



**UNIVERSIDAD LASALLISTA
BENAVENTE**

ESCUELA DE INGENIERÍA EN COMPUTACIÓN

Con estudios incorporados a la
Universidad Nacional Autónoma de México
CLAVE: 8793-16

**“Desarrollo de una INTERFAZ WEB para la
administración de proyectos dirigida
a empresas con multisitios”**

T E S I S

QUE PARA OBTENER EL TÍTULO DE:
INGENIERA EN COMPUTACIÓN

PRESENTA:
ERIKA LÓPEZ LARA

ASESOR:
ING. MARÍA IVONNE VILLAGÓMEZ TORRES

CELAYA, GTO.

SEPTIEMBRE DE 2009



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

Como en todos los retos que la vida nos presenta se tiene que luchar con toda la energía y toda la intensidad para conseguir lo que se quiere y hoy por fin he logrado cumplir con este gran objetivo.

Los siguientes agradecimientos son para todas aquellas personas que estuvieron a mi lado y que consciente o inconscientemente pusieron de su parte para que yo culminara con esta importante etapa en mi vida.

Primero y antes que nada, dar gracias a **Dios**, mi Señor por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente y por haber puesto en mi camino a aquellas personas que han sido mi soporte y compañía durante todas la etapas de mi vida.

A mis padres Raquel y Francisco que son mi mayor y mejor apoyo aquí en la tierra porque gracias a sus esfuerzos, enseñanzas, comprensión y amor he podido salir adelante, librando cualquier obstáculo y convirtiéndome en una mujer de bien, reflejándose ahora en la culminación de este gran esfuerzo.

A mi esposo Luis Enrique gracias por estar a mi lado, gracias por el amor, la paciencia y el apoyo que das día a día y que se ve reflejado en las cosas que hago y en las ganas de cumplir lo que me propongo.

A mi hijo Iván que ha dado un giro radical a mi vida y me ha cambiado la forma de ver las cosas, es la mejor recompensa que se pueda dar a una persona y me ha sido otorgada a mí. Gracias por hacer que todos los días sean diferentes y nuevos para mí.

A mis amigos, los que he cosechado en el camino, esos, los verdaderos, que están conmigo en las buenas y en las malas, ya que con su buena energía me han ayudado a no desanimarme y culminar con mis metas.

A mis profesores y asesores que me hicieron pasar algunas noches de insomnio pero que gracias a ellos y a sus enseñanzas hoy me considero una profesionalista afortunada, una mujer responsable y una persona llena de ideales y objetivos.

A todos ustedes mil gracias sin su colaboración tal vez esto no se hubiera hecho realidad.

Uno puede devolver un préstamo de oro, pero está en deuda de por vida con aquellos que son amables (Proverbio).

Índice

Introducción

Capítulo I. Análisis de la situación actual	5
1.1 Antecedentes.....	6
1.2 Empresa a la que se dirige	7
1.2.1 Identidad o giro	9
1.2.2 Visión y Misión.....	11
1.2.3 Objetivo principal de la empresa	11
1.3 Competencia.....	12
1.3.1 Imagen Corporativa	12
1.3.2 Impacto tecnológico.....	14
1.4 Clientes.....	16
1.4.1 Ventajas.....	19
1.4.2 Acceso y seguridad	20
Capítulo II. Estructura de la interfaz WEB	21
2.1 Objetivo General.....	22
2.2 Objetivos Específicos	22
2.3 Sitios de Internet.....	23
2.3.1 Aspectos que se deben evitar al hacer un sitio WEB	24
2.3.2 Pasos para la realización de un sitio de Internet	25
2.4 Uso de la tecnología .NET para la creación de la interfaz.....	29
2.4.1 Microsoft .NET Framework	29
2.4.2 Introducción a ASP.NET	34
2.4.3 Visual Studio.NET	41
2.5 Uso de bases de datos para aplicaciones Web dinámicas	43
2.5.1 Microsoft SQL Server. Plataforma de Base de Datos	44
2.5.2 Interacción de SQL Server y la tecnología .NET	45
2.6 Otras herramientas	47
Capítulo III. Análisis y Diseño del sitio	50
3.1 Análisis de la página Web	51
3.1.1 Estudio de la situación actual	51
3.1.2 Definición del problema	52
3.1.3 Funcionamiento de la Interfaz Web para la solución del problema.....	53
3.1.4 Estudio de factibilidad.....	57
3.1.5 Mapa de Contenidos del Sitio.....	59
3.2 Diseño del Sitio.....	61
3.2.1 Estilo gráfico	66

3.2.2	Creación de base de datos y tablas	70
3.3	Instalación de Visual Studio y Microsoft SQL Server	76
3.4	Uso de controles y herramientas de ASP.NET en la aplicación.....	79
3.5	Servidor Web	82
Capítulo IV. Implementación y entrega.....		86
4.1	Desarrollo	87
4.1.1	Elaboración de páginas informativas.....	88
4.1.1.1	Página Empresa	89
4.1.1.2	Página Nuestros Servicios.....	91
4.1.1.3	Página Clientes.....	92
4.1.1.4	Página Contáctanos	93
4.1.2	Elaboración de aplicación dinámica	94
4.1.2.1	Página Zona Proyectos	95
4.1.2.2	Página Opciones para el Cliente	100
4.1.2.3	Zona Administrativa	108
4.1.2.4	Página Opciones Administrativas para el Líder.....	110
4.1.2.5	Página Opciones para el Administrador	115
4.2	Seguridad de aplicaciones Web con ASP.NET	123
4.3	Mantenimiento de la aplicación	125

Conclusiones

Bibliografía

INTRODUCCIÓN

Debido al contacto que he tenido con diferentes empresas de desarrollo e instalación de proyectos, llámese eléctrica, de red, telefonía celular, ingeniería civil, etc., cuya característica común es un alto porcentaje de trabajo de campo o dispersión de frentes de trabajo y al observar la carencia que éstas tienen para controlar y actualizar sus proyectos de manera inmediata, provocándoles retrasos, incumplimientos, falta de organización, fuga de materiales, etc.; y si además a ésto le agregamos que no cuentan con una herramienta que proporcione al cliente la facilidad de visualizar el avance de su obra, veo la necesidad de crear una interfaz que funcione a través de Internet, aprovechando el cambio que nuestras actividades diarias han sufrido con el incremento en el uso de esta *vida en línea* y como el término lo refiere la aplicación Web permitirá que se efectúe la actualización de la información en tiempo real y la administración integral de los proyectos en curso.

Para la realización de esta tesis se consideró una empresa con las características antes mencionadas, su nombre GADB Instalaciones cuyo giro principal es la instalación de antenas, radio bases celulares y fibra óptica. Se realizó una investigación de las actividades que la empresa realiza en cuanto a la administración de proyectos y se espera que con el uso de una herramienta tecnológica adecuada se llegue a una buena administración de las tareas que incluye un proyecto mejorando de manera significativa los resultados de la empresa en cuanto a rendimiento y tiempos de entrega.

Por lo anterior, el propósito de este trabajo es la realización de una aplicación Web dinámica que servirá para una mejor administración de la información y por consiguiente una toma de decisiones más oportuna en la ejecución de los proyectos, evitando así, fugas de dinero, de materiales y retrasos en la información y en el cumplimiento ante el cliente, dándole un valor agregado a la empresa y una ventaja competitiva ante las demás, aportando un servicio

tecnológico de vanguardia. Además permitirá la interacción de los usuarios autorizados para la actualización, seguimiento y análisis en línea de los diferentes proyectos con los que cuenta la empresa.

Capítulo I. ANÁLISIS DE LA SITUACIÓN ACTUAL

1.1 Antecedentes.

La administración se define en términos sencillos como la habilidad o la práctica de controlar, dirigir o planear algo. “El acto de dirigir o administrar con un propósito”. Dicho de otra manera es el esfuerzo que se realiza de planear, organizar y movilizar a las personas y los recursos orientados a un propósito dado. En el caso de la administración de proyectos, nos referimos a las capacidades y habilidades que hacen que el proyecto sea factible y real. ¹

Debido a la creciente demanda de empresas que se dedican a desarrollar proyectos de instalación y servicios a terceros, empresas de todo tipo, eléctricas, de comunicaciones, civiles, de telefonía celular, de red, etc. y dado que la administración de proyectos se ha convertido en la piedra angular de las organizaciones y al enfoque de suma importancia que se le ha dado en la toma de decisiones, se considera de manera importante poner énfasis en una buena y efectiva administración.

Los cambios tecnológicos, la necesidad de introducir nuevos productos al mercado, las cambiantes exigencias de los consumidores, entre otras cosas, incrementan el flujo de operaciones en una organización, provocando que los métodos de administración convencionales sean inadecuados. Por esta razón la administración de proyectos es importante y crear una herramienta que ayude a la buena ejecución de ésta es indispensable.

Dado que cada vez son más los equipos de trabajo con proyectos multisitios, es decir que se encuentran dispersos geográficamente, el intercambio eficaz de diseños e información resulta un factor clave para determinar el éxito o fracaso de dichos proyectos. Además que las compañías que trabajan de este modo

¹ Klastorin, Ted. Administración de Proyectos. Editorial Alfaomega.

buscan cada vez más la integración de su empresa y sus usuarios externos (clientes, empleados, proveedores, etc.).

La buena planeación de un proyecto permite a las organizaciones ser más eficientes al poner énfasis en las necesidades concretas de una situación o un grupo de personas. Con una buena herramienta que permita la administración de las tareas de un proyecto y la involucración coherente y coordinada de los diferentes actores; los recursos (materiales y humanos) serán mejor utilizados y la eficiencia en general se incrementará.

Con este tipo de herramienta los usuarios podrán aprovechar de mejor manera los recursos críticos cuando éstos están limitados en cantidad y/o tiempo de disponibilidad, además de que no cuentan con varias versiones de la misma información y todos están trabajando en el mismo documento, ayudando así a realizar acciones concisas y efectivas para obtener el máximo beneficio.

Para poner en práctica esta herramienta, consideré a una empresa cuya actividad preponderante es el trabajo en sitios dispersos geográficamente, que actualmente no cuenta con una herramienta que le facilite la buena planeación de sus proyectos, es por eso que me di a la tarea de crear una interfaz WEB que le ayudara con este cometido.

1.2 Empresa a la que se dirige.

GADB Instalaciones es una empresa constituida en el año 2001, inicialmente para realizar trabajos de suministros, instalación, mantenimientos y asesoría en PC's y redes. En la Fig. 1.1 se muestra el Organigrama de la Empresa.

Fundada por el Ing. Lorenzo Matty Manzo, egresado de la carrera de Ingeniería Industrial y de Sistemas del Instituto Tecnológico de Sonora.

Ha tenido la oportunidad de incursionar en el área de telecomunicaciones y redes inalámbricas con la empresa Bachoco, S.A. de C.V. y Nortel Networks México. Esta última dándole la oportunidad de trabajar como Project Manager en muchos de sus proyectos para sus clientes en México.

La empresa está ubicada en la ciudad de Celaya, Guanajuato. Lugar geográfico por donde estratégicamente se trasladan a cualquier parte del país.

La compañía se preocupa por mantener sus conocimientos y experiencia en continua actualización, evolucionando sistemática y paralelamente al avance vertiginoso de los nuevos desarrollos y tecnologías, con el propósito de estar siempre a la vanguardia y nunca por debajo de sus competidores.

ORGANIGRAMA DE LA EMPRESA

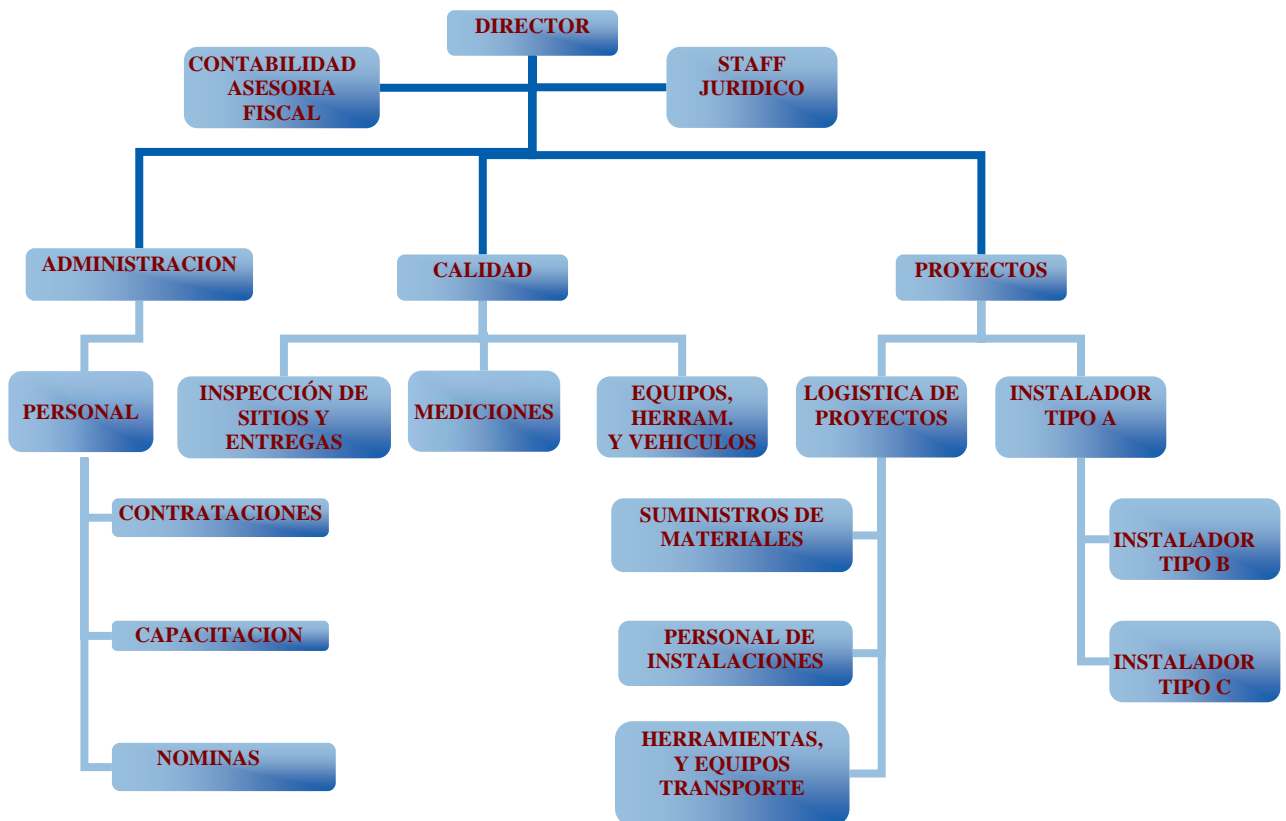


Fig. 1.1 Organigrama de la empresa GDB Instalaciones.

1.2.1 Identidad o giro.

Esta empresa se especializa en la instalación de equipos RTU`s, Newbridge, Radio Bases y Equipos de fibra óptica TN-1X, 2X, 4X.

Sus políticas de servicio incluyen:

- Incorporar personal capacitado para el transporte de los equipos, montaje de radio bases celulares, líneas y antenas, repetidores, plantas de fuerza, bancos de baterías, alarmas, herrajes y escalerillas. Además de trabajos varios como pintura de torres y mantenimiento primario de sitios celulares, etc.
- Ofrecer servicios de instalación, soporte, mantenimiento y capacitación para anticipar y satisfacer las necesidades del cliente, antes, durante y después del servicio. Preocupándose por minimizar los tiempos de instalación y cuidando la economía de los proyectos para hacer éstos más redituables con el ahorro de costos de herramientas, equipos y materiales. Brindando a sus clientes la herramienta más eficaz para la solución de sus problemas.
- Trabajar bajo las más estrictas normas de calidad y funcionalidad que sus clientes requieren. Proporcionando soluciones integrales a la medida de sus necesidades, a través de conductos especializados de distribución, con todo el soporte de la organización y una mejor estrategia.

En las Fig. 1.2, 1.3 y 1.4 se pueden apreciar algunas de las actividades que la empresa realiza, pudiendo constatar que es el trabajo de campo la base del funcionamiento de sus proyectos.



Fig. 1.2 Instalación de torres.



Fig. 1.3 Instalación de equipo satelital.



Fig. 1.4 Instalación de antenas.

1.2.2. Visión y Misión.

Toda empresa que quiera lograr éxito en el campo laboral en estos días, es necesario que esté regida por una Visión y una Misión, a continuación se describen estos elementos fundamentales para el caso de GADB Instalaciones :

Visión:

Ser una empresa líder en el ramo y estar capacitada para incorporar y asimilar rápidamente los avances tecnológicos. Conocemos las herramientas y entendemos como aplicarlas para desarrollar nuestros objetivos.

Misión:

Ser prestadores de servicios de instalación, soporte, mantenimiento y capacitación de la más alta calidad para empresas y compañías dedicadas a proveer servicios de comunicaciones, cuidando estrictamente la productividad, normas y estándares que las rigen.

1.2.3 Objetivo principal de la empresa.

Como empresa prestadora de servicios, su principal objetivo es ofrecer la mayor calidad, funcionalidad y acabados de primera al menor costo para lograr competir de manera efectiva en el mercado actual, teniendo siempre como fin alcanzar la excelencia.

1.3 Competencia.

La ventaja comparativa de una empresa radicaría en su habilidad, recursos, conocimientos, atributos, etc., de los que dispone, los mismos de los que carecen sus competidores o que estos tienen en menor medida y que hace posible la obtención de unos rendimientos superiores a los de aquellos. A eso se refiere la competencia. Una empresa se define como competitiva si presenta mejoras constantes de calidad, innovación y aumenta de forma considerable su productividad.

En la última década, la competencia entre las empresas de servicios ha sido fuerte, agresiva y constante. Intensas guerras de precios, cambios rápidos en la tecnología y un ambiente incierto, han dificultado la retención de clientes. El dinero que se gasta en avances tecnológicos mejora la eficiencia de los servicios o del desempeño; sin embargo, el costo para adquirir cualquier herramienta de vanguardia que proporcione a la empresa mayor proyección resulta ser bastante elevado.

En esta etapa de cambios, las empresas buscan elevar índices de productividad, lograr mayor eficiencia y brindar un servicio de calidad, lo que las está obligando a adquirir herramientas tecnológicas que les permitan mostrar a sus clientes la constante actualización de sus métodos, sin dejar de lado al elemento humano, desarrollando un trabajo en equipo y teniendo como fin cumplir con las expectativas del cliente.

1.3.1 Imagen Corporativa.

Siendo un área generalmente relegada a un segundo plano, la imagen corporativa de una empresa es uno de los más importantes elementos de ventas.

En el mercado tan competitivo y cambiante la imagen corporativa es un elemento definitivo de diferenciación y posicionamiento.

Así como las empresas deben adecuarse a los cambios con velocidad de igual manera deberán adecuar su imagen, para transmitir dichos cambios. La imagen corporativa es la manera por la cual trasmite, quién es, qué es, qué hace y como lo hace.

Imagen Corporativa: es la personalidad de la empresa, su razón de ser, dicha imagen tiene que estar impresa en todas partes que involucren a la empresa para posicionar a ésta en el mercado.

La imagen es utilizada para transmitir y manifestar su identidad, mostrándola al público, dando a conocer sus normas y comportamientos. El logotipo es esa imagen visual que le proporciona a la empresa su identidad.

Al ver su logotipo constantemente este se irá quedando fijo en la mente de las personas. Cuando alguien piensa en algo referente a su producto o servicio se imaginara su logotipo como opción. Las imágenes de empresas tienen una gran influencia en el éxito global de una compañía. Abarcan desde un logo, hasta el estilo de la casa, el diseño o decoración interior y los uniformes de la empresa.

Fig. 1.5 Logotipo de la empresa GADB Instalaciones.

¿Por qué quise hacer referencia o describir un poco el tema de la imagen corporativa de una empresa?, porque parte de la imagen se utilizará en la realización del proyecto, porque además de una página que ayude a administrar los proyectos en campo, también será una página WEB que utilice la empresa como presentación, una página informativa que ayude a hacer esa conexión de personalidad con sus clientes.



Fig. 1.5 Logo de la empresa.

Dentro de la imagen de una empresa también se encuentra como se vió anteriormente, las normas y comportamientos, a continuación se describirán algunas normas que se definieron cuando GADB Instalaciones fue creada.

- Eliminar costos innecesarios que no aporten valor agregado a nuestros clientes a fin de mejorar los rendimientos.
- Desarrollar procesos efectivos de comunicación que permitan el control de todos nuestros procesos internos como externos. La comunicación es la base del buen entendimiento.
- Continuar con el desarrollo de la tecnología y los sistemas de operación para brindar a nuestros clientes un servicio superior y eficiente.
- Establecer la nueva imagen de nuestra empresa a través de la organización para satisfacer todas las necesidades del cliente por medio de nuestros servicios.

1.3.2 Impacto tecnológico.

Una de las normas que se definieron con la empresa al momento de su creación se refiere al desarrollo de tecnología que le permita dar a sus clientes un mejor servicio.

Es de suma importancia que las empresas cuenten con la información oportuna para tomar las mejores decisiones en el momento adecuado. En esta situación las nuevas tecnologías tienen un papel relevante, ya que permiten obtener y procesar mucha más información que los medios manuales. Así que las empresas invierten en ellas.

Sin lugar a dudas, las nuevas tecnologías han llevado consigo un cambio espectacular y drástico en todas las empresas. En los últimos años cabe destacar al Internet como el elemento revolucionario. Con el tiempo, Internet se ha hecho imprescindible en cualquier empresa, independientemente del tamaño de ésta, y es un medio por el cual se pueden dar a conocer servicios con una cobertura inimaginable. Las empresas que cuentan con este tipo de tecnologías y que las utilizan para darse a conocer o como herramienta de trabajo son las más exitosas, porque el hecho de estar a la par de los avances tecnológicos, le aporta un plus a la imagen de éstas.

Ya existen empresas que operan en Internet con un ámbito de operaciones mundial y está comprobado el éxito que han tenido, porque el área que logran abarcar con este tipo de tecnología no lo supera ninguna otra.

La tecnología ha modificado la forma de identificar, crear, desarrollar y servir a la demanda. En los últimos años ha transformado también la oferta a la que puede acceder el consumidor; productos que hace años sólo estaban al alcance de pocos, hoy en día son productos que se utilizan en todos los hogares. El ritmo al que se suceden las innovaciones está transformando el mercado de manera radical.

¿Cuáles serían las ventajas con las que contaría la empresa con una herramienta tecnológica como una Página WEB?, veamos:

- Acceso a la empresa desde cualquier punto en donde exista un equipo de cómputo.
- En la parte comercial, contar con publicidad de forma constante.
- Ampliar su mercado desde un nivel local a un nivel nacional o internacional.
- Promover la comunicación con clientes y proveedores.
- Generar una buena imagen para los clientes.
- Aumentar el número de ventas y de clientes potenciales.
- Contar con una herramienta que le permita estar por arriba de sus competidores.
- El proyecto que se va a realizar además de proporcionar a GADB Instalaciones una página WEB informativa de la empresa, también contendrá una herramienta de control de actividades de los proyectos que desarrollan, es decir, que a través de la Internet y de manera automática, los involucrados en un proyecto (gerente o administrador, líder de proyecto y cliente) podrán dar de alta, eliminar, actualizar, y consultar las tareas de un determinado proyecto, con el fin de optimizar costos, tiempos de respuesta y evitar fugas de materiales y tiempos muertos.

1.4 Clientes

GADB Instalaciones tiene como enfoque principal la satisfacción de los clientes y ciertos principios que están dirigidos hacia ellos:

- Que los clientes reconozca a una compañía que proporciona inmejorables servicios, alto desempeño profesional y calidad en la entrega.
- Que los clientes reconozcan que somos gente dedicada a su trabajo.

- Se sientan satisfechos del producto con el cual contribuimos para su empresa.
- En el momento de pensar en servicios de instalación, soporte, mantenimiento o capacitación para su empresa, nos brinden su preferencia.

Además ofrecen a sus clientes una gama de servicios que se listan a continuación:

Presupuestos acorde a las necesidades de nuestros clientes.

Una de nuestras principales metas dentro de un proyecto, es que nuestros clientes estén satisfechos desde el comienzo de las actividades con lo que económicamente se va a desembolsar, ya que la cotización debe de estar acorde a la logística y planeación de lo que el cliente pide. Además tratamos de estar entre las empresas contratistas proveedoras de servicios de instalación de telefonía más económicas, sin perder el grado de calidad y servicio que GADB Instalaciones proporciona.

Análisis de instalaciones (Site Survey).

Una de nuestras recomendaciones primordiales para la exitosa planificación de las actividades de un proyecto es que personal de GADB Instalaciones visite los sitios contemplados dentro de un proyecto, para hacer el levantamiento de las necesidades de espacio, equipo, logística de traslado y materiales a utilizar para la instalación. Esto da una visión exacta del tiempo y mano de obra a necesitar para la instalación por sitio.

Personal profesional y técnico para los trabajos de instalación.

Contamos con el personal con la experiencia y capacitación para cada uno de los proyectos.

Calidad de instalación, estética y funcionalidad.

Nos hemos caracterizado en todos los proyectos donde hemos trabajado en proporcionar un servicio con organización, cuidando siempre los detalles de la instalación, la estética y sobre todo la funcionalidad e integración.

Control administrativo de los proyectos.

GADB Instalaciones, provee de personal profesional para el control de sus actividades con el cliente. Un coordinador que tenga como tarea de mantener al cliente informado de los avances de las instalaciones en el proyecto y se encargue de administrar los recursos de tiempos, costos y personal. Con esto nuestra empresa y el cliente trabajan juntos manteniendo una integración completa para el manejo de la información.

Garantía sobre los trabajos.

Todas nuestras instalaciones serán garantizadas por un período de 30 días a partir de la firma de aceptación del cliente. Esta garantía puede incrementarse dependiendo de la negociación del proyecto.

Mantenimiento preventivo y correctivo.

Ofrecemos los servicios de mantenimiento preventivo y correctivo y este será según el tipo de negociación que se dé con el cliente. Este podrá ser mediante contrato por un período determinado o bien por llamado y levantamiento de orden de trabajo.

Capacitación.

Podremos proporcionar capacitación sobre nuestro trabajo si el cliente así lo solicitara.

Confidencialidad.

Somos profesionales, y las empresas podrán estar tranquilas de toda la información que nos sea proporcionada será tratada confidencialmente por nuestro personal asignado a sus proyectos.

1.4.1 Ventajas.

Como se puede ver en el punto anterior la satisfacción del cliente para la empresa es lo más importante.

El proyecto de la página WEB que incluye una herramienta de control de proyectos en línea va a ayudar a la empresa a cumplir con esa satisfacción de la que hablabamos.

Algunas de las ventajas para el cliente de contar con esta herramienta serían:

- Permite interactuar con los miembros del equipo de trabajo de la empresa.
- Puede observar las actividades de los líderes del proyecto o de la empresa en sí, qué tienen que hacer, cuándo y en qué consisten.
- Fácil de acceder, se puede hacer desde cualquier lugar que tenga una conexión a Internet.
- Enterarse del avance del proyecto en tiempo real, que le ayudará a la toma de decisiones.
- Puede solicitar presupuestos en línea.

1.4.2 Acceso y seguridad.

El **acceso** se realizará de una forma totalmente segura a través de una interfaz WEB, cada uno de los participantes del proyecto tendrá su propia clave de acceso, con los permisos correspondientes, ya que no es la misma información que podrá acceder un líder de proyecto que la de un cliente, los perfiles de los usuarios estarán bien definidos de tal forma que las funciones y la información a las que cada uno tiene acceso deberán ser determinadas desde el diseño por el Gerente de la Compañía.

En cuanto a la **seguridad** el cliente podrá estar cien por ciento seguro que la información confidencial de su empresa no será utilizada o vista por ningún otro usuario que no tenga los permisos correspondientes. Esto garantizará la confidencialidad del proyecto.

Capítulo II. ESTRUCTURA DE LA INTERFAZ WEB.

2.1 Obetivo General.

Desarrollar una interfaz WEB que permita la interacción de los diferentes usuarios autorizados para la actualización, seguimiento y análisis en línea de los distintos proyectos con los que cuenta la empresa, no importando si éstos se realizan en sitios geográficamente distribuidos.

2.2 Objetivos Específicos.

Dentro de estos objetivos se describen tanto los objetivos técnicos, es decir, los inherentes al desarrollo o creación de la página, como los de contenido.

- Desarrollar una base de datos que contenga la información de los proyectos.
- Definir los niveles de seguridad para acceder a las diferentes herramientas de la página según el tipo de usuario.
- Estructurar las aplicaciones de la página.
- Instalar la página en un servidor WEB.
- Contar con una herramienta de vanguardia en cuanto a que se trata de una página Web que maneja base de datos para el control de proyectos de la empresa.
- Tener una ventana en el mundo WWW que sirva para la promoción de los servicios y novedades del negocio.
- Analizar las ventajas de la herramienta en la toma de decisiones.
- Controlar avance de obra vs. plan proyectado.
- Proporcionar información a líder de proyecto y al administrador para utilización, autorización y/o administración de recursos.
- Proporcionar información al cliente sobre el avance de su proyecto en línea, además de ser un medio para que ellos puedan contactar con la empresa.

2.3 Sitios de Internet.

Un Sitio de Internet o Sitio WEB es un conjunto de páginas de internet que hablan sobre un tema en particular y se almacenan en una misma dirección. Un sitio web puede tener distintos enfoques, puede ser la página personal de alguien, la de una empresa u organización, una página de comercio, una de noticias, de correo electrónico, una revista electrónica, una página social, etc. Actualmente se puede hacer un sitio de cualquier cosa o cualquier tema. La mayoría de los sitios web tienen hiperenlaces o hiperlinks que los llevan a otras páginas u otros sitios y de esta forma se puede navegar en un tiempo mínimo de un lado a otro sin que el usuario lo perciba.

Existen diferentes tipos de sitios como se describió anteriormente, a continuación mencionaré tres de los más comunes:

- **Sitios Institucionales.** Estos se caracterizan por especificar y poner énfasis en sus servicios o actividades y de qué forma van a beneficiar al usuario.
- **Sitios Comerciales.** Estos generalmente tienen un catálogo descriptivo de productos, a veces emplean muchas ayudas visuales para reforzar las características de lo que ofrecen. Algunos cuentan con un “carrito de compras” en línea y la opción de utilizar tarjetas de crédito para facilitar las transacciones con sus clientes.
- **Sitios de empresas de servicios.** Los usuarios utilizan estos sitios para localizar el tipo de servicio que necesitan y generalmente se proporciona un presupuesto aproximado del trabajo que se realizará porque no se cuentan con todos los elementos para dar un costo exacto.

2.3.1 Aspectos que se deben evitar al hacer un sitio WEB.

- Los sitios tediosos, aburridos, de contenido poco interesante no son muy concurridos, la efectividad de un sitio se puede constatar cuando el usuario regresa al sitio, existen formas de realizar estadísticas de visitas del sitio y así nos podremos dar una idea de la popularidad de éste.
- El uso de galerías de imágenes debe ser de uso exclusivo de sitios que contengan información turística, de museos, hoteles, lugares de distracción, etc. Se debe evitar el abuso de éstas, con el fin de “llenar la página”, lo ideal sería poner imágenes descriptivas que vayan con el desarrollo del contenido.
- El abuso de animaciones Flash. Estas generalmente son pesadas y su contenido tarda mucho en cargar, se debe hacer uso de ellas de manera controlada, ya que si bien le dan mucha vida, originalidad y genialidad a las páginas de Internet, también pueden contribuir a que estas se vuelvan aburridas e insoportables para el usuario.
- Las páginas con mucho contenido de texto. Como se vió en los puntos anteriores el abuso de todo es malo, igualmente pasa con el texto, si consideramos que a la mayoría de las personas no les gusta la lectura y a través de una computadora resulta un poco más pesado, entonces se deben omitir grandes bloques de texto, lo mejor es ser concretos en nuestras descripciones y sintetizar lo más que se pueda.
- En páginas que utilizan un cuestionario para dar de alta a un usuario, se debe evitar que éste sea muy extenso, debe contener datos generales e importantes.

