiso/(1-$frPiso)) + ($frTecho/(1-$frTecho)) + ($frPared/(1-$frPared))), 2);
    }

    if ($luxes >= 500)
      { $scumpleNOM025 = 'Si';}
    elseif ($luxes < 500)
      { $scumpleNOM025 = 'No';}
    if ($luxes <= 425)
      {$nivelIluminacion = 'Malo';}
    elseif (($luxes > 425) && ($luxes <= 625))
      {$nivelIluminacion = 'Bueno';}
    elseif (($luxes > 625))
      {$nivelIluminacion = 'Excedido';}

    break;
}

```

Código para insertar los cálculos generados en la tabla de diagnóstico:

```
$modelDiagIllumInt->iddiagnostico_iluminacion_interior = $lastDatoIllum;
$modelDiagIllumInt->altura_recomendada_lamp = $alturaOptima;
$modelDiagIllumInt->densidad_potencia = $densidadDePotencia;
$modelDiagIllumInt->carga_anual = $cargaAnual;
$modelDiagIllumInt->densidad_potencia_nom007 = $densidadDePotenciaNom007;
$modelDiagIllumInt->cumple_nom007 = $cumpleNOM007;
$modelDiagIllumInt->cumple_nom013 = $cumpleNOM013;
$modelDiagIllumInt->cumple_nom025 = $cumpleNOM025;
$modelDiagIllumInt->flujo = $flujo;
$modelDiagIllumInt->ft = $ft;
$modelDiagIllumInt->luxes = $luxes;
$modelDiagIllumInt->nivel_iluminacion = $nivelIluminacion;
$modelDiagIllumInt->mantenimiento_lum = $renLumMantenimiento;
$modelDiagIllumInt->gasto_diario = $gastoDiario;
$modelDiagIllumInt->gasto_mensual = $gastoMensual;
$modelDiagIllumInt->gasto_anual = $gastoAnual;
$modelDiagIllumInt->co2 = $CO2;

$modelDiagIllumInt->save();
```

Código de la función para generar un Ajax que muestra la imagen de la luminaria al seleccionarla:

```
public function actionCargaImagen(){

    $nombreLum = TipoLuminariaDiagnostico::model()->find('idtipo_luminaria_diagnostico=:idTipoLum',array(
':idTipoLum'=>(int) $_POST['DatosIluminacionInterior']['tipo_luminaria']));

    $nomLum = $nombreLum -> tipo_luminaria;

    echo '';
}

public function actionDiagnosticoIluminacionInt ()
{
    $modelHotHasAmb = new HotelHasAmbiente;
    $modelTipoLumDiag = new TipoLuminariaDiagnostico;

    $this->render('DiagnosticoIluminacionInterior',array(
        'modelHotHasAmb'=>$modelHotHasAmb,
        'modelTipoLumDiag'=>$modelTipoLumDiag,
    ));
}
```

Modelos:

“**DatosIluminacionInterior**”; este modelo se encarga de interactuar con la base de datos para guardar la información obtenida en cada ambiente y realizar peticiones para el uso de la misma cuando se realizan los cálculos.

```

public function attributeLabels()
{
    return array(
        'iddatos_iluminacion_interior' => 'Iddatos Iluminacion Interior',
        'hotel_idhotel' => 'Hotel Idhotel',
        'ambiente_idambiente' => 'Ambiente Idambiente',
        'num_espacios' => 'Num Espacios',
        'largo' => 'Largo',
        'ancho' => 'Ancho',
        'altura' => 'Alto',
        'tipo_iluminacion' => 'Tipo de Iluminacion',
        'color_techo' => 'Color del Techo',
        'color_pared' => 'Color de la Pared',
        'color_piso' => 'Color del Piso',
        'tipo_construccion' => 'Tipo de Construccion',
        'tipo_luminaria' => 'Tipo de Luminaria',
        'cantidad_luminaria' => 'Numero de Luminarias en el Ambiente',
        'difusor' => 'Tipo de Cubierta',
        'horas_uso' => 'Cuantas Horas por dia se usa la Luminaria',
        'meses_limpio' => 'Hace Cuantos Meses Limpio la Luminaria',
    );
}
public function search()
{
    $criteria=new CDbCriteria;

    $criteria->compare('iddatos_iluminacion_interior',$this->iddatos_iluminacion_interior);
    $criteria->compare('hotel_idhotel',$this->hotel_idhotel);
    $criteria->compare('ambiente_idambiente',$this->ambiente_idambiente);
    $criteria->compare('num_espacios',$this->num_espacios);
    $criteria->compare('largo',$this->largo,true);
    $criteria->compare('ancho',$this->ancho,true);
    $criteria->compare('altura',$this->altura,true);
    $criteria->compare('tipo_iluminacion',$this->tipo_iluminacion,true);
    $criteria->compare('color_techo',$this->color_techo,true);
    $criteria->compare('color_pared',$this->color_pared,true);
    $criteria->compare('color_piso',$this->color_piso,true);
    $criteria->compare('tipo_construccion',$this->tipo_construccion,true);
    $criteria->compare('tipo_luminaria',$this->tipo_luminaria);
    $criteria->compare('cantidad_luminaria',$this->cantidad_luminaria);
    $criteria->compare('difusor',$this->difusor,true);
    $criteria->compare('horas_uso',$this->horas_uso);
    $criteria->compare('meses_limpio',$this->meses_limpio);

    return new CActiveDataProvider($this, array(
        'criteria'=>$criteria,
    ));
}
}

```

“**Diagnóstico Iluminación Interior**”; este modelo gestiona la información analizada por el algoritmo, principalmente se encarga de guardar los datos calculados de cada ambiente para que se puedan utilizar para obtener la mejor recomendación y posteriormente realizar una comparativa para obtener un histórico de gestiones .

```

public function attributeLabels()
{
    return array(
        'iddiagnostico_iluminacion_interior' => 'Iddiagnostico Iluminacion Interior',
        'altura_recomendada_lamp' => 'Altura Recomendada Lamp',
        'densidad_potencia' => 'Densidad Potencia',
        'carga_anual' => 'Carga Anual',
        'densidad_potencia_nom007' => 'Densidad Potencia Nom007',
        'cumple_nom007' => 'Cumple Nom007',
        'cumple_nom013' => 'Cumple Nom013',
        'cumple_nom025' => 'Cumple Nom025',
        'flujo' => 'Flujo',
        'ft' => 'Ft',
        'luxes' => 'Luxes',
        'nivel_iluminacion' => 'Nivel Iluminacion',
        'mantenimiento_lum' => 'Mantenimiento Lum',
        'gasto_diario' => 'Gasto Diario',
        'gasto_mensual' => 'Gasto Mensual',
        'gasto_anual' => 'Gasto Anual',
        'co2' => 'Co2',
    );
}
public function search()
{
    $criteria=new CDbCriteria;

    $criteria->compare('iddiagnostico_iluminacion_interior',$this->iddiagnostico_iluminacion_interior);
    $criteria->compare('altura_recomendada_lamp',$this->altura_recomendada_lamp,true);
    $criteria->compare('densidad_potencia',$this->densidad_potencia,true);
    $criteria->compare('carga_anual',$this->carga_anual,true);
    $criteria->compare('densidad_potencia_nom007',$this->densidad_potencia_nom007,true);
    $criteria->compare('cumple_nom007',$this->cumple_nom007,true);
    $criteria->compare('cumple_nom013',$this->cumple_nom013,true);
    $criteria->compare('cumple_nom025',$this->cumple_nom025,true);
    $criteria->compare('flujo',$this->flujo,true);
    $criteria->compare('ft',$this->ft,true);
    $criteria->compare('luxes',$this->luxes,true);
    $criteria->compare('nivel_iluminacion',$this->nivel_iluminacion,true);
    $criteria->compare('mantenimiento lum',$this->mantenimiento lum,true);
    $criteria->compare('gasto_diario',$this->gasto_diario,true);
    $criteria->compare('gasto_mensual',$this->gasto_mensual,true);
    $criteria->compare('gasto_anual',$this->gasto_anual,true);
    $criteria->compare('co2',$this->co2,true);

    return new CActiveDataProvider($this, array(
        'criteria'=>$criteria,
    ));
}
}

```



```

<?php
/* @var $this DatosIluminacionInteriorController */
/* @var $model DatosIluminacionInterior */
/* @var $form CActiveForm */
?>

<div class="form">

<?php $form=$this->beginWidget('CActiveForm', array(
    'id'=>'datos-iluminacion-interior-ingresodatosinterior-form',
    'enableAjaxValidation'=>false,
));

$indice = $numAmbAct - 1;
$nombreAmb = $modelsHha[$indice]['tipo'];
$idambiente = $modelsHha[$indice]['ambiente_idambiente'];
$numEspacios = $modelsHha[$indice]['numHabAmbiente'];
$clasificacion = $modelsHha[$indice]['clasificacion'];

?>
    <?php echo $form->errorSummary(array($model)); ?>

    <input name="numAmb" value="<?php echo $numAmbAct;?>" type="hidden"/>
    <input name="idambiente" value="<?php echo $idambiente;?>" type="hidden"/>
    <input name="numEspacios" value="<?php echo $numEspacios;?>" type="hidden"/>
    <input name="clasificacion" value="<?php echo $clasificacion;?>" type="hidden"/>

<div class="span-20 last">
    <h4 class="rojo-sener">ILUMINACION INTERIOR - REGISTRO DE DATOS </h4>
</div>

