

Universidad Nacional de Ingeniería

Facultad de Ingeniería Industrial y de Sistemas



Implementación de una Solución Comercial en una empresa de Distribución Eléctrica

Informe de Suficiencia

Para Optar el Título Profesional de:
INGENIERO DE SISTEMAS

Gustavo Vivanco Carrillo

**Lima – Perú
2002**

*A mi madre y a mi abuela,
por enseñarme el camino correcto en esta vida*

*A mi familia,
por apoyarme y soportarme durante todos estos años*

*A la universidad,
por orientarme y prepararme para la vida profesional*

*A esa chica,
que con su amor, aliento y sonrisa
me apoyó en la culminación de esta etapa*

INDICE

RESUMEN EJECUTIVO	7
INTRODUCCION	8
I. ANTECEDENTES	10
1.1. Diagnóstico Estratégico	10
1.1.1. Fortalezas	10
1.1.2. Debilidades	10
1.1.3. Oportunidades	11
1.1.4. Riesgos (Amenazas)	11
1.2. Diagnóstico Funcional	11
1.2.1. Producto	11
1.2.2. Clientes	12
1.2.3. Proveedores	12
1.3. Organización de la Empresa	12
II. MARCO TEÓRICO	14
2.1. Reglas de Negocio y el Mercado Actual	15
2.2. Reglas de Negocio y Sistemas de Software	16
2.3. Aplicaciones de Dos Capas	17

2.4.	Limitaciones de las Aplicaciones de Dos Capas	18
2.5.	Aplicaciones de Tres Capas	20
2.6.	Ventajas de las Aplicaciones de Tres Capas	21
2.7.	Aplicaciones Distribuidas	22
2.7.1.	Visión General	23
2.7.2.	Beneficios de las Aplicaciones Distribuidas	25
2.7.3.	Beneficios Tácticos	26
2.7.4.	Beneficios Estratégicos	27
2.8.	Arquitectura Windows DNA	27
2.8.1.	COM / DCOM	27
2.8.2.	Ventajas COM	29
2.8.3.	Implementación COM	30
2.8.4.	Componentes	31
2.8.5.	Transacciones	32
2.8.6.	Introduciendo Confiabilidad	34
2.8.7.	Administrando la Complejidad en Tiempo de Ejecución	34
2.9.	Metodología de Desarrollo de Aplicaciones	35
III.	PROCESO DE TOMA DE DECISIONES	37
3.1.	Planteamiento del Problema	37
3.2.	Alternativas de Solución	38
3.3.	Metodología de Solución	40
3.4.	Toma de Decisiones	41
3.5.	Estrategias Adoptadas	42

3.6.	Módulos del Sistema	45
IV.	EVALUACIÓN DE RESULTADOS	53
V.	CONCLUSIONES Y RECOMENDACIONES	58
5.1.	Conclusiones	58
5.2.	Recomendaciones	59
	BIBLIOGRAFÍA	60
	ANEXOS	61

DESCRIPTORES TEMÁTICOS

APLICACIONES DISTRIBUIDAS

N-CAPAS

COMPONENTES

TRANSACCIONES

ARQUITECTURA

DISTRIBUCIÓN

ESTRATÉGIA

COSTO / BENEFICIO

RESUMEN EJECUTIVO

Luz del Sur (antes Edelsur) fue creada al 1ro. de enero de 1994. El 18 de agosto de 1994 Ontario Quinta A.V.V. adquirió del Estado Peruano el 60% de las acciones de la empresa.

En agosto de 1996 la Empresa se escindió creándose la actual Luz del Sur, a la que se le transfirió la concesión de distribución de energía eléctrica. El mismo año, se incorporó la Empresa Eléctrica de Cañete, Edecañete S.A., al adquirir el 100% de su propiedad al Estado Peruano.

En setiembre de 1999, el consorcio Peruvian Opportunity Company S.A.C. adquirió el 22,46% de las acciones de la Empresa en una Oferta Pública de Adquisición realizada en la Bolsa de Valores de Lima.

El giro de negocio principal de la Empresa es realizar actividades de distribución eléctrica, pero también el poder realizar nuevos negocios. Uno de estos negocios pretende ser el de desarrollo de sistemas informáticos, aprovechando el conocimiento acumulado del negocio del sector eléctrico y aplicando la tecnología de punta que actualmente se encuentra en el mercado.

INTRODUCCIÓN

Los objetivos del presente informe son los siguientes:

- ✓ Dar una visión general de mi experiencia, presentándola como una alternativa para el desarrollo de software en el Perú.
- ✓ Mostrar una metodología apoyada en una plataforma tecnológica de punta.
- ✓ Mostrar factores que hay que tener en cuenta para la buena utilización de las herramientas mencionadas.
- ✓ Mostrar la capacidad de diversificación de la cartera de negocios que pueden realizar las empresas en el Perú.

Los logros de este trabajo fueron los siguientes:

- ✓ Generación de un producto base, como Sistema de Gestión Comercial, que puede ser implantado en las diferentes empresas eléctricas del país.
- ✓ Definición de una metodología y estándares para proyectos sobre tecnología Microsoft.
- ✓ Experiencia adquirida en la gestión de proyectos de desarrollo de sistemas para clientes externos.
- ✓ Mejoramiento de los procedimientos comerciales de la empresa en la cuál se implantó el sistema. Esto se puede medir comparando la cantidad de horas hombre necesarias y la cantidad de tareas manuales que han sido automatizadas.

Las limitaciones encontradas durante el desarrollo del sistema fueron las siguientes:

- ✓ Debido a la falta de presupuesto (en tiempo y dinero) para investigación, estas labores tuvieron que realizarse durante el desarrollo mismo del sistema, ocasionando sobretiempos y una labor mas recargada, en especial cuando se acercaban las fechas de entrega.
- ✓ No hubo dedicación exclusiva en la gestión del proyecto, sino hasta la implantación, debido a que se tenían que atender en paralelo las necesidades de Luz del Sur.
- ✓ Los cambios de usuarios ocasionaron cambios de definiciones y por ende, cambios en los módulos.
- ✓ Los datos a migrar no eran consistentes ni confiables.

CAPITULO I

ANTECEDENTES

1.1. DIAGNÓSTICO ESTRATÉGICO

1.1.1. FORTALEZAS

- ✓ Se presenta un mercado potencial grande, principalmente, empresas eléctricas del interior del país.
- ✓ Conocimientos de más de una plataforma tecnológica de punta.
- ✓ Al no estar asociado a algún proveedor tecnológico particular, se puede utilizar cualquier combinación tecnológica.
- ✓ Al no ser una unidad de negocio independiente, con un presupuesto independiente, no tiene un objetivo de ventas que cumplir.

1.1.2. DEBILIDADES

- ✓ El hecho de no contar con un presupuesto exclusivo para desarrollos externos, debido a que es un negocio naciente y no es el giro principal de la empresa.
- ✓ Por ser un negocio naciente, no se tiene una experiencia muy sólida en todas las combinaciones tecnológicas existentes, estando en desventaja con la competencia.

1.1.3. OPORTUNIDADES

- ✓ Generar mayor ingreso económico para la empresa.
- ✓ Aprovechar experiencias propias en el desarrollo de sistemas informáticos comerciales del sector eléctrico.
- ✓ Integrar los conocimientos informáticos con los conocimientos de procedimientos comerciales para generar un paquete completo: procesos-implementación.
- ✓ Realizar la misma integración para los otros rubros de desarrollo: sistemas administrativos y sistemas técnicos.
- ✓ Posibilidad de conocer nuevas plataformas tecnológicas por proyecto solicitado.
- ✓ Absorber el know-how de las empresas cliente para complementar el producto y hacerlo más robusto y flexible.

1.1.4. RIESGOS (AMENAZAS)

- ✓ Distracción de recursos destinados a atender necesidades propias de la empresa, en proyectos externos.
- ✓ El precio de los productos de la competencia. Esto es debido a que el mercado cuenta con productos maduros y estos pueden ser ofrecidos a un precio menor que el de un producto en desarrollo.

1.2. DIAGNÓSTICO FUNCIONAL

1.2.1. PRODUCTO

Luz del Sur no tiene un producto físico que ofrecer. Es una empresa de servicios que brinda el servicio de la distribución eléctrica. Para el negocio específico del desarrollo

de sistemas, el único producto desarrollado hasta el momento y que es razón del presente informe es el SIGECO (Sistema de Gestión Comercial).

1.2.2. CLIENTES

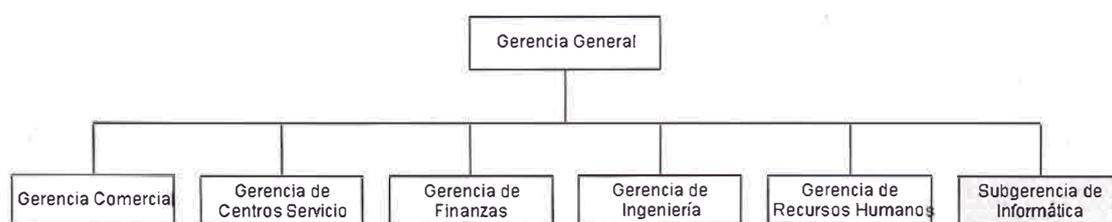
Los clientes de la empresa son todos las zonas con servicio eléctrico de Lima Sur y los clientes potenciales son todos los habitantes de esta misma zona. Para el negocio de software, aún no hay clientes que atender, ya que Edecañete es atendido por su condición de filial de Luz del Sur.

1.2.3. PROVEEDORES

Los proveedores del negocio eléctrico de las empresa son las empresas de mantenimiento, reparaciones, proveedores de servicios especiales, etc. Para el negocio de software, los proveedores son las empresas de ventas de hardware y software.