2.3.2 Pasos para la realización de un sitio de Internet.

Toda página de Internet nace de una idea, sólo que se deben de tener muy claros los objetivos de lo que se quiere alcanzar con la realización de ésta. Contar con un sitio Web le permite a la empresa estar actualizada en cuanto a la tecnología se refiere, además de que es un medio de contacto con sus clientes y también una fuente de ingresos. Para la elaboración de un sitio de Internet se deben tomar en cuenta los siguientes aspectos:

1. Planear la finalidad y el aspecto del sitio.

Primeramente se debe contar con una idea clara y objetivos específicos de lo que se quiere obtener, éstos deben ser cuantificables a corto plazo, algunos podrían ser:

- Mejorar la comunicación con los clientes.
- Tener un medio de apoyo a la venta de los productos de la empresa.
- Difundir las novedades del negocio.
- Tener un espacio para promocionar.
- Contar con un medio en donde los clientes puedan contactar a la empresa.

Al inicio se deben evaluar otros sitios de empresas que tengan el mismo giro. Determinar que tienen de bueno y de malo, la forma en la que fue estructurada la información, el diseño, etc. Deben formularse las siguientes preguntas:

¿Qué dicen de la empresa el diseño, el uso de los gráficos, el color y el tono de la redacción?

¿Ofrece el sitio un acceso fácil a la información que desean conocer de entrada los clientes?

¿Existe algún motivo para volver al sitio?

¿Hay algún indicio de que la información se actualice con regularidad?²

2. Esquematizar el sitio.

Una vez que se tiene la idea de cómo se verá el sitio, se deberá diseñar en papel o con la ayuda de un programa de dibujo, también se tomará en cuenta la distribución de las páginas.

Diseño del sitio. En esta parte se mostrarán todas las páginas que componen el sitio y su vinculación entre éstas. La manera de representar la vinculación varía, se puede realizar con flechas que unan una página y otra o de forma jerárquica, es decir como las ramas de un árbol.

Diseño de las páginas. Crear una formato base para todas la páginas del sitio, es decir, como una plantilla de Power Point, además de que se deben respetar en todas las páginas los colores, las fuentes, la colocación de los vínculos hacia otras páginas, los gráficos, etc. Todo esto con el fin de crear congruencia en el sitio.

3. Registrar un nombre de dominio

Para tener un sitio WEB propio es necesario registrar el dominio o dirección de Internet, para evitar que otras personas o empresas lo usen. Este debe ser fácil de decir, deletrear y recordar, además de que debe de estar disponible. Existen muchos sitios de Internet en donde se puede investigar si el dominio está disponible o no.

² <http://www.microsoft.com/spain/empresas/temas/publicidad/crearweb.msp>

Por ejemplo:

<http://www.register.com>

<http://www.checkdomain.com>

<http://www.domainbank.com>

<http://www.newregistrars.com>

<http://www.internic.com>

Aún si el sitio WEB no se pusiera en marcha de inmediato, es conveniente registrarlo de todas formas para evitar plagio en el nombre del dominio.

4. Buscar alojamiento para el sitio

Una vez hecha la página, se tiene que publicar en Internet, es decir, que todos los archivos relacionados con el sitio (interfaces, programas, gráficos, etc.) deben hospedarse en un servidor, para ésto se deben considerar varios aspectos técnicos, como son:

- **Tipos de acceso al servidor.** Están ligados con los protocolos de acceso que se darán al usuario según la acción que realice en el sitio. Es decir, si se encargará de actualizar diariamente gráficos, información, etc. Se le asignará un protocolo de acceso rápido al servidor.
- **Cantidad de espacio en el disco duro del servidor.** Este aspecto se debe tomar bastante en cuenta si se sabe que la página es bastante pesada en cuanto a gráficos, imágenes Flash, bases de datos, etc.
- **Disponibilidad de cuentas de correo electrónico.** Algunas empresas que se dedican a hospedar páginas ofrecen a sus clientes un número de cuentas de correo electrónico para que éstos a su vez los distribuyan entre los empleados de su empresa, esto le da más proyección a la compañía.

- **Algunas compañías que ofrecen servicio de hosting son:**

<http://www.hostgator.com/>

<http://www.hostmonster.com/>

<http://www.powweb.com/>

Estos ejemplos son de compañías que cobran una renta por ofrecer sus servicios de hospedaje, lo cual es la mejor opción.

5. Crear y comprobar el sitio.

Después de crear la página mediante una herramienta especializada y hacerle las modificaciones necesarias para que quede dinámica y concisa, es importante tomar en cuenta que ésta se debe actualizar constantemente.

El sitio se debe probar de manera exhaustiva para verificar el óptimo rendimiento con diferentes navegadores antes de publicarlo en la Web.

6. Publicar el sitio y realizar mantenimiento.

Después de probarlo, se cargarán los archivos al servidor Web y estará listo para su uso en línea. Es importante evaluar de manera continua el buen funcionamiento de la página, tanto en el aspecto tecnológico como de contenido.

Una vez que está lista y publicada, se debe promocionar de manera que los potenciales visitantes sepan que existe y cuál es su dirección. Una buena forma de publicitar el sitio es incluir la dirección de éste en tarjetas de presentación, facturas, sobres, folletos, fachadas, avisos publicitarios, etc. Además que se puede registrar la dirección del sitio Web en los directorios como Yahoo y Google, que generalmente es gratuito.

Es importante que se actualice de forma dinámica el contenido del sitio para demostrar la actividad y vigencia de la empresa. Sería bueno que la página contara con una herramienta que le permita contar el número de visitantes de ésta par ver realmente que tan efectiva está resultando.

2.4 Utilización de la tecnología .Net para la creación de la interfaz.

Dentro de esta parte del capítulo se describirá en términos generales la nueva tecnología que Microsoft ha creado para el desarrollo de aplicaciones WEB, dicha tecnología ha impactado por su gran número de ventajas y por una marcada vanguardia en lo que a desarrollo se refiere.

2.4.1 Microsoft .NET Framework.

Cada cierto tiempo se produce una revolución en la tecnología empleada en el desarrollo de aplicaciones. Un ejemplo muy claro es la evolución de los sistemas operativos que marcaron grandes ventajas para los usuarios, ahora Microsoft lanza una nueva revolución la llama .NET Framework o más comúnmente .NET.

Microsoft lanza esta nueva tecnología como respuesta a la competencia de la tecnología Java de Sun Microsystems. El .NET Framework tiene grandes similitudes con la plataforma Java. El proyecto de Microsoft consistía en crear una nueva plataforma de desarrollo de software que tuviera transparencia en la cuestión de redes, independencia de plataforma de hardware y además que permitiera un rápido desarrollo de aplicaciones. Con estas ideas crea una herramienta integradora que conjunta desde el sistema operativo, hasta los lenguajes de desarrollo con los que cuenta actualmente. Es un entorno de

desarrollo y ejecución multilenguaje que proporciona bloques básicos para desarrollar Aplicaciones y Servicios Web.

El objetivo de su creación es ofrecer una manera rápida, económica, segura y robusta de desarrollo de aplicaciones, permitiendo un acceso simple desde cualquier tipo de dispositivo. Además de que las aplicaciones ya no correrán sobre un Sistema Operativo, sino sobre una armazón o marco (Framework).

La plataforma .Net puede ser añadida al sistema operativo Windows. Actualmente se incluye en Windows Server 2008 y Windows Vista. Pero puede ser instalada en cualquier otra versión de sistema operativo de Microsoft, inclusive también está disponible para la plataforma Windows Mobile que usan los teléfonos inteligentes. Uno de los proyectos de Microsoft es que la aplicaciones corran sobre la plataforma .NET independientemente del sistema operativo para la cual haya sido implementada.

La plataforma .Net Framework tiene varios componentes como se muestra en la Fig. 2.1. En la capa inferior se refiere a los **sistemas operativos**, que incluyen la variedad de plataformas de Windows, Windows Vista, Windows XP, Windows 2000, Windows Server 2003, etc.

En la siguiente capa se encuentra **el Entorno de Ejecución Común de Lenguajes o CLR** (Common Language Runtime) que es el motor de ejecución de las aplicaciones .NET.

Además al CLR se le conoce como el lenguaje representante del .NET Framework y reúne las ventajas de lenguajes como C#, C++ y VB en uno sólo, la manera de hacerlo es de que compila el código fuente de los diferentes lenguajes soportados por .NET en un código intermedio, MSIL (Microsoft Intermediate Language), para generarlo el compilador se basa en la

especificación CLS (Common Language Specification) que determina las reglas necesarias para crear MSIL compatible con CLR, éste es el encargado de convertir el lenguaje intermedio MSIL en lenguaje máquina del procesador, esto se hace en tiempo real por un compilador JIT (Just-In-Time) que lleva incorporado el CLR.

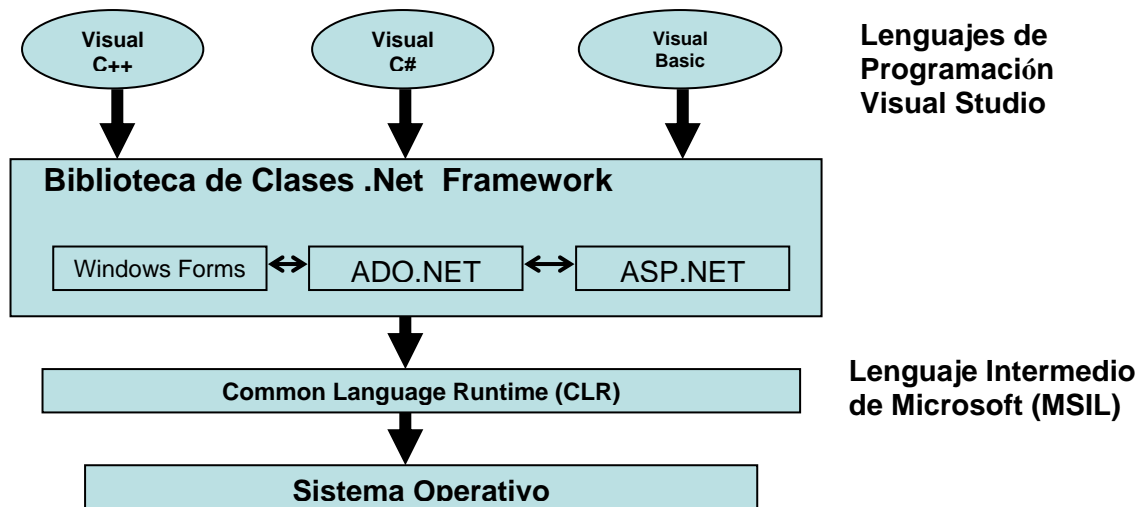


Fig. 2.1 Estructura de la plataforma Microsoft .NET Framework.

El CLR es un motor de ejecución de código de alto rendimiento porque cuando un componente se ejecuta, el runtime es el responsable de crear objetos, invocar llamadas a métodos, gestionar la reserva de memoria (incluyendo la recolección de basura), iniciar y detener hilos y procesos, garantizar la política de seguridad y satisfacer cualquier dependencia que el componente tenga sobre otros componentes, independientemente del lenguaje en el que estén escritos.

Lo que se debe tomar en cuenta como objetivo principal del CLR es proporcionar un entorno unificado para todos los lenguajes de programación.

La capa que le sigue es la llamada **Biblioteca de Clases Base o BCL** , que son un conjunto de clases ya compiladas como ensamblados; .NET nos ofrece cerca de 3000 de las cuales puede hacer uso el desarrollador. Hay clases que encapsulan estructuras de datos, clases que gestionan colecciones de objetos, que acceden a bases de datos, que gestionan la seguridad o la encriptación. El Framework organiza toda la funcionalidad del sistema operativo en un espacio de nombres jerárquico de forma que a la hora de programar resulta bastante sencillo encontrar lo que se necesita.

Para ello, el Framework posee un sistema de tipos universal, denominado Common Type System (CTS). Este sistema permite que el programador pueda interactuar los tipos que se incluyen en el propio Framework (biblioteca de clases de .Net) con los creados por él mismo (clases). De esta forma se aprovechan las ventajas propias de la programación orientada a objetos, como la herencia de clases predefinidas para crear nuevas clases, o el polimorfismo de clases para modificar o ampliar funcionalidades de clases ya existentes.

Dentro de la Biblioteca de Clases Base se encuentra el manejo de datos **ADO.NET (Activex Data Object)** es un conjunto de componentes del software que pueden ser usados por los programadores para acceder y para modificar los datos almacenados en un Sistema Gestor de Bases de Datos Relacionales, aunque también puede ser usado para acceder a datos en fuentes no relacionales. ADO.NET es a veces considerado como una evolución de la tecnología ActiveX Data Objects (ADO), pero fue cambiado tan extensivamente que puede ser concebido como un producto enteramente nuevo.

También dentro de la Biblioteca de Clases Base está el **Windows Forms** que representa una ventana o un cuadro de diálogo que constituye la interfaz de usuario de una aplicación de Windows.

Otro elemento de la Biblioteca de Clases Base es el **ASP.NET y servicios Web XML** , el primero se utiliza para la creación de aplicaciones Web. Los modelos de programación basados en Formularios Web proporcionan un modo fácil y potente de crear interfaces de usuario dinámicos, es decir básicamente se usan para que el usuario y el cliente tengan un contacto. Los Servicios Web de ASP.NET ofrecen los bloques básicos para construir aplicaciones Web distribuídas que utilizan estándares como XML(Extensible Markup Language, Lenguaje de Marcas Ampliable), éstos nos permiten la comunicación a través de Internet entre diferentes computadoras e incluso entre distintos sistemas, permitiendo su compatibilidad para compartir la información de una manera segura, confiable y fácil.

La forma de organizar la Biblioteca de Clases dentro del código es a través de los espacios de nombres (namespaces), donde cada clase está organizada en espacios de nombres según su funcionalidad. Por ejemplo, para manejar archivos se utiliza el espacio de nombres System.IO y si lo que se quiere es obtener información de una base de datos se utilizará el espacio de nombres System.Data.

La principal ventaja de los espacios de nombres es que de esta forma se tiene toda la Biblioteca de Clases centralizada bajo el mismo espacio de nombres (System). Además, desde cualquier lenguaje se usa la misma sintáxis de invocación, ya que a todos los lenguajes se aplica la misma Biblioteca de Clases.

La biblioteca de clases de .NET Framework también formada por una colección de **ensamblados** (o Assembly), son archivos en forma de EXE o DLL(Librería Dinámica).

Un ensamblado consiste en un conjunto de tipos y recursos reunidos para formar la unidad mas elemental de código que puede ejecutar el entorno .NET Framework, contienen toda la funcionalidad de una aplicación de forma encapsulada, una aplicación .NET se compone de varios ensamblados, también se considera a un ensamblado como una DLL lógica porque se trata de una lista de archivos que se referencian en tiempo de ejecución pero que no se compilan (o registran) para producir un archivo físico, a diferencia de las DLL's tradicionales.

La última capa del .NET Framework está constituida por los **lenguajes de programación Visual Studio**, algunos lenguajes que soporta son Visual C++, Visual C# y Visual Basic.NET, ASP.Net y Visual J# que forman un entorno de desarrollo integrado para sistemas Windows. Con estas herramientas se desarrollarán los servicios y aplicaciones WEB y todas ellas parten de la tecnología .NET, comparten los mismos componentes y objetos, incluso se pueden escribir partes distintas de un mismo programa con varios lenguajes .NET, esto gracias al CLS (Common Language Specification) que permite la interoperabilidad entre lenguajes. Microsoft proporcionó los lenguajes que se mencionaron al inicio con compatibilidad con CLS, pero la industria y el sector académico ha desarrollado más de 20 lenguajes compatibles, algunos de ellos son: Delphi, Pascal, Fortran, APL, LISP, Cobol, Perl, Prolog, Eiffel, Mercury, ML, Scheme, RPG, Oberon y algunos más.

2.4.2 Introducción a ASP.NET

La tecnología **Active Server Pages (ASP.NET)** ha sido de las más utilizadas para el desarrollo de sitios Web y aplicaciones Web dinámicas, por eso Microsoft la creó como parte de la estrategia .NET y le proporciona a los desarrolladores los servicios necesarios para la creación de aplicaciones Web empresariales.

Apareció en enero del 2002 con la versión 1.0 del .NET Framework y fue la sucesora de la tecnología ASP (ésta estuvo disponible por primera vez en el IIS- Internet Information Server). ASP.NET está construido sobre el Common Language Runtime (CLR), permitiendo a los programadores escribir código ASP.NET usando cualquier lenguaje admitido por el .NET Framework.

La definición contextual de Microsoft es que *"Las Active Server Pages son un ambiente de aplicación abierto y gratuito en el que se puede combinar código HTML, scripts y componentes ActiveX del servidor para crear soluciones dinámicas y poderosas para la web"*.³

Diferencia entre contenidos Estático y Dinámico de una Web.

Inicialmente las páginas Web eran de contenido estático, una HTML simple, actualmente y con la ayuda del ASP.NET y otras herramientas basadas en .Net Framework el contenido de las páginas se ha vuelto dinámico, muestran al usuario la salida del programa e interactúan internamente con los recursos de servidor, como bases de datos y servicios Web XML. En las Fig. 2.2 y 2.3 se muestra el comportamiento de estos contenidos.

Tipos de tecnologías dinámicas

Existen diferentes tecnologías de contenido dinámico, todas comparten una cosa en común: el uso de la programación lógica, que generan un HTML en el servidor y lo envían de regreso al navegador.

Hay tres tipos de tecnología dinámica de acceso al servidor:

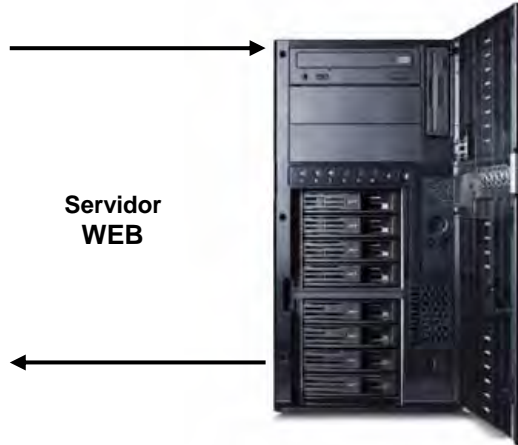
- **Salida directa**

³ <http://es.wikipedia.org/wiki/ASP.NET>

- **Página de secuencia de comandos**
- **Híbrida**



1. El navegador llama a un index.htm al servidor



2. El servidor responde enviando el contenido del index.htm al navegador.



3. El navegador despliega el contenido

Fig. 2.2 Representación de un contenido estático de una Web

La primera generación de tecnologías para la Web de contenido dinámico fue de salida directa. CGI y Java Servlets son ejemplos de este tipo, de esta forma los programadores tenían que escribir el código directamente para la emisión de todos los HTML que salían al cliente.

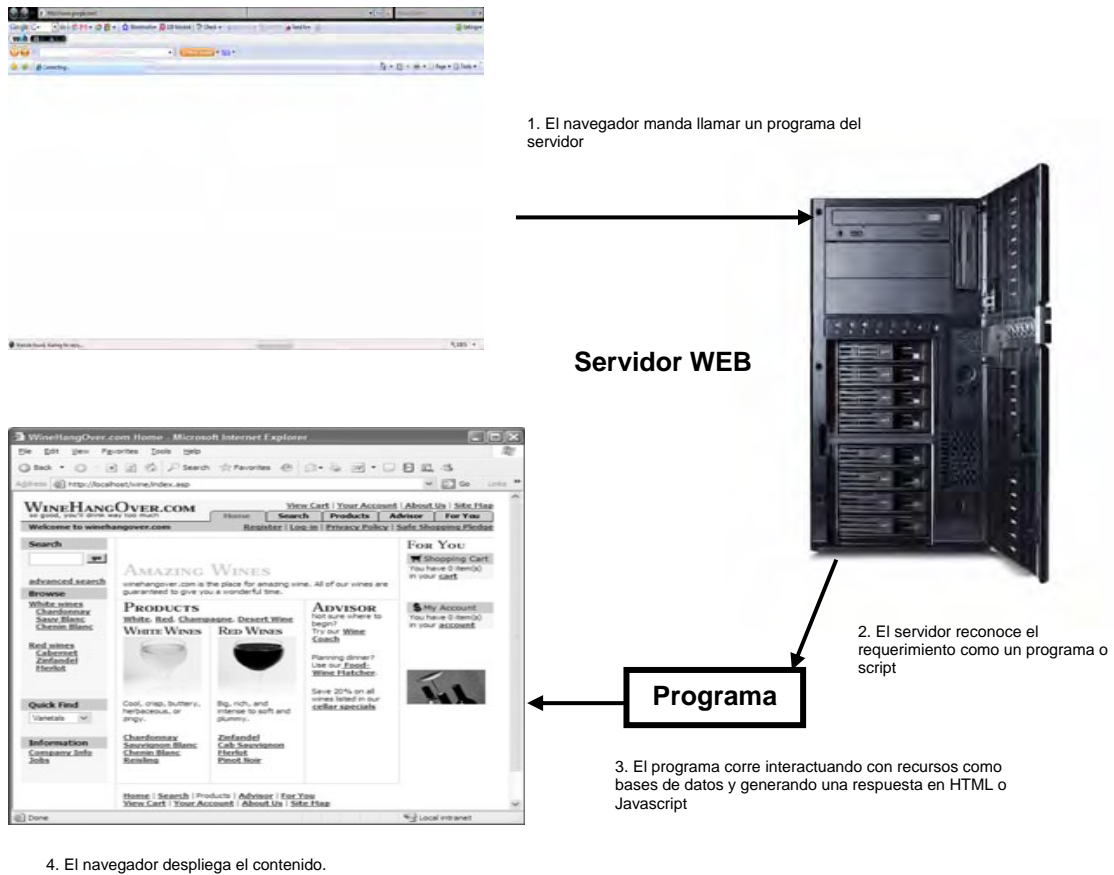


Fig. 2.3 Representación de un contenido dinámico de una Web.

Las páginas ASP.NET son archivos con extensión .aspx que contienen etiquetas *HTML* o *XHTML* estático, *texto*, *controles Web* que se procesan del lado del servidor y *controles de Usuario* que es donde los desarrolladores colocan el código estático o dinámico requerido por la página Web. Las páginas ASP.NET pueden llamar a componentes ActiveX para que realicen tareas, como la conexión con bases de datos o cálculos comerciales. Además se puede agregar contenidos interactivos a las páginas Web o generar aplicaciones Web completas que utilicen páginas HTML como interfaz con el cliente.

Ventajas de la tecnología ASP.NET

Existen varias ventajas que ofrece la nueva tecnología de Microsoft con respecto a su antecesora ASP:

- Mejor desempeño.
- Entorno de desarrollo más potente.
- Más fácil mantenimiento.
- Mejor despliegue y configuración.

El mejor desempeño se debe al compilador poderoso que ASP.NET incluye.

Su entorno es más potente debido a que el desarrollo está plenamente orientado a objetos y utiliza una rica biblioteca de clases enfocadas al desarrollo Web, todo esto reduce la cantidad de código usada por los programadores.

Además es fácil de mantener porque reduce las actividades destinadas a probar y ajustar las páginas para los distintos dispositivos de salida, posee gran parte de la lógica necesaria para la generación de interfaces Web.

Posee un fácil despliegue, con esta herramienta se puede correr una aplicación únicamente subiendo los archivos .NET y los componentes de la aplicación. Ya no es necesario registrar por lo que se evita el “infierno de las DLL”. Además simplifica la configuración con la existencia de archivos basados en XML y un sistema de seguridad integrado.

¿Qué incluye el ASP.NET?

- *Marco de trabajo de página y controles.* Esto se refiere al marcado que realiza el ASP.NET al explorador que le solicitó una página Web dinámica, esta página puede ser solicitada por varios exploradores,

ASP.NET representa el marcado (como HTML) al explorador que realizó la solicitud. Esta herramienta también es compatible con los controles móviles de dispositivos que utilizan páginas Web como son: teléfonos celulares, PC portátiles y asistentes personales digitales (PDA).

Además permite encapsular los controles fáciles de usar y reutilizables para una interfaz de usuario. Los controles se escriben una vez, se pueden utilizar en varias páginas y se integran en la página Web ASP.NET en la que se colocan durante la representación. También son responsables de controlar la apariencia y el funcionamiento de los sitios Web a través de **temas y máscaras**.

- *Compilador*. Es el encargado de compilar todos los elementos de una aplicación ASP.NET (páginas y controles), una vez compilado el código el CLR lo convierte en código nativo, aumentando el rendimiento.
- *Infraestructura de seguridad*. Propociona una estructura de seguridad avanzada para autenticar y dar acceso a los usuarios, para ello utiliza los formularios y la suscripción ASP.NET.
- *Funciones de administración de estado*. Esta función le permite almacenar información entre cada solicitud de la página Web, como información del cliente o del contenido de un carro de compras, puede guardar todo tipo de información: de la aplicación, del usuario, de la página y la definida por el desarrollador, inclusive esta información puede ser distribuída.
- *Configuración de la aplicación*. ASP.NET utiliza un sistema que distingue la configuración de un servidor Web, la de un sitio y la de aplicaciones individuales y es tan práctico que permite agregar o revisar los valores de

la configuración en cualquier momento sin impactar la ejecución de las aplicaciones, estas configuraciones se almacenan en archivos basados en la tecnología XML y son archivos ASCII.

- *Supervisión de estado y características de rendimiento.* La primera provee información sobre eventos clave de la aplicación y las condiciones de error. Y para el segundo punto posee grupos de contadores de rendimiento tanto de sistema como de aplicación, esto con el fin de dar un mejor resultado.
- *Capacidad de depuración.* Permite la depuración entre lenguajes y equipos, tanto de objetos administrados, como de no administrados.
- *Marco de trabajo de servicios Web XML.* A través de los servicios XML le permite a las aplicaciones intercambiar información utilizando la mensajería HTTP y XML.
- *Entorno de host extensible y administración del ciclo de vida de las aplicaciones.* El host extensible se utiliza para controlar el ciclo de vida de una aplicación desde el momento en que se accesa a algún recurso (como una página) hasta el momento de cierre de la aplicación.
- *Entorno de diseñador extensible.* Esto se refiere a que los diseñadores usando ASP.NET pueden crear controles de servidor Web para ser utilizados con una herramienta de diseño visual como **Visual Studio**. Con esta herramienta pueden configurar las propiedades y el contenido del control.

2.4.3 Visual Studio.NET

Es un conjunto poderoso de herramientas para desarrollo de aplicaciones Web ASP, servicios Web XML, aplicaciones para escritorio y aplicaciones móviles. Los lenguajes Visual Basic. NET, Visual C++.NET, Visual C#.NET y Visual J#.NET se unifican en este entorno poderoso (IDE), comparten herramientas y permiten crear soluciones en distintos lenguajes por basarse en la tecnología .NET Framework y por el CLR como se vió anteriormente.

Puntos de interés de Visual Studio .NET

Como se vió Visual Studio es una herramienta muy poderosa para diseño de aplicaciones Web, dentro de las características más interesantes que posee, se encuentran:

- Que se pueden desarrollar aplicaciones Web móviles para dispositivos inteligentes, como Pocket PC y asistentes digitales personales (PDA), teléfonos móviles y otros dispositivos de recursos restringidos.
- Los Web Forms que son parte de esta tecnología, se usan para crear páginas Web programables, se representan como código HTML y secuencias de comandos compatibles con exploradores, que permite ver las páginas en cualquier explorador o plataforma.
- También incluye la Windows Forms que proporcionan un conjunto de clases orientado a objetos que permite desarrollar complejas aplicaciones. Además pueden funcionar como interfaz de un usuario local en una solución distribuída de varios niveles.

- En Visual Studio .Net se pueden crear servicios Web XML que son aplicaciones que pueden recibir solicitudes y datos mediante el uso de XML sobre HTTP.

Visual Studio 2008

Dentro de las diferentes versiones de Visual Studio está la 2008, que es con la que se trabajará en la realización del proyecto de tesis. A continuación se describen las ventajas que tiene esta versión con respecto a la original.

- Esta creado para aprovechar todas las ventajas del Windows Vista, esto lo hace a través de sus subsistemas Windows Communication Foundation (WCF) y Windows Presentation Foundation (WPF), con la primera se pueden realizar aplicaciones orientadas a servicios y con la última se hacen las aplicaciones más dinámicas hasta ahora conocidas.
- Mejora en desempeño, escalabilidad y seguridad.
- Se incluye una herramienta para diagnosticar y optimizar el sistema, con ésta se pueden realizar pruebas como ejecutar perfiles para que corran ciertos módulos y registren su comportamiento. Además cuenta con herramientas integradas para depurar y optimizar.
- Incluye el Visual Studio Tools for Office (VSTO) esta herramienta ayuda a la creación de aplicaciones basadas en la interfaz de usuario (UI) que utilizan todo lo que el Office contiene.
- Permite incorporar el estilo de las aplicaciones Visual al estilo de Windows Vista debido a las mejoras en Microsoft Foundation Class Library (MFC) y Visual C++. Además permite mejorar la interoperabilidad

entre el código nativo y el código .NET, simplificando el trabajo de diseño y codificación.

- Contiene el LINQ (Language Integrated Query) es un conjunto de herramientas que reduce la complejidad del acceso a Bases de Datos, con extensiones para C++, Visual Basic y para .NET Framework.
- Permite la creación de soluciones multiplataformas que funcionan con las diferentes versiones de .Net Framework.
- Incluye una biblioteca ASP.NET AJAX (combinación de Javascript y XML, la aplicaciones son más rápidas para el usuario porque las “idas y vueltas” entre el servidor Web y el navegador se hacen sin que la página se regargue), para desarrollo de aplicaciones más eficientes, interactivas y personalizadas.
- Posee la capacidad de especificar el Framework sobre el cuál se desea compilar. Con la opción Target Framework se puede elegir entre .Net Framework 2.0, .Net Framework 3.0 y .Net Framework 3.5.
- Se agregó una herramienta para el diseño de hojas de estilo (CSS), también contiene una tecnología llamada **Intellisense** que se utiliza para los proyectos de Jscript en AJAX.

2.5 Uso de bases de datos para aplicaciones Web dinámicas.

Si una aplicación Web se considera dinámica debe tener acceso al servidor de bases de datos para generar sus procesos. Estas aplicaciones se dividen en **operativas y analíticas**, las primeras almacenan datos, permitiendo crear,

obtener, actualizar o eliminar datos, a esto se le conoce como CRUD, esto se hace a través de formularios Web donde los datos quedarán en la base de datos central (no en excel o en una pc). Las analíticas convierten esos datos en información, es decir generan gráficas o análisis a través de “datashapes” (registros jerárquicos), “cubos de datos” (se hace una selección de variables de la base de datos, en las que desea buscar patrones o comportamientos de interés, según el número de variables seleccionadas es la dimensión del cubo formado, a cada variable se le denomina una *dimensión* o un eje del cubo), “dashboards” (tablero de comando integrado, muestra exactamente la información que se solicita de manera gráfica) y otros mecanismos.

2.5.1 Microsoft SQL Server. Plataforma de Base de Datos.

El MS SQL Server es un manejador de bases de datos relacionales (creado por Microsoft) capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea. Se utiliza por los programadores para desarrollar sus aplicaciones y explotar por completo una base de datos.

SQL Server propociona métodos para definir la base datos, para manipular la información y para gestionar los permisos de acceso a dicha información.

A continuación se listan algunas ventajas de esta tecnología:

- Escalabilidad, estabilidad y seguridad.
- Soporta procedimientos almacenados.
- Incluye un potente entorno gráfico de administración.
- Permite trabajar de modo cliente-servidor, en donde todos los datos permanecen en el servidor y no en las terminales, las cuales sólo los accesan.
- Permite administrar información de otros sevidores de datos.

Existen varias versiones de MS SQL Server la última es la 2008, ésta aporta valor y funcionalidad, además de una gama de características, desde alta disponibilidad y sólida escalabilidad hasta herramientas avanzadas de inteligencia comercial, diseñadas para ofrecer a los usuarios de una organización una plataforma de gestión de datos más segura, fiable y productiva, con estrictos controles de seguridad, SQL Server 2008 representa un avance espectacular al ser compatible con los sistemas empresariales más exigentes del mundo.