<div class="span-17 prepend-1 append-2 last">
<fieldset style="display:inline; padding:0.5em 3em; margin-left:2em;">
    <legend><strong>Datos de <?php echo $nombreAmb;?></strong></legend>
    <table border="0" style="margin:0em;">
        <tr>
            <td style="text-align:center;">
                <div class="row">
                    <?php echo $form->labelEx($model,'largo',array('class'=>'inline')); ?>
                    <?php echo $form->textField($model,'largo',array('size'=>4,'maxlength'=>4)); ?> m.
                    <?php echo $form->error($model,'largo'); ?>
                </div>
            </td>
            <td>
                <div class="row">
                    <?php echo $form->labelEx($model,'ancho',array('class'=>'inline')); ?>
                    <?php echo $form->textField($model,'ancho',array('size'=>4,'maxlength'=>4)); ?> m.
                    <?php echo $form->error($model,'ancho'); ?>
                </div>
            </td>
            <td>
                <div class="row">
                    <?php echo $form->labelEx($model,'altura',array('class'=>'inline')); ?>
                    <?php echo $form->textField($model,'altura',array('size'=>4,'maxlength'=>4)); ?> m.
                    <?php echo $form->error($model,'altura'); ?>
                </div>
            </td>
        </tr>
    </table>

```

```

<table border="0" style="margin:0em;">
  <tr>
    <th style="padding-left:0em;"><?php echo $form->labelEx($model,'tipo_iluminacion'); ?></th>
    <th style="padding-left:0em;"><?php echo $form->labelEx($model,'tipo_construccion'); ?></th>
    <th style="padding-left:0em;"><?php echo $form->labelEx($model,'difusor'); ?></th>
  </tr>
  <tr>
    <td style="padding-left:2em;">
      <?php echo $form->radioButton($model,'tipo_iluminacion', array(
        'uncheckedValue' => NULL,
        'value' => 'Directa',
      )); ?>
      <?php echo 'Directa';?><br />
      <?php echo $form->radioButton($model,'tipo_iluminacion', array(
        'uncheckedValue' => NULL,
        'value' => 'Semidirecta',
      )); ?>
      <?php echo 'Semidirecta';?>
    </td>
    <td style="padding-left:2em;">
      <?php echo $form->radioButton($model,'tipo_construccion', array(
        'uncheckedValue' => NULL,
        'value' => 'T1',
      )); ?>
      <?php echo 'Con Envolvente';?><br />
      <?php echo $form->radioButton($model,'tipo_construccion', array(
        'uncheckedValue' => NULL,
        'value' => 'T2',
      )); ?>
      <?php echo 'Sin Envolvente';?><br />
      <?php echo $form->radioButton($model,'tipo_construccion', array(
        'uncheckedValue' => NULL,
        'value' => 'T3',
      )); ?>
      <?php echo 'Con Reflector';?>
    </td>
    <td style="padding-left:2em;">
      <?php echo $form->radioButton($model,'difusor', array(
        'uncheckedValue' => NULL,
        'value' => 'Si',
      )); ?>
      <?php echo 'Con Difusor';?><br />
      <?php echo $form->radioButton($model,'difusor', array(
        'uncheckedValue' => NULL,
        'value' => 'No',
      )); ?>
      <?php echo 'Sin Difusor';?>
    </td>
  </tr>
</table>

<?php echo $form->error($model,'tipo_iluminacion'); ?>
<?php echo $form->error($model,'tipo_construccion'); ?>
<?php echo $form->error($model,'difusor'); ?>

```

Campos mostrados en la interface del ingreso de datos

```

<legend><strong>Color de Superficies:</strong></legend><br />
<table border="0" style="margin:0em;">
  <tr>
    <th style="text-align:center;"><?php echo $form->labelEx($model,'color_techo'); ?></th>
    <th style="text-align:center;"><?php echo $form->labelEx($model,'color_pared'); ?></th>
    <th style="text-align:center;"><?php echo $form->labelEx($model,'color_piso'); ?></th>
    <th style="text-align:center;"><label>Muestra de Colores</label></th>
  </tr>
  <tr>
    <td>
      <?php echo $form->radioButton($model,'color_techo', array(
        'uncheckedValue' => NULL,
        'value' => 'C1',
      )); ?>
      <?php echo 'Blanco';?>
    </td>
    <td></td>
    <td></td>
    <td rowspan="4">
      
    </td>
  </tr>
  <tr>
    <td>
      <?php echo $form->radioButton($model,'color_techo', array(
        'uncheckedValue' => NULL,
        'value' => 'C2',
      )); ?>
      <?php echo 'Claro';?>
    </td>
    <td>
      <?php echo $form->radioButton($model,'color_pared', array(
        'uncheckedValue' => NULL,
        'value' => 'C4',
      )); ?>
      <?php echo 'Claro';?>
    </td>
    <td>
      <?php echo $form->radioButton($model,'color_piso', array(
        'uncheckedValue' => NULL,
        'value' => 'C7',
      )); ?>
      <?php echo 'Claro';?></td>
  </tr>
  <tr>
    <td>
      <?php echo $form->radioButton($model,'color_techo', array(
        'uncheckedValue' => NULL,
        'value' => 'C3',
      )); ?>
      <?php echo 'Medio';?>
    </td>
    <td>
      <?php echo $form->radioButton($model,'color_pared', array(
        'uncheckedValue' => NULL,
        'value' => 'C5',
      )); ?>
      <?php echo 'Medio';?>
    </td>
    <td></td>
  </tr>
  <tr>
    <td></td>
    <td></td>
    <td></td>
  </tr>

```

Campos mostrados en la interface del ingreso de datos

```

<tr>
  <td></td>
  <td>
    <?php echo $form->radioButton($model,'color_pared', array(
      'uncheckedValue' => NULL,
      'value' => 'C6',
    )); ?>
    <?php echo 'Oscuro';?>
  </td>
  <td>
    <?php echo $form->radioButton($model,'color_piso', array(
      'uncheckedValue' => NULL,
      'value' => 'C6',
    )); ?>
    <?php echo 'Oscuro';?>
  </td>
</tr>
</table>

<?php echo $form->error($model,'color_techo'); ?>
<?php echo $form->error($model,'color_pared'); ?>
<?php echo $form->error($model,'color_piso'); ?>
</fieldset>
</div>
<div class="span-20 last">

<fieldset style="display:inline">
  <legend><strong>Datos de la Luminaria de <?php echo $nombreAmb;?></strong></legend>

<div class="span-12">
<table border="0" style="margin:0em;">
  <tr>
    <td>
      <div class="row">
        <?php echo $form->labelEx($model,'tipo_luminaria',array('class'=>'inline')); ?>
        <?php echo $form->dropDownList($model,'tipo_luminaria',
CHtml::listData(TipoLuminariaDiagnostico::model()->findAll(),'idtipo_luminaria_diagnostico','descripcion',
'tipo_luminaria'),

array (
  'prompt'=>'----- Selecciona una Luminaria -----',
  'ajax'=>array(
    'type'=>'POST',
    'url'=>CController::createUrl('iluminacionInterior/cargaImagen'),
    'update'=>'#imagenLuminaria',
  ),
); ?>
        <?php echo $form->error($model,'tipo_luminaria'); ?>
      </div>
    </td>
  </tr>
  <tr>
    <td>
      <div class="row">
        <?php echo $form->labelEx($model,'cantidad_luminaria',array('class'=>'inline')); ?>
        <?php echo $form->dropDownList($model,'cantidad_luminaria',array('prompt'=>'Selecciona',1=>1,2=>2,3
=>3,4=>4,5=>5,6=>6,7=>7,8=>8,9=>9,10=>10,11=>11,12=>12,13=>13,14=>14,15=>15)); ?>
        <?php echo $form->error($model,'cantidad_luminaria'); ?>
      </div>
    </td>
  </tr>
</tr>

```

Campos mostrados en la interface del ingreso de datos

Imagen que se muestra al seleccionar la luminaria

“DiagnósticolluminacionInterior” muestra el diagnóstico en una tabla mostrando todos los ambientes relacionándolos con sus respectivas cargas.

```
<?php
/* @var $this IluminacionInteriorController */

//$this->breadcrumbs=array(
// 'Iluminacion Interior',
//);

$sambientes = $modelHotHasAmb->getAmbientesInteriores();
$numAmbientes = count($sambientes);

//echo $numAmbientes;

$conn = Yii::app()->db;
$idhotel = Yii::app()->user->id;
$sql = "SELECT a.tipo, di.cantidad_luminaria, di.horas_uso, di.altura, di.tipo_iluminacion,
di.difusor, (SELECT descripcion FROM {{color_superficie}} WHERE idcolor_superficie = di.color_techo), (SELECT
descripcion FROM {{color_superficie}} WHERE idcolor_superficie = di.color_pared), (SELECT descripcion FROM
{{color_superficie}} WHERE idcolor_superficie = di.color_piso), tc.construccion, tl.descripcion FROM
{{datos_iluminacion_interior}} di JOIN {{ambiente}} a ON(di.ambiente_idambiente = a.idambiente) JOIN
{{tipo_luminaria_diagnostico}} tl ON(di.tipo_luminaria = tl.idtipo_luminaria_diagnostico) JOIN
{{tipo_construccion}} tc ON(di.tipo_construccion = tc.tipo_construccion) ORDER BY iddatos_iluminacion_interior
DESC LIMIT $numAmbientes";
$command = $conn->createCommand($sql);
$datosIluminacionInt = $command->queryAll();