1.3. ORGANIZACIÓN DE LA EMPRESA

A continuación se muestra el organigrama de la empresa, indicando donde se encuentra ubicada la Subgerencia de Informática:



La Subgerencia de Informática tiene tres departamentos:

- ✓ *Departamento de Telecomunicaciones.* Encargado de la infraestructura de comunicaciones de la empresa, tanto internas como externas.

- ✓ *Departamento de Soporte Técnico.* Encargado de lo que es la gestión del hardware de la empresa
- ✓ *Departamento de Desarrollo de Sistemas.* Encargado del mantenimiento y desarrollo de los sistemas de software de la empresa. Sobre este departamento recae la responsabilidad del desarrollo de proyectos externos que involucran el desarrollo de sistemas de información.

CAPITULO II

MARCO TEÓRICO

Dos conceptos destacan la relación entre el entorno de negocios y el entorno de los sistemas de software: las reglas de negocio y los requerimientos de negocio.

Las reglas de negocio abarcan aquellas practicas y políticas que definen el comportamiento de una corporación. Mientras a menudo el sentido común es ampliamente usado por las empresas, las reglas de negocio también generan aquellos comportamientos únicos que resultan en una ventaja competitiva en el mercado.

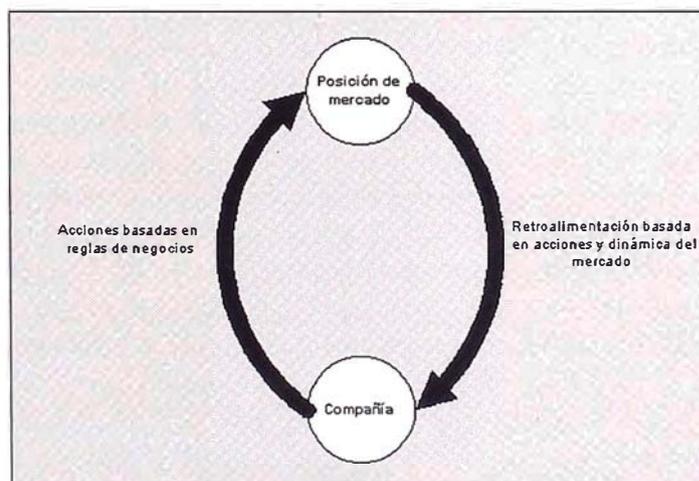
Los requerimientos de negocios son coacciones, por lo general auto impuestos, que las empresas usan para ayudarse a operar en su particular ambiente de negocios. Los requerimientos de negocio a menudo definen una línea de base para los requerimientos de aplicaciones y proporcionan pautas para el desarrollador. En términos prácticos, estos requerimientos de negocio son metas que los desarrolladores procuran reunir para sus aplicaciones.

Juntos, las reglas y los requerimientos de negocio proporcionan un contexto para desarrollar aplicaciones que puedan ayudar a una empresa a alcanzar sus objetivos o metas. Estas metas son únicas para cada empresa, pero pueden ser generalizadas para lo siguiente:

- ✓ Acceso apropiado a la información de manera oportuna.
- ✓ Generar mejores decisiones de negocios.
- ✓ Responder rápidamente a cambios en el entorno.
- ✓ Mejorar la comunicación entre la organización y los clientes.
- ✓ Alentar la innovación entre los empleados.
- ✓ Crear un mejor retorno de inversión.

2.1. REGLAS DE NEGOCIO Y EL MERCADO ACTUAL

Las reglas de negocio son aquellas series de practicas y políticas, algunas explícitas y otras implícitas, que definen como una empresa hace negocio en el mercado. La siguiente ilustración muestra la interacción entre una empresa y su posición en el mercado:



Los detalles de dicha interacción son a menudo enmarañados y muy complejos. Para una ejecución satisfactoria, un estado de flujo seguro y productivo de retroalimentación, se debe seguir lo siguiente:

- ✓ Extraer lo más posible, los datos pertinentes al entorno o segmento del mercado.
- ✓ Condensar, consolidar y priorizar este flujo de datos en información útil.

- ✓ Analizar y priorizar la información de acuerdo a su utilidad en la compañía.
- ✓ Crear un plan de acción que use esta información para ayudar a realizar sus metas.
- ✓ Llevar a cabo el plan.

Esta actividad es continua, desarrollada para reunir las necesidades del mundo mercantil. La simplificación de la secuencia anterior asume que la empresa es de “una sola opinión”, lo cual no es siempre verdad excepto para las empresas pequeñas.

Además, la empresa debe tener las siguientes características internas:

- ✓ Comunicación precisa y suficiente entre las partes apropiadas.
- ✓ Entendimiento consistente de las metas dentro de la empresa.
- ✓ Una sola persona responsable de cada plan.

2.2. REGLAS DE NEGOCIO Y SISTEMAS DE SOFTWARE

Los sistemas de software juegan varios roles importantes en la realización de las metas de las reglas de negocios y los beneficios de la interacción de la empresa con el mercado. Los sistemas de software proporcionan la facultad para:

- ✓ Almacenar los datos del mercado no depurados y la información procesada.
- ✓ Encapsular las reglas de negocios en software.
- ✓ Cambiar rápida y fácilmente las reglas para responder a un cambio en el mercado.
- ✓ Actuar automáticamente sobre las reglas.

Una aplicación típica que interactúa con un usuario, desde hojas de cálculo en computadoras personales hasta sistemas de cuentas de pago en mainframes,

consisten de tres elementos: lógica de presentación, lógica de negocios y lógica de datos.

La lógica de presentación enfoca la interacción con el usuario. La lógica de negocios desempeña cálculos y determina el flujo de la aplicación. La lógica de datos administra información que debe persistir a través de las sesiones o ser compartida entre los usuarios.

Con respecto a la configuración y distribución, los desarrolladores tienen varias opciones. Las aplicaciones de escritorio por lo general consisten en una aplicación monolítica que es ejecutada en una máquina. Las aplicaciones de mainframe tradicional son también monolíticas, pero son compartidas entre múltiples clientes. Algunas aplicaciones separan la lógica de presentación, de la lógica de negocios y de la de datos (juntan estas dos últimas). Otras aplicaciones separan los tres elementos uno del otro. Estas capas no corresponden necesariamente a ubicaciones físicas en la red. Por ejemplo, las tres capas pueden existir en dos máquinas solamente o ser distribuidas en cinco.

2.3. APLICACIONES DE DOS CAPAS

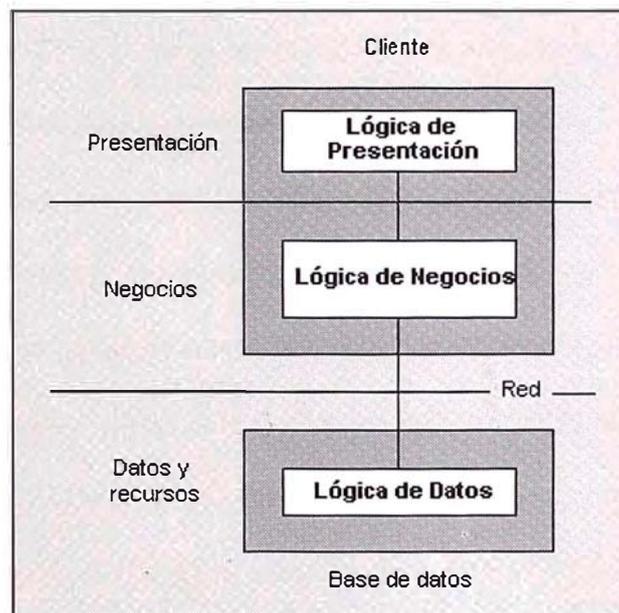
Las aplicaciones de dos capas o cliente/servidor, agrupan los componentes de lógica de presentación y de negocios en la máquina del cliente y accesan a una fuente de datos compartidos (lógica de datos) usando una conexión de red.

La ventaja de dicha configuración es que los datos son centralizados. Esta centralización beneficia la organización mediante:

- ✓ Distribución de datos.
- ✓ Proporcionando consistencia en los datos accesados.

- ✓ Reduciendo la duplicación y mantenimiento.
- ✓ Simplificando la generación de reportes.
- ✓ Facilitando actualizaciones de hardware.

Las aplicaciones de dos capas trabajan bien en aplicaciones de escala departamental con un pequeño número de usuarios.



2.4. LIMITACIONES DE LAS APLICACIONES DE DOS CAPAS

Las clásicas arquitecturas de dos capas han llevado eficiencia a los negocios pero hay también un buen número de limitaciones:

- **Aplicación de cliente monolítico.**

Las aplicaciones de dos capas tienden a tener componentes de cliente monolíticos, los cuales impiden mejoras incrementales (actualizaciones y correcciones de error) para la aplicación.

- **Dificultad para escalar.**

La escalabilidad de la aplicación es pobre debido a la cantidad limitada de conexiones de base de datos disponibles para los clientes. Las conexiones que requieren mas allá de este límite son simplemente rechazadas.

- **Dificultad para mantener.**

Es difícil mantener la lógica de negocios del lado del cliente, debido a que tiene que ser distribuido para cada cliente. Cualquier cambio en ella debe ser redistribuido para todos los clientes.

- **Compromiso de confidencialidad.**

La lógica de negocios en el cliente potencialmente expone las reglas de negocio a los usuarios.

- **Dificultad para volver a usar.**

Es difícil volver a usar ampliamente la lógica de negocios en dos capas ya que las aplicaciones son limitadas a formatos específicos de bases de datos.

- **Limitado a la fuente de datos.**

El cliente es a menudo configurado para una base de datos particular, la migración a una base de datos diferente es más difícil. (Aquí ODBC puede brindar alguna ayuda).