2.5.2 Interacción de SQL Server y la tecnología .NET.

Las aplicaciones de ASP .Net y Visual Studio pueden conectar a múltiples fuentes de datos, como SQL Server, fuentes de datos ODBC y una variedad de proveedores OLEDB, mediante la nueva tecnología de acceso a datos ADO.NET.

Microsoft SQL Server 2005 integra el Common Language Runtime (CLR). Esto permite a los desarrolladores escribir procedimientos, triggers, y funciones en cualquiera de los lenguajes CLR (Visual C++, Visual C#, Visual Basic) y también pueden ampliar la base de datos con nuevas funciones y tipos definidos por el usuario. El CLR también le permite utilizar códigos distintos a los de Microsoft para desarrollar aplicaciones rápidas de base de datos. SQL Server es compatible con .NET Framework y el Visual Studio lo que le permite crear bases de datos seguras y eficaces a un menor costo.

Algunos beneficios de la integración SQL Server 2005 y CLR :

- **Mejor modelo de programación.** Los lenguajes de programación compatibles con .NET Framework son mejores que Transact-SQL(es un

productos de Microsoft para manejar bases de datos) en muchos aspectos, ofreciendo capacidades novedosas para los desarrolladores de SQL Server.

- **Más seguridad.** El código administrado se ejecuta en un entorno CLR, alojado en el motor de la base de datos. Esto permite que los objetos de la base de datos .NET Framework sean más seguros que los procedimientos almacenados extendidos de versiones anteriores de SQL Server.
- **Tipos definidos por el usuario y agregados.** Estos dos objetos se habilitan al alojar el CLR y amplían las capacidades de consulta y almacenamiento del SQL Server.
- **Entorno de desarrollo común.** El desarrollo de bases de datos está integrado con el entorno de Microsoft Visual Studio 2005, usando las mismas herramientas para desarrollar y depurar secuencias de comandos en la base de datos que las utilizadas para escribir componentes y servicios .NET Framework a nivel medio y de cliente.
- **Rendimiento y escalabilidad.** Debido a que se compila el código administrado a código nativo antes de la ejecución, esto implica una mejora en el rendimiento.

Además, con la funcionalidad disponible en la biblioteca de clases de .NET Framework, se tiene disposición total a miles de rutinas y clases prearmadas a las que se puede tener acceso fácilmente desde cualquier procedimiento almacenado o función definida por el usuario, también se puede tener acceso fácilmente a funciones matemáticas, operaciones con fechas, acceso a los recursos de sistema, algoritmos avanzados de cifrado, acceso a archivos, procesamiento de imágenes y manipulación de datos XML.

También puede agregar los objetos de base de datos en CLR al mismo sistema de control de origen que utiliza para todos sus proyectos en Visual Studio, y de

esta manera el nivel de integración y seguridad de los procesos de desarrollo serán aún mayores.

2.6 Otras herramientas.

A continuación se describen brevemente otras herramientas que se utilizarán durante la realización del proyecto.

ASP. NET AJAX . es el nombre que recibe el proceso de utilizar un objeto *JavaScript* para intercambiar información en formato *XML* con el servidor sin tener que hacer *submit* de un formulario o poner una URL en el navegador: el famoso **XmlHttpRequest**. La llegada de Ajax ha puesto fin a la árdua lucha del desarrollo de aplicaciones web para encontrar una tecnología que no sólo mejore los tiempos de respuesta, sino también que permita el procesamiento asíncrono (es una operación vinculada a un proceso de entrada y salida, el CPU se encuentra en modo de espera mientras los dispositivos entrada/salida terminan su trabajo), esto es debido a que las interfaces de usuario se pueden actualizar por partes sin tener que actualizar toda la página completa.

Como se dijo anteriormente 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.

Ajax es una tecnología que se ha convertido en la herramienta más utilizada para el desarrollo de aplicaciones WEB con mejora de tiempos de respuesta, ya que aumenta la interactividad, la velocidad y la usabilidad (la tarea, el usuario y el tipo de contextos sean lo más cercano al diseño deseado) en las aplicaciones. Se puede usar Ajax en cualquier navegador web, tales como IE, Mozilla, Firefox, etc.

Debido a este avance y para utilizar lo último en la tecnología de desarrollo de páginas Web, se van a implementar en este proyecto algunas aportaciones de Ajax.

Crystal Reports.

Es una aplicación utilizada para diseñar y generar informes tomando como fuente una Base de datos, es una llamada **aplicación de inteligencia empresarial**. Visual Studio tienen una versión OEM (generada por un tercero) de Crystal Reports y es utilizada para generar informes o reportes.

Ha sido el generador de reportes de Visual Basic desde sus inicios y ahora que se transformó en la tecnología .NET también aparece en Visual Studio, este producto está dirigido de manera sencilla al usuario final, es decir que cualquier persona puede generar un reporte sin la ayuda de un desarrollador.

Uno de los conceptos introducidos con el nacimiento de la tecnología .NET es el de Web Services (Servicios en la Web), los reportes creados en Crystal también pueden ser publicados y consumidos como Web Services, es decir que proveen alguna funcionalidad a través de la Web para que pueda ser utilizada por otra aplicación.

Crystal Reports es una herramienta para diseño de reporte creada mucho antes del nacimiento de la tecnología .NET, gracias a ésta el diseño de reportes para las aplicaciones se hace mas sencillo, Crystal Reports puede ser utilizado con diferentes base datos: MS Access, SQL Server, Oracle, Informix, etc., se pueden diseñar reportes tanto desde el ambiente de desarrollo de Visual Studio .NET como desde el ambiente de diseño de Crystal Reports, los reportes creados desde el ambiente de diseño de Crystal pueden ser compilados a .EXE

para poder ser ejecutados sin necesidad de crear un programa o interfaz para ésto.

Esta herramienta para generar reportes es la que se utilizará para presentar los informes derivados de la Interfaz Web presentada en esta tesis.

Capítulo III. ANÁLISIS Y DISEÑO DEL SITIO.

3.1 Análisis de la Página Web.

Este es el primer paso para la creación de la interfaz web, en esta etapa se revisarán la idea sugerida con el planteamiento del problema actual, la factibilidad de la herramienta y el uso de ésta para la solución del mismo, todo con el fin de desarrollar un sitio Web efectivo y bien planeado.

3.1.1 Estudio de la situación actual.

El mercado actual de las compañías que se dedican a la instalación de equipos de comunicación y fibra óptica, como es el caso de **GADB Instalaciones** está experimentando cambios trascendentales en los últimos tiempos. Estos cambios están propiciados por diversos factores, entre ellos se encuentra la rápida evolución tecnológica, no sólo relacionada específicamente a las tecnologías (equipos) que la empresa maneja, sino que el mismo concepto de imagen tecnológica de la empresa ha evolucionado.

Quizás el cambio principal lo está representando la irrupción masiva de la información en los negocios y en la sociedad en general, impulsada en buena medida por el fenómeno Internet.

Debido a este fenómeno la empresa ha decidido actualizarse y tener como muchas otras de la competencia una página Web comercial en donde informe sobre la historia de la empresa, los servicios que ofrece, los objetivos, etc., además de contar con un medio en donde el usuario pueda contactar al prestador de servicios. Todas estas características se ofrecen actualmente en casi todas las páginas Web. Actualmente no cuenta con estos servicios que son "indispensables" hoy en día para cualquier compañía que quiera estar a la par de los rápidos movimientos empresariales en línea, tomando en consideración que la mayoría de los clientes reales y cautivos utilizan el Internet o la computadora de manera cotidiana.

3.1.2 Definición del Problema.

Actualmente GADB Instalaciones no cuenta con una página de Internet que funja como medio de enlace informativo a través de la red para sus clientes.

Otro punto a considerar es que la empresa, como se describió en el Capítulo I, trabaja de forma dispersa, es decir, que todas sus actividades las realiza en diversos sitios de trabajo y cuenta con varios líderes de proyectos y trabajadores que les reportan a estos líderes. Actualmente el control administrativo de los proyectos, la asignación de recursos, el avance de tareas programadas y los reportes a los clientes se hace a través de teléfono, de correo electrónico y de hojas de cálculo en Excel, convirtiendo a estas actividades en una tarea pesada, tomando en cuenta el volumen de proyectos que maneja la empresa y que cualquier falla en el equipo de cómputo que utilizan podría significar una gran pérdida de valiosa información, causando problemas con los tiempos de entrega, fallas con los clientes y pérdidas económicas.

Por otra parte toda la captura administrativa de los proyectos la realiza una sola persona, el Administrador, ésto lo hace al finalizar los trabajos cuando los líderes regresan de sus viajes y entregan una hoja de papel con las actividades realizadas. Esto presenta una desventaja porque la carga de trabajo es mucha para una sola persona, pudiendo responsabilizar a cada líder de la administración de su proyecto y de la captura de las tareas incluidas en éstos casi desde el momento de su realización y no permanecer tanto tiempo en línea de espera.

Otra área de oportunidad son los clientes, que para la empresa es algo muy importante, porque de la forma y el cumplimiento del proyecto dependerá que éstos le sigan ofreciendo proyectos futuros. Actualmente los clientes se enteran del cumplimiento de su proyecto al finalizar el mismo o durante la realización de

éste, a través de llamadas telefónicas, no resultando ser un medio muy apto, porque algunas veces no encuentran al Gerente o les gustaría conocer más a detalle el por qué del avance o del retraso.

3.1.3 Funcionamiento de la Interfaz Web para la solución del problema.

Se están planteando en esta tesis las ventajas que se proporcionarían a la empresa al usar una Interfaz Web, tomando en cuenta que se solucionarían un gran porcentaje de sus problemas administrativos, aunque también tendrían que generar una cultura de trabajo un tanto diferente para los líderes y los clientes.

Como primer punto se tendría una Interfaz Web informativa, que ofreciera de manera clara los servicios que la empresa proporciona, que trate de enganchar al cliente, mostrándole la información que él busca y la forma como puede contactar el servicio. Esto solucionaría la parte que corresponde a proyección tecnológica ya que actualmente la mayoría de empresas del giro cuentan con una herramienta de este tipo para la promoción de sus productos o servicios, de tal forma que tengan un alcance mundial y estén accesibles para cualquier persona que lo necesite y navegue por la WEB.

Posteriormente se haría dentro de la misma página un apartado que incluyera la administración de proyectos, a este apartado tendrán acceso: el administrador, los líderes de proyectos y los clientes; cada uno tendrá diferentes niveles de acceso y podrán realizar actividades diferentes, todo a través de la Web y desde el sitio en donde trabajen. Se podrá integrar esta nueva herramienta a los métodos de trabajo existentes, para conseguir las ventajas completas del uso compartido de la información y el análisis de datos.

Este tipo de solución pretende que todos los usuarios involucrados en el proyecto participen de forma directa y en tiempo real, haciendo que la información se utilice de manera efectiva y oportuna para la toma de decisiones.

Ventajas de la Interfaz Web.

- Fácil acceso, se solicita un usuario y un password, que se proporcionan en el momento del registro de datos, se hace una distinción entre los clientes y los líderes, cada uno con una página de acceso distinta. El registro del cliente lo puede hacer él mismo, el registro del líder lo puede hacer únicamente el Administrador (el usuario y password de éste son únicos y se dan de alta directamente en la Base de Datos).
- Buena herramienta de planificación que permite la captura, control y visualización de todas las tareas que incluyen los proyectos de la empresa.
- Publicación automática de los datos, ya que éstos se capturan en línea, por lo tanto su consulta también puede ser en línea.
- Facilita la buena comunicación con todos los miembros del equipo del proyecto (Administrador, Líderes y Clientes).
- Centralización de la información relacionada con los proyectos en una sola base de datos, lo que facilitará la administración de los mismos y una reducción de tiempo en el proceso administrativo. Además permitirá una descentralización de todas las actividades, distribuyendo cargas y responsabilidades entre los miembros.
- Permite una buena administración de plazos, recursos y calidad.
- Mejora la imagen de la empresa haciéndola más competitiva.
- Proporciona un mayor alcance del mercado potencial, trayéndole más beneficios económicos al aumentar la prestación de servicios.

¿A quiénes va dirigida la Interfaz Administrativa?

Este punto se va a describir en base a los involucrados en los proyectos, hablaremos de cuatro tipos de usuarios, les llamaremos Administrador, Líder de Proyecto, Cliente Potencial y Cliente. A continuación se describirá el perfil de cada uno de ellos y las ventajas que obtendrán con la Interfaz.

Administrador.

Es el Gerente de la Compañía, el encargado de liderar todos los proyectos de su empresa, además asigna los proyectos a los líderes que considera más competentes para el caso, reasigna y aprueba cargas de trabajo, hace negociaciones con los clientes para la planeación conjunta de los proyectos, se encarga de cotizar los requerimientos que le solicitan los clientes y de conseguir nuevas plazas de trabajo.

Ventajas para el Administrador:

- Revisión y seguimiento en tiempo real de todas las tareas involucradas en los proyectos.
- Puede planear la carga de trabajo a medio y largo plazo según las necesidades de cada proyecto.
- Llevar un control sobre sus líderes y cuadrillas, dar de alta en el sistema a los Líderes de Proyectos.
- Asignar de forma rápida y fácil los proyectos a los líderes.
- Contactar a clientes potenciales que entren a la página a solicitar información y dar seguimiento a los mismos.
- Realizar reportes de servicio.

Líder de Proyectos.

Es la cabeza de un proyecto, tiene bajo su orden a varios empleados instaladores, se encarga de asignarles tareas, administrar tiempos, recursos y materiales, es el responsable de recibir el sitio y entregarlo ya con el proyecto terminado.

Ventajas para el Líder de Proyectos:

- Podrá actualizar las tareas realizadas en tiempo real, y estimar el trabajo restante.
- Accesar a toda la información relacionada con el proyecto.
- Mantener una estandarización en la forma de trabajo.
- Marcar los motivos por los que no se realizó una tarea o cualquier tipo de observación, generando una comunicación efectiva con el administrador y el cliente en tiempo real.
- Realizar reportes de servicio.

Cliente Potencial.

Es el que aún no hace una contratación con la empresa, pero accesa a la página en busca de información o porque requiere algún servicio relacionado.

Ventajas para el Cliente Potencial:

- Conocer a la empresa a través de una página que informa de manera clara y efectiva los servicios que presta.
- Evaluar la página que consultó, lo que permite el mejoramiento de la misma.
- Enviar comentarios, observaciones, quejas o solicitudes de servicio (requerir cotizaciones) en una página que quedará registrada en la base de datos para que el administrador le de el seguimiento correspondiente.

Cliente.

Es el que hace la contratación de la compañía para la realización del proyecto, mantiene contacto directo con el Administrador, está presente al inicio y término del proyecto.

Ventajas para el Cliente:

- Consultar la información del proyecto bajo un acceso restringido.
- Obtener reportes actualizados en todo momento.

Con estos puntos se puede observar que cada miembro del proyecto trabaja con sólo la información que necesita y las funciones que le hacen falta.

3.1.4 Estudio de factibilidad.

Dentro de este estudio se analizará el proyecto de la **Interfaz Web** en tres aspectos: Operacional, Técnico y Económico.

Factibilidad Operacional:

Este estudio se hizo junto con el Administrador de la empresa y se llegó a la conclusión que se quería una herramienta sencilla, que fuera fácil de operar, ya que de lo contrario, podríamos tener resultados no deseados, como que ésta sea ignorada o que cause errores o fallas mientras se usa.

Se habló también de que se generará en los usuarios una nueva cultura de trabajo en lo que corresponde a la metodología que seguían para la administración de los proyectos, esta herramienta cambiará el patrón antes utilizado, a lo mejor al principio causará algunos inconvenientes porque tienen que actualizar la información en tiempo real o en su defecto en el menor tiempo posible. Pero el Gerente considera que no habrá resistencia al cambio.

Factibilidad Técnica:

Como se ha estado mencionando la herramienta que se realizará es una página WEB dinámica y para ello es necesario contar con un servidor Web potente, posteriormente en este mismo capítulo se listarán las características de un servidor de este tipo. Se recurrirá al arrendamiento de hosting para alojar la aplicación Web, esto debido a que las empresas que ofrecen este servicio tienen los equipos más adecuados.

También se necesita contar con un dominio que como se vió en el Capítulo II le proporciona una identificación o nombre a la página, la mayoría de las empresas que ofrecen el servicio de hosting también ayudan a proporcionar el dominio para la aplicación. También se pagará por este servicio.

Para la elaboración de la aplicación se utilizará un manejador de bases de datos y una herramienta para diseño de aplicaciones Web, el manejador será el SQL Server 2000 y la herramienta de diseño el Visual Studio .Net versión 2008, las características de ambos fueron ya expuestas en el capítulo anterior.

Para la utilización, actualización y consulta de la herramienta, el administrador y cada líder cuenta con su equipo de cómputo con conexión a Internet, lo que facilitará el uso de la misma.

Con lo que respecta al cliente, éste debe contar con su propio equipo de cómputo con acceso a Internet para poder ingresar a su módulo.

Además del equipo, el personal de la empresa maneja desde hace tiempo su trabajo a través de computadoras, ésto hace más fácil el uso de la herramienta, y con la capacitación que recibirán a la entrega, estarán preparados para utilizarla.

Factibilidad Económica:

Este estudio se hizo en base a las ventajas que traerá la herramienta a nivel de clientes, primero le proporcionará a la empresa una imagen actualizada y competitiva dentro del comercio cibernético; segundo, seguramente atraerá a nuevos clientes que estén interesados en sus servicios y ésto se traduce en mayores ventajas económicas.

Otro punto importante es que los clientes que tienen actualmente se verán beneficiados con la herramienta porque podrán dar seguimiento a sus proyectos y utilizar la información que se genere a partir de éstos. Esta herramienta obligará a GADB Instalaciones a cumplir con los tiempos y condiciones pactadas en el inicio, generando en el cliente satisfacción con el trabajo y por consiguiente podrán surgir nuevos proyectos para la empresa por los buenos resultados obtenidos.

3.1.5 Mapa de Contenidos del Sitio.

A continuación se planteará los diferentes enlaces o contenidos que la Interfaz Web contendrá:

Empresa

Este menú contendrá dos submenús:

- **Quiénes somos?**. Detalles de la empresa
- **Nuestra Filosofía**. Misión, Visión y Objetivo Principal.

Nuestros Servicios

Descripción los servicios que la empresa proporciona, como son: instalación y mantenimiento de Radio Bases Celulares, instalación de antenas y repetidores,

tierras físicas y equipos de fibra óptica. Además relaciona detalladamente las principales políticas de servicio de la empresa.

Clientes

En este menú se incluirán algunos puntos específicos que GADB Instalaciones ofrece a sus clientes de forma que puedan diferenciarla como empresa competitiva. Algunos de estos puntos son: presupuestos acorde a las necesidades, análisis de instalaciones, garantía en los trabajos y confidencialidad.

Contáctanos

Se llenará un pequeño formulario con los datos del cliente o la persona que requiera información, éste se guardará en una tabla de la base de datos y posteriormente esta información podrá ser usada por el administrador, éste contactará con el cliente directamente si así se requiere. Además del formulario se presenta una pequeña encuesta para la evaluación de la página, con el fin de que el administrador considere cambios en la misma.

Zona Proyectos

En esta parte del menú se tendrá anexo el módulo administrativo de Control de Proyectos, al cual podrán acceder Clientes, Líderes de Proyectos y el Administrador, se le dio el nombre de **Zona Proyectos**, por estar enfocada esta parte del módulo a la administración de los mismos, por parte de todos los elementos que conforman el equipo de trabajo. A continuación se muestra en la figura 3.1 la distribución de los módulos a los que tiene acceso un Cliente, en la figura 3.2 se describe lo concerniente al Líder de Proyectos y en la figura 3.3 se muestran las opciones para el Administrador.

3.2 Diseño del Sitio.

La organización de la información en un sitio Web incide de manera directa en el éxito o fracaso de ésta.

Una vez que se tiene definida la información que irá en el sitio, se tiene que clasificar de manera apropiada. Aquí hay que encontrar un adecuado balance entre la linealidad y la jerarquización.⁴

Se dice que es lineal a aquella información que requiere que la persona que los lee vaya avanzando poco a poco en el conocimiento de algo, como el caso de un libro. Para ésto se pondrá toda la información en una sola página. Ahora el tipo jerárquico se refiere a trozos de información que sean complementarios o que dependan uno de otro, éstos se pondrán en diferentes páginas. Para ello se utilizarán links (ligas) para pasar de uno a otro.

La cualidad de las estructuras jerárquicas es que brindan una manera simple y familiar de organizar la información y permite al usuario sentirse cómodo en la navegación, aunque usadas sin control pueden causar el efecto contrario y perder al usuario al momento de estar navegando. En la figura 3.4 se mostrará una estructura jerárquica en donde cada bloque del organigrama lleva a la descripción del departamento correspondiente.

⁴ http://www.wikilearning.com/curso_gratis/una_web_es_mas_que_codigo-como_se_puede_estructurar_un_sitio_web/10112-6

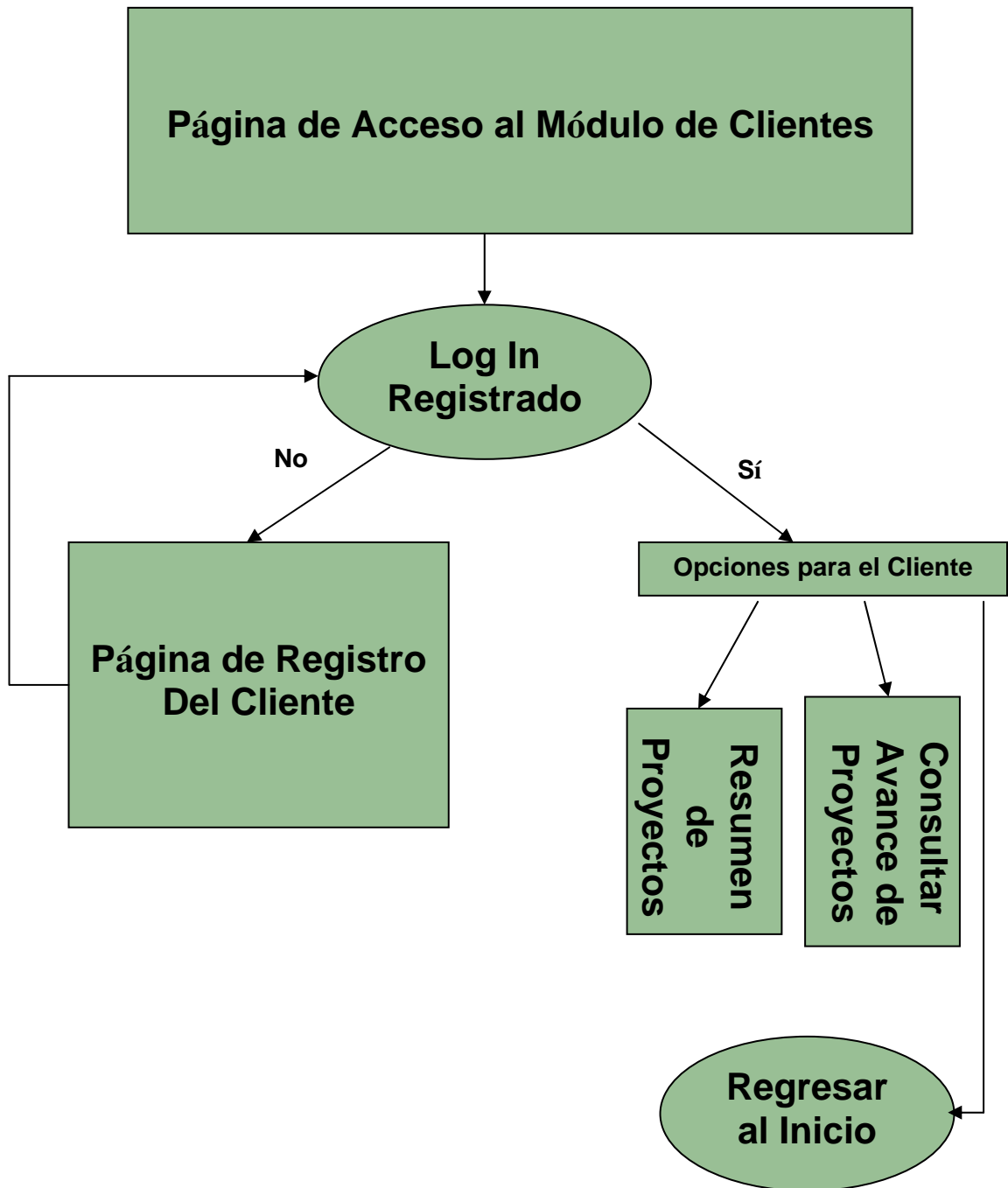


Fig. 3.1 Contenidos del Módulo de Clientes para la Administración de Proyectos.

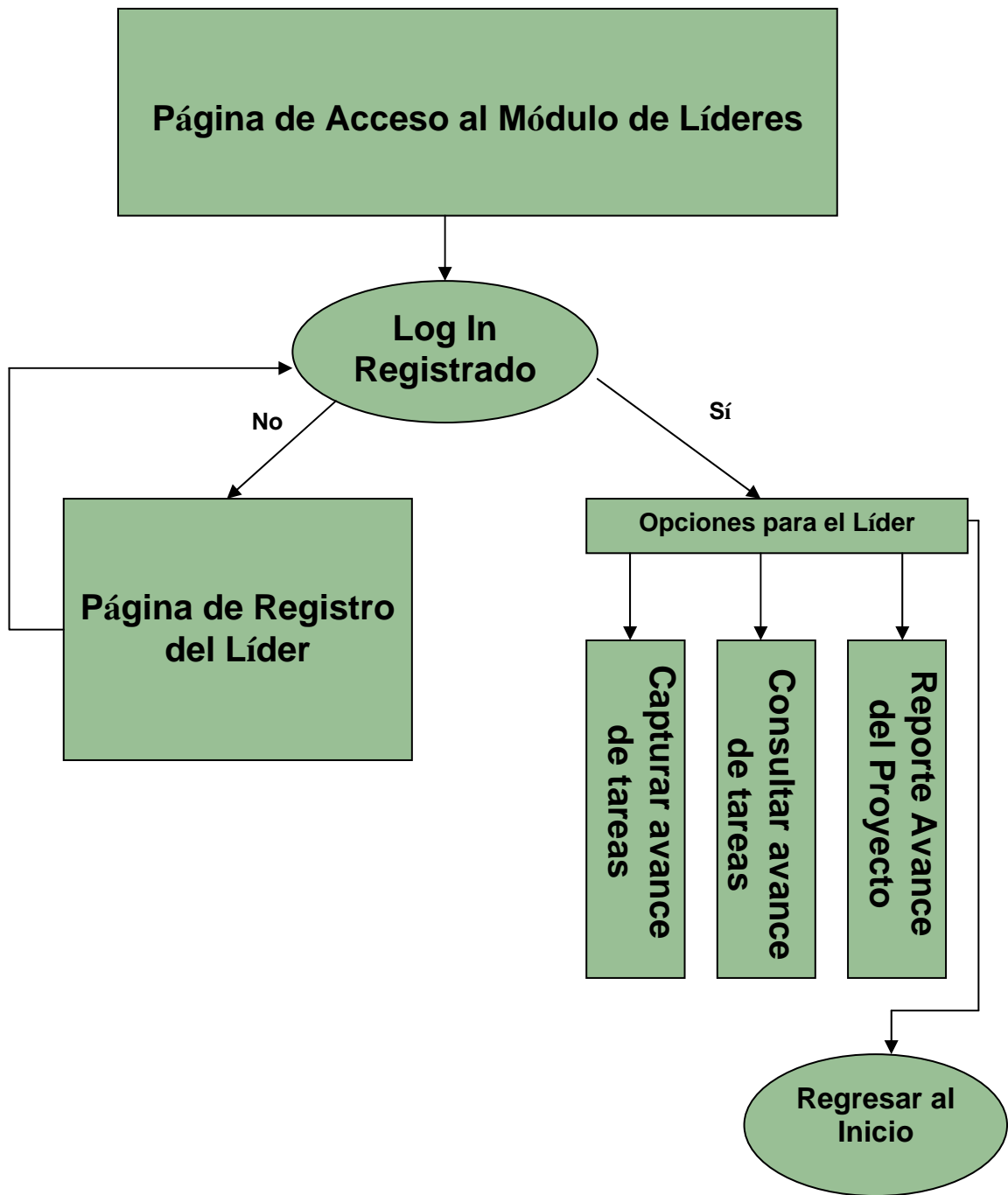


Fig. 3.2 Contenidos del Módulo de Líderes para la Administración de Proyectos.

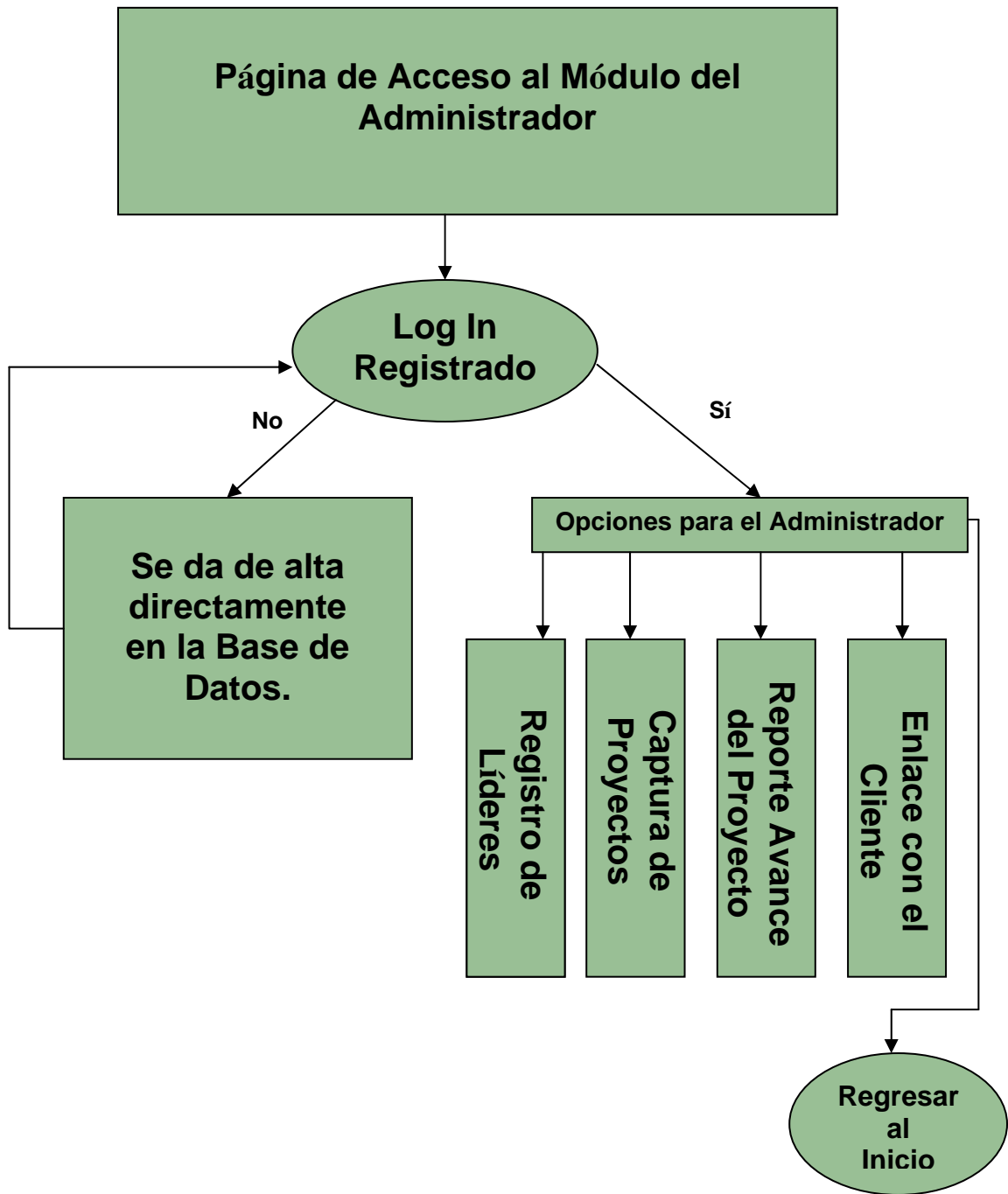


Fig. 3.3 Contenidos del Módulo del Administrador para la Administración de Proyectos.