$conn = Yii::app()->db;
$idhotel = Yii::app()->user->id;
$sql = "SELECT a.tipo, diag.cumple_nom007, diag.cumple_nom025, diag.cumple_nom013,
diag.nivel_iluminacion, diag.carga_anual, diag.gasto_mensual, diag.gasto_anual, diag.co2,
diag.iddiagnostico_iluminacion_interior FROM {{diagnostico_iluminacion_interior}} diag JOIN
{{datos_iluminacion_interior}} di ON(diag.iddiagnostico_iluminacion_interior =
di.iddatos_iluminacion_interior) JOIN {{ambiente}} a ON(di.ambiente_idambiente = a.idambiente) ORDER BY
iddatos_iluminacion_interior DESC LIMIT $numAmbientes";
$command = $conn->createCommand($sql);
$diagnosticoIluminacionInt = $command->queryAll();
?>
<div class="form">

<div class="span-20 last">
<h4 class="rojo-sener">ILUMINACION INTERIOR - DIAGNOSTICO</h4>
</div>

<div class="span-20 last" style="margin-right:1em;">
<fieldset style="display:inline; margin:0em; ">
<legend><strong>Datos de Ambientes</strong></legend>

<table border="0">
<tr style="font-size:12px"> <!-- El numero indica la posicion dle arreglo -->
<th style="text-align:center;">Ambiente</th>
<th style="text-align:center;">Tipo de Lum*</th>
<th style="text-align:center;">Numero Lum*</th>
<th style="text-align:center;">Horas de uso</th>
<th style="text-align:center;">Tipo Iluminacion*</th>
<th style="text-align:center;">Tipo Construcción*</th>
<th style="text-align:center;">Difusor</th>
```

```

        </tr>
        <?php
        foreach ($datosIluminacionInt as $fila) {
        echo '<tr style="font-size:12px">
            <td style="text-align:center;">'. $fila['tipo'].'</td>
            <td style="text-align:center;">'. $fila['descripcion'].'</td>
            <td style="text-align:center;">'. $fila['cantidad_luminaria'].'</td>
            <td style="text-align:center;">'. $fila['horas_uso'].'</td>
            <td style="text-align:center;">'. $fila['tipo_iluminacion'].'</td>
            <td style="text-align:center;">'. $fila['construccion'].'</td>
            <td style="text-align:center;">'. $fila['difusor'].'</td>
        </tr>';
        }
        ?>
    </table>
</fieldset>
</div>
<span class="nota">*Luminaria.</span>

<div class="span-20 last" style="margin-right:1em;">
<br /> <br />
<fieldset style="display:inline; margin:0em; ">
    <legend><strong>Diagnóstico de Ambientes</strong></legend>

    <table border="0" >
        <tr style="font-size:12px">
            <th style="text-align:center;">Ambiente</th>
            <th style="text-align:center;">Cumple NOM007</th>
            <th style="text-align:center;">Cumple NOM025</th>
            <th style="text-align:center;">Nivel Iluminacion</th>
            <th style="text-align:center;">Carga Anual (Kw)</th>
            <th style="text-align:center;">Gasto Anual (Kw)</th>
            <th style="text-align:center;">Emision Anual CO2 (Kg)</th>
        </tr>
    <?php
    foreach ($diagnosticoIluminacionInt as $fila) {
    echo '<tr style="font-size:12px">
        <td style="text-align:center;">'. $fila['tipo'].'</td>
        <td style="text-align:center;">'. $fila['cumple_nom007'].'</td>
        <td style="text-align:center;">'. $fila['cumple_nom025'].'</td>
        <td style="text-align:center;">'. $fila['nivel_iluminacion'].'</td>
        <td style="text-align:center;">'. $fila['carga_anual'].'</td>
        <td style="text-align:center;">'. $fila['gasto_anual'].'</td>
        <td style="text-align:center;">'. $fila['co2'].'</td>
    </tr>';
    }
    ?>
</table>

</fieldset>

</div>

<div class="span-19 append-1 last row buttons" style="text-align: right;">
    <br /><br />
    <?php echo CHtml::link('Proceder a las Recomendaciones',array('iluminacionInterior/recomendacionesInt'));
    ?>

</div>

</div><!-- Fin Form-->

```

“**RecomenIluminaInterior**” muestra las recomendaciones del diagnóstico por cada ambiente mostrando la luminaria recomendada.

```
<?php
/* @var $this IluminacionInteriorController */

$this->breadcrumbs=array(
    'Iluminacion Interior'=>array('/iluminacionInterior'),
    'RecomenIluminaInterior',
);

$ambientes = $modelHotHasAmb->getAmbientesInteriores();
$numAmbientes = count($ambientes);

$conn = Yii::app()->db;
$idhotel = Yii::app()->user->id;
$sql = "SELECT a.tipo, diag.cumple_nom007, diag.cumple_nom025, diag.cumple_nom013,
diag.nivel_iluminacion, diag.carga_anual, diag.gasto_mensual, diag.gasto_anual, diag.co2,
diag.iddiagnostico_iluminacion_interior FROM {{diagnostico_iluminacion_interior}} diag JOIN
{{datos_iluminacion_interior}} di ON(diag.iddiagnostico_iluminacion_interior =
di.iddatos_iluminacion_interior) JOIN {{ambiente}} a ON(di.ambiente_idambiente = a.idambiente) ORDER BY
iddatos_iluminacion_interior DESC LIMIT $numAmbientes";
$command = $conn->createCommand($sql);
$diagnosticoIluminacionInt = $command->queryAll();

?>

<div class="form">

<div class="span-20 last">
    <h4 class="rojo-sener">ILUMINACION EXTERIOR - RECOMENDACIONES</h4>
</div>

<div class="span-20 last" style="margin-right:1em;">

<fieldset style="display:inline; margin:0em; ">
    <legend><strong>Diagnostico de Ambientes</strong></legend>

    <table border="0" >
        <tr style="font-size:12px">
            <th style="text-align:center;">Ambiente</th>
            <th style="text-align:center;">Cumple NOM007</th>
            <th style="text-align:center;">Cumple NOM025</th>
            <th style="text-align:center;">Nivel Iluminacion</th>
            <th style="text-align:center;">Carga Anual (Kw)</th>
            <th style="text-align:center;">Gasto Anual (pesos)</th>
            <th style="text-align:center;">Emision Anual CO2 (Kg)</th>
        </tr>
```

```

<?php
foreach ($diagnosticoIluminacionInt as $fila) {
    echo    '<tr style="font-size:12px">
            <td style="text-align:center;">'. $fila['tipo']. '</td>
            <td style="text-align:center;">'. $fila['cumple_nom007']. '</td>
            <td style="text-align:center;">'. $fila['cumple_nom025']. '</td>
            <td style="text-align:center;">'. $fila['nivel_iluminacion']. '</td>
            <td style="text-align:center;">'. $fila['carga_anual']. '</td>
            <td style="text-align:center;">'. $fila['gasto_anual']. '</td>
            <td style="text-align:center;">'. $fila['co2']. '</td>
            <td style="text-align:center;">
                <input type="button" value="Recomendaci&oacute;n" class="AmbienteExt" />
                <input type="hidden" value="'. $fila['iddiagnostico_iluminacion_interior']. '" />
            </td>
        </tr>';
    }
?>
</table>