- **Pobre desempeño de la red.**

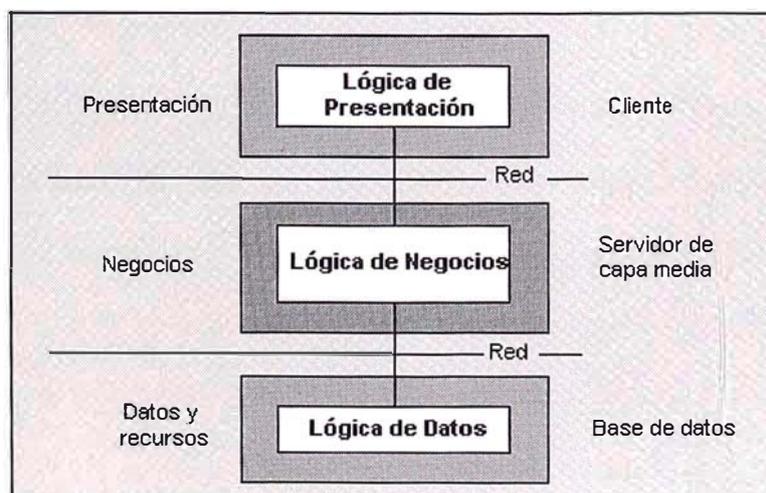
El rendimiento de la red es deficiente debido a la cantidad de registros que son transferidos a través de ella.

2.5. APLICACIONES DE TRES CAPAS

En la arquitectura de tres capas, la lógica de presentación, la lógica de negocios y la lógica de datos están separados conceptualmente. Los componentes de la presentación administran la interacción del usuario y solicitan los servicios de la lógica de negocios por medio del llamado de componentes de capas intermedias. Los componentes de lógica de negocios ejecutan las reglas de negocios y hacen solicitudes a la base de datos.

Los diseños de aplicación llegan a ser más flexibles debido a que los clientes pueden llamar a los componentes alojados en el servidor tantas veces como sea necesario para completar un requerimiento, y los componentes pueden llamar a otros componentes para mejorar la reusabilidad de código. Las aplicaciones de tres capas implementadas usando múltiples servidores a través de una red son conocidas como *aplicaciones distribuidas*.

Nota: El termino “aplicación de n-capas” es otra manera de referirse a las aplicaciones distribuidas. Una aplicación de n-capas representa un caso especial de una aplicación de tres capas, donde una o más de las capas son separadas en capas adicionales proporcionando otras capas de abstracción. Lo que se aprende del diseño e



implementación de aplicaciones de tres capas pueden ser extendidas a un diseño de n-capas.

2.6. VENTAJAS DE LAS APLICACIONES DE TRES CAPAS

Las arquitecturas de tres capas son a menudo llamadas servidores-centricos ya que permiten únicamente componentes de aplicación para ser ejecutados en servidores de capas medias, independiente de la interface de presentación y de la implementación de la base de datos. La independencia de la lógica de presentación, lógica de negocios y lógica de datos ofrece muchos beneficios:

- **Soporte multi-lenguaje.**

Los componentes de aplicación pueden ser desarrollados usando lenguajes de programación general.

- **Componentes centralizados.**

Los componentes pueden ser centralizados para fácil desarrollo, mantenimiento y distribución.

- **Balance de carga.**

Los componentes de aplicación pueden ser distribuidos a través de múltiples servidores permitiendo mejor escalabilidad.

- **Eficiente acceso de datos.**

Los problemas de limitación de conexiones a la base de datos son minimizados desde que la base de datos actual ve solamente el componente de aplicación y no todos sus clientes. También las conexiones de base de datos y drivers no son requeridos en el cliente. Las conexiones de base de datos de aplicaciones de dos capas son creadas al inicio y luego mantenidas por largo tiempo, mientras que en

las aplicaciones de tres capas, ellas son creadas mas tarde y liberadas rápidamente.

- **Seguridad mejorada.**

La seguridad de los componentes de aplicación de capa media puede ser configurada centralizadamente usando una infraestructura común. El acceso puede ser otorgado o negado componente por componente, simplificando su administración.

- **Acceso simplificado a fuentes externas.**

El acceso a fuentes externas tales como aplicaciones mainframe y otras bases de datos se simplifica; un servidor externo llega a ser un componente que es usado por la aplicación.

2.7. APLICACIONES DISTRIBUIDAS

El argumento para usar las aplicaciones distribuidas es simple: mediante la división de una aplicación compleja en lógica de presentación, lógica negocios y lógica de datos, el resultado aumentará la eficiencia, escalabilidad y confiabilidad del sistema.

Una aplicación distribuida segmenta lógicamente los componentes de un sistema cliente/servidor en tres capas: presentación, negocios y datos. Estas capas no corresponden necesariamente a ubicaciones físicas en la red. Por ejemplo, las tres capas pueden existir en dos maquinas solamente o ser distribuidas en cinco. Diseñar aplicaciones de esta forma da mayor flexibilidad cuando se distribuyen los procesos y datos en la red con desempeño máximo y fácil mantenimiento.

2.7.1. VISIÓN GENERAL

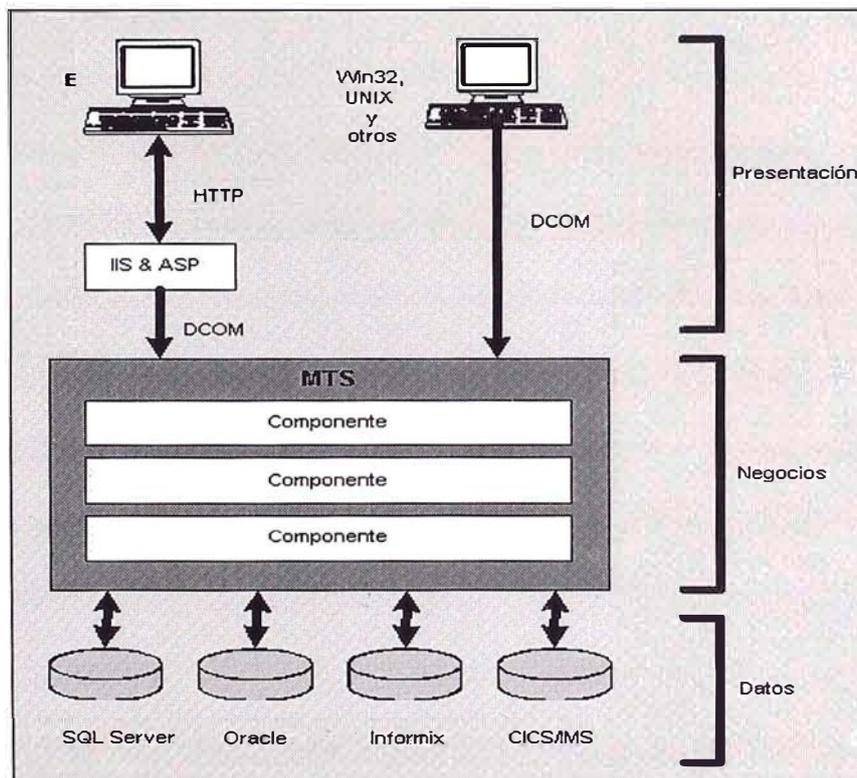
Antes de definir que es una aplicación distribuida, vale la pena definir que es una aplicación. Una aplicación es un programa de computadora que es usado para resolver un problema particular o una serie de ellos. Las aplicaciones básicas corren en un espacio del proceso y a menudo obtienen utilidad o ayuda, de funciones a través de librerías de enlace dinámico (DLL).

Los servidores son aplicaciones especializadas que proporcionan servicios a otras aplicaciones. En dicho contexto, la aplicación requiere un servicio que es llamado "cliente". El cliente y los servidores a menudo se ejecutan en maquinas separadas conectadas por una red. Una aplicación cliente que usa una aplicación servidor es llamada "aplicación cliente/servidor".

Frecuentemente, los servidores se comunican simultáneamente con los múltiples clientes. Para el cliente, el servidor parece estar interactuando exclusivamente con ese cliente únicamente. En realidad, los servidores son recursos compartidos que son usados por muchos clientes.

Para satisfacer el requerimiento de un cliente, un servidor puede llegar a ser un cliente para uno o más servidores creando una cadena de procesos desde el cliente original a muchos servidores. Esto es lo que significa una aplicación distribuida. Una aplicación distribuida es hecha de múltiples aplicaciones especializadas que trabajan juntas para ejecutar algunas metas útiles.

La lógica de presentación, la lógica de negocio y la lógica de base de datos en la aplicación distribuida pueden todas ellas ser ejecutadas en diferentes computadoras. A continuación podemos ver una ilustración de una aplicación distribuida y algunas de las tecnologías que pueden ser usadas para implementarla:



Para extender la definición a la empresa, una aplicación distribuida a nivel empresa es una aplicación distribuida robusta que es flexible para cambiar de acuerdo a las necesidades del negocio y responder a la creciente demanda del usuario. Puede ser distribuida en una variedad de plataformas a través de redes corporativas, intranets o internet. Debería ser amigable para el usuario y reunir estrictos requerimientos de seguridad, administración y mantenimiento. En resumen, es un sistema altamente complejo.

2.7.2. BENEFICIOS DE LAS APLICACIONES DISTRIBUIDAS

Como cualquier aplicación, una aplicación distribuida debe proporcionar una interface de usuario intuitiva, reunir las necesidades de los usuarios, etc. Mas allá de estas cualidades comunes, puede caracterizarse mediante tres requerimientos adicionales.

Una aplicación distribuida es:

- **Escalable**

Cuando el número de usuarios o la cantidad de trabajo se incrementa, el desempeño de la aplicación no se degrada significativamente.

- **Confiable**

Las aplicaciones confiables no impiden al usuario continuar con su trabajo debido a una falla de hardware o software. Las aplicaciones confiables también tienen un alto nivel de confianza por parte de los usuarios de la corrección de su operatividad y disponibilidad.

- **Eficiente**

Las aplicaciones eficientes hacen su trabajo rápido y son efectivas para ayudar al usuario a alcanzar sus metas.