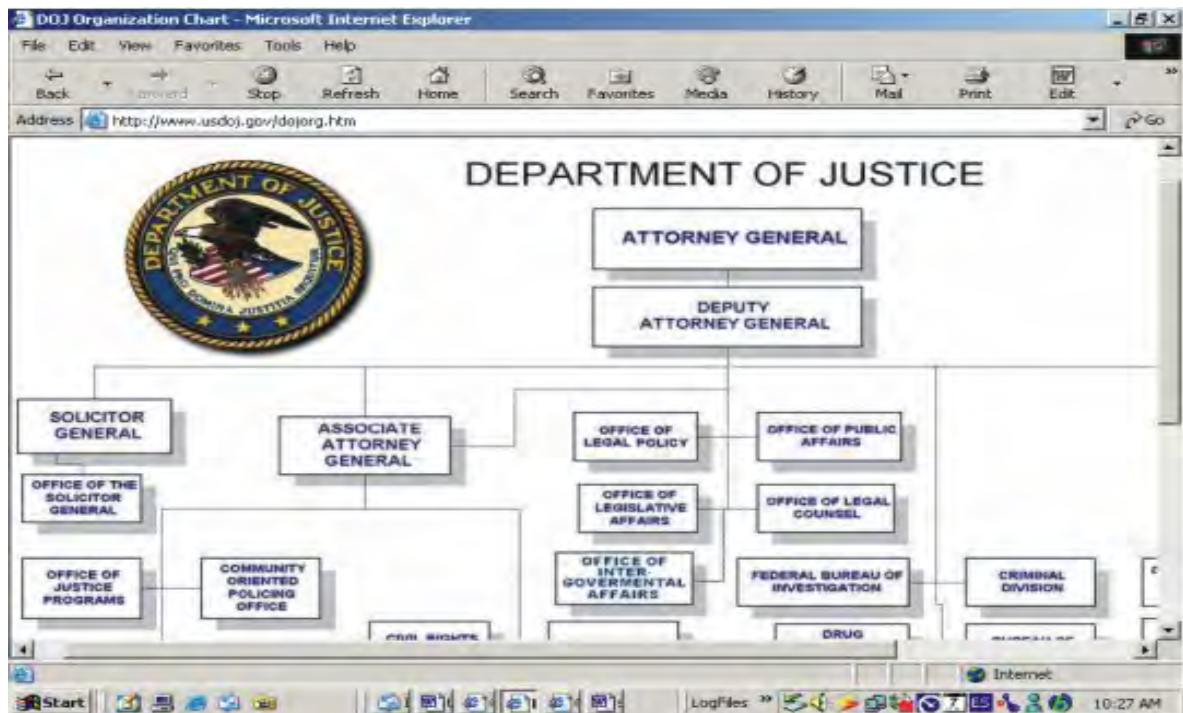


Fig. 3.4 Ejemplo de estructura jerárquica.

Dentro del proyecto de tesis que se está proponiendo se utilizará el tipo de estructura jerárquica, debido al tamaño y tipo de la aplicación, sería poco factible hacerlo de manera lineal. Se pasará de una página a otra a través de menús o de ligas que se manejarán con **hipertexto**.

El hipertexto es una vía no lineal de estructurar la información. Un sistema de hipertextos contiene dos componentes primarios: los elementos o nodos de información que se enlazarán y los enlaces o vínculos entre estos. Los nodos de hipertextos pueden conectarse de manera jerárquica, no jerárquica o en ambas.

En la figura 3.5 se muestra el modelo jerárquico para la parte informativa de la aplicación, la parte administrativa o Zona Proyectos ya se diagramó en las figuras 3.1, 3.2 y 3.3.

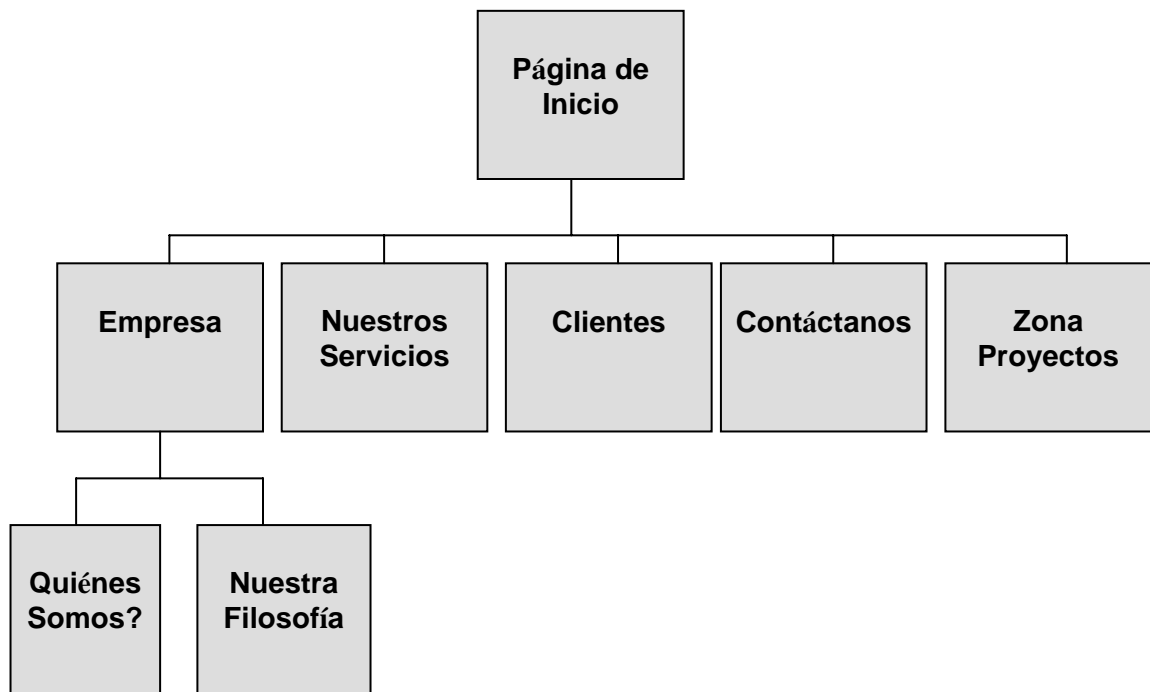


Fig. 3.5 Estructura Jerárquica de la parte informativa de la aplicación Web.

3.2.1 Estilo gráfico.

Un estilo gráfico adecuado puede ser algo importantísimo a la hora de hacer que la persona que nos visita se sienta cómoda, y como una manera de aliviar el contenido y hacerlo más digerible. Un sitio por más contenido que tenga si no tiene un buen diseño gráfico difícilmente logrará un lugar destacado entre otros sitios del mismo tipo.⁵

Otro punto importante en la etapa de definición de la parte gráfica, es intentar en lo posible mantener una cierta coherencia gráfica, y atreverse a innovar en cuanto a ella, además proporcionarle al sitio una apariencia profesional. Un color de fondo, un fondo o una distribución interesante de los elementos dentro de una página es algo que no se olvida con facilidad.

⁵ <http://es.kioskea.net/contents/web/webdesign.php3>

Al estar diseñando un sitio web se realizan **plantillas**, que representan el esquema gráfico de la página, en la figura 3.6 se muestra la estructura básica que tendrá la Página Web que se realiza en el proyecto de tesis. A continuación se describen las partes de que consta esta estructura:

- El logotipo en la esquina superior izquierda. Haciendo clic en éste los lleva a la página de inicio.
- Un menú localizado a la izquierda, que cambia según la opción en la que se encuentren.
- Un encabezado que contiene un header y un banner de navegación, con las diferentes opciones de acceso a la página.
- El cuerpo de la página que contiene la información, la cual va a cambiar según la opción elegida del menú.
- Un pie de página que incluye información de la empresa, su dirección y teléfono.

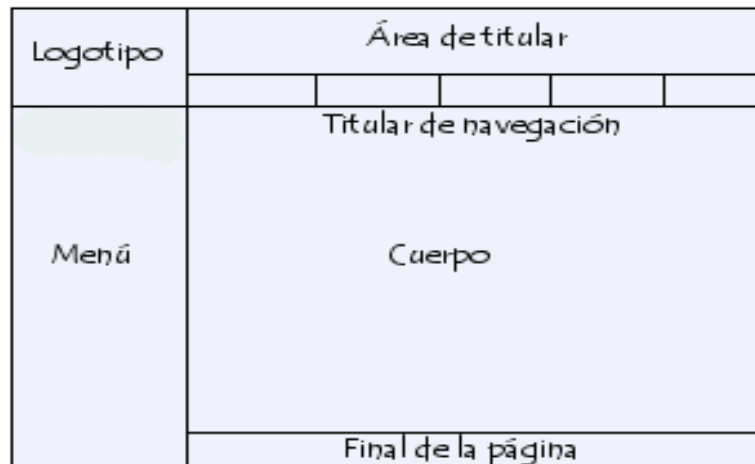


Fig. 3.6 Plantilla básica del sitio Web.

Tamaño de la Página.

El ancho de una página depende de la definición de la pantalla de usuario, de preferencia se debe realizar de un tamaño menor a la definición horizontal para que se pueda ver la información fácilmente. Con respecto a lo largo no debe exceder a cinco veces lo largo de la pantalla, las páginas excesivamente largas suelen ser tediosas y toma mucho tiempo el descargarlas. Algunos puntos a considerar en el tamaño de la página son:

- Elegir un ancho de página entre 600 y 800 píxeles de ancho.
- Utilizar un script para detectar la resolución de la pantalla del usuario.

Posicionamiento de la información.

Debido a la forma que tenemos de leer que es en diagonal, de la esquina superior izquierda hacia la esquina inferior derecha, es conveniente colocar la información más importante en la parte superior, que es donde se comienza a leer, es fundamental centrar la atención en ese punto de la página. Con lo que respecta al proyecto, la parte publicitaria de la empresa, así como los servicios que ofrece se colocarán en esta posición.

Elección de colores.

Este manejo de colores se usará dependiendo del tipo de página, no se utilizarán las mismas combinaciones para una página de niños, o de arte, o turística, en donde es conveniente llamar la atención del usuario; que para una página de una empresa, en este caso no es recomendable utilizar tanta variedad de colores, lo más sencillo le da más seriedad a la página, generalmente los colores que se utilizan deben corresponder con los colores del logo de la empresa, debe elegirse un color dominante y usarse como color principal de la página web. Teniendo el color dominante se seleccionarán otros colores afines (dos o tres) a éste y usarse sólo para destacar algunos

elementos de la página. En el caso de la página web que se está realizando se utilizará el azul como color dominante y diferentes gamas de este color, pasando por el marino, azul-gris, azul cielo, etc. Esto con el fin de que vaya acorde a los colores de la empresa y proporcionarle una imagen seria.

Selección de imágenes.

Como se vio en el capítulo anterior tanto el abuso de las imágenes como la carencia de éstas provoca una mala impresión de una página web, algunas veces se quiere suplir la carencia de información con imágenes, provocando que la descarga de la página se vuelva muy lenta, esto también ocurre con el abuso de imágenes .gif (animadas) que suelen ser molestas para el usuario. Por el contrario la carencia de imágenes y el abuso de información en texto, también suele ser tediosa para quien visita un sitio web, lo ideal es que se mantenga el equilibrio con las imágenes.

En el caso particular de la página web que se está desarrollando se usarán fotografías de las diferentes instalaciones o servicios que la empresa proporciona y éstas irán mezcladas con la información. En la etapa de desarrollo se mostrará un ejemplo de cómo la página queda terminada.

Tipografía.

Generalmente es conveniente usar máximo de dos a tres tipos de letra, unas que se utilicen solamente para realzar los títulos y una fuente clásica como arial, verdana, helvética, etc. para la mayor parte del contenido. Algunas fuentes no clásicas pueden visualizarse de forma diferente dependiendo de la definición de la pantalla del usuario. Para este proyecto se utilizaron 2 tipos de letra que me parecieron claros, legibles y que van muy relacionados con la tipografía que se usa en la documentación de la empresa, estos tipos son:

- **Tempus Sans ITC.** Para los títulos y menús.

- **Verdana.** Para el grueso de la información.

En la figura 3.7 se muestra el primer prototipo de lo que será el esquema básico de la página, aclarando que puede sufrir cambios durante el desarrollo.

3.2.2 Creación de base de datos y tablas.

Cómo se ha venido describiendo el sitio web que se quiere desarrollar además de ser una página que promueva sus servicios y ser medio de contacto con los clientes, también se utilizará como herramienta de control de proyectos para el administrador, los líderes de proyectos y los clientes. Para la realización de esta herramienta se utilizará el ASP.Net con el Visual Studio para los programas de ejecución en Internet, éstos deberán interactuar con una base de datos hecha en Microsoft SQL Server 2000. A continuación se mostrará la base de datos que fue creada para el proyecto, así como las tablas y los campos que las componen, éstos últimos con los tipos de datos que contendrán.

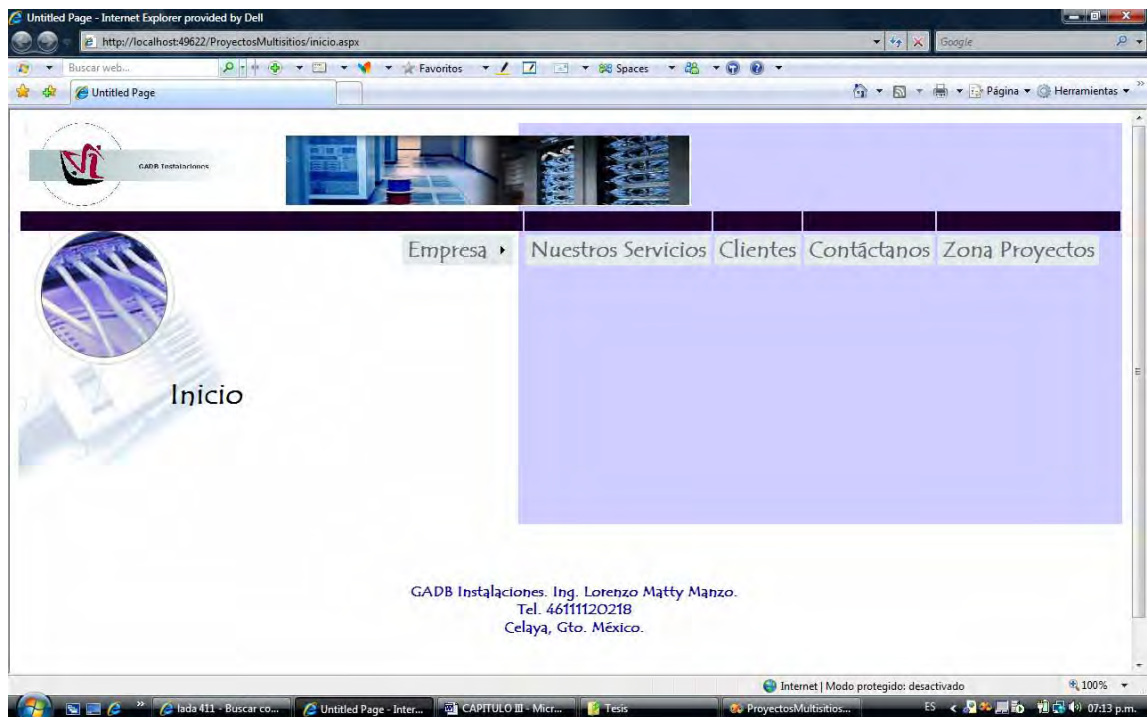


Fig. 3.7 Prototipo del sitio web

Base de datos.

Las bases de datos en aplicaciones Web permiten almacenar datos, recuperarlos, modificarlos, eliminarlos y consultarlos, como cualquier aplicación de base de datos convencional. Aunque en una de tipo Web los componentes multimedia permiten una representación más amena y más efectiva de la información.

Primeramente se creó la base de datos en el Administrador Corporativo de SQL Server 2000, se le puso el nombre de: **ProyectoMultisitios**. En la figura 3.8 se muestra la base de datos ya creada en el SQL Server.

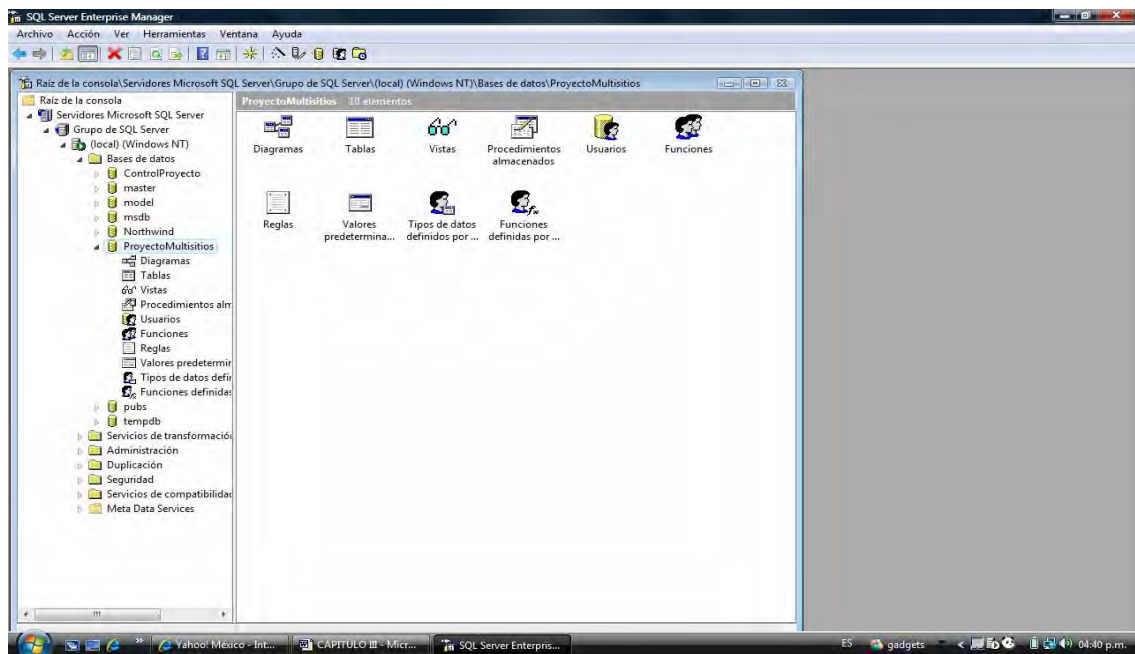




Fig. 3.8 Base de Datos del Proyecto.

Tablas.


Estas conformarán la base de datos, es el lugar en donde se guardan los datos recolectados por un programa, están compuestas de campos de diferentes tipos, según el dato que se quiera almacenar. Para la creación de nuestras tablas se utilizará el Administrador Corporativo del SQL Server. A continuación se

muestran las tablas que contendrá el proyecto, junto con los nombres de los campos y los tipos de datos que almacenarán, el símbolo  indica que ese campo será la llave primaria de la tabla.


Usuarios

Nombre de la columna	Tipo comprimido
 idUsuario	int
NomUsuario	varchar(60)
Empresa	varchar(50)
Cargo	varchar(50)
Telefono	varchar(40)
email	varchar(80)
password	varchar(50)
TipoUsuario	varchar(20)


Lider

Nombre de la columna	Tipo comprimido
 idLider	int
NomLider	varchar(60)
Direccion	varchar(60)
Telefono	varchar(40)
email	varchar(120)
NoCuadrilla	int
NoEmpleados	int
password	varchar(50)
Tipo Usuario	varchar(50)

Proyecto

Nombre de la columna	Tipo comprimido
 idProyecto	int
NomProyecto	varchar(120)
FechaInicio	smalldatetime
FechaFin	smalldatetime
NomSitio	varchar(70)
Ciudad	varchar(60)
Estado	varchar(60)
IdLider	int
IdUsuario	int

TareaProyecto

Nombre de la columna	Tipo comprimido
 idTareaProyecto	int
Descripcion	varchar(300)
FechaInicioProy	smalldatetime
FechaFinProy	smalldatetime
FechaInicioReal	smalldatetime
FechaFinReal	smalldatetime
Observaciones	varchar(300)
idProyecto	int

A continuación se relacionan los tipos de datos que contendrán los campos de la tabla según la especificación definida:

int. Almacena valores numéricos enteros de -2,147,483,648 a 2,147,483,647.

tinyint. Almacena valores numéricos enteros de 0 a 255.

varchar. Almacena cadena de caracteres de longitud variable con un máximo de 8000 caracteres.

smalldatetime. Almacena fecha y hora en el mismo campo del 1 de enero de 1900 al 6 de junio de 2079 con una precisión de un minuto. Este es un formato reducido de almacenar fecha y hora, a diferencia del datetime éste almacena fecha y hora pero sólo hasta llegar a los minutos, los segundos y milisegundos son siempre 0. Internamente se almacenan como enteros de 2 bytes.

Las tablas están relacionadas por llaves primarias o índices, la relación que se establece entre las tablas creadas se muestra en la figura 3.9 este diagrama lleva por nombre ControlMultisitios.

Adicionalmente a estas tablas se crearon 3 más que van a ser utilizadas en la parte del contacto con el cliente o cliente potencial con el propósito de que la empresa tenga almacenada información que le sirva para tener comunicación con éstos, resolver dudas, hacer cotizaciones, nuevas contrataciones, citas, etc.

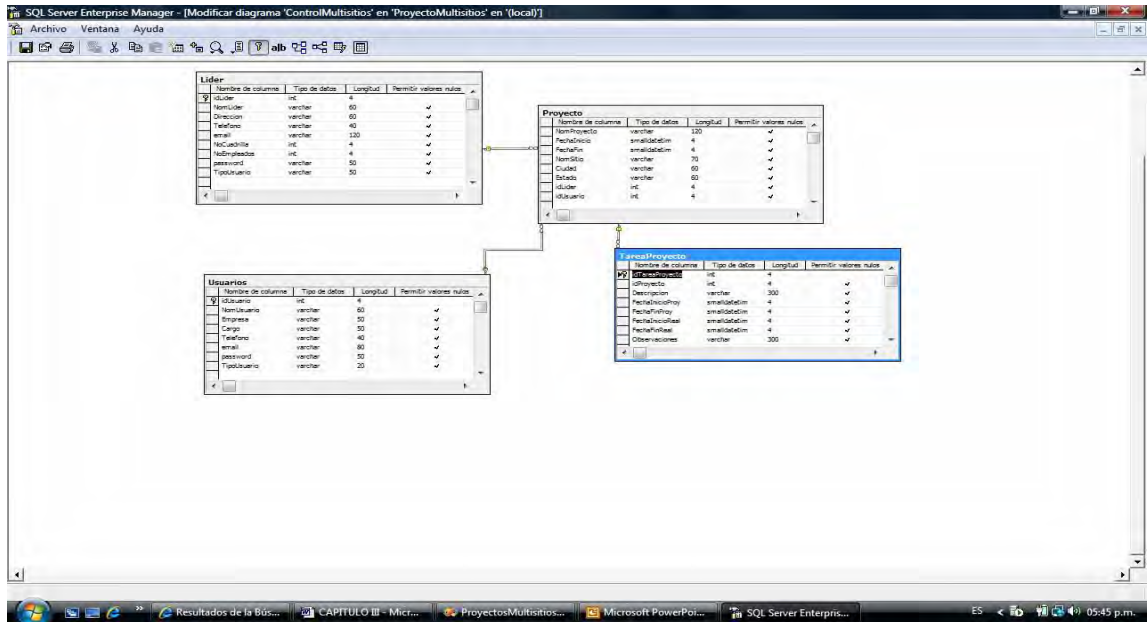





Fig. 3.9 Diagrama ControlMultisitios.

En dichas tablas se va a guardar información del cliente y comentarios que éste tenga que hacerle a la empresa, así como una serie de preguntas y respuestas de un pequeño formulario que se le planteará y se utilizará para evaluar la página. A continuación se muestran estas tablas con los datos que contendrán y el tipo utilizado:

Pregunta

Nombre de la columna	Tipo comprimido
 idPregunta	tinyint
Pregunta	varchar(200)

Respuesta

Nombre de la columna	Tipo comprimido
 idPregunta	tinyint
 idRespuesta	tinyint
Respuesta	varchar(100)

Contacto

Nombre de la columna	Tipo comprimido
email	varchar (120)
nombre	varchar (120)
telefono	varchar (50)
idPregunta	tinyint
idRespuesta	tinyint
comentario	varchar(1024)
fecha	smalldatetime

En la figura 3.10 se muestra el diagrama Contactanos con las tablas usadas para el formulario y el envío de la información a la empresa, así como sus relaciones con las llaves primarias.

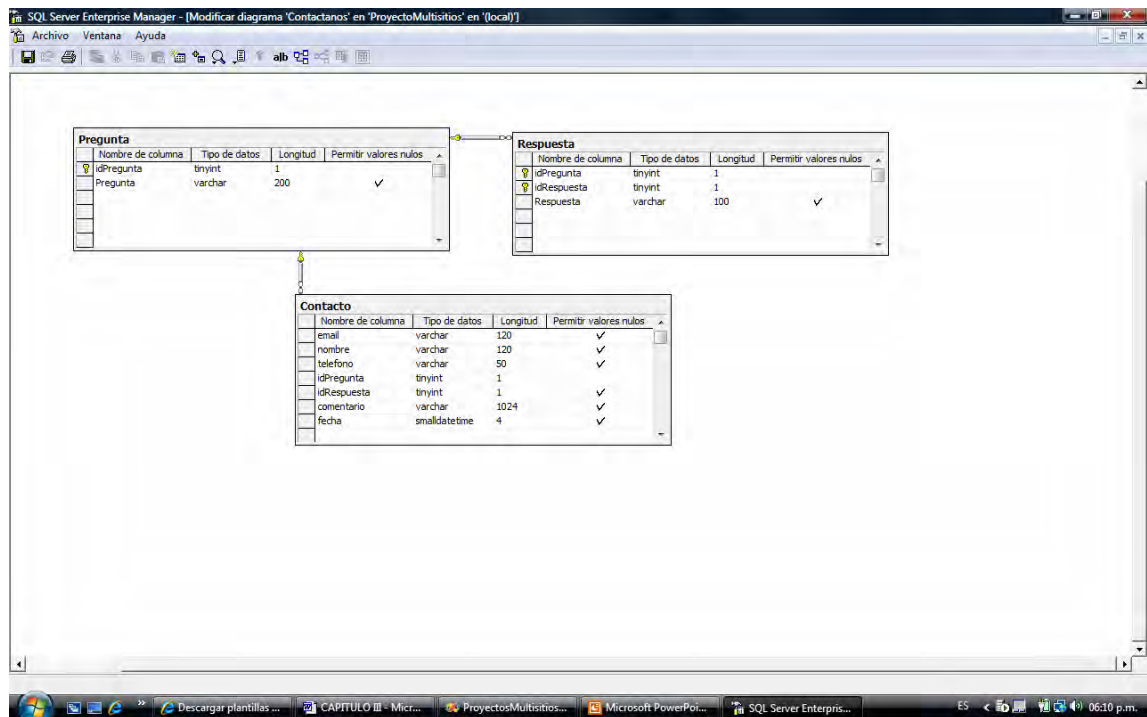


Fig. 3.10 Diagrama Contactanos

3.3 Instalación de Visual Studio y Microsoft SQL Server.

Aunque estas herramientas se utilizaron en la parte del diseño, dedico este espacio para marcar algunas especificaciones que se deben considerar al momento de instalarlas.

Microsoft SQL Server 2000.

En este caso hablamos de esta versión porque es la que se utilizó en la realización del proyecto, debido a que es una versión nueva de Microsoft SQL Server, no la más actualizada pero si bastante robusta, además de que es un potente motor de bases de datos de alto rendimiento y es compatible con la versión de Visual Studio que se usó para la realización de la aplicación Web.

Como dije SQL Server 2000 es una potente base de datos que aumenta y expande las posibilidades, rendimiento, fiabilidad, calidad y facilidad de uso de su anterior versión, SQL Server 7.0. Incluye un buen número de novedades que reafirman a SQL Server 2000 como la plataforma idónea para aplicaciones de comercio electrónico, data warehousing y OLTP (procesos transaccionales en línea).

Ofrece escalabilidad y presenta nuevas soluciones para reducir la carga del servidor, asegurar el funcionamiento continuo y aporta una administración avanzada y alta capacidad de configuración.

Requerimientos mínimos de sistema:

- Procesador Intel Pentium III Xeon a 733 MHz o un AMD Athlon a 1000 MHz o compatibles
- 512MB de RAM
- 20 GB de espacio libre en disco, según el tipo de instalación.

Requerimientos óptimos de sistema:

- Procesador Intel Dual a 1 GHz
- 4 GB de RAM
- 50 GB de disco duro

Sistema Operativo:

- Windows 2000/NT
- Windows XP
- Windows Vista

Existen tres tipos de instalación:

- **Local:** Se instala SQL Server en la máquina que se está utilizando en esos momentos para la instalación.
- **Remota:** Se puede instalar SQL Server en otra máquina que se encuentre en la red.
- **Desatendida:** Se puede instalar SQL Server automáticamente con los parámetros que hayamos marcado previamente. No es necesario estar presente para realizar este tipo de instalación.

En nuestro caso será del tipo Local, ya que se hará en la PC en la que se realizará el proyecto, este equipo cuenta con las especificaciones necesarias para un buen funcionamiento del SQL Server.

Lo más importante a destacar durante la instalación es que se necesita tener presente en todo momento la seguridad del servidor o del equipo. Para ello durante la instalación cuando se elija el modo de autenticación es recomendable el modo Mixto (autenticación de Windows y autenticación de

SQL Server) y es recomendable también no dejar la contraseña del sa (administrador) en blanco, como se mencionó, por motivos de seguridad.

Visual Studio.Net.

La herramienta que se va a utilizar para hacer la aplicación dinámica en ASP.Net es el Visual Studio; para su instalación se necesitan los siguientes requerimientos mínimos de sistema:

- PC con procesador Pentium II a 600 MHz o mayor.
- Microsoft Windows 2000 con Service Pack 3 o posterior, Microsoft XP Professional con Service Pack I o posterior, o Microsoft Windows Server 2003. Para el caso de Windows Vista todavía se están desarrollando Service Pack para que pueda funcionar de forma adecuada.
- 128 MB de memoria en RAM o mayor.
- 4 GB de espacio disponible en disco duro.
- Microsoft Internet Explorer con Service Pack 1.

Método optimizado de instalación:

- **Usar la cuenta de administrador local.** Esto evita problemas relacionados con los permisos. La cuenta de administrador local es la cuenta de mantenimiento. Utilice esta cuenta cuando instale y borre programas.
- **Detener todos los servicios y programas necesarios.** Muchos programas, particularmente los antivirus u otros programas como los administradores de servicio de bases de datos, interrumpen el proceso de instalación.
- **Vacíe la carpeta System\Temp.** Los asistentes de instalación utilizan la carpeta System\Temp con mucha frecuencia. Si la instalación deja

archivos o carpetas dañadas en esta carpeta, volverá a aplicar las versiones dañadas, en lugar de buscar copias nuevas en el origen de la instalación.

De esta forma se evitarán problemas posteriores debido a una mala instalación de la aplicación.

3.4 Uso de controles y herramientas de ASP.NET en la aplicación.

Control **Panel**. Es un contenedor de otros controles. Resulta especialmente útil si se desea generar controles mediante programación, ocultar, mostrar o adaptar un grupo de éstos. Este tipo de contenedor crea grupos lógicos de controles y sirve para aplicar estilos a áreas específicas de las páginas y le ayuda a ASP.Net a la hora de representar páginas.

Control **TextBox**. Las aplicaciones utilizan el control TextBox para que los usuarios puedan escribir una cadena o dar una entrada. Ellos aparecen como un cuadro y permite al usuario introducir texto, como el nombre de usuario, contraseña, cantidades, fechas, etc. El control TextBox puede enmascarar los datos correspondientes a contraseñas si el dispositivo del usuario admite esta funcionalidad. Un control TextBox debe colocarse en un control Form o Panel.