</fieldset>

</div>
</fieldset>
</div>

</div> <!-- Fin de Form -->

```

Las partes de modelos, vistas y controladores mencionados anteriormente en conjunto forman el Sistema web Inteligente para la gestión del ahorro de energía en refrigeración e iluminación en hoteles de México con el cual se pueden realizar diagnósticos energéticos para las diferentes áreas obteniendo una óptima recomendación las cuales si se ponen en marcha lograrán un ahorro económico significativo.

PRUEBAS Y GENERACIÓN DE RESULTADOS

Capítulo 6

VI.- Pruebas y Generación de Resultados

Para garantizar que el sistema web es funcional y no genera errores he elaborado una serie de pruebas de aceptabilidad, antes de su puesta en producción, evitando así sorpresas indeseables al provocar la insatisfacción de nuestros usuarios por un inadecuado dimensionamiento.

Los siguientes aspectos fueron puestos a prueba para garantizar la funcionalidad:

- ✓ Ortografía y Gramática para tener una clara percepción de la información.
- ✓ Revisión de toda la información que he proporcionado para que sea clara y precisa.
- ✓ Prueba de navegación donde se revisaron todos los caminos dentro del sistema.
- ✓ Chequeo de enlaces rotos para que la operatividad y navegación no decrezca.
- ✓ Revisión de campos requeridos donde se garantiza que los campos pedidos se ingresen a la base de datos para su correcto uso.
- ✓ Inspección de Mensajes de error para que se muestren los más significativos.
- ✓ Compatibilidad con diferentes navegadores para garantizar la correcta visualización.
- ✓ La usabilidad del servicio, entendiendo por usabilidad la utilidad, facilidad de uso y satisfacción de los usuarios del sitio web.

Las pruebas antes mencionadas se realizaron de forma correcta con lo que se pudo corregir la ortografía, la navegación para que resultara más amigable para el usuario, la compatibilidad con diferentes navegadores y dispositivos móviles además de la usabilidad del sistema web con lo que pudimos garantizar que el sistema realiza un diagnostico energético confiable ya que verifica que cada elemento encaja de forma adecuada y que se alcanza la funcionalidad y el rendimiento del sistema total.

VI.1 Demostración del Sistema web Inteligente para el Ahorro de Energía.

A continuación se realiza la presentación del sistema iniciando con la interface general donde se podrá llevar a través del registro y autenticación del usuario con su respectivo hotel, hasta la generación de consultas para obtener los cálculos realizados para el diagnóstico energético y sus respectivas recomendaciones.

La página de inicio será el punto de partida para comenzar a trabajar con el sistema, desde la visualización de noticias, objetivos, organismos, contactos, eventos, páginas amarillas, etc. Es decir, información disponible a todo público.

OBJETIVO

Sistema Web para la Gestión de la Eficiencia Energética en Hoteles. Que busca mejorar el consumo y eficiencia energética en el sector hotelero.

ORGANISMOS

- UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
- UNIVERSIDAD AUTÓNOMA DE CAMPECHE
- UNIVERSIDAD VERACRUZANA
- FONDO SECTORIAL

CONTACTO

Sección que incluirá información de contacto, para el envío de comentarios y sugerencias sobre el uso del sistema, así como el reporte de posibles errores que puedan existir.

NOTICIAS SOBRE AHORRO DE ENERGÍA EN HOTELES

Sección dedicada a la publicación de experiencias e implementaciones que han tenido diversos hoteles con el sistema. Se presentan casos de éxito, en donde se resumen las medidas de ahorro más significativas que el sistema ha propuesto para las diversas áreas con las que cuenta el hotel.

Apartados adicionales para la presentación de los reportes más significativos efectuados en otros hoteles, que de igual manera han sido beneficiados, gracias a la utilización de los diversos módulos para la gestión de la eficiencia energética que implementa el sistema.

SIMAE

PÁGINAS AMARILLAS

MEJORES PRÁCTICAS

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

Para poder trabajar con los módulos y áreas energéticas, será necesario estar registrado dentro del sistema:



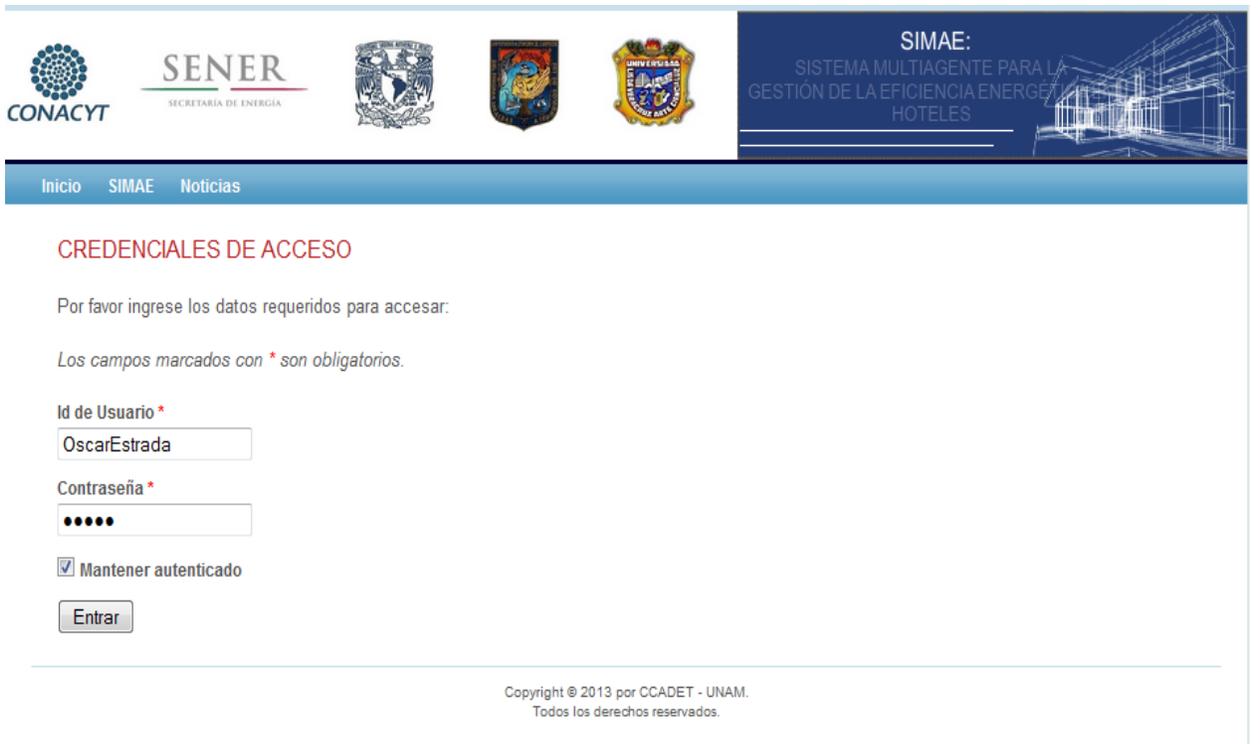
ACCESO AL SISTEMA

Para usar el Sistema Multiagente para la Gestión de la Eficiencia Energética en Hoteles (SIMAE), debes acceder al sistema, hazlo ahora:

NUEVO USUARIO **USUARIO EXISTENTE**

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

En caso de ya estar registrado solo basta con ingresar con el usuario y contraseñas creadas con anterioridad:



CREDENCIALES DE ACCESO

Por favor ingrese los datos requeridos para acceder:

Los campos marcados con * son obligatorios.

Id de Usuario *
OscarEstrada

Contraseña *
●●●●●

Mantener autenticado

Entrar

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

En caso de no estar registrado se procede al llenado del formulario de registro para los usuarios, en caso de faltar algún campo obligatorio o de que ambas contraseñas no coincidan, se mostrará un mensaje de error temporal, impidiendo el registro del usuario hasta su corrección.

CONACYT **SENER** SECRETARÍA DE ENERGÍA

SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio SIMAE Noticias

DATOS DEL RESPONSABLE DEL HOTEL

Los campos marcados con * son obligatorios.

Nombre *

Id de Usuario *

Contraseña *

Vuelva a escribir su contraseña *

lfe *

E-mail

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

Una vez registrado el usuario, se procede a pedir la información del hotel la cual si no es correctamente llenada o hay campos faltantes se mostrara un mensaje error aclarando que es lo que falta y no se podrá guardar el registro hasta que sean corregidos los errores:



SENER
SECRETARÍA DE ENERGÍA



SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES



[Inicio](#) [SIMAE](#) [Noticias](#) [Salir \(oscarestrada\)](#)

DATOS DEL HOTEL

Los campos marcados con * son obligatorios.

Nombre del Hotel *

Ozkar-in

RFC *

EAVO870410JR5

Calle *

AV.Tlalpan

Colonia *

Portales

Delegación *

Coyoacan

Código Postal *

67894

Telefono *

56978985

Pagina Web

http://www.ozkarin.com

Área del terreno (m2)

4000

Área Construida (m2)

3000

Número de neveras

1

Categoría *

Cinco estrellas

Estado *

DISTRITO FEDERAL

Ciudad *

Mexico

Zona de Ubicación

Turistica

Create

Una vez registrado o autenticado se puede trabajar con los módulos del sistema:

CONACYT SENER SECRETARÍA DE ENERGÍA

UNIVERSIDAD DEL ESTADO DE MÉXICO

SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)

Módulos del Sistema

Hotel: [Ozkar-in](#)

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

BIENVENIDO A LA PÁGINA PRINCIPAL DE LOS MÓDULOS DEL SISTEMA

Para comenzar, dar click en el botón activo del menú de la izquierda.

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

La primera opción justo al ingresar al sistema es proceder a la encuesta donde se recabara información general para completar el registro de ambientes, la facturación eléctrica que tiene el hotel y la ocupación.

CONACYT SENER SECRETARÍA DE ENERGÍA

UNIVERSIDAD DEL ESTADO DE MÉXICO

SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)

Módulos del Sistema

Hotel: [Ozkar-in](#)

ENCUESTA

- [REGISTRO DE AMBIENTES](#)
- [FACTURACIÓN ELÉCTRICA](#)
- [OCUPACIÓN](#)

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

El registro de ambientes es fundamental ya que describe las áreas con las que cuenta el hotel y las características de cada una, lo que ayuda al diagnóstico energético







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA
HOTELERÍA



Inicio
Salir (oscurestrada)

Módulos del Sistema

Hotel: [Ozkar-in](#)

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

REGISTRO DE AMBIENTES

Seleccione los ambientes con los que cuenta el hotel, así como el número de espacios para cada uno y si cuenta o desea acondicionamiento.

<input checked="" type="checkbox"/> Habitación Simple Numero de Espacios * <input type="text" value="3"/> Acondicionamiento * Si <input checked="" type="radio"/> No <input type="radio"/>	<input checked="" type="checkbox"/> Habitación Doble Numero de Espacios * <input type="text" value="2"/> Acondicionamiento * Si <input checked="" type="radio"/> No <input type="radio"/>	<input checked="" type="checkbox"/> Habitación Triple Numero de Espacios * <input type="text" value="1"/> Acondicionamiento * Si <input checked="" type="radio"/> No <input type="radio"/>
<input type="checkbox"/> Habitación Cuadruple <input type="checkbox"/> Lobby	<input type="checkbox"/> Suite <input type="checkbox"/> Sala de Recepcion	<input type="checkbox"/> Bungalow <input checked="" type="checkbox"/> Pasillo Numero de Espacios * <input type="text" value="3"/> Acondicionamiento * Si <input type="radio"/> No <input checked="" type="radio"/>