Una aplicación distribuida a nivel empresa o aplicación empresarial es una aplicación de gran escala. Las aplicaciones empresariales tienen características adicionales mas allá de una aplicación distribuida. Una aplicación empresarial es:

- **Extensiva**

Una aplicación multiusuario que puede manipular enormes cantidades de datos y utilizar procesamiento paralelo, recursos distribuidas de red y lógica compleja es común en la empresa. Generalmente es desarrollada por un equipo de

desarrolladores y puede ser distribuida a través de múltiples plataformas e interoperar con muchas otras aplicaciones. Las aplicaciones empresariales son de larga vida.

- **Orientada al negocio**

Su propósito es satisfacer los requerimientos de negocios específicos. Codifica las políticas de negocio, procesos, reglas y entidades, es desarrollada dentro de una organización y puede ser distribuida de manera que pueda dar una respuesta adecuada a las necesidades del negocio.

- **De misión crítica**

Una aplicación empresarial debe ser suficientemente fuerte para sustentar su continua operación. Esta debe ser extremadamente flexible para su distribución y escalabilidad, y permitir un eficiente mantenimiento, monitoreo y administración.

2.7.3. BENEFICIOS TÁCTICOS

Los beneficios tácticos de aplicaciones distribuidas son:

- ✓ Una estrategia dividida puede ser usada en el desarrollo de la aplicación.
- ✓ Es posible centralizar la administración de las reglas de negocio.
- ✓ Uso eficiente del “pool” de conexiones a la base de datos.
- ✓ Distribución flexible de datos (el cliente no debe saber donde están los datos).
- ✓ Máxima utilización del servidor (muchos clientes pueden estar usando un mismo servidor)
- ✓ Mejor modelo de seguridad (roles).
- ✓ Fácil adaptación de protocolos (entre cliente y base de datos).

- ✓ Los clientes pueden ser personalizados (browser internet y PC clientes accedendo al mismo componente de capa media).

2.7.4. BENEFICIOS ESTRATÉGICOS

Los beneficios estratégicos (largo plazo) de las aplicaciones distribuidas son:

- ✓ Incremento en la eficiencia del trabajador.
- ✓ Nuevos conocimientos (tales como minería de datos).

2.8. ARQUITECTURA WINDOWS DNA

Las aplicaciones Windows DNA utilizan un conjunto estándar de servicios basados en Windows que dirigen los requerimientos de todas las capas de la aplicación: la interface de usuario y/o navegación, procesos de negocio y almacenamiento de datos. Estos servicios pueden ser componentes COM, DHTML, browser y servidores Web, transacciones, scripting, cola de mensajes (MSMQ), seguridad, directorios, acceso de datos y base de datos, administración de sistemas e interface de usuario, los cuales a su vez, son mostrados de una manera unificada en todas las capas para que las aplicaciones las puedan usar.

2.8.1. COM / DCOM

La evolución de la arquitectura Cliente / Servidor, el crecimiento de Internet y la continua innovación en el hardware y el software han proporcionado potentes y sofisticadas aplicaciones para los usuarios. Con todos estos adelantos, los problemas han recaído en los desarrolladores, los proveedores de software y los usuarios:

- ✓ Aumenta la complejidad y tamaño de las aplicaciones, aumenta el tiempo de desarrollo, se hace más difícil y costoso el mantenimiento; los riesgos se disparan cuando se pretende añadir funcionalidad a ese software.
- ✓ Las aplicaciones vienen “prefabricadas” con muchas características, la mayoría de las cuales no pueden ni suprimirse, ni actualizarse independientemente, ni sustituirse por alternativas.
- ✓ Estas aplicaciones son difíciles de integrar porque los datos y la funcionalidad de una aplicación no está disponibles para otras, aunque las aplicaciones se hubiesen desarrollado con el mismo lenguaje programación y se ejecutasen en el mismo ordenador.

La solución es un sistema en el que los desarrolladores creen componentes de software reutilizable. Un componente es un trozo de código reutilizable que puede integrarse con otros componentes –de otros proveedores o desarrolladores- con relativa facilidad. Así, un componente puede tratar una petición de un cliente (buscar un registro específico en una base de datos, por ejemplo) y devolver el resultado al cliente. El Modelo de Objetos Componentes (COM) constituye un modelo de programación basado en objetos y diseñado para promover el trabajo conjunto. Permite que dos o más aplicaciones –o componentes- cooperen fácilmente, aunque hayan sido escritas por diferentes equipos en diferentes momentos y se ejecuten con diferentes sistemas operativos. Para soportar esta interoperabilidad, COM define e implementa los mecanismos que posibilitan la conexión de las aplicaciones como si de objetos se tratase.

2.8.2. VENTAJAS COM

Las principales ventajas de COM son las siguientes:

- **Compatibilidad binaria**

Cualquier lenguaje de programación que pueda crear estructuras con punteros y realizar, ya sea de forma explícita o implícita, llamadas a las funciones apuntadas por esos punteros, pueden crear y utilizar componentes COM. Estos componentes pueden implementarse con una gran variedad de lenguajes de programación y utilizarse posteriormente desde clientes escritos en otros lenguaje completamente diferente. COM asegura la compatibilidad binaria entre el cliente desarrollado en Visual Basic y los componentes desarrollados en otro lenguaje como Visual C++, Visual J++ o cualquier otra herramienta de terceros.

- **Compatibilidad en el trabajo con sistemas operativos distintos**

Los componentes COM deben desarrollarse para una plataforma específica, como Windows NT o UNIX ya que no pueden ejecutarse en una plataforma que no sea para la que fueron escritos; sin embargo, si que puede comunicarse con otros objetos COM situados en otras plataformas, incluyendo Windows 95, Windows NT, Macintosh y UNIX.

- **Localización transparente**

Permite usar un componente desde un cliente sin importar si ese componente pertenece al mismo proceso, a otro proceso en la misma máquina o a otro proceso en una máquina distinta.

- **Reutilización del código**

Es la mayor ventaja de COM. Los componentes COM contienen clases que exponen grupos de métodos, conocidos como interfaces, a través de los cuales los

clientes se comunican con objetos. Dado que estas interfaces están bien documentadas, el código que implementan puede reutilizarse con facilidad.

- **Control de versiones**

Las interfaces implementadas en las clases son estáticas y, una vez definidas, no varían. Por lo tanto, para añadir nuevas funcionalidades habrá que añadir nuevas interfaces a las clases. Se puede cambiar la implementación de las viejas interfaces siempre que se mantenga la compatibilidad con los clientes que ya las usaban. No se pierde funcionalidad cuando se actualiza o modifica un componente COM, siempre se mejora o se añaden nuevas.

2.8.3. IMPLEMENTACIÓN COM

El Modelo de Objetos Componentes (COM) es una especificación de cómo estos objetos interactúan con su clientes. Como tal especificación, COM define una funcionalidad estándar, no define cómo debe implementarse sino qué es lo que debe tener. Entre lo que especifica COM se incluye lo siguiente:

- ✓ Cómo se instancian los objetos COM.
- ✓ Cómo se acceden los clientes a las características de estos objetos.
- ✓ La responsabilidad de su auto-destrucción cuando ya no quedan instancias suyas.

Además de lo que es meramente la especificación, existen “bibliotecas COM” que contienen:

- ✓ Una reducida API para facilitar la creación de aplicaciones COM, tanto clientes como servidoras. Esta API ofrece, por un lado, las funciones para trabajar con clientes y, por otro, las que permiten a los servidores proporcionar objetos COM.

- ✓ Servicios de localización, a través de los cuales se averigua en qué servidor se encuentra el objeto y la ubicación dentro de éste. Indirectamente esto incluye un soporte entre el identificador de objeto y el paquete en el que se encuentra su código, que normalmente se obtiene a través del Registro de Windows. De esta manera, los clientes quedan totalmente independientes de los posibles cambios de ubicación de los objetos COM, ya sean cambios de paquete o -incluso- cambios de máquina servidora.
- ✓ Servicios para la transparencia en las llamadas a procedimiento remotos, es decir, cuando un objeto está ejecutándose en un proceso aparte o en una máquina distinta desde la que se usa.

Existen versiones de estas bibliotecas para Windows 3.1, 95, 98, NT, para Apple, Macintosh y para Solaris y se espera incluir más sistemas operativos en esta lista.

2.8.4. COMPONENTES

Los componentes de software son unidades de código, que en tiempo de ejecución proporcionan funcionalidad específica para una aplicación. Hay muchas razones para usar componentes en una aplicación:

- **Reusabilidad**

Una vez creado un componente, otros desarrolladores pueden usarlo. Esto permite el fácil acceso a las características de cualquier componente en otras aplicaciones sin requerir que se re-escriban códigos extensos.

- **Reducción de la complejidad**

Pueden crearse componentes que escondan su complejidad. Los otros programadores que quieran utilizar estos componentes sólo necesitarán saber que información proporcionar al componente y que información recuperar.

- **Facilita las actualizaciones**

Los componentes hacen fácil la revisión y actualización de las aplicaciones. Por ejemplo, se puede crear un componente que encapsule las reglas de negocio. Si las reglas de negocio cambian, se actualiza solo el componente y no todas las aplicaciones que usan el componente.

- **Facilita el desarrollo**

Separando el código en componentes da la habilidad de desarrollar y probar pequeñas piezas encapsuladas de la aplicación independientemente. Esto también permite que múltiples desarrolladores trabajen en un proyecto juntos, permitiendo distribuir las tareas entre los miembros del equipo de desarrollo.

- **Neutralidad del lenguaje**

Los componentes pueden relacionarse con independencia del lenguaje; esto permite utilizar los lenguajes más eficientes según la funcionalidad que deba aportar el componente.

2.8.5. TRANSACCIONES

Al trabajar en aplicaciones multiusuario distribuidas se pueden involucrar cambios a tablas de base de datos y alguna otra entidad que no usa una base de datos (tales como mensajes en cola). A menudo, mas de un objeto en una actividad inicia el cambio. Ya que el trabajo en cualquier objeto puede fallar, la aplicación debe poder manejar las situaciones donde algunos objetos fallan pero otros tienen éxito. El trabajo

no puede ser dejado en un estado inconsistente donde algunos de los trabajos se han hecho pero otros no han sido hechos.