Control **DropDownlist**. Es igual que el control ListBox, admite una única selección y muestra los elementos de manera desplegable. Este se configura con la base de datos, es aquí de donde saca la información a desplegar.

Control **GridView**. Su función es mostrar los valores de un origen de datos en una tabla o rejilla tabular en donde cada columna representa un campo y cada fila representa un registro. El control **GridView** es muy poderoso ya que permite seleccionar, ordenar y editar los datos que contiene.

Controles de Validación.

Este tipo de Controles se utilizan para evitar que una aplicación falle por una mala captura del usuario, lo que se hace es comprobar si los datos proporcionados son correctos de acuerdo con la información solicitada. Existen varios tipos de controles de Validación capaces de realizar comprobaciones útiles como:

- Dejar en blanco un campo obligatorio. (Controles de campo requerido).
- Las contraseñas escritas 2 veces no coinciden. (Controles de comparación)
- Lo tecleado se sale del un rango o difiere de un valor o es igual a un valor. (Controles de comparación)
- Lo tecleado se adapta a un patrón determinado. (Controles de expresiones regulares)

Los **controles de Validación de ASP.Net** permiten:

- Capturar y validar cualquier cosa enviada por el usuario desde un formulario Web.
- Personalizar los avisos de error cuando los datos no pasan la validación.
- Escribir validaciones personalizadas de acuerdo a nuestras preferencias.

Los tipos de **controles de Validación** de que dispone **ASP.Net** son:

RequiredFieldValidator: comprueba que no se deja en blanco un campo obligatorio.

CompareValidator: compara lo tecleado por el usuario con:

- Un valor constante.
- El valor de otro control.
- Un determinado tipo de datos (String, Date, Integer, etc.)
- Además puede usar operadores relacionales (mayor que, menor que, igual, etc.)

Rangevalidator: compara lo tecleado por el usuario con un rango de valores.

RegularExpressionValidator: compara lo tecleado por el usuario con un patrón definido en una expresión regular.

ValidationSummary: muestra resúmenes de la validación con los mensajes de error.

Crystal Reports.

El **Crystal Reports** se incluye dentro del Visual Studio 2002 y posterior, permite a los desarrolladores crear informes interactivos para sitios Web y aplicaciones de cliente inteligente que se pueden escalar para adaptarse a las necesidades de los usuarios finales.

Crystal Reports para Visual Studio proporciona a los desarrolladores una forma rápida y productiva de crear e integrar informes con calidad de presentación en las aplicaciones, sin salir del entorno de desarrollo de Visual Studio conocido.

A continuación se listan algunas ventajas que se tienen al incluir los reportes con esta herramienta dentro de una aplicación Web:

- El reporte forma parte del proyecto, de manera que no se tiene que agregar ningún archivo de reporte externo.
- Existe una mayor flexibilidad para manejar objetos.
- Se tiene mayor control sobre los objetos que forman parte del reporte.
- Tener una vista previa en tiempo de diseño que permite ver una versión de sólo lectura del informe de Crystal con sus datos predeterminados. Se mostrará la vista previa independientemente de si hay fallos de inicio de sesión.

Dentro del Crystal Reports se pueden realizar gráficas para la representación de los datos contenidos en el reporte, para eso se utiliza el **Chart Expert**. En el **Chart Expert** se puede seleccionar entre varios tipos de gráficos el que se quiere presentar en el reporte, en una parte están los diferentes tipos de gráficos y en otra los estilos de gráficos por cada tipo, éstos se configuran automáticamente tomando los valores que se le indiquen.

3.5 Servidor Web.

Si queremos tener un servidor Web en un equipo es necesario que cuente con ciertas especificaciones para que pueda albergar una aplicación de ASP.Net y sea compatible con éste, entre ellas están:

- Que cuente con Windows 2000, Windows XP, Windows Server 2003 o Windows Vista Business o Windows Vista Ultimate.
- Que el equipo cuente con IIS (Internet Information Server); los sistemas operativos mencionados anteriormente ya traen una versión de IIS, para que sea compatible con ASP.Net se tiene que usar la versión de IIS 5.0 o

superior. El acceso al IIS se realiza mediante el ícono de “ Servicios de Internet Information Server “ situado en las “ Herramientas administrativas “ dentro del Panel de Control.

- El servidor Web IIS permite administrar las aplicaciones Web y comunicarse con los navegadores cliente mediante protocolo http (protocolo de transferencia de hipertexto). El IIS también ofrece otros servicios de protocolo, como transferencia de archivos (FTP), servicio de correo electrónico (SMTP) y servicio de noticias (NNTP).
- Se necesita que el servidor Web tenga instalado .Net Framework para poder procesar código de ASP.Net, al instalar el Visual Studio.Net se especifica la instalación del .Net Framework. Es importante decir que los navegadores cliente que accedan a la aplicación Web no necesitan tener instalado IIS ni .Net Framework ya que es el servidor Web el que tiene que saber interpretar el código de ASP.Net.
- Tener los archivos .aspx que corresponden a las páginas Web.
- Un archivo de ensamblado (DLL) situado en la carpeta Bin de la aplicación Web, que contiene el código de servidor que necesitan las páginas aspx.
- Un archivo llamado Global.asax que sirve para el control general de la aplicación durante su ejecución.
- Un archivo llamado Web.config donde se establece la configuración de la aplicación. Aunque este fichero es opcional se necesita cuando se quieren establecer parámetros de configuración que no sean los de por defecto.
- De manera adicional también puede aparecer en la carpeta Web otro tipo de archivos como:
 - Archivos .ascx (controles personalizados de usuario de ASP.Net)
 - Archivos .asmx (servicios Web XML de ASP.Net).
 - Páginas .htm ó .html (páginas Web estáticas)

- Páginas .asp (páginas activas de servidor)
- Archivos .css (hojas de estilo CSS, Cascade Style Sheet).
- Documentos, imágenes, etc.

En la figura 3.11 se muestra de manera gráfica el contenido de la carpeta de la aplicación de una página Web.⁶

Como punto final cabe mencionar que el servidor web también puede funcionar como Servidor de Datos, es decir, que puede albergar la aplicación Web dinámica y la base de datos a la que se hace referencia en sus páginas.

Para fines de desarrollar y probar la aplicación se utilizará un equipo que tendrá las funciones de un servidor Web, el Visual Studio.Net al ejecutar las páginas lo que hace es abrir un puerto y emular como si se tuviera un site en Internet.

Es difícil contar con un equipo lo suficientemente robusto para que funcione como servidor Web y servidor de datos, tomando en cuenta además que el equipo tendría que permanecer encendido todo el tiempo para que la conexión con la página no se perdiera, y contar con mucha seguridad que impida la entrada de hackers, lo más recomendable es rentar un servicio de Hosting con una empresa especializada que trabaje con páginas en ASP.Net y pagar una renta mensual, si queremos que nuestra página funcione eficientemente.

Algunos ejemplos de empresas que albergan sites con programación ASP.Net son:

<http://www.elenlacehost.com/>

<http://www.servidor2.net/>

⁶ <http://www.desarrolloweb.com/articulos/1519.php>

<http://winnehost.net/>

<http://www.whoosting.com.mx/>

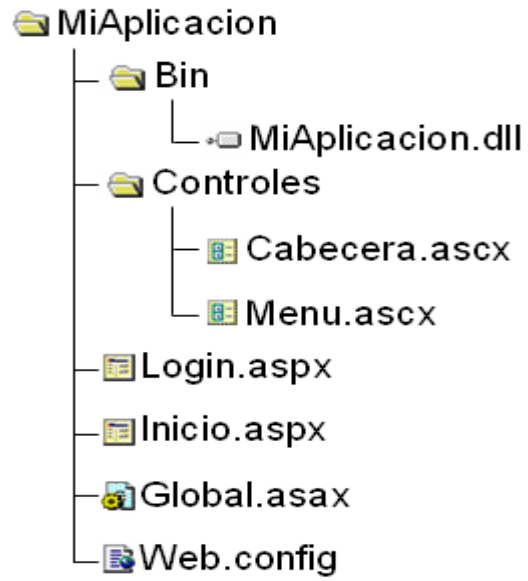


Fig. 3.11 Contenido de la carpeta de una aplicación de ASP.Net

Capítulo IV. IMPLEMENTACIÓN Y ENTREGA.

4.1 Desarrollo.

En esta etapa se realizará la aplicación de acuerdo a lo presentado en las etapas de análisis y diseño, es decir, se llevará a la práctica la creación de la Interfaz Web.

Como se vió en el Capítulo II, el objetivo de esta tesis es realizar una aplicación Web, es decir una página que ofrezca servicios avanzados de administración de proyectos, y se construye utilizando un lenguaje de programación del lado del servidor, en este caso ASP.Net con ayuda del Visual Studio.Net, a través de este lenguaje se construirá la aplicación con datos extraídos de una base de datos (realizada en MS SQL Server) y será capaz de dar seguimiento a las acciones del cliente, el líder y el administrador y guardar información valiosa durante las visitas que se realicen a esa página y de los movimientos o interacciones que tengan con la aplicación.

En este capítulo se incluirán fragmentos de la programación en conjunción con el diseño que llevarán cada una de las páginas que conformarán el sitio Web.

En la figura 4.1 se muestra la herramienta del Visual Studio en donde se harán los diseños de las páginas y las programaciones en ASP.Net del sitio dinámico.⁷ Los archivos con extensión **.aspx** contendrán el diseño de la página Web, su distribución, tipo de letra, tablas, menús, hiperlinks, gridviews, dropdownlist, etc., es decir estos archivos contienen solamente contenido HTML y código ASP.Net con controles de servidor y controles HTML. Ahora bien a los archivos con extensión **.aspx.vb**, se les llama archivos de código de apoyo y tendrán la programación para acceso a la base de datos y que el resultado de ésta se vea reflejada en la página Web, además contienen declaraciones de variables,

⁷ http://asp.com-e.net/template_permalink.asp?id=97

funciones y procedimientos; es el ambiente de programación de la página. De forma predeterminada, ambos archivos deben estar juntos en la misma carpeta.

4.1.1. Elaboración de páginas informativas.

Como se planteó en la etapa del análisis se pretende que la Interfaz Web también contenga una parte informativa, que promueva el negocio, que presente a los clientes actuales y potenciales la información de la empresa y de los servicios que presta. A continuación se presentarán las páginas que contendrán este bloque informativo con una síntesis del contenido de la página.

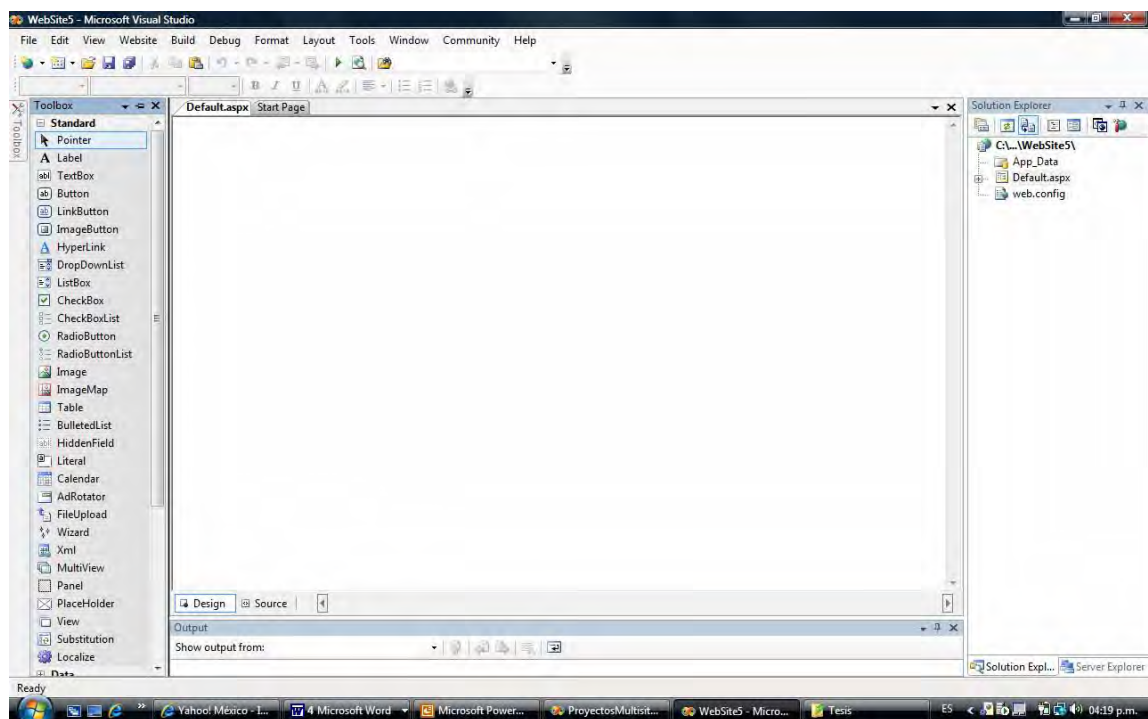


Fig. 4.1 Herramienta para diseño y programación de páginas, Visual Studio.

Página de Inicio.

En esta página únicamente se muestra el menú de la aplicación informativa y los datos de la empresa, ésto se puede ver en la figura 4.2.

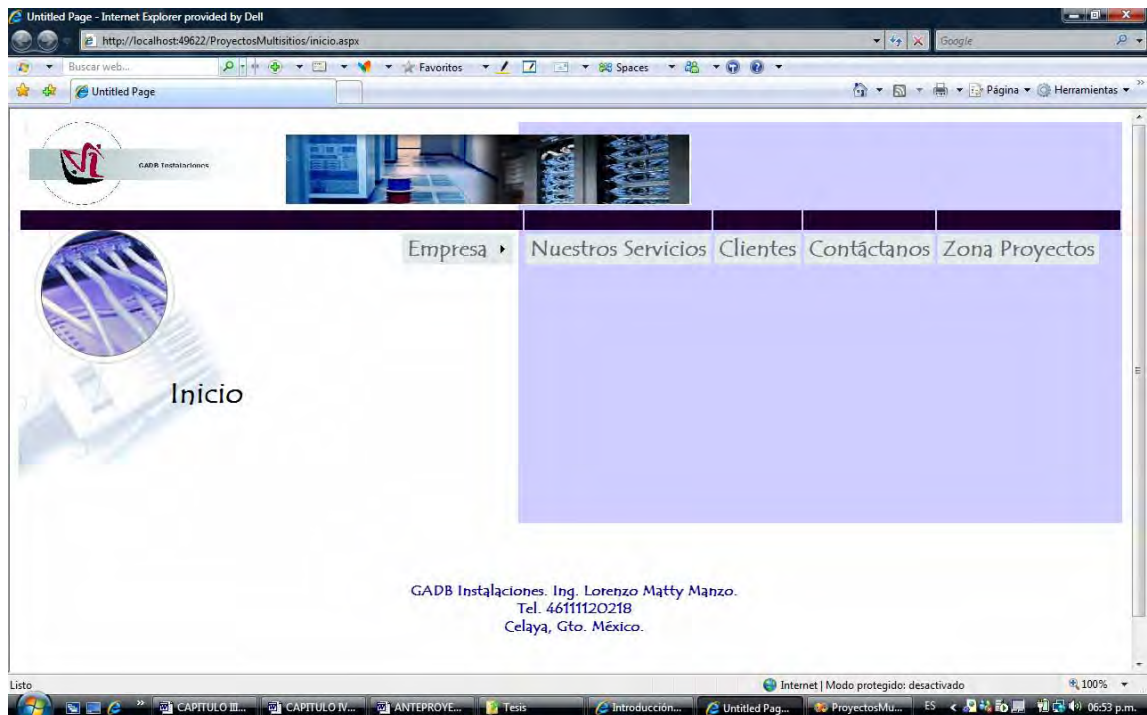


Fig. 4.2 Página de Inicio

4.1.1.1 Página Empresa.

Dentro de esta página sólo se presenta el menú con dos opciones a escoger:

- **¿Quiénes somos?**. Que contiene la información desde que inició la empresa, su fundador, los principales servicios que prestan, así como una descripción breve de las actividades que realizan y que le dan una ventaja en relación con sus competidoras. En la figura 4.3 se muestra el resultado final de ésta página.

- **Nuestra Filosofía.** En esta página se mostrará la misión, visión y objetivos primordiales que la empresa tiene y quiere dar a conocer a sus clientes. En la figura 4.4 se presenta como queda terminada la página.

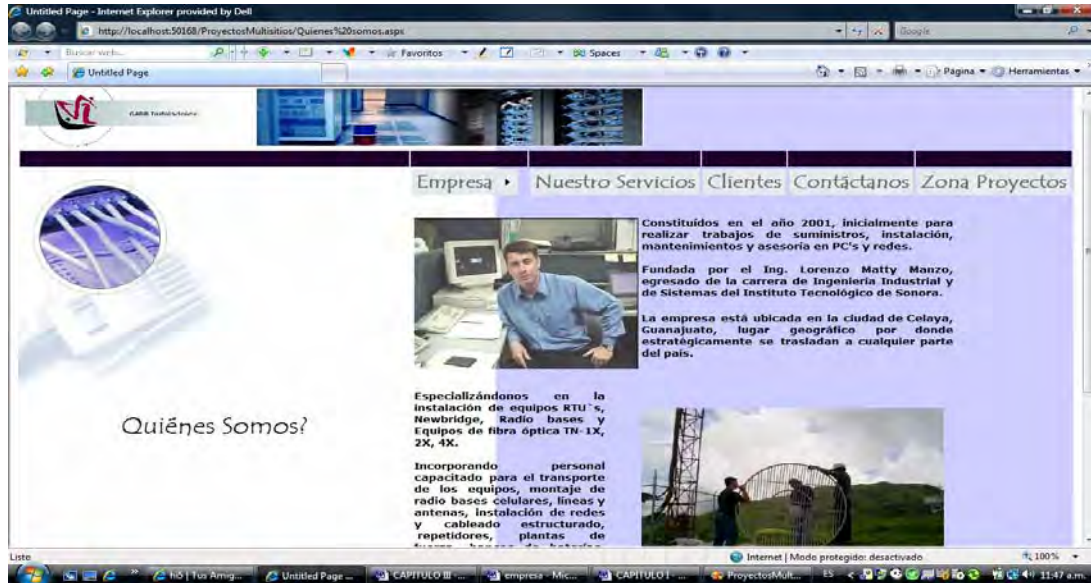


Fig. 4.3 Página "Quiénes Somos?" del menú Empresa.



Fig. 4.4 Página "Nuestra Filosofía" del menú Empresa.

4.1.1.2 Página Nuestros Servicios.

En esta página se hace una descripción a detalle de los servicios que la empresa ofrece, así como de las principales políticas que rigen a dichos servicios para la plena satisfacción del cliente. En la figura 4.5 se muestra el resultado de la página ya concluida.



Fig. 4.5 Página "Nuestros Servicios".

4.1.1.3 Página Clientes.

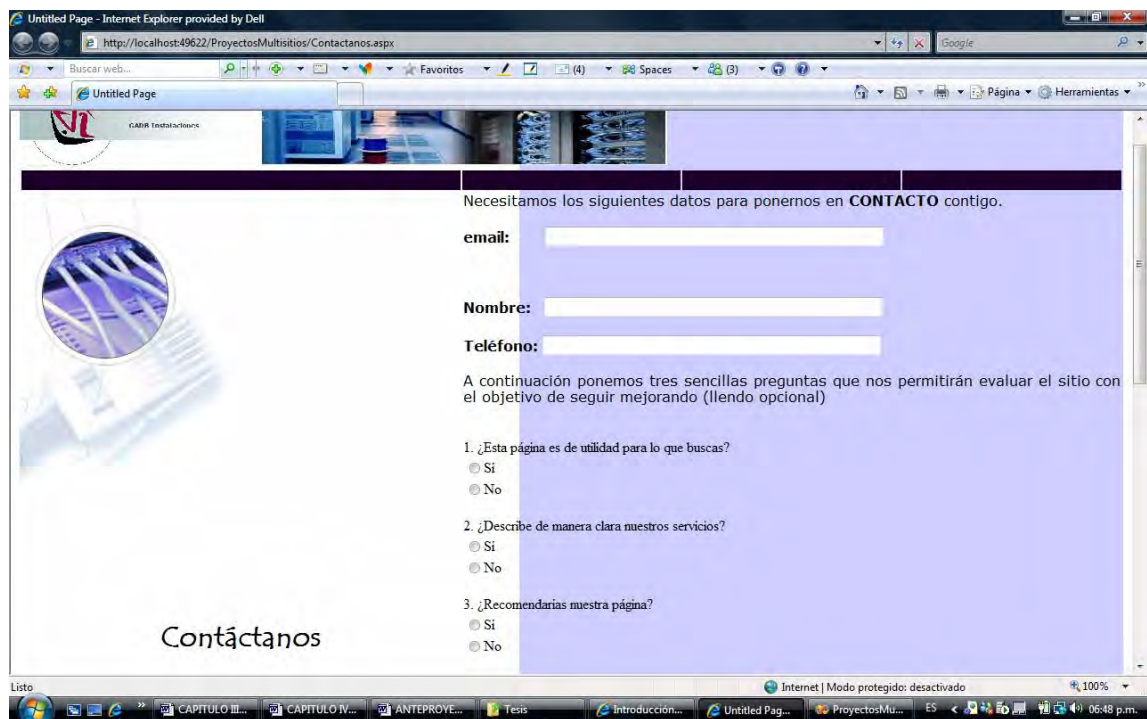
Esta página contiene los puntos que la empresa considera esenciales para que el servicio prestado al Cliente esté completo y le brinde a la empresa una mejor imagen ante éste. En la figura 4.6 se muestra la página terminada.



Fig. 4.6 Página "Clientes"

4.1.1.4 Página Contáctanos.

En esta página consta de dos partes en la figura 4.7 se muestra un formulario breve que el cliente tiene que llenar para saber si la página ha cubierto sus expectativas, y un campo en donde colocará su correo electrónico, nombre y teléfono los cuales quedarán grabados en la base de datos en la tabla Contacto y el administrador podrá revisar en el momento que quiera estos datos para contactar al cliente; la parte del formulario se realizó con un contenedor de controles de ASP.Net, llamado **panel** para mostrar las respuestas, el uso de este control se explicó en el capítulo III.



The screenshot shows a web browser window with the URL `https://localhost:49622/ProyectosMultisitios/Contactanos.aspx`. The page has a light blue background and a dark header. On the left, there is a circular image of a network switch. The main content area contains the following text and form elements:

Necesitamos los siguientes datos para ponernos en **CONTACTO** contigo.

email:

Nombre:

Teléfono:

A continuación ponemos tres sencillas preguntas que nos permitirán evaluar el sitio con el objetivo de seguir mejorando (llendo opcional)

1. ¿Esta página es de utilidad para lo que buscas?
 Sí
 No
2. ¿Describe de manera clara nuestros servicios?
 Sí
 No
3. ¿Recomendarías nuestra página?
 Sí
 No

At the bottom left of the form area, the word "Contáctanos" is written in a large, dark font.

Fig. 4.7 Formulario de la página "Contáctanos"

En la segunda parte que se muestra en la figura 4.8 en donde el cliente puede poner el comentario que quiera ya sea para la realización de algún servicio, alguna cotización, queja, etc. cualquier cosa que se le quiera hacer llegar

directamente al Administrador. Es el medio por el cual se mantendrá comunicación directa entre el Cliente actual o potencial y la empresa.

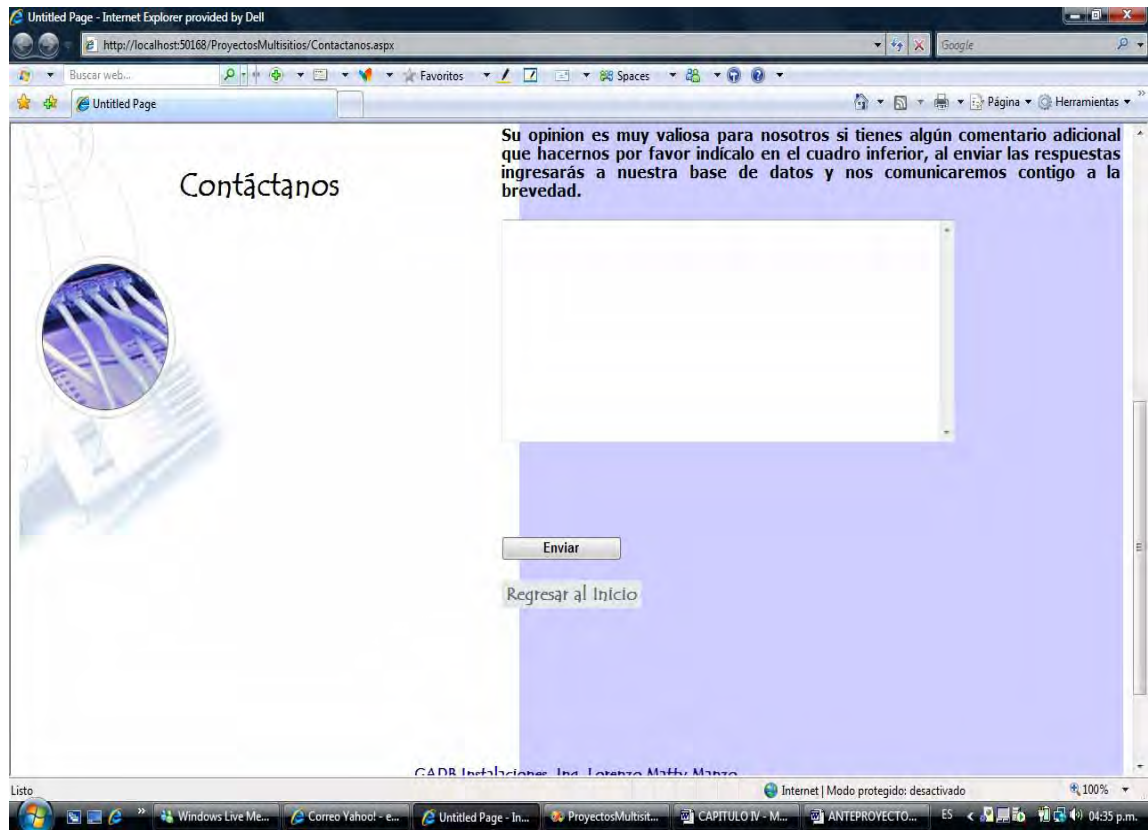


Fig. 4.8 Comentarios de la página "Contáctanos".

4.1.2. Elaboración de aplicación dinámica.

Toda aplicación dinámica Web como se explicó en el Capítulo II tiene una interacción entre el usuario, la aplicación o sitio Web y una base de datos, en este caso ya se presentaron las tablas de la base de datos ProyectoMultisitios y sus relaciones, realizadas en Microsoft SQL Server a continuación se dará seguimiento a las páginas dinámicas que conforman la aplicación.

4.1.2.1 Página Zona Proyectos.

Dentro del menú que aparece en la página de inicio hay una opción para entrar en la Zona Proyectos, que ofrece una aplicación para la administración, consulta y reportes de los diferentes proyectos y sus tareas relacionadas, tanto para el administrador, el líder de proyectos y el cliente. Según sea el tipo de usuario que acceda a la aplicación, así serán las operaciones que podrán realizar.

La primera parte de la aplicación presenta una página de acceso, en donde se solicita un usuario, en este caso un email o cuenta de correo (que resulta más seguro por el tamaño que representa que un usuario común y corriente) y un password. Después de capturarlo y al apretar el botón de **Aceptar** se hace una búsqueda en la base de datos, en la tabla de Usuarios, si éste no existe o es incorrecto, manda un mensaje de error pidiendo que se registre. A continuación muestro el código en ASP que da acceso al usuario a la aplicación, éste se captura al dar doble click al botón en el archivo .aspx y automáticamente se abrirá en el archivo correspondiente .aspx.vb la siguiente subrutina:

```
Protected Sub Aceptar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Aceptar.Click
    Dim Cn As New OleDbConnection("Provider=SQLOLEDB;userid=sa;Password=;
initial Catalog=ProyectoMultisitios;Data Source=ERIKAPERSONAL")
    Dim DR As OleDbDataReader
    Dim Comando As OleDbCommand
    Cn.Open()
    Comando = (New OleDbCommand("Select isNull(count(*),0),TipoUsuario
from Usuarios where email='" & Usr.Text & "'
And password='" & Pwd.Text & "' Group By TipoUsuario", Cn))
    DR = Comando.ExecuteReader
    If DR.Read Then
        If DR(0) = 1 Then
            If DR(1) = "Cliente" Then
                Session("conectado") = 52 'Bandera
                Response.Redirect("MenuCliente.aspx")
            End If
        End If
    End If
Else
```

```
Estado.Text = "Datos Incorrectos. Regístrate"  
End If  
End Sub
```

Cabe aclarar que esta página de acceso es sólo para el Cliente, en la figura 4.9 se muestra dicha página de acceso.

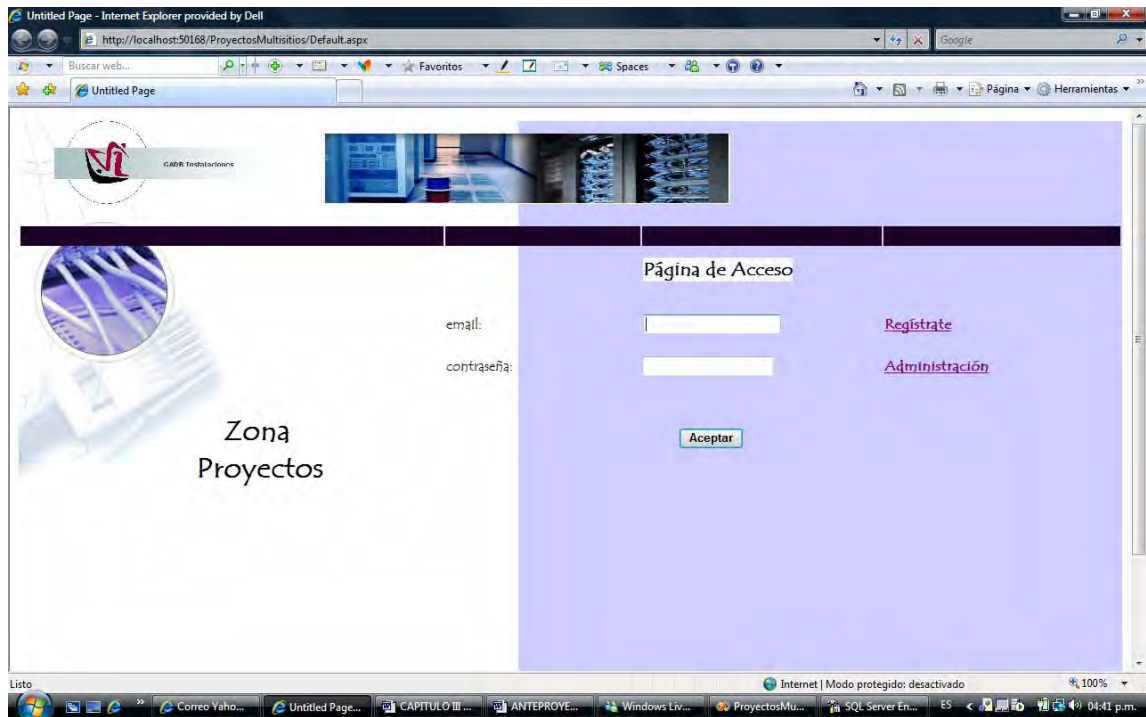


Fig. 4.9 Página de Acceso del Cliente a la aplicación

Como se mencionó anteriormente si el Cliente no teclea los datos correctos, o éstos no existen en la base de datos, marca un error y se solicita que se registre, además para validar que los datos capturados en el campo del email correspondiera a una dirección de correo electrónico se utilizó el **RegularExpressionValidator**, control de validación explicado en el capítulo III. Ahora si el cliente no está registrado, la página de acceso tiene un **hiperlink** (Control que muestra un vínculo a otra página Web) llamado **Regístrate**, en donde podrá darse de alta. En la figura 4.10 se muestra el formulario que se tiene que llenar para dar de alta un nuevo cliente.

Básicamente se utilizaron controles TextBox para esta página. Estos ya fueron explicados en el capítulo anterior.