<input type="checkbox"/> Bar <input type="checkbox"/> Cocina <input type="checkbox"/> Gimnasio <input type="checkbox"/> Oficina	<input type="checkbox"/> Restaurante <input type="checkbox"/> Lavanderia <input type="checkbox"/> Cuarto de Maquinas <input type="checkbox"/> Sala de Internet	<input type="checkbox"/> Auditorio <input type="checkbox"/> Spa <input type="checkbox"/> Sala Multiuso <input checked="" type="checkbox"/> Alberca Numero de Espacios * <input type="text" value="1"/> Acondicionamiento * Si <input type="radio"/> No <input checked="" type="radio"/>
<input checked="" type="checkbox"/> Estacionamiento Numero de Espacios * <input type="text" value="1"/> Acondicionamiento * Si <input type="radio"/> No <input checked="" type="radio"/>	<input checked="" type="checkbox"/> Jardin Numero de Espacios * <input type="text" value="1"/> Acondicionamiento * Si <input type="radio"/> No <input checked="" type="radio"/>	<input type="checkbox"/> Cancha Deportiva <input type="checkbox"/> Patio

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

Una vez registrados los ambientes podemos proceder a la realización del diagnóstico energético:

Al seleccionar Iluminación interior se muestra la clasificación de los ambientes interiores que se registraron y el número de espacios de cada uno:

AMBIENTE	NO. DE ESPACIOS
Habitacion Simple	3
Habitacion Doble	2
Habitacion Triple	1
Pasillo	3
Estacionamiento Cubierto	1

Al proceder al ingreso de datos se inicia un ciclo el cual va registrando ambiente por ambiente y para cada uno se ingresan los datos necesarios, en caso de faltar un dato o no llenar un campo se marca un error el cual no dejara registrar los datos hasta ser corregido:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN INTERIOR - REGISTRO DE DATOS

Datos de Estacionamiento Cubierto

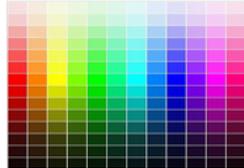
Largo * m. Ancho * m. Alto * m.

<p>Tipo de Iluminacion *</p> <p><input checked="" type="radio"/> Directa</p> <p><input type="radio"/> Semidirecta</p>	<p>Tipo de Construccion *</p> <p><input checked="" type="radio"/> Con Envolverte</p> <p><input type="radio"/> Sin Envolverte</p> <p><input type="radio"/> Con Reflector</p>	<p>Tipo de Cubierta *</p> <p><input checked="" type="radio"/> Con Difusor</p> <p><input type="radio"/> Sin Difusor</p>
--	--	---

Color de Superficies:

<p>Color del Techo *</p> <p><input type="radio"/> Blanco</p> <p><input checked="" type="radio"/> Claro</p> <p><input type="radio"/> Medio</p>	<p>Color de la Pared *</p> <p><input type="radio"/> Claro</p> <p><input checked="" type="radio"/> Medio</p> <p><input type="radio"/> Oscuro</p>	<p>Color del Piso *</p> <p><input type="radio"/> Claro</p> <p><input checked="" type="radio"/> Oscuro</p>
--	--	--

Muestra de Colores



Datos de la Luminaria de Estacionamiento Cubierto

Tipo de Luminaria *

No. de Luminarias en el Ambiente *

No. de Luminarias funcionales en el Ambiente *

Cuantas Horas por día se usa la Luminaria *

No. de apagadores en el Ambiente *



Los Campos con * son requeridos.

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

En la imagen anterior se muestran los datos ingresados para estacionamiento cubierto y en la siguiente para el ambiente pasillo:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELERÍA

Inicio Salir (oscurestrada)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

Hotel: [Ozkar-in](#)

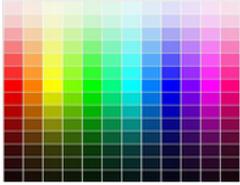
ILUMINACIÓN INTERIOR - REGISTRO DE DATOS

Datos de Pasillo

Largo * m. Ancho * m. Alto * m.

Tipo de Iluminacion *	Tipo de Construccion *	Tipo de Cubierta *
<input type="radio"/> Directa <input checked="" type="radio"/> Semidirecta	<input type="radio"/> Con Envolverte <input checked="" type="radio"/> Sin Envolverte <input type="radio"/> Con Reflector	<input type="radio"/> Con Difusor <input checked="" type="radio"/> Sin Difusor

Color de Superficies:

Color del Techo *	Color de la Pared *	Color del Piso *	Muestra de Colores 
<input type="radio"/> Blanco	<input checked="" type="radio"/> Claro	<input checked="" type="radio"/> Claro	
<input checked="" type="radio"/> Claro	<input type="radio"/> Medio	<input type="radio"/> Oscuro	
<input type="radio"/> Medio	<input type="radio"/> Oscuro	<input type="radio"/> Oscuro	

Datos de la Luminaria de Pasillo

Tipo de Luminaria *

No. de Luminarias en el Ambiente *

No. de Luminarias funcionales en el Ambiente *

Cuantas Horas por día se usa la Luminaria *

No. de apagadores en el Ambiente *



Los Campos con * son requeridos.

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

En la imagen siguiente se muestran los datos ingresados para habitación triple:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN INTERIOR - REGISTRO DE DATOS

Datos de Habitación Triple

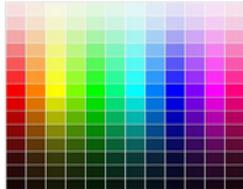
Largo * m. Ancho * m. Alto * m.

<p>Tipo de Iluminacion *</p> <p><input checked="" type="radio"/> Directa</p> <p><input type="radio"/> Semidirecta</p>	<p>Tipo de Construccion *</p> <p><input type="radio"/> Con Envolverte</p> <p><input type="radio"/> Sin Envolverte</p> <p><input checked="" type="radio"/> Con Reflector</p>	<p>Tipo de Cubierta *</p> <p><input checked="" type="radio"/> Con Difusor</p> <p><input type="radio"/> Sin Difusor</p>
--	--	---

Color de Superficies:

<p>Color del Techo *</p> <p><input checked="" type="radio"/> Blanco</p> <p><input type="radio"/> Claro</p> <p><input type="radio"/> Medio</p>	<p>Color de la Pared *</p> <p><input type="radio"/> Claro</p> <p><input checked="" type="radio"/> Medio</p> <p><input type="radio"/> Oscuro</p>	<p>Color del Piso *</p> <p><input type="radio"/> Claro</p> <p><input checked="" type="radio"/> Oscuro</p>
--	--	--

Muestra de Colores



Datos de la Luminaria de Habitación Triple

Tipo de Luminaria *

No. de Luminarias en el Ambiente *

No. de Luminarias funcionales en el Ambiente *

Cuántas Horas por día se usa la Luminaria *

No. de apagadores en el Ambiente *



Los Campos con * son requeridos.

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

En la siguiente imagen se muestran los datos ingresados para habitación doble:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscarsestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN INTERIOR - REGISTRO DE DATOS

Datos de Habitación Doble

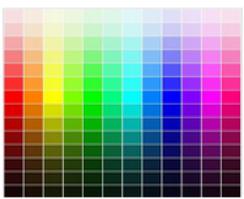
Largo * m. Ancho * m. Alto * m.

<p>Tipo de Iluminación *</p> <p><input type="radio"/> Directa</p> <p><input checked="" type="radio"/> Semidirecta</p>	<p>Tipo de Construcción *</p> <p><input checked="" type="radio"/> Con Envolverte</p> <p><input type="radio"/> Sin Envolverte</p> <p><input type="radio"/> Con Reflector</p>	<p>Tipo de Cubierta *</p> <p><input type="radio"/> Con Difusor</p> <p><input checked="" type="radio"/> Sin Difusor</p>
--	--	---

Color de Superficies:

<p>Color del Techo *</p> <p><input type="radio"/> Blanco</p> <p><input type="radio"/> Claro</p> <p><input checked="" type="radio"/> Medio</p>	<p>Color de la Pared *</p> <p><input checked="" type="radio"/> Claro</p> <p><input type="radio"/> Medio</p> <p><input type="radio"/> Oscuro</p>	<p>Color del Piso *</p> <p><input type="radio"/> Claro</p> <p><input checked="" type="radio"/> Oscuro</p>
--	--	--

Muestra de Colores



Datos de la Luminaria de Habitación Doble

Tipo de Luminaria *

No. de Luminarias en el Ambiente *

No. de Luminarias funcionales en el Ambiente *

Cuántas Horas por día se usa la Luminaria *

No. de apagadores en el Ambiente *



Los Campos con * son requeridos.

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

En la imagen siguiente se muestran los datos ingresados para habitación simple:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio
Salir (oscurestrada)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

Hotel: [Ozkar-in](#)

ILUMINACIÓN INTERIOR - REGISTRO DE DATOS

Datos de Habitación Simple

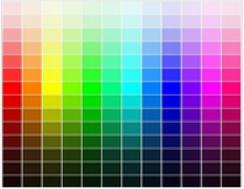
Largo * m.
Ancho * m.
Alto * m.

<p>Tipo de Iluminación *</p> <p><input checked="" type="radio"/> Directa</p> <p><input type="radio"/> Semidirecta</p>	<p>Tipo de Construcción *</p> <p><input type="radio"/> Con Envoltente</p> <p><input checked="" type="radio"/> Sin Envoltente</p> <p><input type="radio"/> Con Reflector</p>	<p>Tipo de Cubierta *</p> <p><input checked="" type="radio"/> Con Difusor</p> <p><input type="radio"/> Sin Difusor</p>
--	--	---

Color de Superficies:

<p>Color del Techo *</p> <p><input checked="" type="radio"/> Blanco</p> <p><input type="radio"/> Claro</p> <p><input type="radio"/> Medio</p>	<p>Color de la Pared *</p> <p><input checked="" type="radio"/> Claro</p> <p><input type="radio"/> Medio</p> <p><input type="radio"/> Oscuro</p>	<p>Color del Piso *</p> <p><input checked="" type="radio"/> Claro</p> <p><input type="radio"/> Oscuro</p>
--	--	--

Muestra de Colores



Datos de la Luminaria de Habitación Simple

Tipo de Luminaria *

No. de Luminarias en el Ambiente *

No. de Luminarias funcionales en el Ambiente *

Cuantas Horas por día se usa la Luminaria *

No. de apagadores en el Ambiente *



Los Campos con * son requeridos.

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

Al concluir el ciclo para procesar los datos de los ambientes se muestra en una tabla el diagnóstico energético donde por referencia también se muestran los datos ingresados de cada ambiente.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA
HOTELERÍA



Inicio
Salir (oscurestrada)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

Hotel: [Ozkar-in](#)

ILUMINACIÓN INTERIOR - DIAGNÓSTICO

Datos de Ambientes

Ambiente	Tipo de Lum*	No. Lum*	Horas de uso	Tipo Iluminación	Tipo Construcción	No. Lum Funcionales
Habitacion Simple	Incandescente de 75 watts	1	12	Directa	Sin envolvene	1
Habitacion Doble	Incandescente de 75 watts	2	11	Semidirecta	Con envolvene	2
Habitacion Triple	Incandescente de 50 watts	3	10	Directa	Con reflector	3
Pasillo	Incandescente de 60 watts	2	14	Semidirecta	Sin envolvene	2
Estacionamiento Cubierto	Incandescente de 100 watts	4	12	Directa	Con envolvene	4

*Luminaria.

Diagnóstico de Ambientes

Ambiente	Cumple NOM007	Cumple NOM025	Nivel Iluminacion	Carga Anual (Kw)	Gasto Anual (pesos)	Emision Anual CO2 (Kg)
Habitacion Simple	Si	Si	Excedido	324.00	388.80	13.55
Habitacion Doble	Si	Si	Excedido	594.00	712.80	27.10
Habitacion Triple	Si	Si	Excedido	540.00	648.00	27.10
Pasillo	No	Si	Excedido	604.80	727.20	21.68
Estacionamiento Cubierto	No	No Aplica	No Aplica	1728.00	2073.60	72.27

[Proceder a las Recomendaciones](#)

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

Al proceder a las recomendaciones se accede a una página donde se muestra una lista con los ambientes procesados y al seleccionar uno de la lista se muestran sus respectivas recomendaciones:

The screenshot shows the SIMAE web application interface. At the top, there are logos for CONACYT, SENER (Secretaría de Energía), and three university logos. The main header area includes the text 'SIMAE: SISTEMA MULTIAGENTE PARA LA GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN HOTELES' and a wireframe image of a building. Below the header is a navigation bar with 'Inicio' and 'Salir (oscurestrada)'. On the left, a sidebar lists 'Módulos del Sistema' including 'ENCUESTA', 'DIAGNÓSTICO ENERGÉTICO' (highlighted), 'RECOMENDACIÓN DE MEDIDAS DE AHORRO', 'EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO', 'MONITOREO DE LAS MEDIDAS DE AHORRO', and 'FACTURACIÓN DE CONSUMO'. The main content area is titled 'ILUMINACIÓN INTERIOR - RECOMENDACIONES' and contains a 'Lista de Ambientes' section with a dropdown menu for 'Ambiente:' showing options: 'Todos', 'Habitacion Simple', 'Habitacion Doble', 'Habitacion Triple', 'Pasillo', and 'Estacionamiento Cubierto'. A 'Ver Recomendaciones' button is positioned to the right of the dropdown. The top right corner of the main area shows 'Hotel: Ozkar-in'. At the bottom, a copyright notice reads 'Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.'

Las siguientes imágenes muestran las recomendaciones para cada ambiente donde se recomienda cambiar a un cierto tipo de luminaria la cual es mostrada como referencia también aparecen los ahorros de energía eléctrica, la demanda de esta nueva luminaria, las emisiones y ahorros de CO2:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA
HOTELERÍA

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN INTERIOR - RECOMENDACIONES

Lista de Ambientes

Ambiente: Habitacion Simple Ver Recomendaciones

Se recomienda sustituir por el siguiente tipo de lámpara:



LÁMPARA PROPUESTA

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	3	Sí	Sí

Usando este tipo de lampara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.003	96

EMISIONES DE CO2

Con el uso de este tipo de lámpara, las emisiones de CO2 se ven disminuidas de la siguiente forma:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.45	0.02	95.6

Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA
HOTELERÍA

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN INTERIOR - RECOMENDACIONES

Lista de Ambientes

Ambiente: Habitacion Doble Ver Recomendaciones

Se recomienda sustituir por el siguiente tipo de lámpara:



LÁMPARA PROPUESTA

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	5	Sí	Sí

Usando este tipo de lampara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.01	93.3

EMISIONES DE CO2

Con el uso de este tipo de lámpara, las emisiones de CO2 se ven disminuidas de la siguiente forma:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.82	0.05	93.9

Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA
HOTELERÍA

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN INTERIOR - RECOMENDACIONES

Lista de Ambientes

Ambiente:

Se recomienda sustituir por el siguiente tipo de lámpara:



LÁMPARA PROPUESTA

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	3	Sí	Sí

Usando este tipo de lampara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.009	94

EMISIONES DE CO2

Con el uso de este tipo de lámpara, las emisiones de CO2 se ven disminuidas de la siguiente forma:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.74	0.04	94.6

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA
HOTELERÍA

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN INTERIOR - RECOMENDACIONES

Lista de Ambientes

Ambiente:

Se recomienda sustituir por el siguiente tipo de lámpara:



LÁMPARA PROPUESTA

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	3	Sí	Sí

Usando este tipo de lampara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.006	95

EMISIONES DE CO2

Con el uso de este tipo de lámpara, las emisiones de CO2 se ven disminuidas de la siguiente forma:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.83	0.04	95.2

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

The screenshot displays the SIMAE (Sistema Multiagente para la Gestión de la Eficiencia Energética en Hoteles) interface. At the top, there are logos for CONACYT, SENER (Secretaría de Energía), and the Mexican coat of arms. The main header identifies the system as SIMAE: SISTEMA MULTIAGENTE PARA LA GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN HOTELES. The user is logged in as 'Hotel: Ozkar-in'. The left sidebar contains navigation options: 'Inicio', 'Salir (oscurestrada)', 'Módulos del Sistema', 'ENCUESTA', 'DIAGNÓSTICO ENERGÉTICO' (highlighted), 'RECOMENDACIÓN DE MEDIDAS DE AHORRO', 'EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO', 'MONITOREO DE LAS MEDIDAS DE AHORRO', and 'FACTURACIÓN DE CONSUMO'. The main content area is titled 'ILUMINACIÓN INTERIOR - RECOMENDACIONES'. It features a 'Lista de Ambientes' with a dropdown menu set to 'Estacionamiento Cubierto' and a 'Ver Recomendaciones' button. Below this, a message states: 'Se recomienda sustituir por el siguiente tipo de lámpara:'. An image of a modern LED lamp is shown. To the right, a table titled 'LÁMPARA PROPUESTA' provides technical specifications:

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	3	Sí	Sí

Usando este tipo de lampara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.006	95

Below the table, the section 'EMISIONES DE CO2' explains that CO2 emissions are reduced. A summary table shows:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.83	0.04	95.2

At the bottom, a copyright notice reads: 'Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.'

Al finalizar el diagnóstico energético para iluminación interior procedemos al de iluminación exterior esto se hace por medio de la selección del área en la parte izquierda de la página.

Como iluminación exterior es muy similar a iluminación interior podemos decir que siguen los mismos pasos descritos con anterioridad.

Iluminación Exterior

Al seleccionar Iluminación exterior se muestra la clasificación de los ambientes exteriores que se registraron y el número de espacios de cada uno mostrados en la siguiente imagen y en las siguientes a esa se muestra el ciclo el cual va registrando ambiente por ambiente y para cada uno se ingresan los datos necesarios:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN EXTERIOR - AMBIENTES