Las transacciones proporcionan un modelo simple de administrar el trabajo de “todo o nada”. O todos los objetos tienen éxito y todo el trabajo es llevado a cabo, o uno o mas objetos fallan y ningún trabajo es llevado a cabo. Basado en los resultados cualquier tabla de base de datos o archivos afectados por el trabajo serán todos cambiados, o de lo contrario, ninguno cambiará. No serán dejados en un estado inconsistente.

Una transacción proporciona un servicio de falta de tolerancia para un conjunto de recursos y es el elemento esencial para las aplicaciones distribuidas a nivel empresarial.

Una transacción tiene las propiedades siguientes:

- **Atomicidad**

Las transacciones deben ser indivisibles, es decir, o se ejecutan todas las acciones o no se ejecuta ninguna.

- **Consistencia**

Toda transacción debe llevar el sistema de un estado consistente a otro.

- **Independencia**

Para ser consistentes debemos imaginar que somos los únicos que estamos realizando operaciones en un momento dado; las demás transacciones deben ser invisibles para nosotros.

Como ejemplo citamos el siguiente caso: un viajero pide cambiar un billete de un vuelo a otro distinto. Una transacción se ocuparía de cancelar el billete del primer vuelo, al tiempo que otra transacción intentaría añadir un pasajero nuevo en el otro vuelo. Supongamos ahora que la primera transacción falla, el viajero no podría

volver a su “asiento” anterior porque ahora estaría ocupado con el nuevo viajero.

Este tipo de anomalías se evitan gracias a la independencia de las transacciones.

- **Durabilidad**

Cuando una transacción se confirma, permanece así aunque se produzcan errores del sistema, de las comunicaciones o del propio ordenador servidor.

2.8.6. INTRODUCIENDO CONFIABILIDAD

Las transacciones permiten a los participantes, recursos y procesos, tomar parte en los resultados de una operación transaccional. Un participante, tal como un administrador de base de datos, puede terminar con éxito o fracaso su trabajo. Una desavenencia por cualquier participante invalida una transacción, causando la recuperación del estado del sistema antes que empiece la transacción. Si todos los participantes tienen éxito, la transacción tendrá éxito y el sistema entrará a un nuevo estado.

Las transacciones, junto con los monitores de proceso de transacciones (TP), reemplazan los métodos clásicos de proporcionar un ambiente confiable con una plataforma bien definida que pueda ser usada sobre múltiples procesos y máquinas.

2.8.7. ADMINISTRANDO LA COMPLEJIDAD EN TIEMPO DE EJECUCIÓN

De la misma manera que los componentes proporcionan una manera de administrar el desarrollo de una aplicación estática, las transacciones ayudan a que los desarrolladores administren la complejidad en tiempo de ejecución de las aplicaciones. Las transacciones encapsulan los detalles de operaciones en tiempo de ejecución y proporcionan un resultado de éxito o falla que puede actuar sobre todo el proceso de la aplicación.

2.9. METODOLOGÍA DE DESARROLLO DE APLICACIONES

El proceso de desarrollo de aplicaciones abarca el análisis, diseño, desarrollo, distribución y mantenimiento. A continuación se describen los siete pasos principales de este proceso:

- **Análisis de los requisitos del software**

Se trata de extraer información acerca de los objetivos de la empresa y de los requisitos de los usuarios: características, preferencias y terminología. El objetivo del análisis es tener un conocimiento completo del entorno en el que la aplicación realizará su trabajo.

- **Definición de la arquitectura de la aplicación**

El objetivo de este paso es seleccionar la arquitectura más apropiada para el problema planteado. Ésta vendrá definida por el tipo de aplicación (de escritorio, C/S de 2 o 3 capas), las tecnologías que se usarán y los métodos para almacenar los datos.

- **Diseño de la solución**

En este paso, se generarán los diseños conceptual y lógico de la solución. El diseño conceptual está basado en los requisitos del software, incluye escenarios, diagramas de flujo, secuencias de tareas y el entorno físico. Una vez tenemos el diseño conceptual, extraemos de él la información para el diseño lógico, que incluirá los datos, las interfaces de usuario, componentes y servicios que tendrá la solución final.

- **Codificación e implementación de la solución**

Se incluye aquí la selección de la mejor herramienta para codificar la solución. Una vez seleccionada, se crearán las interfaces de usuarios, los componentes COM y los servicios de datos.

- **Prueba de la solución**

En este paso, debe crearse un plan para probar la aplicación, scripts para las pruebas, y ejecutar el plan.

- **Distribución**

Una vez que se ha probado la solución, llega el momento de ponerla en producción, es decir, de repartir los componentes COM por los servidores, las interfaces de usuario para los clientes, etc.

- **Mantenimiento**

El mercado hace que las empresas cambien y con ellas sus aplicaciones. El mantenimiento de las aplicaciones, además de cubrir estas continuas actualizaciones, abarca el soporte para los usuario, el mantenimiento de Web sites, sistemas gestores de datos, etc.

CAPITULO III

PROCESO DE TOMA DE DECISIONES

El desarrollo del sistema que se va a presentar como resultado de este informe, surgió debido a una necesidad de Edecañete, empresa filial de Luz del Sur, y aprovechada por esta última como una oportunidad de negocio.

El problema en esta caso, podemos decir que era de Edecañete.

3.1. PLANTEAMIENTO DEL PROBLEMA

La empresa tenía un contrato de licenciamiento de uso de un Sistema Comercial, el cual le obligaba a un pago mensual y le daba derecho a actualizaciones del software en forma gratuita, además de atenciones a los problemas que ocurrieran durante la operatividad diaria del mismo. Pero en los últimos meses, los problemas habían empezado a ser cada vez más frecuentes y los tiempos de atención de los mismos eran demasiado largos. Además, el hecho de no contar con los programas fuente y no conocer en su totalidad las reglas del negocio que este sistema manejaba, debido a la falta de documentación, hacía difícil el hecho de explicar algunos resultados que dejaban dudas.

Este conjunto de acontecimientos hizo tomar a la jefatura, la decisión de cambiar de sistema. El nuevo sistema debería de reunir características de confiabilidad, flexibilidad y rendimiento que el actual no tenía. La empresa debería tener los derechos de propiedad sobre los programas fuente o, en su defecto, conocer al detalle las reglas de negocio que manejaría y que las atenciones de nuevos requerimientos sean rápidas. Para realizar el proyecto se plantearon alternativas, las cuales son analizadas más adelante.

3.2. ALTERNATIVAS DE SOLUCIÓN

Para solucionar el problema que se presentaba en Edecañete, se plantearon y analizaron las siguientes alternativas:

- **Conexión Directa a Luz del Sur para utilizar su sistema. Servicio DIGIRED.**

Esta alternativa consistía en la contratación de un servicio de canal dedicado de 128 kbps entre Edecañete y Luz del Sur, sede de San Isidro. DIGIRED es un circuito basado en cableado de cobre que se contrata con la condición de asegurar calidad de atención y eficiencia del circuito CIR. Este servicio lo brinda Telefónica y el costo era US \$ 3200 mensual.

- **Conexión Directa a Luz del Sur para utilizar su sistema. Servicio IP-VPN.**

Esta alternativa, al igual que la anterior consistía en la contratación de un servicio de canal dedicado de 128 kbps entre Edecañete y Luz del Sur, sede de San Isidro. IP es un servicio alternativo a DIGIRED, que consiste en una red basada en IP, la cual puede brindar más servicios, como voz y audio. Se solicitó la cotización de la modalidad más

alta del servicio: Calidad Oro, ya que sólo esta modalidad era conveniente por el costo que tiene. Las otras calidades son Plata y Bronce. Este servicio es brindado por Telefónica y el costo era de US \$ 2500 mensual.

- **Propuesta de Desarrollo del Sistema Comercial de un Proveedor Externo.**

Esta propuesta incluía el desarrollo de casi todos los módulos incluidos en la solución final. Se utilizaría Visual FoxPro 2.6, tanto para el desarrollo de las aplicaciones como para gestionar la base de datos. Esta alternativa incluía la entrega de los programas fuente del sistema. La inversión total era de US \$ 9450.00 los cuales se irían pagando a razón de US \$ 1350 mensual durante todo el tiempo de desarrollo, que era de 7 meses.

- **Desarrollo de un nuevo Sistema Comercial en sociedad con Luz del Sur.**

Esta propuesta incluía el desarrollo de todos los módulos incluido en la solución final. Se utilizaría tecnología Microsoft: Windows NT 4, Visual Basic 6.0, SQL Server 7.0, MTS y el desarrollo sería utilizando las técnicas de n-capas. Para el desarrollo de este sistema se creó una sociedad, donde Edecañete aportaba dos programadores, y Luz del Sur un Jefe de Proyecto y un programador. El sistema sería propiedad de Luz del Sur y Edecañete en forma conjunta, pero Luz del Sur tenía la potestad de poder venderlo a otra empresa, participando Edecañete de esta venta en un 30%. Esta alternativa incluía la entrega de los programas fuente del sistema. El tiempo de desarrollo era de 8 meses a un costo de US \$ 3000 mensuales, la inversión por la compra de equipo ascendía a US \$ 7500, haciendo un total de US \$ 31500.00. Para el

desarrollo propio, Edecañete utilizaría personal de planta, el cual ya laboraba en sus instalaciones.

- **Renovación del Contrato de Licenciamiento con el Proveedor del Sistema Comercial Actual.**

Este sistema es el que actualmente utilizaba Edecañete. Presenta continuas fallas que no eran corregidas a tiempo por la empresa que prestaba el servicio, teniendo en cuenta que Edecañete no contaba con los programas fuentes del sistema. El costo mensual de la licencia de uso era de US \$ 1500.00.