Se utilizaron **Controles de Validación** para el llenado de los datos solicitados, si se observa en la figura, todos los campos son de llenado obligatorio así que se uso el control **RequiredFileValidator** para obligar al cliente o usuario al llenado de las cajas de texto.

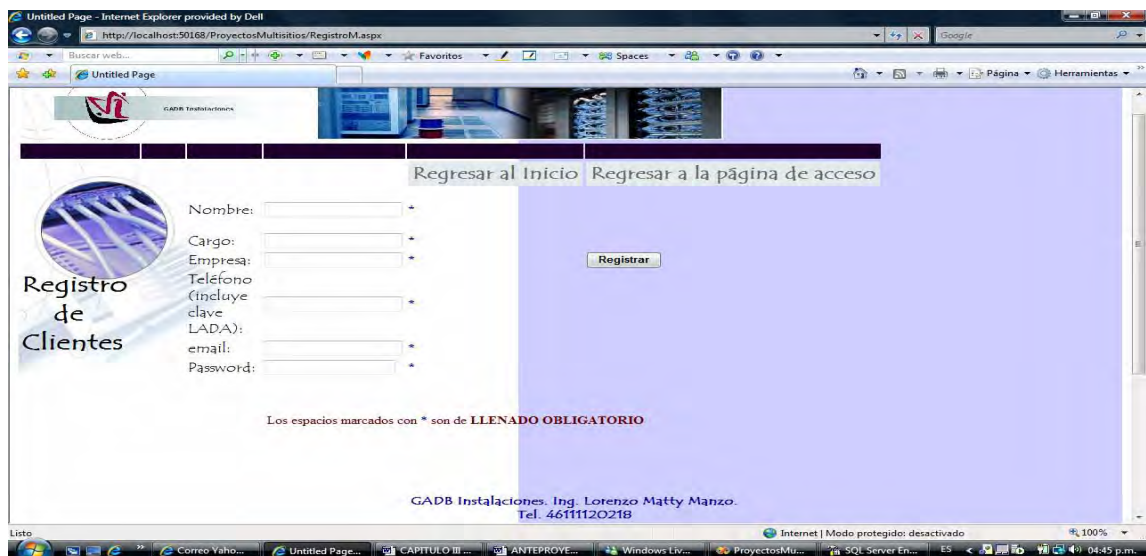


Fig. 4.10 Página de Registro de Clientes

En la figura 4.11 se muestra cómo en las propiedades del control **RequiredFileValidator** en **ErrorMessage** se puede poner el mensaje que se muestra si el campo es dejado en blanco.

Dentro del llenado de este formulario se encuentra un campo que es el email, al igual que en la página de Contáctanos, a éstos se le puso un control de Validación llamado **RegularExpressionValidator**, que compara el contenido del campo con un formato predeterminado para una dirección de correo electrónico. En la figura 4.12 se muestra el contenido del ErrorMessage y del ValidationExpression, en éste último se pueden escoger distintas opciones

como se muestra en la figura, en este caso se eligió Internet e-mail address. Dentro de las propiedades existe un campo que se llama ControlToValidate en donde se seleccionará el nombre del TextBox a validar en este caso el TextBox se llama email.

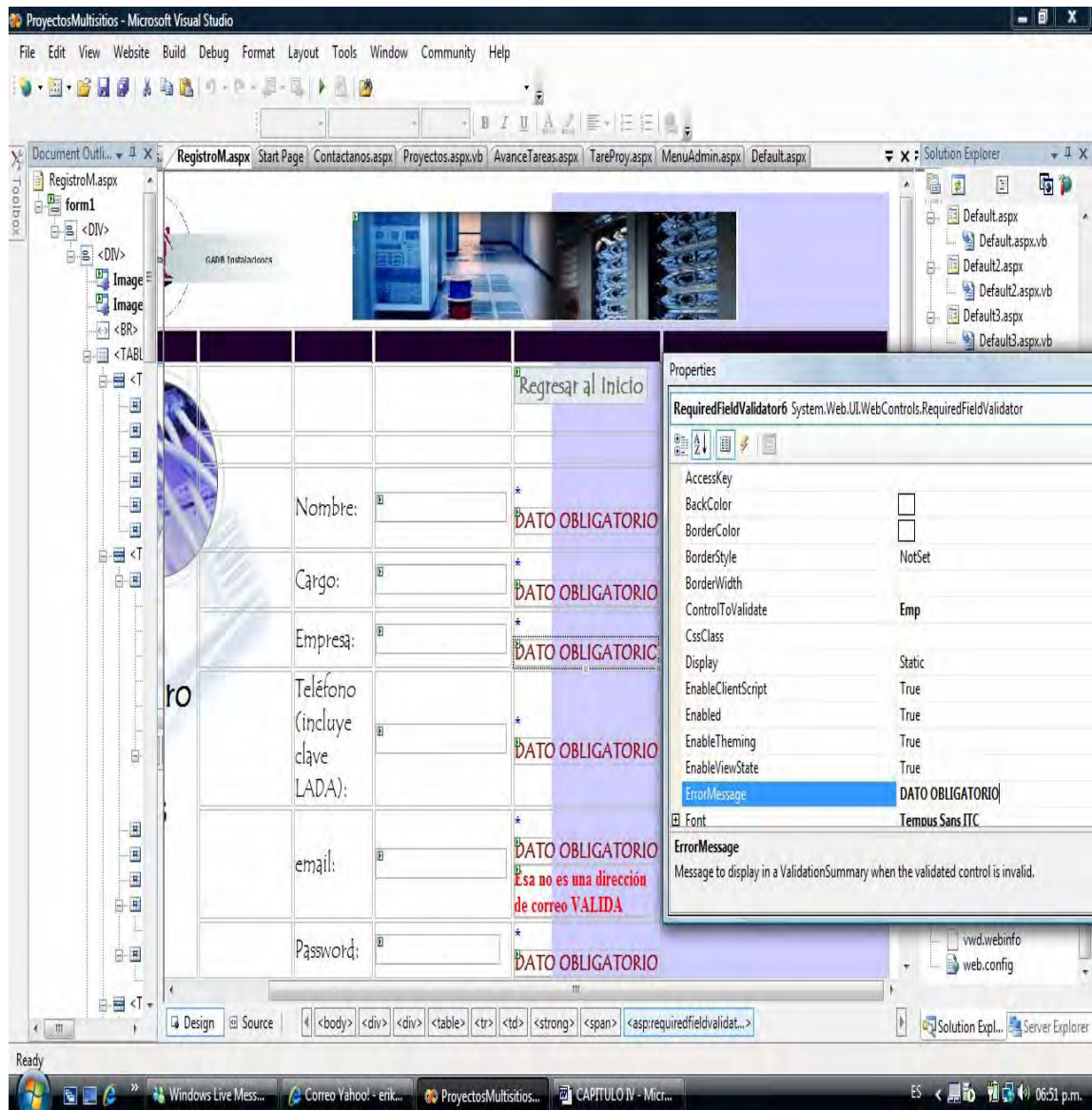


Fig. 4.11 Propiedades del RequiredFileValidator.

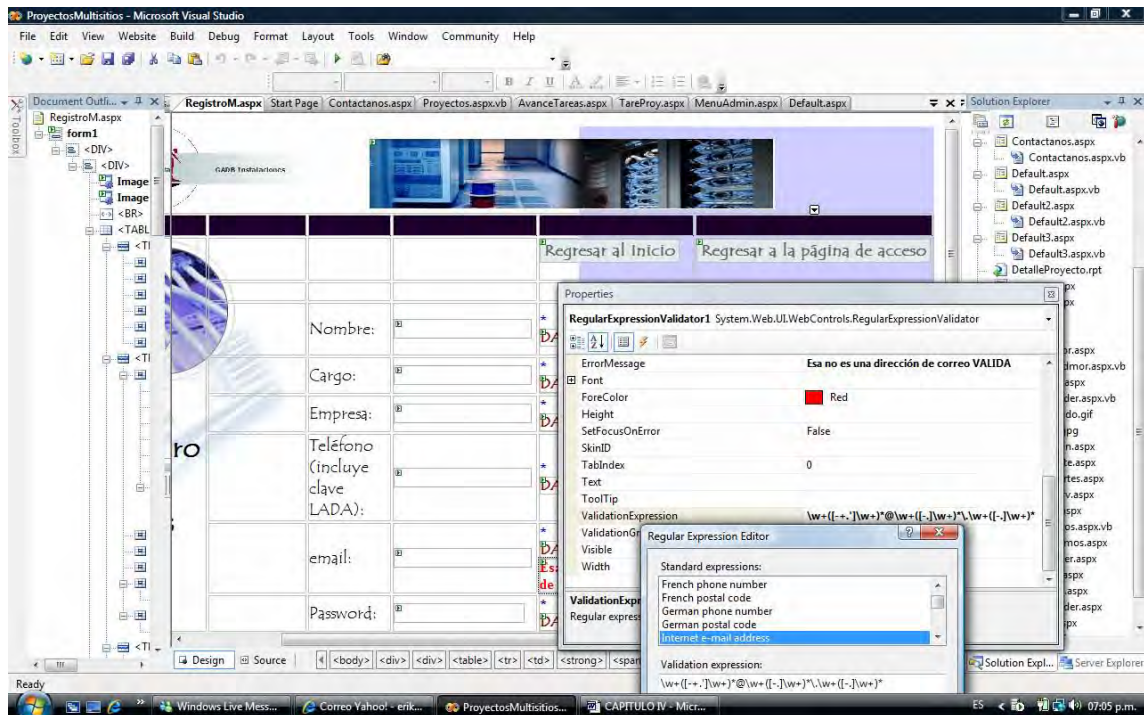


Fig. 4.12 Propiedades del RegularExpressionValidator.

Una vez completados todos los campos que se solicitan, la información se agrega a la base de datos al oprimir el botón **Registrar**, nuevamente a la tabla de Usuarios, he aquí el fragmento de código que realiza esta acción:

```
Protected Sub Registrar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Registrar.Click
    Dim Cn As New OleDbConnection("Provider=SQLOLEDB;userid=sa;
Password=;initial Catalog=ProyectoMultisitios;
DataSource=ERIKAPERSONAL")
    Dim Comando As OleDbCommand
    Cn.Open()
    Comando = New OleDbCommand("Insert into Usuarios(NomUsuario,
Cargo,Empresa,Telefono,email,password,TipoUsuario)
values ('" & Nom.Text & "','" & Carg.Text & "','" & Emp.Text & "','" & Tel.Text & "','" & email.Text & "','" & Pwd.Text & "','Cliente')", Cn)
    Comando.ExecuteNonQuery()
    Response.Redirect("Default.aspx")
End Sub
```

Antes de seguir avanzando con fragmentos de código en ASP se debe mencionar que al inicio de cualquier página que lleve algo de programación, se debe incluir la siguiente instrucción:

```
Imports System.data.OleDb
```

Esta se refiere al espacio de nombres, además es el proveedor de datos del .Net Framework, su función es describir una colección de clases que se utilizan para obtener acceso a un origen de datos OLE DB en el espacio administrado.⁸

Una vez que el Cliente registró sus datos tendrá dos opciones, regresar a la página de inicio figura 4.2 o ir a la página de acceso a la aplicación figura 4.9.

4.1.2.2 Página Opciones para el Cliente.

Si los datos de acceso ya son los correctos entonces el Cliente podrá acceder a la aplicación con un nuevo menú, el cual se muestra en la figura 4.13.

La primera opción de este menú es una consulta que el cliente puede hacer a la base de datos para obtener los datos generales de un Proyecto, para lo cual se utilizará la tabla Proyectos, todas la tablas que se mencionan fueron ya incluídas en el Capítulo III en la parte del diseño; junto con los campos y los tipos de datos que las conforman.

Este tipo de consulta se mostrará a través de un reporte realizado en Crystal Reports incluido dentro del Visual Studio 2005, en el capítulo anterior se describió con más detalle esta herramienta.

⁸ [http://msdn.microsoft.com/es-es/library/system.data.oledb\(VS.80\).aspx](http://msdn.microsoft.com/es-es/library/system.data.oledb(VS.80).aspx)

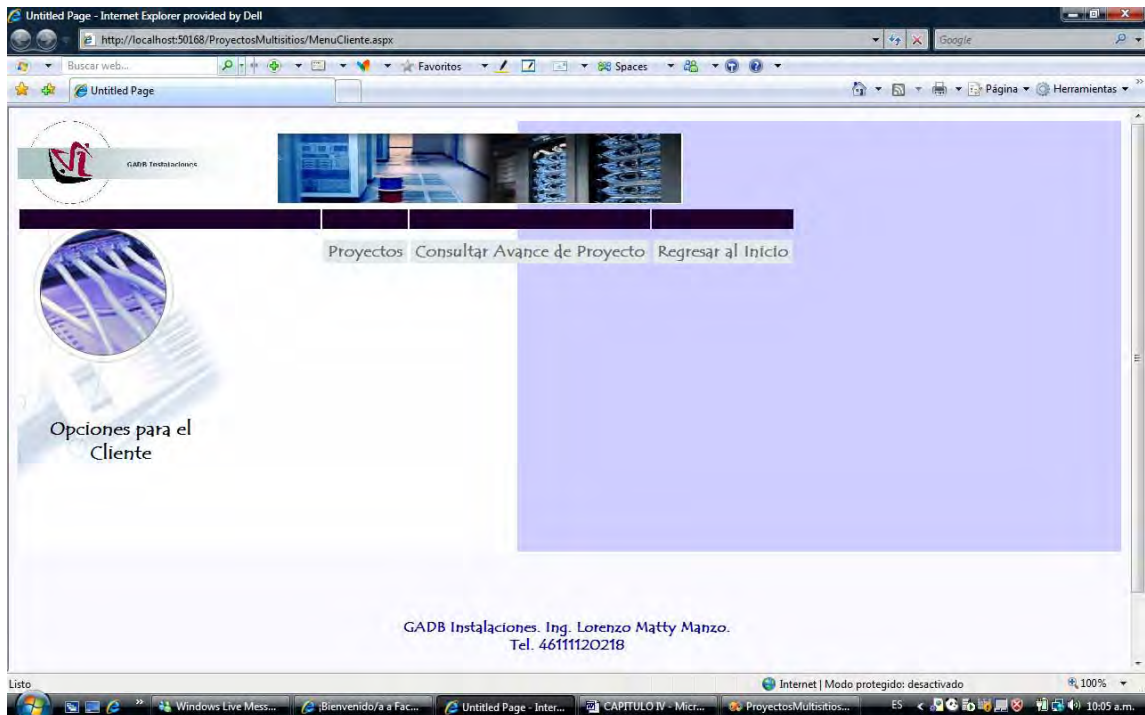


Fig. 4.13 Menú para el Cliente dentro de la Aplicación

Uno de los puntos por los que escogí hacer una consulta a través de un reporte de Crystal Reports y no una página de consulta típica, es porque el reporte da una mejor presentación a la información y tiene un formato definido, además el usuario puede guardarlo en su computadora como archivo o imprimirlo.

Dentro del menú Cliente existen dos submenús de consulta el de Proyecto y el de Avance de Proyecto.

A continuación se muestra en la figura 4.14 el reporte llamado **Resumen de Proyectos** realizado con Crystal Reports mediante el Crystal Reports Gallery, el cual es un wizard que te lleva paso a paso para la realización del reporte.

Para este reporte se tomó información de la tabla Proyectos como se mencionó anteriormente y el campo NomUsuario de la tabla Usuarios, al inicio del reporte

en la esquina superior derecha aparece el nombre del cliente. Se restringió el reporte para que aparecieran sólo los proyectos correspondientes al cliente que inició sesión. Para hacer esta restricción se utilizó el siguiente código dentro de la página .aspx.vb correspondiente a la que manda llamar el reporte:

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
    Me.CrystalReportViewer1.LogOnInfo.Item(0).ConnectionInfo.UserID = "sa"
    Me.CrystalReportViewer1.LogOnInfo.Item(0).ConnectionInfo.Password = ""
    Me.CrystalReportViewer1.SelectionFormula = "{Usuarios.idUsuario}="
    & Session("idUsuario")
End Sub
```

Las primeras dos líneas permiten el acceso del Crystal Report a la base de datos mediante el usuario y password de ésta y la tercera línea restringe que la información que se muestre en el reporte sea únicamente la del usuario que inició sesión.

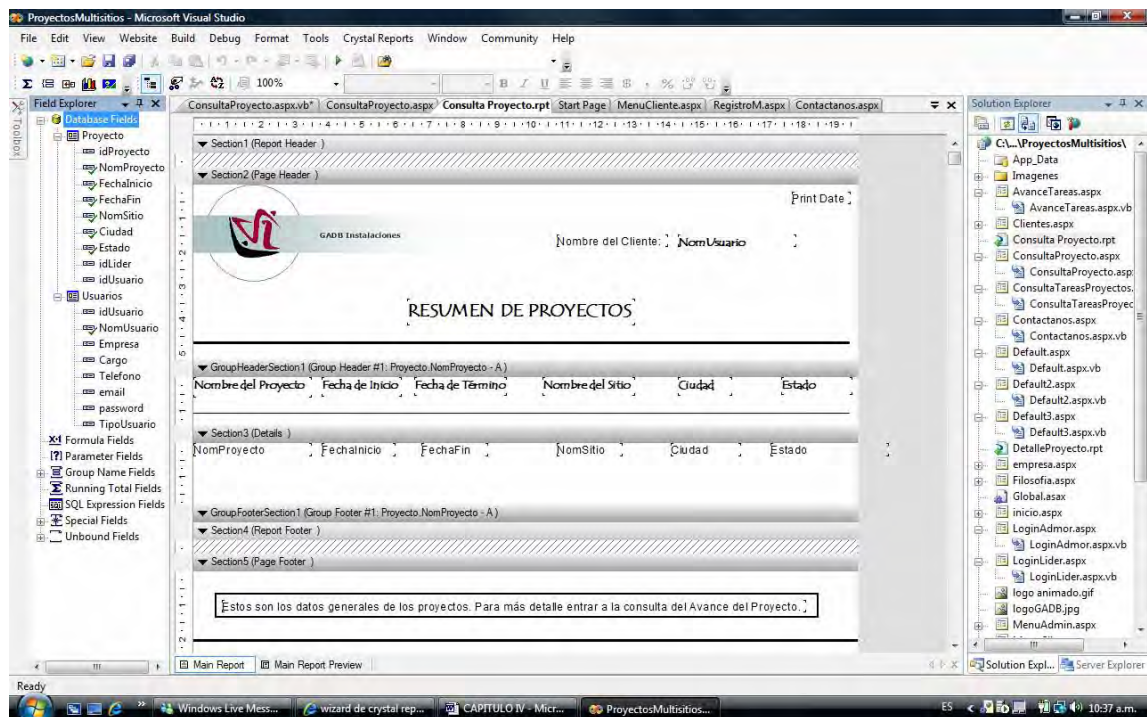


Fig. 4.14 Estructura del reporte RESUMEN DE PROYECTOS.

Una vez terminado el archivo .rpt, correspondiente al reporte en Crystal Reports se inserta en el archivo .aspx para poderlo visualizar en la aplicación, esto se hace a través de un control llamado **CrystalReportViewer** que se usa en .NET Framework para enlazar y mostrar informes, de tal forma que éstos sean visibles en la aplicación. Este control se muestra en la figura 4.15.

El reporte terminado se muestra en la figura 4.16

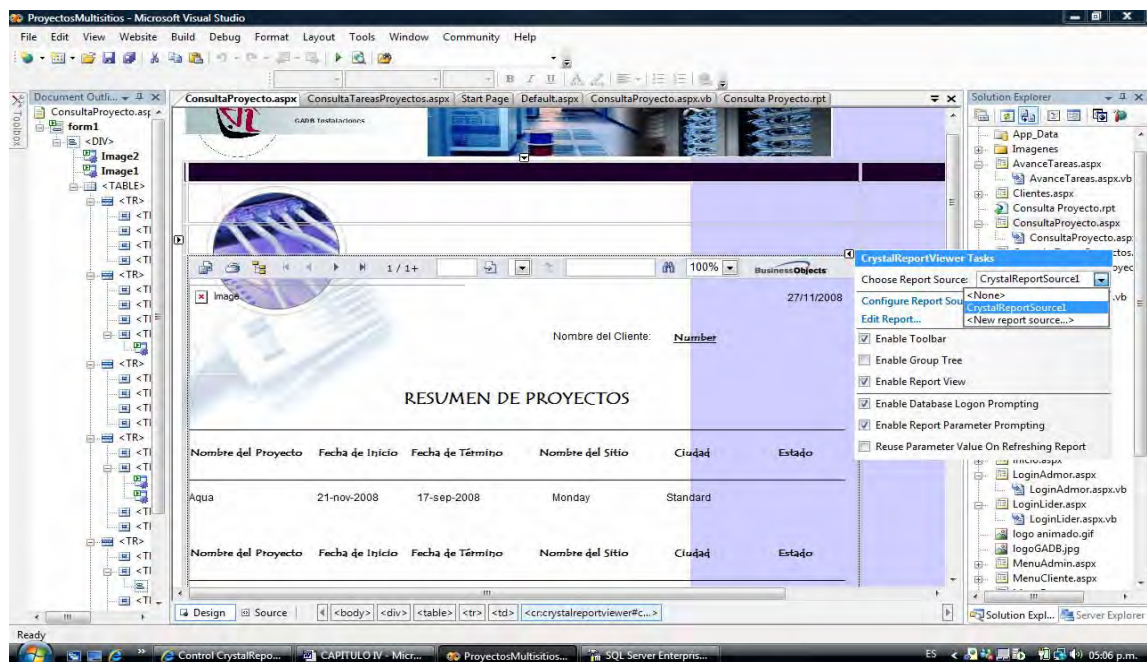


Fig. 4.15 Control CrystalReportViewer.

Dentro de la página con opciones para el cliente de la figura 4.13 aparece un segundo menú Consultar Avance de Proyecto, al acceder a este menú aparecerá una página en donde se podrá seleccionar un proyecto de un **dropdownlist**, éste contendrá únicamente los proyectos relacionados con el cliente que inició la sesión. Este tipo de control ya fue descrito en el capítulo anterior.

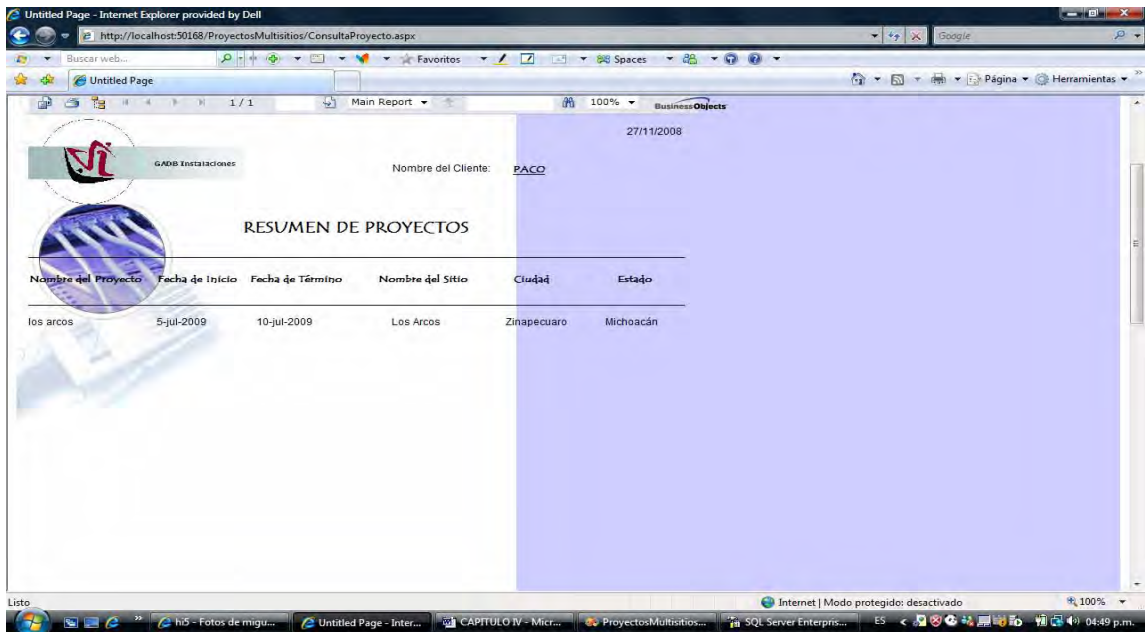


Fig. 4.16 Reporte Resumen de Proyectos.

En la figura 4.17 se muestra cómo se configuró el **DropDownList** para que aparecieran los proyectos, tomados de la tabla Proyecto.

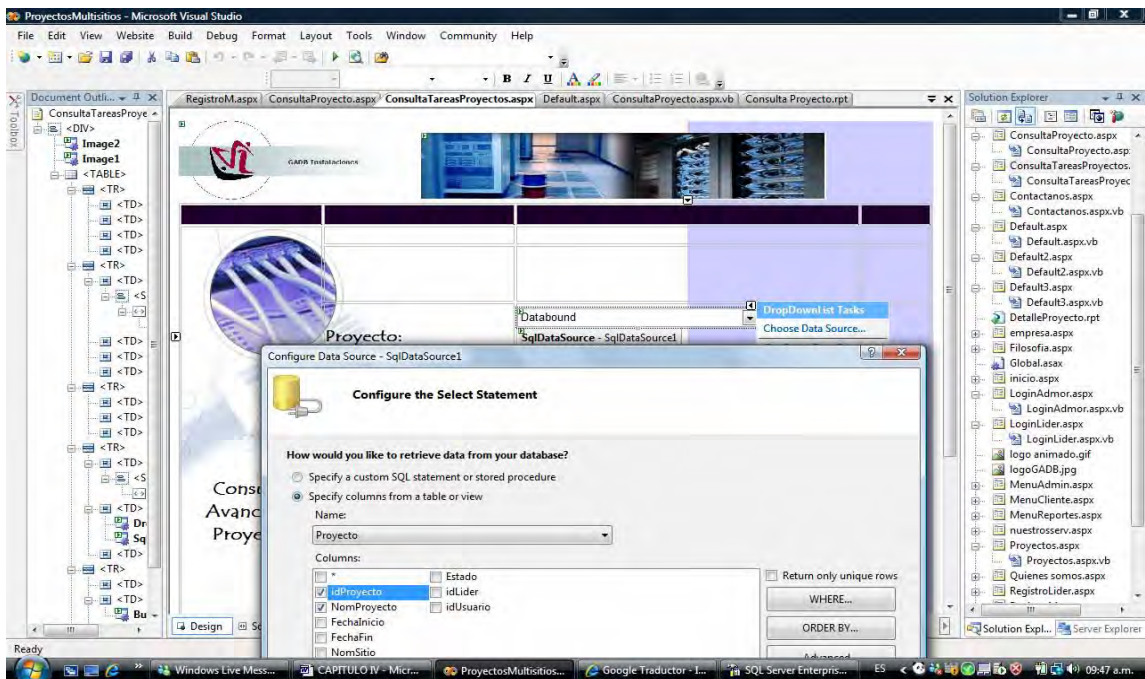


Fig. 4.17 Configurar DropDownList

Se seleccionaron dos campos de la tabla, el idProyecto y el NomProyecto como se puede observar, uno va a ser el campo visible en este caso el NomProyecto y el otro por el que se lleve el control.

En la figura se observa un botón WHERE es aquí en donde se pondrá la condición para que aparezcan únicamente los proyectos relacionados con el usuario que inició sesión. Esto se muestra en la figura 4.18.

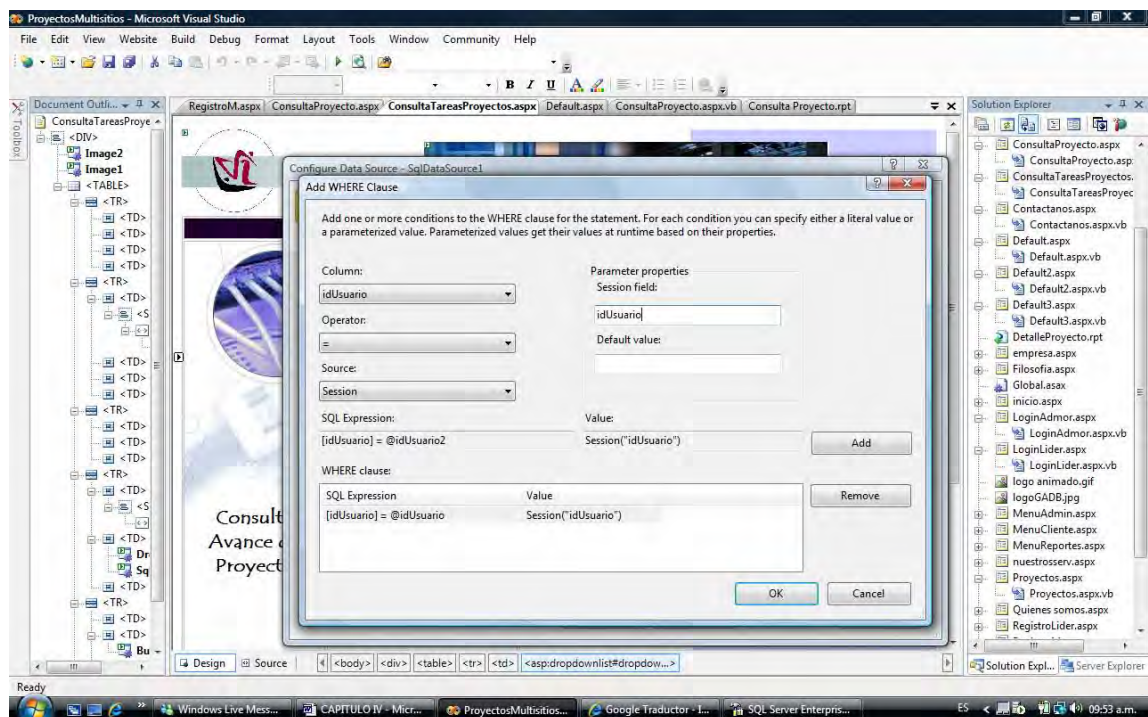


Fig. 4.18 Condición Where para DropDownList de la consulta

En la figura 4.19 se muestra la página de la forma en que el usuario la visualizará.

Al haber seleccionado el proyecto y oprimir el botón **Mostrar** se ejecutará el siguiente código del botón:

```

Protected Sub Button1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Response.Redirect("RepAvance.aspx?idp="
    & Me.DropDownList1.SelectedValue)
End Sub

```

Lo que hace esta instrucción es redireccionar a una página llamada RepAvance.aspx y ligar a una variable llamada idp el contenido del DropDownList.

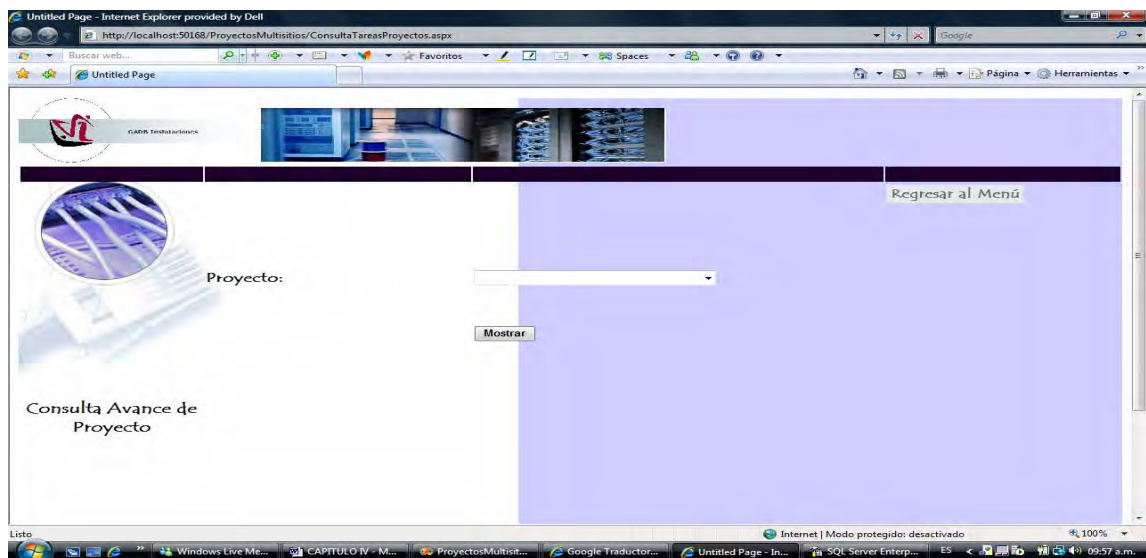


Fig. 4.19 Página Consulta Avance Proyecto.

Se creó un archivo .rpt que lleva como encabezado el nombre del proyecto seleccionado y en los detalles el contenido de las tareas relacionadas con ese proyecto, todos estos datos fueron tomados de la tabla TareaProyecto, a éste reporte se le anexa en la parte inferior, dos gráficas, la primera muestra las fechas de inicio y término proyectadas de cada tarea y la segunda las fechas inicio y término reales de cada tarea. Para las gráficas se utilizó el **Chart Expert** que tiene el Crystal Reports, y el tipo de gráfica es la de Gantt en donde muestra con una barra el inicio y fin de la tarea. En las figuras 4.20 y 4.21 se muestra el previo del reporte. El uso del Chart Expert se detalla en el capítulo III.

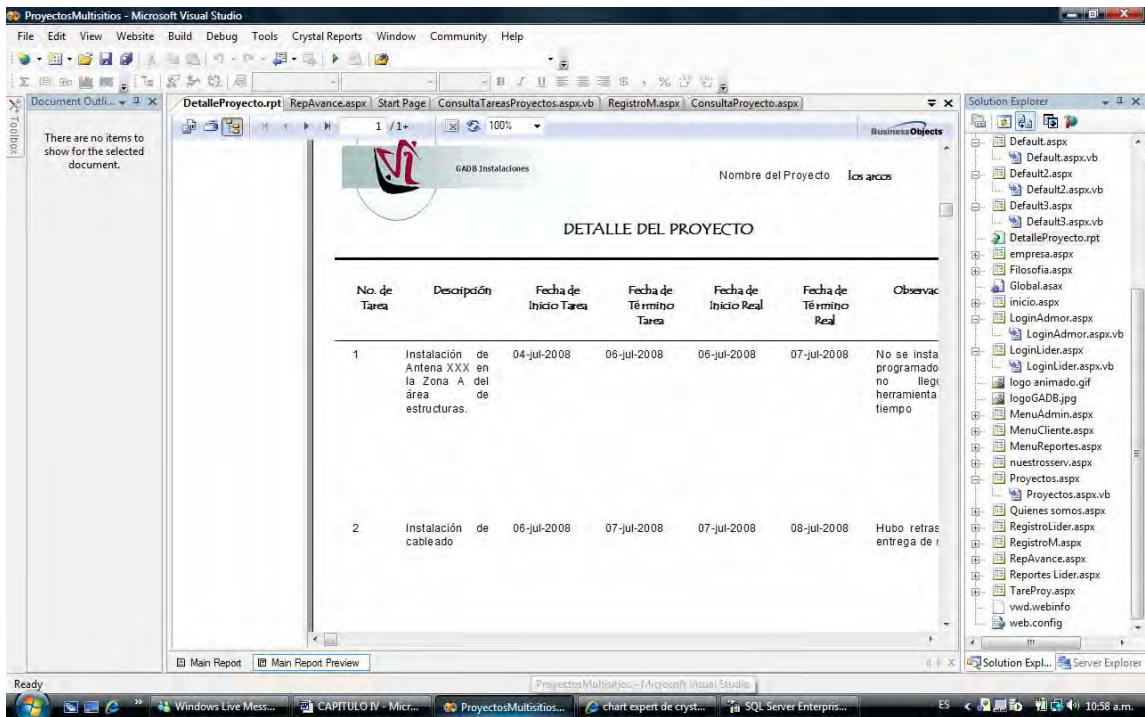


Fig. 4.20 Vista previa del reporte DETALLE DEL PROYECTO

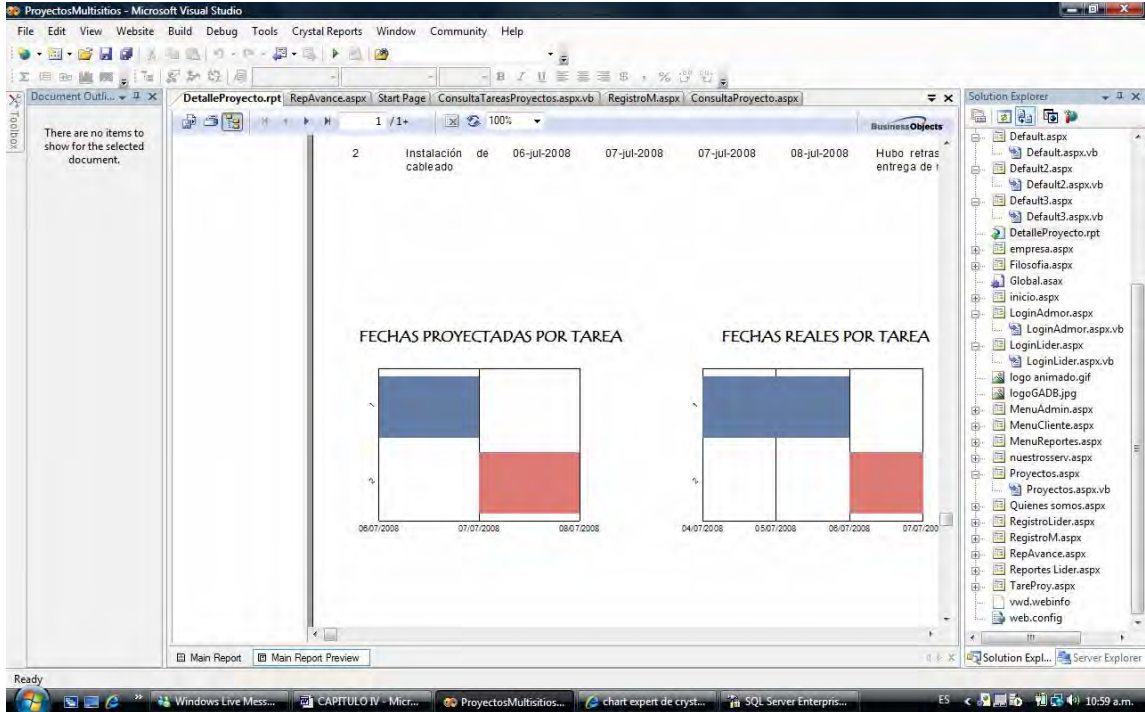


Fig. 4.21 Vista previa del reporte DETALLE DEL PROYECTO

Posteriormente se presenta el reporte en la aplicación dentro del archivo RepAvance.aspx a través del CrystalReportViewer y dentro del archivo RepAvance.aspx.vb llevará un código que servirá para acceder la base de datos y para mostrar únicamente las tareas del proyecto seleccionado en el DropDownList, a continuación se muestra:

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
    Me.CrystalReportViewer1.LogOnInfo.Item(0).ConnectionInfo.UserID = "sa"
    Me.CrystalReportViewer1.LogOnInfo.Item(0).ConnectionInfo.Password = ""
    Me.CrystalReportViewer1.SelectionFormula =
        "{TareaProyecto.idProyecto}=" & Request.QueryString("idp")
End Sub
```

4.1.2.3 Zona Administrativa.

De la misma forma que en la página de Zona Proyectos, al presionar el hiperlink de **Administración** que se encuentra en la página de acceso del cliente, aparecerá la página de acceso pero ahora es exclusiva de los Líderes de Proyectos y dentro de ésta un hiperlink para el acceso únicamente del Administrador de la Empresa, dentro de esta página se solicita el email y el password para poder acceder al menú administrativo del líder, en la figura 4.22 se muestra esta página de acceso.

Si alguno de los datos que se piden para entrar no son correctos o el usuario no existe, se envía un mensaje de error y se tiene que registrar pero esto únicamente lo puede hacer el Administrador. En el campo del email se utilizó el **RegularExpressionValidator**, para validar que los datos capturados correspondieran a una dirección de correo electrónico.

La parte de código en donde se busca el usuario en la base de datos es muy similar a la del cliente, sólo se tiene que destacar que a éste se le asigna una bandera diferente para identificarlo como líder y la tabla en la que se hace la

búsqueda se llama Lider, todo este código se ejecuta cuando se presiona el botón de **Aceptar**.

```
Protected Sub Aceptar_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Aceptar.Click
    Dim Cn As New OleDbConnection("Provider=SQLOLEDB;userid=sa; Password=;initial Catalog=ProyectoMultisitios; DataSource=ERIKAPERSONAL")
    Dim DR As OleDbDataReader
    Dim Comando As OleDbCommand
    Cn.Open()
    Comando = (New OleDbCommand("Select isNull(count(*),0), TipoUsuario from Lider where email='" & Usr.Text & "' And password='" & Pwd.Text & "' Group By TipoUsuario", Cn))
    DR = Comando.ExecuteReader
    If DR.Read Then
        If DR(0) = 1 Then
            If DR(1) = "Lider" Then
                Session("conectado") = 77 'Bandera
                Response.Redirect("MenuAdmin.aspx")
            End If
        End If
    Else
        Estado.Text = "Datos Incorrectos"
    End If
End Sub
```

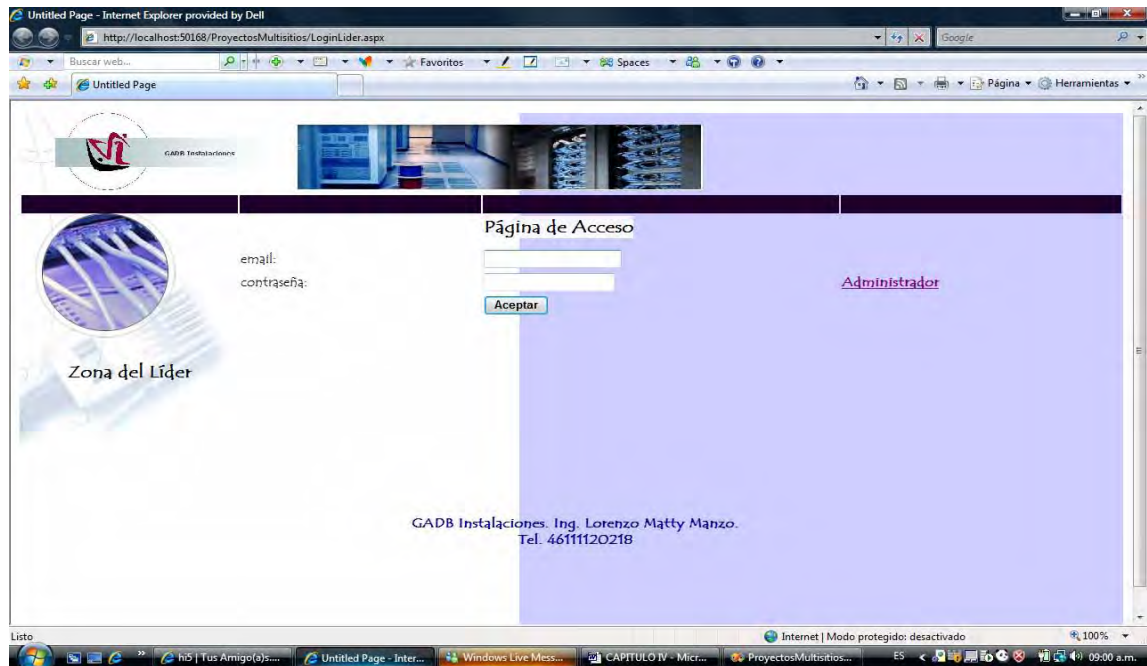


Fig. 4.22 Página de acceso a la Zona del Líder

4.1.2.4 Página Opciones Administrativas para el Líder.

Después de que el líder teclea sus datos correctos accesa al menú administrativo que le corresponde en la figura 4.23 se presenta dicho menú.

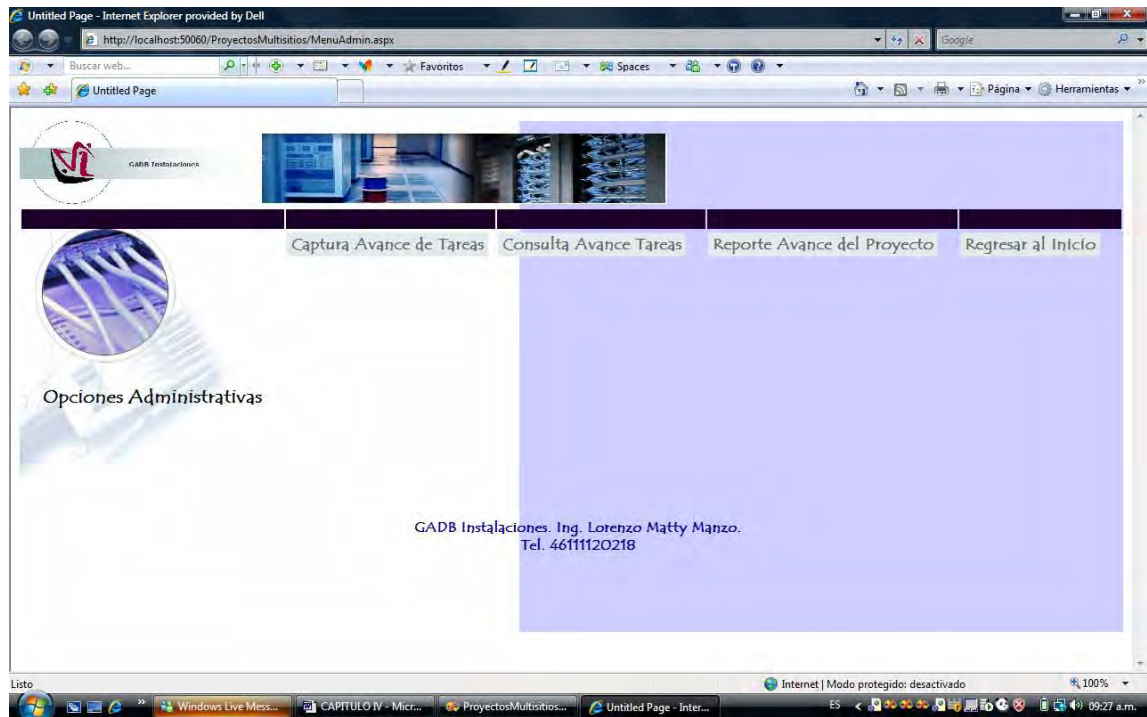


Fig. 4.23 Página Opciones Administrativas para el Líder de Proyectos.

Página Captura Avance de Tareas.

Una de las opciones del menú administrativo para el Líder de Proyectos es la captura de las distintas tareas que conforman un proyecto, esto se realiza en esta página. La captura se hace en base a los proyectos dados de alta con anterioridad en el menú de Captura de Proyectos que realiza el Administrador y sólo para los proyectos que le corresponden a ese líder si no fuera así el proyecto no aparecería en el DropDownList y no se podrían capturar sus tareas. Los campos en donde se capturan fechas, están validados con un

RegularExpressionValidator para que se respeten los formatos de las fechas que serían:

dd/mm/aaaa

En la figura 4.24 se muestra la Página de Captura Avance de Tareas.

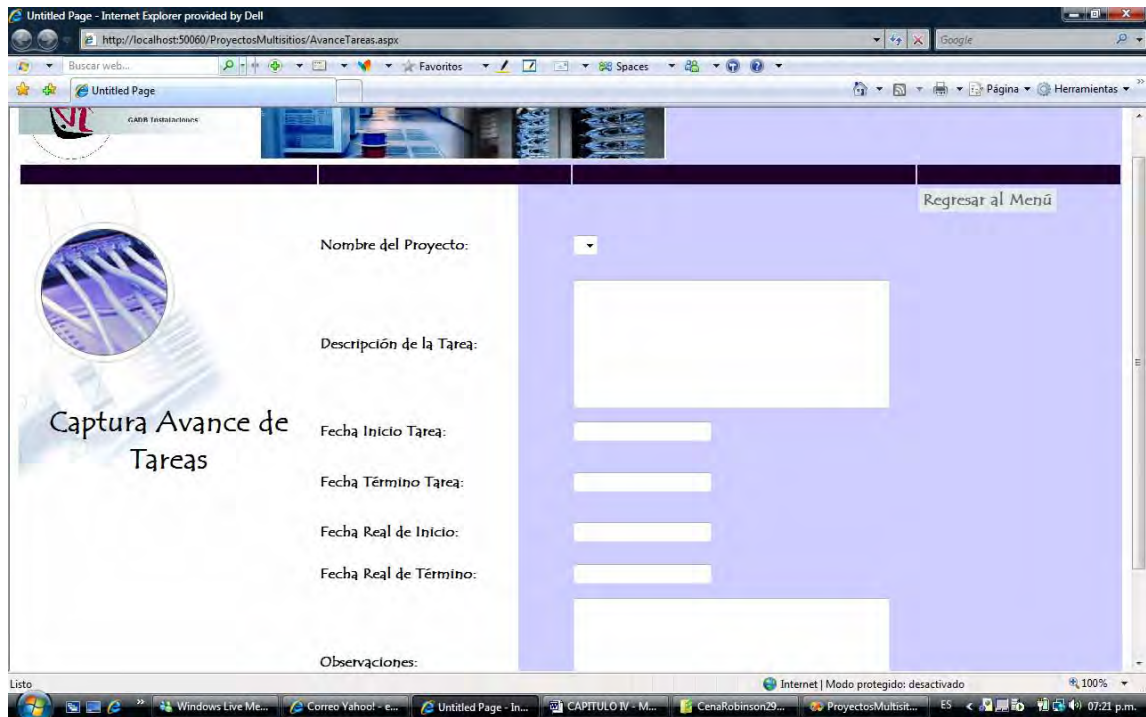


Fig. 4.24 Página de Captura de Avance de Tareas de los Proyectos existentes

A continuación se presenta el código de captura de los datos correspondientes a los proyectos, los cuales se almacenan en la tabla TareaProyecto al momento de pulsar el botón **Guardar**:

```
Protected Sub Guardar_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Guardar.Click
    Dim Cn As New OleDbConnection("Provider=SQLOLEDB;userid=sa;Password=;initial Catalog=ProyectoMultisitios;DataSource=ERIKAPERSONAL")
    Dim Comando As OleDbCommand
    Cn.Open()
    Comando = New OleDbCommand("Insert into TareaProyecto(idProyecto, Descripcion, FechaInicioProy, FechaFinProy, FechaInicioReal, FechaFinReal, Observaciones) values (" & idProyecto.SelectedValue & ",
```

```

' " & Desc.Text & " ', ' " & Fini.Text & " ', ' " & Ffin.Text & " ',
' " & FRini.Text & " ', ' " & FRfin.Text & " ', ' " & Obs.Text & " ' )", Cn)
Comando.ExecuteNonQuery()
Desc.Text = " "
Fini.Text = " "
Ffin.Text = " "
FRini.Text = " "
FRfin.Text = " "
Obs.Text = " "
Desc.Focus()
End Sub

```

Al finalizar la captura o si se decide no hacerla se puede regresar al menú.

Página Consulta Avance de Tareas.

En esta página el líder podrá revisar por proyecto el avance a través de un **GridView** que despliega todas las tareas capturadas para ese proyecto, esto será posible siempre y cuando el líder tenga algún proyecto asignado a él, ya que al inicio de la página aparecerá un **DropDownList** en donde se listarán sólo los proyectos que le corresponden. Escogerá uno de ellos del **DropDownList** y las tareas asignadas a ese proyecto se desplegarán en el **GridView**. La función del control **GridView** ya fue descrita en el capítulo anterior.

Se utilizó este tipo de control porque una vez capturadas las tareas del proyecto se pueden editar y borrar directamente en la consulta.

En la figura 4.25 se muestra la página con un ejemplo de las tareas correspondientes a un proyecto.

Página Reporte Avance del Proyecto.

De igual forma que en el menú del Cliente se reportan los avances de las tareas que conforman un proyecto, aquí se realiza algo similar, con la diferencia que

en el módulo de Clientes se podía elegir en un **DropDownList** los proyectos en los que el cliente esta involucrado y en este caso se muestran los proyectos en los que el líder que inicia la sesión está llevando a cabo, de igual forma se selecciona sólo uno del **DropDownList** y las tareas correspondientes a ese proyecto son las que se muestran en el reporte junto con las gráficas de Gantt como se muestra en la figura 4.26.

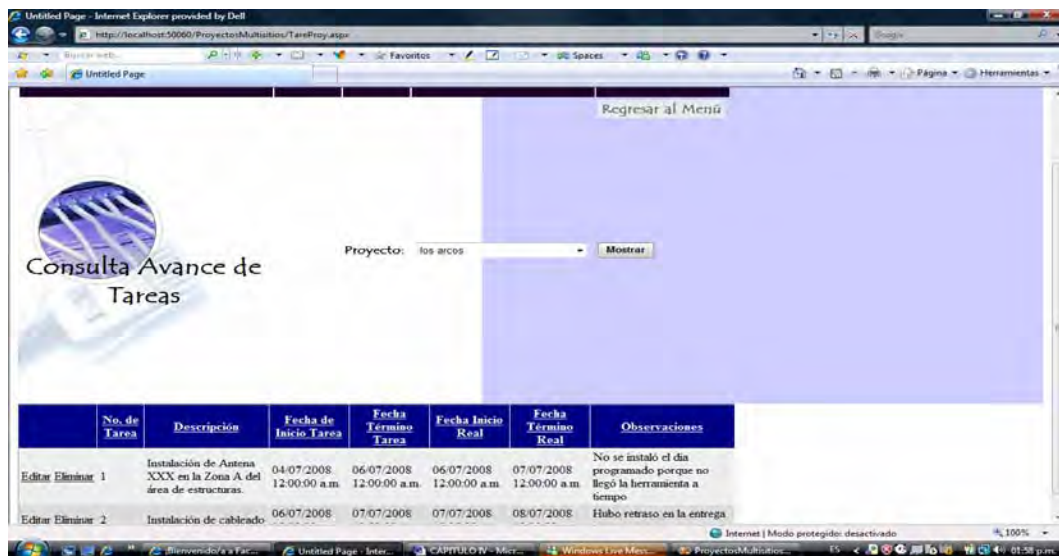


Fig. 4.25 Página Consulta Avance de Tareas.

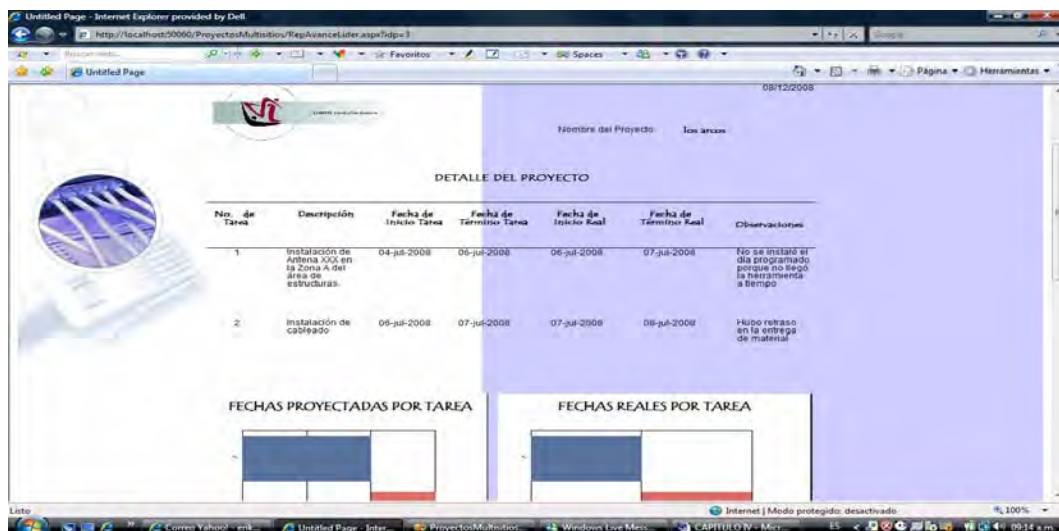


Fig. 4.26 Reporte de Detalle de Proyecto para módulo del Líder.

La última opción del menú del líder es la de **Regresar al inicio** que envía a la página de Login del Líder y de ésta puede regresar al Inicio de la aplicación.

Dentro de la página de inicio de sesión del Líder, mostrada en la figura 4.27, hay un hiperlink (**Administrador**) que lleva a la página del Login del Administrador, este usuario y password son únicos porque sólo existe un administrador en la compañía y fue dado de alta directamente en la tabla de Usuarios en el SQL Server. Si cualquiera de los dos campos son llenados incorrectamente marca un mensaje de error, pero si estos son introducidos correctamente y al momento de pulsar el botón de **Aceptar** se ejecuta el siguiente código:

```
Protected Sub Aceptar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Aceptar.Click
    Dim Cn As New OleDbConnection("Provider=SQLOLEDB;userid=sa;
    Password=;initial Catalog=ProyectoMultisitios;
    DataSource=ERIKAPERSONAL")
    Dim DR As OleDbDataReader
    Dim Comando As OleDbCommand
    Cn.Open()
    Comando = (New OleDbCommand("Select isNull(count(*),0),
    TipoUsuario,idUsuario from Usuarios where email='" & Usr.Text &
    "'And password='" & Pwd.Text & "' Group By TipoUsuario,idUsuario", Cn))
    DR = Comando.ExecuteReader
    If DR.Read Then
        If DR(0) = 1 Then
            If DR(1) = "Administrador" Then
                Session("conectado") = 100 'Bandera
                Session("idUsuario") = DR(2)
                Response.Redirect("MenuAdmor.aspx")
            End If
        End If
    Else
        Estado.Text = "Datos Incorrectos"
    End If
End Sub
```

Como se ve en el código una vez validados los datos se accesa al archivo MenuAdmor.aspx el cual contiene el Menú del Administrador a continuación se muestra éste en la figura 4.28.

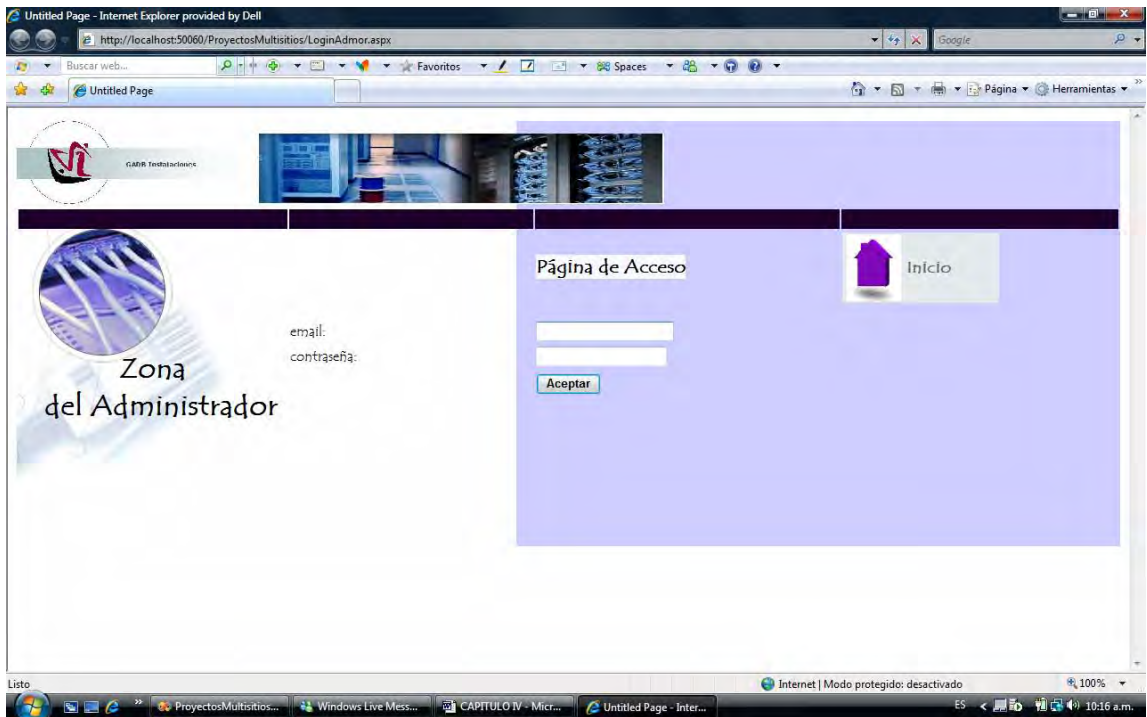


Fig. 4.27 Página de inicio de sesión para el Administrador.

4.1.2.5 Página Opciones para el Administrador.

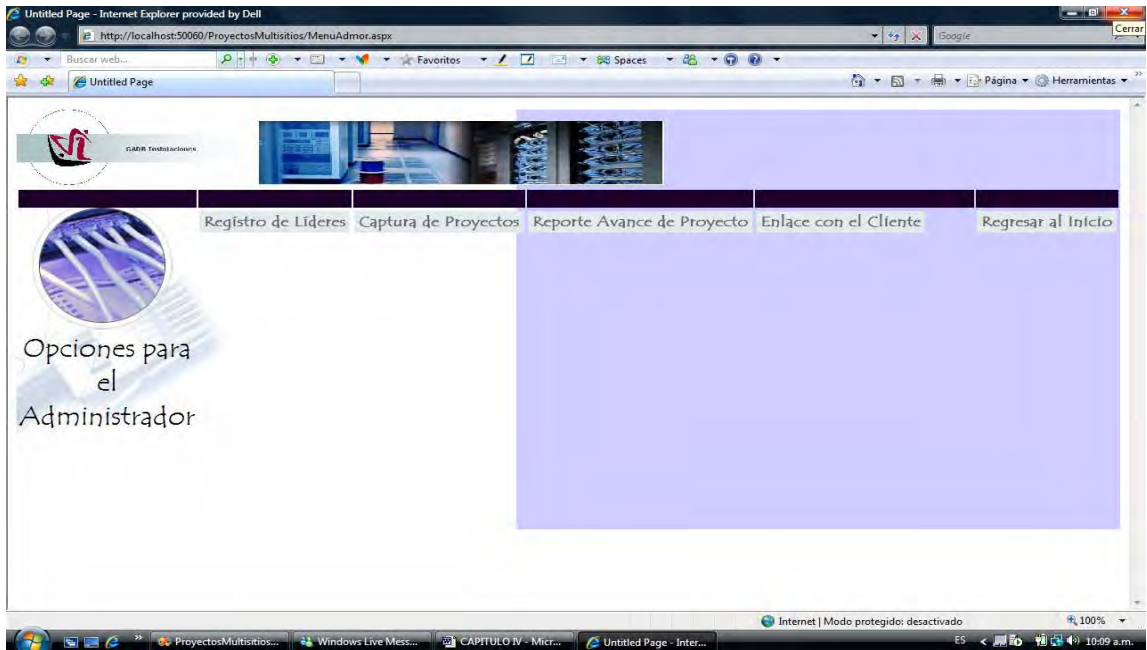


Fig. 4.28 Menú para el Administrador.

Página Registro de Líderes.

Como se mencionó anteriormente el administrador es el único indicado para dar de alta a los usuarios Líderes de Proyectos, si estos no tienen un registro para entrar a la aplicación deberán pedir al administrador que llene el formulario correspondiente para su alta. En la figura 4.29 se muestra este formulario en el cual todos los datos se tienen que llenar de forma obligatoria, para ello se utilizó el control **RequiredFieldValidator** el cual impide que el campo se deje en blanco y en el caso del campo que contiene el email se usó el control **RegularExpressionValidator** , el cual valida que la información que se está ingresando corresponda al formato de una dirección de correo electrónico.