Con la finalidad de realizar un diagnóstico lo más efectivo posible será necesario procesar ambiente por ambiente, lo que se puede llevar un tiempo al capturar los datos en los siguientes ambientes exteriores:

AMBIENTE	NO. DE ESPACIOS
Alberca	1
Estacionamiento	1
Jardín	1

[Proceder al Ingreso de Datos](#)

Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN EXTERIOR - REGISTRO DE DATOS

Datos de la Luminaria en Jardín

Largo * m. Ancho * m. Tipo de lámpara * Incandescente de 50 watts

No. de lámparas en el ambiente *

No. de lámparas funcionales en el ambiente *

Horas de uso por día de la lámpara

Mantenimiento en un periodo de 6 meses * Una vez

No. de apagadores en el ambiente *

La lámpara cuenta con difusor *

Con difusor Sin difusor



Los Campos con * son Requeridos.

Procesar los Datos Exteriores

Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN EXTERIOR - REGISTRO DE DATOS

Datos de la Luminaria en Estacionamiento

Largo * 10 m. Ancho * 5 m. Tipo de lámpara * Fluorescente Compacta de 9 watts

No. de lámparas en el ambiente * 5

No. de lámparas funcionales en el ambiente * 4

Horas de uso por día de la lámpara 10

Mantenimiento en un período de 6 meses * Dos veces

No. de apagadores en el ambiente * 4

La lámpara cuenta con difusor *

Con difusor Sin difusor



Los Campos con * son Requeridos.

Procesar los Datos Exteriores

Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulo del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN EXTERIOR - RECOMENDACIONES

Lista de Ambientes

Ambiente: Alberca [Ver Recomendaciones](#)

Se recomienda sustituir por el siguiente tipo de lámpara:



LÁMPARA PROPUESTA

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	3	Sí	Sí

Usando este tipo de lámpara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.012	80

EMISIONES DE CO2

Con el uso de este tipo de lámpara, las emisiones de CO2 se ven disminuidas de la siguiente forma:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.03	0.01	66.7

Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.

Al concluir el ciclo para procesar los datos de los ambientes se muestra en una tabla el diagnóstico energético donde por referencia también se muestran los datos ingresados de cada ambiente.



CONACYT **SENER** SECRETARÍA DE ENERGÍA

SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

Hotel: [Ozkar-in](#)

ILUMINACIÓN EXTERIOR - DIAGNÓSTICO

Datos de Ambientes

Ambiente	Tipo de Lámpara	No. Lámparas	No. Lámparas funcionales	No. Apagadores
Alberca	Incandescente de 15 watts	4	4	2
Estacionamiento	Fluorescente Compacta de 9 watts	5	4	4
Jardin	Incandescente de 50 watts	2	2	2

Diagnóstico de Ambientes

Ambiente	Cumple NOM007	Cumple NOM025	Nivel Iluminación	Carga Anual (Kw)	Gasto Anual (Kw)	Emisión Anual CO2 (Kg)
Alberca	No	Si	Excedido	259.20	309.60	10.84
Estacionamiento	Si	No Aplica	No Aplica	162.00	194.40	8.13
Jardin	No	Si	Excedido	360.00	432.00	18.07

*Luminaria.

[Proceder a las Recomendaciones](#)

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

Al proceder a las recomendaciones se accede a una página donde se muestra una lista con los ambientes procesados y al seleccionar uno de la lista se muestran sus respectivas recomendaciones para cada ambiente donde se recomienda cambiar a un cierto tipo de luminaria la cual es mostrada como referencia también aparecen los ahorros de energía eléctrica, la demanda de esta nueva luminaria, las emisiones y ahorros de CO2:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN EXTERIOR - RECOMENDACIONES

Lista de Ambientes

Ambiente:

Se recomienda sustituir por el siguiente tipo de lámpara:



LÁMPARA PROPUESTA

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	3	Sí	Sí

Usando este tipo de lámpara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.012	80

EMISIONES DE CO2

Con el uso de este tipo de lámpara, las emisiones de CO2 se ven disminuidas de la siguiente forma:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.03	0.01	66.7

Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio Salir (oscurestrada)
Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

ILUMINACIÓN EXTERIOR - RECOMENDACIONES

Lista de Ambientes

Ambiente:

Se recomienda sustituir por el siguiente tipo de lámpara:



LÁMPARA PROPUESTA

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	3	Sí	Sí

Usando este tipo de lámpara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.006	94

EMISIONES DE CO2

Con el uso de este tipo de lámpara, las emisiones de CO2 se ven disminuidas de la siguiente forma:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.05	0	100

Copyright © 2013 por CCADET - UNAM. Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES



Inicio
Salir (oscurestrada)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

Hotel: [Ozkar-in](#)

ILUMINACIÓN EXTERIOR - RECOMENDACIONES

Lista de Ambientes

Ambiente:

Se recomienda sustituir por el siguiente tipo de lámpara:



LÁMPARA PROPUESTA

Tipo de Lámpara	Potencia (kW)	Cumple NOM007	Cumple NOM025
Led	3	Sí	Sí

Usando este tipo de lámpara su demanda sería de:

Demanda (kW)	Ahorro (%)
0.006	94

EMISIONES DE CO2

Con el uso de este tipo de lámpara, las emisiones de CO2 se ven disminuidas de la siguiente forma:

Emisión actual (kg)	Emisión utilizando esta lámpara (kg)	Reducción de emisión (%)
0.05	0	100

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.

Refrigeración:

Al seleccionar el módulo de refrigeración de inicio se muestra una primera página para el ingreso de datos relacionados con las características de la nevera como medidas, espesor, aislamiento, puertas de vidrio y motores y al llenarla se pasa a una segunda la cual contiene las características de los productos como la temperatura en el cuarto de enfriamiento, la temperatura a la que se encuentra el producto refrigerado y el nuevo producto a ingresar, el sistema puede admitir una gran cantidad de productos:







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio
Salir (oscurestrada)

Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

REFRIGERACIÓN - DATOS ENTRADA

Cuarto Frío o Nevera

Medidas

Largo del Cuarto Frío o Nevera: m.

Altura del Cuarto Frío o Nevera: m.

Ancho del Cuarto Frío o Nevera: m.

Esesor y Aislamiento

Techo => Esesor: m. Aislamiento:

Pared => Esesor: m. Aislamiento:

Piso => Esesor: m. Aislamiento:

Puertas de Vidrio

Número de Puertas de Vidrio:

NOTA: Si no tiene puerta de vidrio coloque cero.

Motores (Ventiladores)

Carga Total de los Motores (Ventiladores): HP.

NOTA: La Carga total es de los Motores que estan dentro del cuarto frío o nevera.

[Proceder al Ingreso de Datos](#)

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.







SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES

Inicio
Salir (oscurestrada)

Hotel: [Ozkar-in](#)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

REFRIGERACIÓN - DATOS ENTRADA PRODUCTO

Enfriamiento

Temperatura

Temperatura del Cuarto de Enfriamiento: °C

Espacio a Refrigerar

Volumen del Espacio a Refrigerar es: X = 1793.43 ft³.

Producto

A Temperatura Ambiente

kgs/día de

kgs/día de

kgs/día de

kgs/día de

A Temperatura Refrigerada

kgs/día de

kgs/día de

kgs/día de

kgs/día de

[Proceder a las Recomendaciones](#)

Copyright © 2013 por CCADET - UNAM.
Todos los derechos reservados.



SENER
SECRETARÍA DE ENERGÍA



SIMAE:
SISTEMA MULTIAGENTE PARA LA
GESTIÓN DE LA EFICIENCIA ENERGÉTICA EN
HOTELES



Inicio Salir (oscurestrada)

Módulos del Sistema

ENCUESTA

DIAGNÓSTICO ENERGÉTICO

RECOMENDACIÓN DE MEDIDAS DE AHORRO

EVALUACIÓN / IMPLEMENTACIÓN DE MEDIDAS DE AHORRO

MONITOREO DE LAS MEDIDAS DE AHORRO

FACTURACIÓN DE CONSUMO

Hotel: [Ozkar-in](#)

REFRIGERACIÓN - DIAGNÓSTICO

Datos Ingresados

Medidas:

El Largo del Cuarto Frío o Nevera es: 8.53 m.
El Altura del Cuarto Frío o Nevera es: 2.44 m.
El Ancho del Cuarto Frío o Nevera es: 2.44 m.

Espesor y Aislamiento:

Techo => Espesor: 0.1 m. Aislamiento: Poliestireno.
Pared => Espesor: 0.1 m. Aislamiento: Poliestireno.
Piso => Espesor: 0.15 m. Aislamiento: Cemento.

Puertas de Vidrio:

El Número de Puertas de Vidrio es: 1.

Motores (Ventiladores):

La Carga Total de los Motores (Ventiladores): 0.2 HP.

Ciudad y Temperatura:

El Estado Donde se Encuentra es: Mexico
La Temperatura Ambiente (B.S.) es: 28.2 °C.
La Temperatura Piso (B.H.) es: 15 °C.
Temperatura del Cuarto de Enfriamiento es: 1.7 °C.

Espacio a Refrigerar:

El Volumen del Espacio a Refrigerar es X = 1793.43 ft³.
El Valor Inmediato Superior a X es: 2000 ft³.
El Valor Inmediato Inferior a X es: 1500 ft³.

Productos:

Temperatura Ambiente

1 => 100 kgs/día de Naranjas
2 => 30 kgs/día de Camotes
3 => 50 kgs/día de jamon curado
4 => 40 kgs/día de Cordero fresco