3.3. METODOLOGÍA DE SOLUCIÓN

La metodología utilizada es la descrita en el último punto del Capítulo Marco Teórico. A continuación se realiza una descripción breve de las acciones realizadas en cada paso de la metodología, para el desarrollo del proyecto:

- **Análisis de los requisitos del software**

En este caso se analizó el sistema actual y se realizaron reuniones con los usuarios para ver que funcionalidades nuevas requerían.

- **Definición de la arquitectura de la aplicación**

Para el proyecto se escogió una arquitectura de aplicación distribuida de 3 capas, servidor transaccional y base de datos relacional, utilizándose tecnología Microsoft para soportarla.

- **Diseño de la solución**

Para este proyecto, debido a que era un reemplazo de un sistema por otro, la mayoría de los procesos comerciales ya estaban definidos, ya existían los modelos

lógicos y físicos. Lo que se hizo fue un mejoramiento de dichos modelos, que en algunos casos consistió en pequeñas modificaciones y en otros un diseño casi nuevo.

- **Codificación e implementación de la solución**

La herramienta seleccionada fue Visual Basic 6.

- **Prueba de la solución**

Se realizó un proceso de pruebas de un mes en las instalaciones de Edecañete, donde el usuario probó todos los módulos de las aplicaciones.

- **Distribución**

Los componentes y la base de datos se instalaron en los servidores adquiridos para tal fin, configurándose adecuadamente para optimizar el rendimiento del sistema en Producción.

- **Mantenimiento**

Este punto correrá por cuenta de Edecañete, ya que el objetivo es que realicen sus propios desarrollos, habida cuenta que el producto les fue entregado con fuentes.

3.4. TOMA DE DECISIONES

Para tomar la decisión se tuvo en cuenta el factor del costo y principalmente el hecho de que Edecañete y Luz del Sur pertenecen al mismo grupo empresarial, lo que le daba gran confiabilidad a este proyecto.

La evaluación de las alternativas se dividió en dos: Evaluación Técnica y Evaluación Económica:

Para la Evaluación Técnica, se establecieron CRITERIOS generales, con un peso determinado para cada uno, y dentro de ellos un conjunto de CARACTERÍSTICAS, las

cuales fueron evaluadas para todas las alternativas y promediadas otorgaron un puntaje a cada CRITERIO, y estos a su vez fueron ponderados para obtener el puntaje final de cada alternativa. Para el caso de la característica FUNCIONALIDADES, si bien es cierto que entre las alternativas se ofrecen casi los mismos módulos, estos tienen un “paquete de opciones” diferente, por lo que los puntajes asignados a los módulos de cada alternativa se calcularon en base a este “paquete de opciones”.

Para la Evaluación Económica se planteó un horizonte de 5 años (60 meses), asumiendo que esa sería la vida útil del sistema. Se consideraron los costos de cada alternativa, obteniendo el Valor Presente de todas ellas.

Finalmente, con los resultados anteriores se generó el indicador Costo/Beneficio, es decir, Total Evaluación Económica / puntaje final de Evaluación Técnica.

En ANEXO 1 y ANEXO 2 se puede observar el proceso de evaluación y las características tomadas en cuenta.

La decisión elegida fue la alternativa 4, que contemplaba el desarrollo de un nuevo sistema en sociedad con Luz del Sur. Como resumen del proceso de selección podemos decir que las alternativas 1 y 2 fueron descartadas por el alto costo que implicaba la interconexión, debido principalmente a la distancia geográfica y el alto costo que tienen los servicios en Cañete. La alternativa 3 no ofrecía la suficiente garantía y respaldo necesarias para un proyecto de esta envergadura. La alternativa 5 era la que se iba a tomar en caso no hubiera alguna otra alternativa mejor.

3.5. ESTRATEGIAS ADOPTADAS

El proyecto se desarrolló sobre un entorno tecnológico de n-capas, bajo Plataforma Abierta, donde concurren los siguientes componentes:

- **Servidor de Aplicaciones**

En este servidor se realizarán las tareas de gestión de las transacciones del sistema, las cuales se encuentran embebidas en los componentes de negocios y de datos. Aquí también se ubicarán los componentes del cliente, que serán los formularios y módulos del sistema. En este servidor se instaló Windows NT 4.0 SP6 como sistema operativo y como servidor de transacciones el Microsoft Transaction Server (MTS) 2.0. Este último es la herramienta utilizada para gestionar las transacciones del sistema, por lo que los componentes de negocios y de datos fueron matriculados aquí. Los componentes del cliente no fueron matriculados en el MTS, sino que son directamente instanciados desde las Estaciones PC de los usuarios. Las características técnicas son las siguientes:

- ✓ Servidor IBM NETFINITY 3000 Pentium III 933 MHz
- ✓ 256 MB de RAM
- ✓ 2 Discos duros de 18 GB cada uno

- **Servidor de Base de Datos**

En este servidor se instaló la Base de Datos, para la cuál se utilizó el SQL Server 7.0. En este servidor también se ubicaron los programas batch, que fueron hechos en C, los cuales son utilizados para la realización de procesos que implican la gestión de grandes cantidades de datos, como los procesos de facturación, actualización de saldos de los clientes, etc. En este servidor se instaló Windows NT 4.0 SP6 como sistema operativo. Las características técnicas son las siguientes:

- ✓ Servidor IBM NETFINITY 3000 Pentium III 933 MHz

- ✓ 512 Mhz de RAM
- ✓ 2 Discos duros de 36 GB cada uno
- ✓ 1 Unidad de respaldo DAT DDS3 de 4 mm Tape Backup

- **Estaciones PC de los Usuarios**

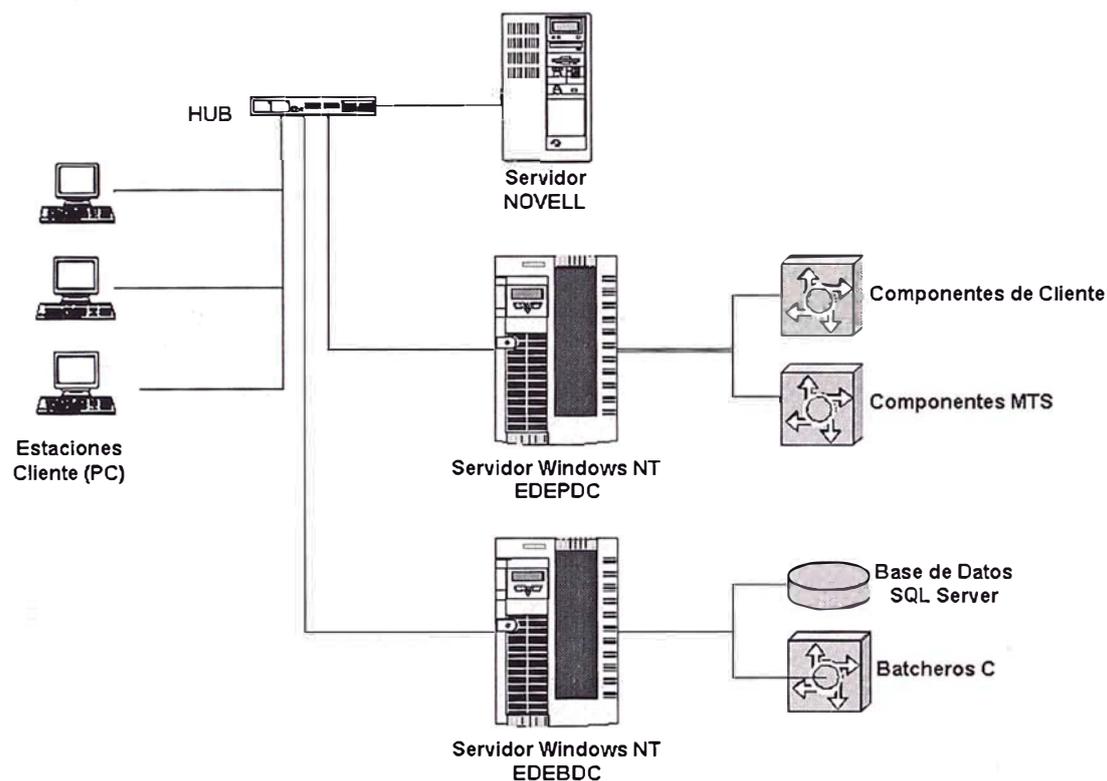
Para las PC se instalaron máquinas que tenían como mínimo de un procesador 486 con 16 Mb de memoria RAM, monitor color SVGA y Windows 95, esto garantizará que al usar estaciones de trabajo más poderosas se mejore significativamente el rendimiento del sistema.

Las aplicaciones serán desarrolladas sobre dicha arquitectura (n-capas). Los programas a ejecutar en las estaciones clientes se desarrollarán en Visual Basic 6, se tendrá un servidor de aplicaciones sobre el cuál funcionará el Microsoft Transaction Server (MTS) y la Base de Datos será SQL Server 7. El Sistema Operativo será Windows NT 4.0. En caso de ser necesarios procesos batch, estos serán desarrollados en Visual C++ 6 o en C estándar con sentencias SQL embebidas y que serán ejecutadas en el servidores de Base de Datos.

Todo el desarrollo está basado en la explotación de las interfaces gráficas que dispone el lenguaje Visual Basic, esto permite que la operación del sistema sea intuitiva para los usuarios, facilitando el proceso de aprendizaje de uso del sistema.

Asimismo se debe tener en cuenta la gestión de componentes que hace el MTS, lo cual permite la reutilización de componentes en varios módulos, lo cual reduce el tiempo de mantenimiento.