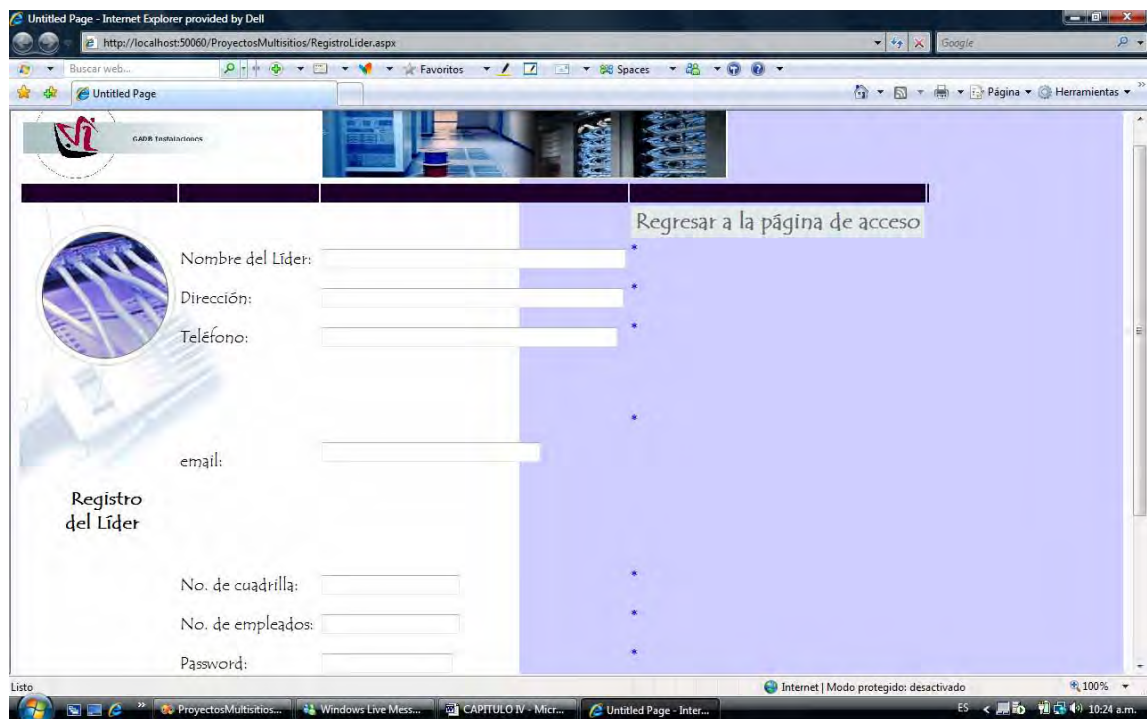


Fig. 4.29 Página Registro del Líder.

En esta página a diferencia de la de Registro del Cliente en la parte inferior se muestra un **GridView** con los datos de los líderes que se van dando de alta y

tiene la opción de editar o eliminar algún campo o el registro completo. Esto se puede ver en la figura 4.30. A continuación se muestra el código que hace que los datos se agreguen a la tabla de Lider en el momento que se pulsa el botón de **Registrar**:

```
Protected Sub Guardar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Guardar.Click
    Dim Cn As New OleDbConnection("Provider=SQLOLEDB;userid=sa;
    Password=;initial Catalog=ProyectoMultisitios;
    DataSource=ERIKAPERSONAL")
    Dim Comando As OleDbCommand
    Cn.Open()
    Comando = New OleDbCommand("Insert into Lider(NomLider,Direccion,
    Telefono,email,Nocudadrilla,Noempleados,password,TipoUsuario)
    values('" & NomLid.Text & "','" & DirLid.Text & "','" &
    TelLid.Text & "','" & email.Text & "','" & NCuad.Text & "','" &
    Nemp.Text & "','" & pwd.Text & "','Lider')", Cn)
    Comando.ExecuteNonQuery()
    NomLid.Text = ""
    DirLid.Text = ""
    TelLid.Text = ""
    email.Text = ""
    NCuad.Text = ""
    Nemp.Text = ""
    pwd.Text = ""
    GridView1.DataBind()
    NomLid.Focus()
End Sub
```

En la última parte del código todos los campos se inicializan en blanco para que al momento de registrar los datos se limpien las cajas de texto.

Página Captura de Proyectos.

En esta página se capturarán los datos principales del proyecto como: Nombre del Proyecto, Fecha de Inicio, Fecha de Término, Nombre del Sitio, Ciudad, Estado y a través de un DropDownList se elige el Cliente y el Líder de Proyecto que previamente ya fueron dados de alta en la base de datos.

El responsable de esta alta es el administrador de la compañía, ya que es él, el encargado de la distribución de los recursos materiales y humanos, y de la

asignación de los proyectos. En la figura 4.31 se muestra ésta página y al final también un **GridView** con los proyectos que se tienen en la base de datos, dando la opción de eliminar o editar algún campo o el registro completo.

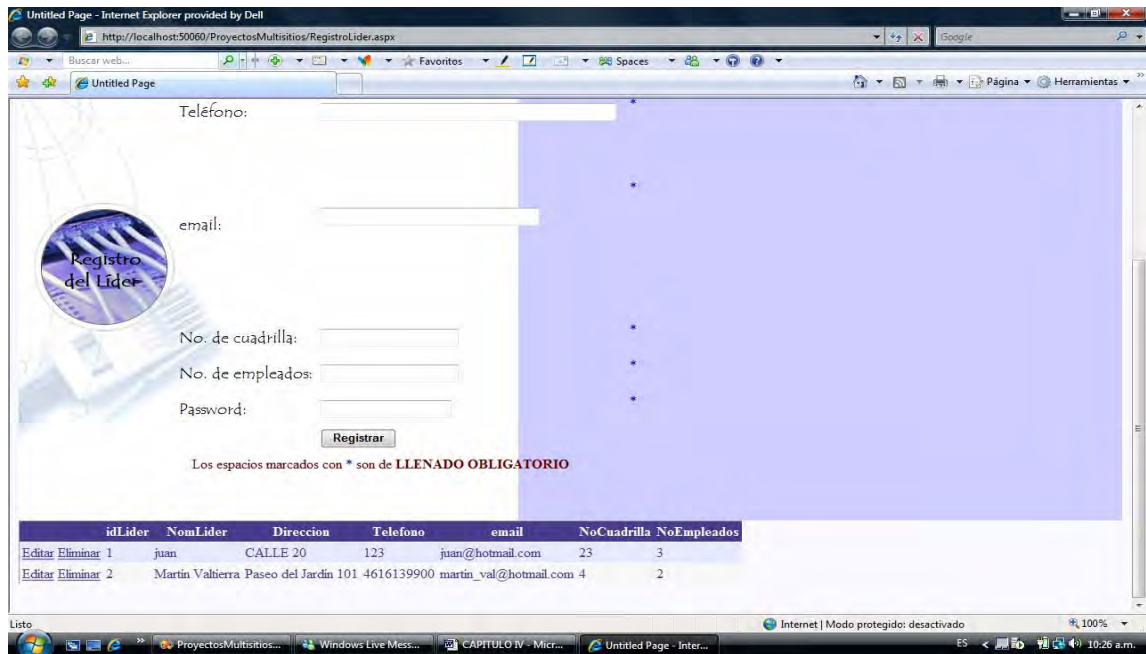


Fig. 4.30 GridView de la Página de Registro de Líderes.

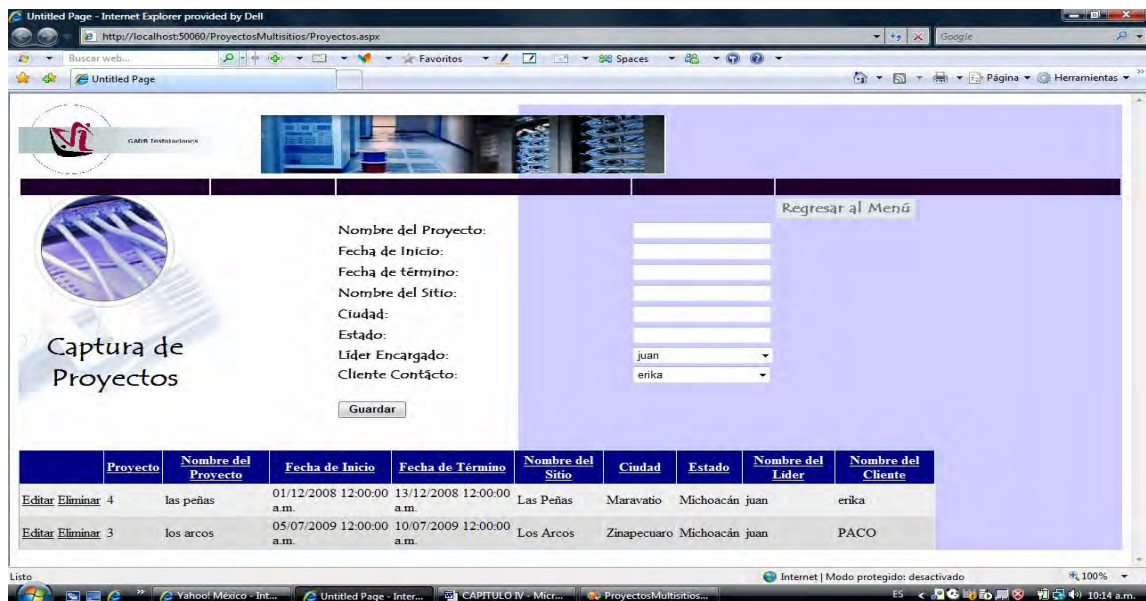


Fig. 4.31 Página de Captura de Proyectos

Cabe señalar que en los campos en donde se captura la Fecha de Inicio y Fecha de Término se utilizó el **RegularExpressionValidator** para limitar el formato de fecha a : dd/mm/aaaa.

A continuación se muestra el código que hace que los datos capturados en la página se graben en la base de datos, en este caso se usó la tabla Proyecto y los campos del líder y el cliente de los DropDownList los toma de las tablas Lider y Usuarios respectivamente, para lo cual se utilizan llaves en tablas relacionales. Este código se ejecuta al momento de pulsar el botón de **Guardar**.

```
Protected Sub Guardar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Guardar.Click
    Dim Cn As New OleDbConnection("Provider=SQLOLEDB;userid=sa;
    Password=;initial Catalog=ProyectoMultisitios;
    DataSource=ERIKAPERSONAL")
    Dim Comando As OleDbCommand
    Cn.Open()
    Comando = New OleDbCommand("Insert into Proyecto(NomProyecto,
    FechaInicio,FechaFin,NomSitio,Ciudad,Estado,idLider,idUsuario)
    values ('" & NomProyecto.Text & "',''" & FechaInicio.Text & "','
    '" & FechaFin.Text & "',''" & NomSitio.Text & "','
    '" & Ciudad.Text & "',''" & Estado.Text & "','
    '" & idLider.SelectedValue & "',''" & idUsuario.SelectedValue & "')", Cn)
    Comando.ExecuteNonQuery()
    NomProyecto.Text = ""
    FechaInicio.Text = ""
    FechaFin.Text = ""
    NomSitio.Text = ""
    Ciudad.Text = ""
    Estado.Text = ""
    GridView1.DataBind()
    NomProyecto.Focus()
End Sub
```

Página Reporte Avance de Proyecto.

De igual forma que con el Cliente y el Líder de Proyectos, el Administrador podrá generar un reporte en donde se muestren de manera detallada las tareas de un proyecto, éste se elegirá en un DropDownList, en el siguiente código se

muestra la asignación del proyecto elegido a una variable idp y el redireccionamiento a un archivo .aspx, todo ésto al pulsar el botón de **Mostrar**:

```
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Response.Redirect("RepAvanceAdmor.aspx?idp="
    & Me.DropDownList1.SelectedValue)
End Sub
```

A continuación se muestra en las figuras 4.32 y 4.33 el reporte generado por detalle para un proyecto seleccionado, la diferencia con los reportes anteriores es que en la esquina superior derecha muestra además del Nombre del Proyecto, el Nombre del Líder y el Nombre del Cliente Contacto, en el cuerpo del reporte se muestra el detalle de las tareas y la representación gráfica de éstas (mediante Gráficas de Gantt).

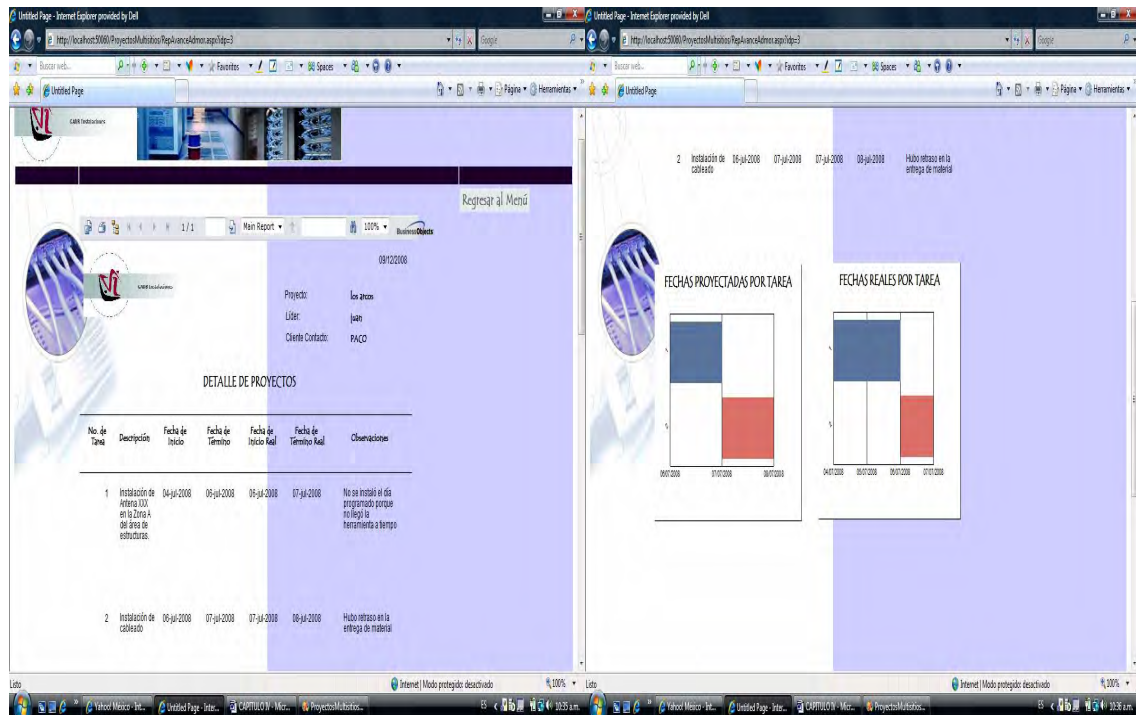


Fig. 4.32 y 4.33 Reporte Detalle de Proyectos para el Administrador

Página Enlace con el Cliente.

Esta página es derivada de la página de Contáctanos y es el enlace que tiene el administrador con sus clientes actuales y potenciales, la información que los clientes capturaron se guardó en la tabla Contacto de la base de datos y es de ahí de donde se toma la información, ésta se representará a través de un reporte generado a partir de un rango de fechas, para escoger las fechas se utilizó el control **Calendario** de ASP.Net localizado en la barra de herramientas Standard, este control muestra un calendario de un mes y permite seleccionar el mes siguiente o el anterior, por defecto muestra el mes actual. En la figura 4.34 se muestra el uso de este control.

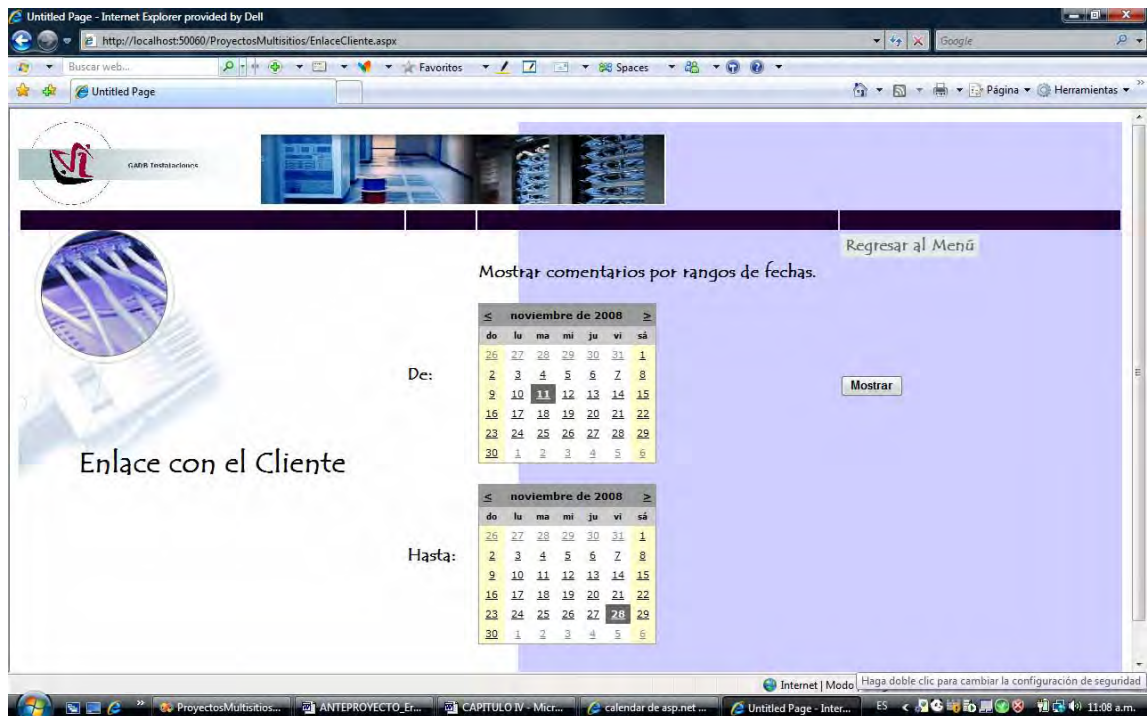


Fig. 4.34 Página Enlace con el Cliente

Una vez seleccionado el rango de fechas (De-Hasta) se pulsa el botón de **Mostrar** generándose el siguiente código:

```

Protected Sub Mostrar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Mostrar.Click
    Response.Redirect("RepContacto.aspx?fec1="
    & Mid(Calendar1.SelectedDate.ToString(), 1, 10) & "&fec2="
    & Mid(Calendar2.SelectedDate.ToString(), 1, 10))
End Sub

```

Lo que hace es redireccionar a la página RepContacto.aspx y asignar a dos variables fec1 y fec2 lo seleccionado en el calendario correspondiente, extrayendo del formato Date (Mid), los 10 primeros caracteres, empezando del 1, ésto se hizo para que sólo extrajera la fecha y no la hora.

En el archivo RepContacto.aspx.vb se pondrá el siguiente código:

```

Protected Sub Page_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
    fec1 = Request.QueryString("fec1")
    fec2 = Request.QueryString("fec2")
    Me.CrystalReportViewer1.LogOnInfo.Item(0).ConnectionInfo.
    UserID = "sa"
    Me.CrystalReportViewer1.LogOnInfo.Item(0).ConnectionInfo.
    Password = ""
    Me.CrystalReportViewer1.SelectionFormula = "{Contacto.fecha} >=
    Date (" & Mid(fec1, 7, 4) & "," & Mid(fec1, 4, 2) & "," &
    Mid(fec1, 1, 2) & ") and {Contacto.fecha} <= Date (" & Mid(fec2, 7, 4)
    & "," & Mid(fec2, 4, 2) & "," & Mid(fec2, 1, 2) & ")"
End Sub

```

Las primeras dos instrucciones hacen uso de la clase **Request.QueryString** cuya función es pasar datos entre páginas en ASP.Net asignando a las variables fec1 y fec2 el contenido de de las selecciones hechas en los calendarios en la página anterior. Posteriormente se abre la base de datos y en la última instrucción se selecciona de la tabla Contacto la información contenida entre ese rango de fechas, (campo fecha de la tabla >= fec1 y <=fec2), aquí se aplica el formato de fórmula para manejo de fechas que utiliza Crystal Reports, porque si no el reporte no se generaría. De igual forma se utiliza el Mid para extraer de una cadena sólo lo que le indicamos y en la posición que corresponde, en este caso: aaaa, mm, dd, Crystal lo maneja en ese orden.

El reporte finalizado se muestra en la figura 4.35.

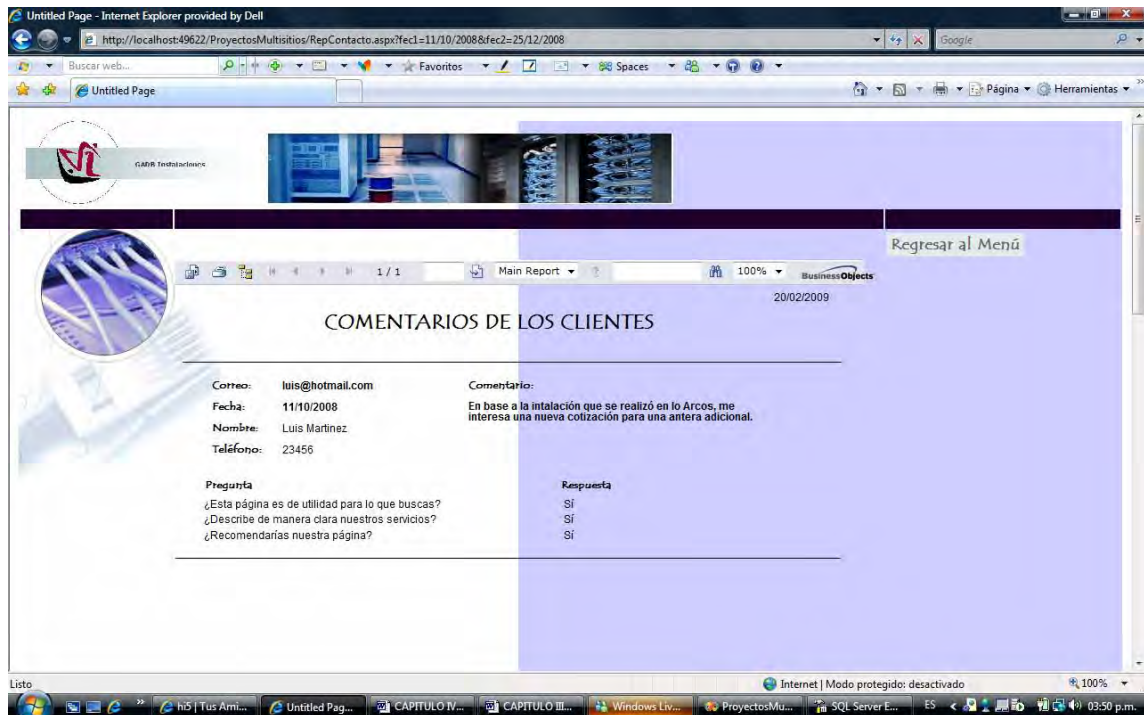


Fig. 4.35 Reporte Comentarios de los Clientes

4.2 Seguridad de aplicaciones Web con ASP.NET.

La seguridad de las aplicaciones Web es un tema crítico y complejo. Un sistema seguro requiere una cuidadosa planificación y Microsoft ASP.NET está sincronizado con el .NET Framework e Internet Information Server (IIS) para proporcionar seguridad a las aplicaciones Web.

Es importante tomar en cuenta las posibles amenazas presentes cuando tenemos en producción un sitio Web. Estas amenazas van desde el ingreso no autorizado al sitio, así como la manipulación indebida de datos de alta confidencialidad. Para ésto existe una solución consistente en aplicar membresías, roles, almacenamiento encriptado de datos, administración de

usuarios, manejo de procedimientos almacenados para realizar consultas parametrizadas. Manipulación de archivos de configuración, manejo de clases utilizando orientación a objetos, desarrollo de la aplicación por capas entre otros.

Dentro del ASP.Net existe una herramienta de administración de sitios Web en donde se pueden configurar y editar usuarios, funciones y permisos de acceso para el sitio, utilizando para ello la autenticación de Windows.

En el caso particular de la aplicación que se desarrolló, se asignaron tres tipos de usuarios: los clientes, los líderes de proyectos y el administrador, cada uno con permisos diferentes y menús diferentes, cada usuario tiene su página de logeo independiente y acceden a una tabla de la base de datos en particular. De todas formas en la parte de programación a cada usuario al momento de ingresar se le asigna una bandera o etiqueta que lo distingue como cliente, líder o administrador según sea el caso y se le restringe la entrada a cualquier página que no corresponda a su nivel de usuario. A continuación se presenta el ejemplo del usuario cliente, el primer fragmento muestra la bandera que se asigna al momento que un cliente ingresa:

```
Protected Sub Aceptar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Aceptar.Click
    Dim Cn As New OleDbConnection("Provider=SQLOLEDB;user id=sa;
Password=;initial Catalog=ProyectoMultisitios;
Data Source=ERIKAPERSONAL")
    Dim DR As OleDbDataReader
    Dim Comando As OleDbCommand
    Cn.Open()
    Comando = (New OleDbCommand("Select isNull(count(*),0),
TipoUsuario,idUsuario from Usuarios where email='" & Usr.Text &
"'And password='" & Pwd.Text & "' Group By TipoUsuario,idUsuario",Cn))
    DR = Comando.ExecuteReader
    If DR.Read Then
        If DR(0) = 1 Then
            If DR(1) = "Cliente" Then
                Session("conectado") = 52 'Bandera
                Session("idUsuario") = DR(2)
                Response.Redirect("MenuCliente.aspx")
            End If
        End If
    End If
```

```
Else
    Estado.Text = "Datos Incorrectos. Regístrate"
End If
End Sub
```

A este usuario se le asigna la bandera 52 al momento de conectarse con los datos correctos en el inicio de sesión. Al líder se le asigna la bandera 77 y al administrador el 100, los números correspondientes a las banderas fueron seleccionados aleatoriamente; en el código siguiente se muestra la condición que tiene que cumplir el usuario para dar de alta un proyecto, tomando en cuenta que esto sólo lo pueden hacer los líderes de proyecto:

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
    If Session("conectado") <> 77 Then Response.Redirect("Default.aspx")
End Sub
```

Para cargar esta instrucción se tiene que hacer doble click en el cuerpo de la página y automáticamente se abrirá la subrutina **Page_Load**, si el usuario tiene una bandera diferente a la 77 (que es la que corresponde al líder) se enviará de forma automática a la página de inicio de sesión en este caso Default.aspx, y de ahí no podrá seguir avanzando, con esta instrucción se protegerán todas la páginas para que los usuarios que no tienen el permiso no puedan acceder.

4.3 Mantenimiento de la aplicación.

Casi en todas las aplicaciones y las de tipo Web no son la excepción, una vez que se tienen en producción tarde o temprano se les tiene que hacer algún tipo de mantenimiento. Ello implica muchas veces retoques sencillos que no interfieren en el trabajo normal de éstas, pero en ocasiones hay que hacer cambios significativos, como el cambio de la estructura de la base de datos o el

montaje de un módulo nuevo. En estos casos es conveniente detener la aplicación.

Por suerte ASP.Net ofrece un atajo sencillo para todo esto. Se trata de crear un archivo llamado **App_Offline.htm** y colocarlo en la raíz de la aplicación.

Siempre que ASP.Net detecte la existencia de este archivo en la raíz, cerrará la aplicación, descargará el dominio de aplicación de la memoria del servidor y desbloqueará las bases de datos adjuntadas dinámicamente. Además, cualquier petición que llegue a la aplicación, sea de usuarios nuevos o de gente que ya la estuviera usando, obtendrá como respuesta el contenido de ese archivo que normalmente contendrá un mensaje de "**Página en mantenimiento**" o el que se quiera poner.

Se trata de un modo sencillo y conveniente de poner la aplicación en mantenimiento.

El borrado del archivo **App_Offline.htm** restablece de forma automática el funcionamiento normal de ASP.Net.

Este recurso fue introducido en la **versión 2 de ASP.Net** y es de gran ayuda para el mantenimiento de entornos en producción.

CONCLUSIONES

Durante este trabajo de tesis se ha tenido la posibilidad de conocer más a fondo cómo labora una empresa cuyo rol principal son las actividades en campo, de qué forma administra las diferentes tareas de un proyecto, la manera en que los implicados en dichos proyectos interactúan, así como las dificultades a las que se enfrentan como son: incumplimiento en las fechas de entrega, mala administración de mano de obra, fuga de materiales, mala comunicación con el cliente, etc., todos estos factores influyeron para el desarrollo de una aplicación Web que permitiera una buena administración de los proyectos en cualquier sitio, a través del uso de Internet.

Además he podido diferenciar el cambio en los controles que se tenían antes y después de que la aplicación Web para la administración de proyectos empezara a funcionar, aunque ha sido poco el tiempo de prueba y uso; ésta ha resultado bastante efectiva ya que todos los integrantes del proyecto: administrador, líder y cliente han podido realizar las actividades que les corresponden según sus roles, en el tiempo que les corresponden y eso les ha traído buenos resultados mejorando sobre todo en el tiempo de entrega y cuando se trata del caso contrario, se tiene una justificación bien respaldada, siendo ésta del conocimiento de cliente desde el momento en que se generó.

Otro punto de gran importancia es que la empresa carecía de una página Web informativa y se creó junto con la aplicación una página de éste tipo, además se agregó un apartado en donde el cliente real o potencial puede contactar con la empresa para cualquier comentario, sugerencia, solicitud, requerimiento, etc. y la empresa contará con una base de datos de estos clientes para contactarlos posteriormente. En este mismo apartado se agregó una pequeña evaluación que el cliente deberá hacer sobre la página que está consultando, esto con la finalidad de mejorar el aspecto y funcionalidad de la misma.

Para la realización de la aplicación se utilizaron tecnologías de vanguardia el ASP.Net con el Visual Studio 2008 para la elaboración de la aplicación Web y el Microsoft SQL Server 2000 como potente motor de bases de datos de alto rendimiento, se eligieron estas herramientas por considerarse de última tecnología y bastante robustas para la elaboración de aplicaciones cliente/servidor.

La realización de esta tesis me trajo como experiencia el profundizar en el aprendizaje de una nueva tecnología revolucionaria la .Net, que con el conocimiento del ASP.Net me hizo adentrar a un mundo con una gran gama de posibilidades para el desarrollo de aplicaciones Web y si a eso le agregamos que la aplicación fue dinámica, es decir que tiene interacción con una base de datos, me resultó bastante satisfactorio y más satisfactorio fue que el resultado se transformara en una herramienta práctica, de mucha utilidad para la empresa en cuestión, además de que pudiera adaptarse a cualquier empresa cuyo actividad principal esté basada en la administración de proyectos.

BIBLIOGRAFÍA

- Connolly Randy, “Core Internet Application Development with ASP.NET 2.0”, Ed. Prentice Hall, Estados Unidos, 2007, p.p. 1049.
- Hoang Lam, Thuan L. Thai, “.NET Framework Essentials”, Ed. O’Reilly, 2003, p.p. 530.
- Kanjilal, J., Putrevu, S., “ASP.NET Ajax in 24 Hours”, Ed. SAMS, Estados Unidos, 2008, p.p. 408.
- Klastorin, Ted, “Administración de Proyectos”, Ed. Alfaomega, 2005, p.p. 265.
- Liberty Jesse, Hurwitz Dan, McDonald Brian, “Learning ASP.NET 3.5”, Ed. O’Reilly, Estados Unidos, 2008, p.p. 588.
- Ryan D., Ryan T., “ASP.NET”, Ed. Hungry Minds, Estados Unidos, 2002, p.p. 318.

OTRAS FUENTES

- Internet:
<http://www.desarrolloweb.com/articulos/1328.php>
- Internet:
[http://msdn.microsoft.com/es-es/library/k1s94fta\(VS.80\).aspx](http://msdn.microsoft.com/es-es/library/k1s94fta(VS.80).aspx)
- Internet:
http://www.adrformacion.com/curso/puntonet/leccion1/tecnologia_punto_net.htm
- Internet:
http://andresnaranjo.typepad.com/implementando/2007/04/aplicando_tecno.html

- Internet:
<http://www.microsoft.com/spain/empresas/temas/publicidad/crearweb.msp>
- Internet:
<http://es.kioskea.net/contents/web/webdesign.php3>
- Internet:
<http://www.desarrolloweb.com/articulos/1519.php>
- Internet:
<http://www.yoreparo.com/nav/?url=http://support.microsoft.com/?scid=kb;es;319714>
- Internet:
<http://thinkingindotnet.wordpress.com/2007/06/17/trucos-creacion-de-programas-de-instalacion-para-aspnet-con-vs-2005/>
- Internet:
<http://nelsongomez.wordpress.com/2008/11/19/como-aplicar-seguridad-a-nuestras-aplicaciones-web-utilizando-tecnologas-aspnet/>
- Internet:
http://asp.com-e.net/template_permalink.asp?id=97