Temperatura Refrigerada:

1 => 200 kgs/día de Atun
2 => 150 kgs/día de Chuleta de ternera fresca
3 => 300 kgs/día de Frutas secas
4 => 100 kgs/día de Ganso fresco

Diagnóstico Obtenido

Diferenciales de Temperatura

El Diferencial de Temperatura de la Pared es: 26.5 °C.
El Diferencial de Temperatura del Piso es: 13.3 °C.

Aislamiento

Carga del Calor del Techo 168.54 Kcal/m²/24h.
Carga del Calor del Pared 168.54 Kcal/m²/24h.
Carga del Calor del Piso 368.14 Kcal/m²/24h.

Transmisión por Cargas

La Transmisión por Cargas del Techo es: 3507.86 kcal/24hrs.
La Transmisión por Cargas del Pared Norte es: 3507.86 kcal/24hrs.
La Transmisión por Cargas del Pared Sur es: 3507.86 kcal/24hrs.
La Transmisión por Cargas del Pared Este es: 1003.42 kcal/24hrs.
La Transmisión por Cargas del Pared Oeste es: 1003.42 kcal/24hrs.
La Transmisión por Cargas del Piso es: 7662.25 kcal/24hrs.
La Transmisión por Cargas Total es: 20192.66 kcal/24hrs.

Carga por Cambios de Aire

El Cambio de Aire para el Equipo es de: 12.83 hrs.
El Factor de calor remivido: 0.41 BTU/ft³.
La Carga por Cambios de Aire es: 9431.24 BTU/24hrs.
La Carga por Cambios de Aire es: 2375.73 kcal/24hrs.

Cargas Adicionales

La Carga para los Motores Eléctricos es: 15000 BTU/24hrs.
La Carga para la Iluminación Eléctrica es: 18370.58 BTU/24hrs
La Cargas por puertas de Vidrio es: 19200 BTU/24hrs
Las Cargas Adicionales Totales son: 52570.58 BTU/24hrs
Las Cargas Adicionales Totales son: 13242.53 kcal/24hrs

Productos

Carga a Temperatura Ambiente:

La Carga Total de los Productos es: 17824.52 BTU/24hrs.
La Carga Total de los Productos es: 4490 kcal/24hrs.

1 => 220.46 lbs/día a 47.7 °F = 9464.35 BTU/24hrs.
2 => 66.138 lbs/día a 47.7 °F = 2492.28 BTU/24hrs.
3 => 110.23 lbs/día a 47.7 °F = 2839.3 BTU/24hrs.
4 => 88.184 lbs/día a 47.7 °F = 3028.59 BTU/24hrs.

Carga a Temperatura Refrigerada:

La Carga Total de los Productos es: -1393.6 BTU/24hrs.
La Carga Total de los Productos es: -351.05 kcal/24hrs.

1 => 440.92 lbs/día a -0.072000000000003 °F = -24.13 BTU/24hrs.
2 => 330.69 lbs/día a -1.062 °F = -256.37 BTU/24hrs.
3 => 661.38 lbs/día a -3.06 °F = -728.58 BTU/24hrs.
4 => 220.46 lbs/día a -3.06 °F = -384.53 BTU/24hrs.

La Carga Total de los Productos es: 4138.95 kcal/24hrs

Resultados Finales

La Carga Total de Refrigeración es: 39949.87 BTU/24 hrs.
Se Agregar 10% por Factor de Seguridad: 3994.99 BTU/24 hrs.
La Carga Total con Factor de Seguridad es: 43944.86 BTU/24 hrs.
Requerimientos de Enfriamiento en Ton/Ref: 0.91 kcal/hrs.

La imagen anterior muestra el diagnóstico energético de la nevera la cual calcula las cargas por transmisión, por cambio de aire, por cargas eléctricas de motores, por puertas de vidrios y de los productos obteniendo una carga general para poder dar las recomendaciones, en este caso el algoritmo sigue en construcción por lo que la recomendaciones aún no están formuladas pero ya se tiene un gran avance en la generación del diagnóstico.

CONCLUSIONES

Capítulo 7

VII.- Conclusiones

Con el desarrollo del Sistema Web Inteligente para la gestión del ahorro de energía enfocado en las áreas de refrigeración e iluminación es capaz de beneficiar al sector hotelero en materia de gestión eficiente de la energía logrando los siguientes objetivos:

- ✓ Fomentar el ahorro de energía.
- ✓ Mejorar la eficiencia energética de los procesos de forma sistemática.
- ✓ Reducir los costes y mejorar la competitividad.
- ✓ Disminuir las emisiones de gases que provocan el cambio climático.
- ✓ Demostrar el cumplimiento legal en materia energética.
- ✓ Fomentar la innovación tecnológica.
- ✓ Incrementar el aprovechamiento de energías renovables o energías excedentes propias o de terceros.
- ✓ Mejora continua gracias a la integración con otros sistemas de gestión de calidad y ambientales.
- ✓ Cultura energética.
- ✓ Mejor control energético.
- ✓ Mejora continua.

Cabe señalar que se trata de un sistema nuevo para gestionar la eficiencia energética que con el tiempo y uso que se le dé, irá mejorándose para tal vez convertirse en una herramienta de gran utilidad y de gran difusión con la cual podamos estandarizar la gestión energética de nuestro país, y por qué no, llevarlo a hoteles en todo el mundo.

México se renovó y redefinió en un nuevo contexto al reducir el consumo de energía a través de la implementación de un Sistema Web Inteligente de Gestión de la Energía ya que fue la clave para combatir la escasez de energía y el consecuente aumento del costo de la misma de igual forma ayuda a reducir las emisiones de Gases del Efecto Invernadero logrando un mayor equilibrio entre economía y medio ambiente aportando un gran beneficio a nuestro país.

Al desarrollar el Sistema Web Inteligente con el cual se efectúa el diagnóstico, monitoreo y evaluación de las medidas de ahorro de energía se logró obtener el menor impacto al medio ambiente, una mejor administración de la eficiencia energética, un uso racional y eficiente de la energía, gestión de mejora continua del sistema, una gran reducción del consumo de energía y asegurar el cumplimiento de normas y reglamentos técnicos , todo esto, contribuye a la formación de una cultura energética y ambiental que pondrá en alto el nombre de nuestro país.

BIBLIOGRAFÍA

Capítulo 8

VIII.- Bibliografía

- 1.- Artículo “pyme”, se define en el “Diccionario de la lengua española”. Madrid: Real Academia Española, vigésima segunda edición, 2001.
- 2.- “Las PYMES su erigen como las principales creadoras de empleo”, artículo en el diario *Aproximate*, de mayo de 2010.
- 3.-Secretaria de turismo http://www.sectur.gob.mx/es/sectur/sect_Programa_de_Apoyo_a_la_Competitividad_de_las_pymes. [julio 2013]
- 4.- Monteagudo, J., Armas, M. (2005). *La gestión energética: una alternativa eficaz para mejorar la competitividad empresarial*. Energética
- 5.- KENNEY, W.F. *Energy Conservation in the Process Industries*. / .F.Kenney./s.1/: Academia Press, 1969, 329 p.
- 6.- Modelo del Sistema de Gestión de la Energía ISO 50001.
- 7.- Norma ISO 50001. www.iso.org/iso/iso_50001_energy-es.pdf [agosto 2013]
- 8.- Renate Lippert , Teaching problem solving in mathematics and science with expert systems, *Journal of Artificial Intelligence in Education* Volume 1, Issue 3.
- 9.- *Sistemas Expertos y Modelos de Redes Probabilísticas* (Academia Española de Ingeniería, 1998) E. Castillo, J.M. Gutiérrez, and A.S. Had.
- 10.- Rajan, S., Jairath, A. (2011). *Cloud Computing: The Fifth Generation of Computing*. Communication Systems and Network Technologies (CSNT).
- 11.- Conpas.net. Definición SaaS: Software como Servicio en la empresa. 2011. < http://www.conpas.net/por_que_saas.html>. [Agosto 2013]
- 12.- Tata Communications. *Infrastructure-a- a-Service, Fulfilling the Promise of Cloud Computing*. 2012.
- 13.-Eguíluz, J. *Introducción a XHTML*. 2010. < <http://www.librosweb.es/xhtml/index.html>>. [Agosto 2013]
- 14.- World Widw Web consorcium www.w3.org/TR/xhtml1/diffs.html#h-4.4. [Agosto 2013]

- 15.- «Cascading Style Sheets (CSS) Snapshot 2010». World Wide Web Consortium [Agosto 2013]. Definiciones del CSS (2013)
- 16.- Página oficial de javascript <http://librosweb.es/javascript/> [Agosto 2013]
- 17.- Página oficial de jquery <https://jquery.org/> [Agosto 2013]
- 18.- Página oficial de php <http://php.net/> [Agosto 2013]
- 19.- “PHP 5.5.0 released” <http://php.net/archive/2013.php#id2013-06-20-1> (en inglés). [Agosto 2013].
- 20.- “What is Web 2.0, Ideas, technologies and implications for education”, escrito por Paul Anderson.
- 21.- Characteristics of Web 2.0 Technology TechPluto Staff [Agosto 2013]
- 22.- Planeta Web 2.0, libro de Cristóbal Cobo Romani y Hugo Pardo 2010.
- 23.- Framework para aplicaciones web <http://www.phpframeworks.com/> [Agosto 2013]
- 24.- Winesett, Jeffrey (2010). *Agile de desarrollo de aplicaciones Web con Yii1.1 y PHP5*. Packt Publishing. ISBN 978-1-84719-958-4 .
- 25.- Makarov, Alexander (2011). *Yii Cookbook 1.1 Desarrollo de aplicaciones*. Packt Publishing. ISBN 978-1-84951-548-1.
- 26.- Yii framework <http://yiiframework.wordpress.com/> [Agosto 2013]
- 27.- inicio rapido de Yii framework <http://www.yiiframework.com/doc/guide/quickstart.what-is-yii> [Agosto 2013]
- 28. -The DCI Architecture: A New Vision of Object-Oriented Programming - Trygve Reenskaug and James Coplien - March 20, 2009.
- 29.- Página oficial de apache. http://httpd.apache.org/ABOUT_APACHE.html [Agosto 2013]
- 30.- Pagina oficila de php donde describe el uso de sql .<http://php.net/manual/es/book.mysql.php> [Agosto 2013]
- 31.- Wikipedia. SQL. 2010. < <http://es.wikipedia.org/wiki/SQL>>. [Agosto 2013]