A continuación se muestra la configuración de la plataforma de Edecañete:



3.6. MÓDULOS DEL SISTEMA

A continuación se muestra una breve descripción de los módulos que componen la solución desarrollada:

- **Módulo Técnico**

Este módulo tiene por objetivo el registro y el mantenimiento de los datos correspondientes a los elementos de la red eléctrica de la empresa. Esta información es muy importante, ya que servirá para los diferentes procesos que ejecuten los diferentes Sistemas de la empresa (Comercial, Calidad, Técnico), los cuales utilizarán los datos para realizar consultas, cálculos de compensaciones, etc.

Entrega las herramientas necesarias para el mantenimiento de los elementos que componen la red eléctrica. Se definieron los siguientes elementos:

- ✓ SET
- ✓ Alimentador
- ✓ SED

- ✓ Llave
- ✓ Tramo

El Módulo permite registrar las relaciones jerárquicas entre cada elemento, de modo que se puedan utilizar estos datos para la representación gráfica de la red eléctrica, lo cual permite llegar desde el suministro del cliente, hasta el SET de origen.

- **Módulo de Solicitudes**

El objetivo del Módulo de Solicitudes es el de ingresar, modificar, controlar y procesar todas las solicitudes comerciales que los clientes hagan llegar a las oficinas de la empresa. Las solicitudes que se atienden por este módulo tienen la característica de necesitar el cálculo de un presupuesto o de algún de interés adicional a la deuda del cliente. Otro tipo de solicitudes, como cambios de nombre o direcciones, son atendidas por el Módulo de Modificaciones.

Las solicitudes comerciales que pueden ser procesadas por este módulo son las siguientes:

- ✓ Nuevo Suministro. Cuando se ingresa un nuevo cliente a la empresa.
- ✓ Modificación de Contrato. Cuando se realiza una modificación referida a cambio de tarifa, cambio de modalidad de facturación, aumento o disminución de potencia, cambio de fase, los cuales pueden cambiar las condiciones del contrato vigente del cliente
- ✓ Reincorporación de Suministro. Cuando los clientes retirados (cuyo medidor ya fue retirado físicamente del terreno después de alcanzar ocho meses de deuda) o congelados (a los cuales ya no se les factura, después de alcanzar siete meses de

deuda) solicitan el restablecimiento del servicio eléctrico, previa negociación con la empresa.

- ✓ Servicios Generales. Se refiere a servicios diversos que puede ofrecer la empresa, distintos de la distribución eléctrica. Estos servicios pueden ser otorgados tanto a clientes como a no clientes de la empresa.

- **Módulo de Modificaciones**

Este módulo tiene como objetivo el realizar modificaciones de los datos de los clientes de la empresa y sus medidores asociados. En este módulo se procesan las solicitudes de modificaciones que no involucran la generación de un presupuesto o algún otro cálculo. Estas modificaciones pueden ser solicitadas por el cliente o pueden ser hechas por iniciativa de la empresa respondiendo a alguna necesidad.

Este módulo permite la modificación de nombres, direcciones, teléfonos, rutas de lectura, datos de conexión, datos de medidores, cambio / retiro de medidor, etc. Todos los cambios posibles estarán describiéndose en la parte de funcionalidades.

Cuando se realiza alguna modificación de datos en este módulo, se graba un registro de control, en donde se indica que dato se modificó, el dato anterior y el nuevo, el usuario y la fecha de la modificación. De manera similar, cuando se cambia o retira un medidor, se graba un registro de control similar. Esto es muy importante para funciones de auditoría.

- **Módulo de Convenios**

Este módulo tiene por objetivo registrar y controlar todas las solicitudes de convenio que el cliente realice. Estas solicitudes pueden ser ingresadas por este módulo, así

como por el Módulo de Solicitudes, cuando el cliente desea pagar los presupuestos generados por medio de un convenio.

Los convenios están divididos en tipos. Para la operatividad del módulo se han definido los siguientes tipos:

- ✓ Deuda. Son los que se originan para financiar deuda vencida de energía del cliente.
- ✓ No deuda. Son los creados debido a deuda no facturada inicialmente, como recupero de energía o consumo no registrado.
- ✓ Conexión. Son los originados como opción de financiamiento para los clientes que soliciten nuevas conexiones.
- ✓ Servicios Generales. Son los originados como opción de financiamiento de los servicios adicionales que brinda la empresa.
- ✓ Deuda de Servicios. Son utilizados para financiar deuda vencida originada por servicios adicionales impagos.

Basándonos en estos tipos, el usuario puede crear las opciones de convenio que son las que finalmente se van a utilizar directamente en el Módulo.

A continuación se muestra el diagrama Entidad-Relación correspondiente:

- **Módulo de Recaudación**

El objetivo de este módulo es registrar, controlar y procesar todos los pagos ingresados a la empresa, por cualquier concepto, ya sea energía o servicios adicionales. Permite el registro de pagos recibidos por cajas de la empresa, bancos u oficinas recaudadoras. Este módulo se encarga de realizar los cuadros de caja, actualización de los saldos del cliente, generación de reconexiones automáticas.

Los procedimientos de este módulo se apoyan en un sistema de ingreso masivo de pagos, incluyendo lectura de código de barras y control de autorizaciones de cajeros. También permite el ingreso de pagos a cuenta, cuotas iniciales de convenios y la generación de reportes e informes de control.

- **Módulo de Cortes y Reconexiones**

El objetivo principal consiste en realizar las acciones de corte y reconexión, además de mantener la información actualizada respecto de la situación del estado de suministro de cada cliente en particular.

El Módulo entrega herramientas para la localización y selección de los clientes candidatos a corte del suministro, selección que puede realizarse según parámetros tales como sector, antigüedad de deuda, tarifa y consumo promedio, todos ellos modificables dinámicamente.

Tanto los cortes y reconexiones pueden realizarse mediante procesos masivos o individuales, generando los listados y archivos planos correspondientes para los contratistas encargados de la labor en campo, así como la carga masiva o individual de los resultados de dicha labor, al sistema.

Los cobros por corte se manejan relacionándolos con el tipo de corte y la acción realizada en el suministro, siendo esta parte totalmente paramétrica para el usuario.

- **Módulo de Lecturas**

Este Módulo tiene como objetivo la ejecución del proceso de lectura de cada ciclo de facturación en la empresa, empezando desde el establecimiento del calendario de

facturación, hasta la validación de las lecturas entregadas por el contratista para ser cargadas en el sistema, mediante una consulta de consistencia de consumos.

- **Módulo de Facturación**

El Módulo tiene como objetivos principales la valorización del consumo registrado, de cálculo de intereses y moras por no pago, cargos por cortes y convenios de energía, otros servicios, deuda vencida, impuestos, cálculo de la fecha de corte y antigüedad de deuda, emisión de la boleta correspondiente y la actualización de la información histórica de los clientes.

Este Módulo entrega herramientas para estructurar todo el esquema tarifario de la empresa, entendiéndose por esto, la creación de nuevos cargos, precios, tarifas, cobros por arriendo, entre otros.

Los diferentes procesos del ciclo permiten el cobro de los consumos, potencias, impuestos, intereses, cuotas de convenios vigentes, conceptos esporádicos, consumos o valores de servicios comunes de prorrateo y otros. Automáticamente, y en forma transparente, el sistema administra cualquier tipo de cliente, sin importar cual fuese su nivel de consumo. Permite además la coexistencia de clientes especiales (mayores), así como de clientes de tipo general y masivo (comunes), todos dentro del mismo sector de facturación si se desea. La separación de sectores y clientes dentro de cada ciclo de facturación, es de decisión del usuario.

Maneja en forma automatizada el reproceso de la facturación de los clientes desde puntos definidos del proceso, los cuales son:

- ✓ Generación de Libros de Lectura
- ✓ Cálculo de Consumo

- ✓ Cálculo de Importe
- ✓ Proceso de Facturación
- ✓ Cálculo de Intereses

- **Módulo de Refacturaciones**

El Módulo de Refacturaciones tiene como objetivo registrar y procesar los movimientos adicionales respecto a la cuenta del cliente por motivos varios. Estos motivos pueden ser:

- ✓ Refacturaciones. Propiamente dichas y que están orientadas a corregir alguna omisión o recargo ocurrido durante la facturación.
- ✓ Diferencia de Lectura: Son las ocasionadas por errores de lectura en terreno, ocasionando una facturación incorrecta.
- ✓ Débitos Varios: Cargos adicionales que son cobrados por algún servicio adicional brindado y que el cliente quiere que se le cobre por medio de su boleta de luz.

- **Consulta de Clientes**

El objetivo de este módulo es proporcionar un fácil acceso a toda la información actual e histórica de los clientes: sus antecedentes personales, las características del servicio que se le entrega, sus estados de pago y el estado de las órdenes asociadas a él.

El módulo tiene por función general acceder y consultar la información de los clientes a través de la implementación de una serie de consultas. La información obtenida se presenta principalmente en forma de tablas, en los cuales se podrán generar archivos del tipo *.xls (Excel) que contengan la información mostrada y para que el usuario pueda trabajar con ella en su PC.

Como es un Módulo de Consulta, no es posible realizar modificaciones de la información de la base de datos. La única modificación que se puede realizar es en el monto a pagar en la emisión de comprobantes de pago. Esto sólo modifica la base de datos para registrar el Comprobante de Pago emitido.

CAPITULO VI

EVALUACIÓN DE RESULTADOS

El sistema fue implantado en Octubre del año pasado, por lo que aún no hay una evaluación formal de los resultados y el impacto que tuvo en los Estados Financieros de Edecañete. Esta evaluación se podrá realizar recién a finales del 2002.

Para realizar una evaluación preliminar, mencionamos los puntos más importantes que permitieron aumentar ingresos o disminuir costos en la empresa:

- **Modificación del Procedimiento de Corte**

Cuando un cliente no paga su boleta después de dos meses, se le genera una orden de corte del servicio eléctrico. Este corte y la posterior reconexión (en caso de que el cliente pague la deuda) tienen un costo el cuál está normado por ley. Edecañete cobraba este concepto mediante la boleta de luz, cuando el cliente se acercaba a pagar el corte para obtener la reconexión. Esto implicaba que la empresa tenía que esperar el tiempo que el cliente tomara para acercarse a pagar (que llegaban inclusive a varios meses).

Con el nuevo sistema, este procedimiento cambió. Cuando el corte se efectuaba y este era registrado en el sistema, inmediatamente se cargaba el monto del corte y la reconexión en la siguiente boleta del cliente, independientemente de si este se acercaba a pagar o no.

A continuación mostramos un cuadro donde se indica la variación de los ingresos por concepto de corte (en soles) de los meses de operación con el nuevo sistema respecto a los meses correspondientes del año anterior:

	2000	2001	Variación
Octubre	14000	14500	3,45%
Noviembre	11000	13500	18,52%
Diciembre	12380	11000	-12,55%
Enero	14470	15000	3,53%

- **Modificación en el Cálculo del Interés**

El Procedimiento del Cálculo de Interés contemplaba la realización de dicho cálculo empleando el método de interés compuesto. El sistema nuevo empleaba el método de interés simple, ya que mediante este método, se obtenía un mejor resultado para la empresa. Para explicar mejor esto, lo graficaremos con un ejemplo:

Para obtener las tasas de interés en un período de tiempo en base a la TEA (Tasa Efectiva Anual) que es publicada por la SUNAT, se utilizan las siguientes fórmulas:

$$\text{Tasa Interés Simple} = TEA * \text{Días} / 360$$

$$\text{Tasa Interés Compuesto} = ((1 + TEA)^{1/360})^{\text{Días}} - 1$$

Asumiremos una TEA de 14% y períodos de 30 días. Este último valor lo escogimos debido a que la facturación de los clientes de esta empresa es mensual, por lo que el cálculo de intereses se hace cada 30 días.

Con estos datos, generamos el siguiente cuadro:

Días	Tasa Interés Simple	Tasa Interés Compuesto
30	1,1667%	1,0979%
60	2,3333%	2,2078%
90	3,5000%	3,3299%
120	4,6667%	4,4644%
150	5,8333%	5,6113%
180	7,0000%	6,7708%
210	8,1667%	7,9430%
240	9,3333%	9,1281%
270	10,5000%	10,3262%
300	11,6667%	11,5374%
330	12,8333%	12,7620%
360	14,0000%	14,0000%
390	15,1667%	15,2516%
420	16,3333%	16,5169%
450	17,5000%	17,7961%
480	18,6667%	19,0894%

Como podemos ver, las tasas se igualan para períodos de cálculo de un año. Pero como el cálculo de interés se realiza cada 30 días, la tasa de interés simple es mayor que la obtenida por el interés compuesto.

Como resultado de la aplicación de este nuevo tipo de cálculo, mostramos un cuadro con los valores facturados respecto a los correspondientes del año anterior y su variación (los montos están en soles):

	2000	2001	Variación
Octubre	21000	24900	18,57%
Noviembre	20000	22500	12,50%
Diciembre	22000	23500	6,82%
Enero	23000	24000	4,35%

- **Registro de hora de reconexión y fechas de atención de solicitudes**

El sistema anterior no permitía registrar la hora en la que se producía la reconexión, lo cual era requisito indispensable para medir la calidad del servicio de la empresa. El nuevo sistema permite este registro y con este dato la empresa puede controlar que la atención de las reconexiones no exceda las 24 horas, que es el límite impuesto por ley. De manera similar, se establecieron controles para los procedimientos de atención de solicitudes y reclamos, basados en el registro de las fechas de recepción, generación de presupuesto y atención de dichas solicitudes, comparándolas con los límites impuestos por ley. Estos controles tampoco estaban implementados en el sistema anterior.

La implementación de estos registros y controles permitió a la empresa evitar multas contempladas en la Escala de Multas aplicadas a las Infracciones a la Ley de Concesiones Eléctricas.

Como ejemplo citamos el Numeral 5 del Anexo 2 de dicha Escala: 'Cuando el usuario paga lo adeudado y el concesionario no cumple con reponer el servicio eléctrico dentro de las 24 horas de haberlo el usuario cancelado'. Dicha multa podrá ser hasta de 140000 kwh, lo cual multiplicado por la tarifa (0.3 soles / kwh aproximadamente) llegan a ser 42000 soles. Considerar que estamos citando sólo un ejemplo y que las posibles infracciones pueden ser varias.

- **Formatos pre-impresos**

Los formatos impresos, tales como Contrato de Nuevo Suministro, Solicitudes, etc., eran impresos en papel en blanco o membretado. Esto quiere decir, que se imprimía todo el contrato, se dibujaban líneas y demás detalles que debían tener estos

documentos. Esto ocasionaba una demora en el tiempo de impresión y un consumo elevado de cintas de impresora.

Para la implementación del nuevo sistema, se diseñaron formatos pre-impresos, en los cuales el sistema sólo imprimía unos cuantos datos más para que esté listo para la firma o entrega al cliente. Los beneficios fueron obvios, mejorando el tiempo de atención al cliente y disminuyendo los costos del procedimiento. La empresa venía operando con un tiempo promedio de 15 minutos por cada atención al cliente. Actualmente, este tiempo es de sólo 5 minutos.

Además de las ventajas cuantitativas, hay que considerar las ventajas cualitativas como:

- ✓ Propiedad de los programas fuente.
- ✓ Tecnología de desarrollo del producto.
- ✓ Mejora en los procesos comerciales debido a asesoría de Luz del Sur.

Los beneficios para Luz del Sur son los siguientes:

- ✓ Propiedad de un producto que puede empezar a implantarse en otras empresas.
- ✓ Conocimiento y experiencia adquiridos durante el proyecto.
- ✓ Empresa filial utilizando tecnología de punta.

Este sistema será mejorado como consecuencia de las observaciones realizadas durante su implantación, lo cuál lo hará más robusto, confiable y portable.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

- ✓ Se deben escoger las herramientas adecuadas y necesarias, los usuarios y analistas que conocen bien la funcionalidad deseada, y sobre todo, dejar en blanco y negro las responsabilidades de cada parte participante, compromisos con fechas y requerimientos con actas firmadas, así como si estos escapan o no al alcance del proyecto
- ✓ Es conveniente seguir una metodología formal de principio a fin de modo que se pueda llevar un control adecuado durante todo el desarrollo del proyecto.
- ✓ La tecnología es muy útil cuando es bien utilizada. No se debe utilizar una tecnología por el simple hecho de que “esté de moda” o por que se le ocurrió a alguien. Esto debe ser fruto de una evaluación seria y que determine cuál o cuáles son las que realmente satisface las necesidades de la empresa.

RECOMENDACIONES

- ✓ El equipo que se forme para la ejecución del proyecto debe estar totalmente identificado uno con otro. Se debe realizar un trabajo integrador del equipo previo al inicio del proyecto.
- ✓ En los proyectos de implantación de sistemas encontramos diferentes tipos de usuarios, a los cuales hay que saber identificar, tratar y entender. Hay de los que son colaboradores, que entienden que ningún sistema es perfecto y que te ayudan a resolver los problemas. Los que no son colaboradores, que entienden que el sistema es una amenaza para sus puestos de trabajo, por lo que ponen trabas para dar su aprobación final. Por estas razones, es muy importante la comunicación y explicar el objetivo de la implantación en la empresa y los beneficios que se van a obtener
- ✓ Cuando se realiza el levantamiento de requerimientos, siempre es importante que estén el usuario operativo, por que es el que conoce el detalle de la función a implementar, y alguien que desempeñe el cargo de analista funcional, ya que él puede ver el marco global de las funciones a implementar. Puede que el usuario solicite algo que sea muy beneficioso para su trabajo, pero que interfiera con alguna funcionalidad de otra área, o esta sea demasiado costosa comparándola con los beneficios a obtener. O puede que el analista solicite la implementación de un procedimiento muy bueno, pero que dificulte en algún sentido el trabajo de los usuarios.
- ✓ Los acuerdos que se tomen durante el desarrollo del proyecto deben ser refrendados mediante la suscripción de actas formales, ya que estas sirven luego para sustentar el trabajo realizado y definir responsabilidades.

BIBLIOGRAFIA

- ✓ Luz del Sur S.A.A. *Memoria Anual 2000*. Biblos S.A. – Lima, Perú, 2000.
- ✓ Microsoft Corporation. *Mastering Distributed Applications Design Using Visual Studio*. Cargraphics S.A. – Bogotá, Colombia, 1998
- ✓ Grupo EIDOS. *Programación DNA con Visual Basic 6*. Grupo EIDOS – Madrid, España, 2001.

ANEXO 1

Alternativas Evaluadas		Conexión a LDS por DIGIRED		Conexión a LDS por VPN-IP		Desarrollo por Proveedor Externo		Desarrollo por LDS		Renovación de Contrato		
Peso	Característica	Puntaje	Puntaje	Puntaje	Puntaje	Puntaje	Puntaje	Puntaje	Puntaje	Puntaje	Puntaje	
3	PLATAFORMA											
	Sistemas Operativos	Windows, Unix	7	Windows, Unix	7	Windows	5	Windows, WNT	6	Windows	5	
	Base de Datos	Informix	8	Informix	8	Fox Pro	2	SQL Server	7	Fox Pro	2	
	Arquitectura	Cliente / Servidor	7	Cliente / Servidor	7	Cliente / Servidor	5	COM / DCOM	9	Cliente / Servidor	5	
	Configuración y Ejecución de Procesos Batch	Si	8	Si	8	No	0	Si	7	No	0	
	Sistema de Seguridad Integrado	Si	7	Si	7	Si	5	Si	8	Si	5	
	Herramientas de desarrollo cliente	Visual Basic 3	4	Visual Basic 3	4	Visual Fox Pro 6	5	Visual Basic 6	8	Visual Fox Pro 6	5	
	Herramientas de desarrollo servidor	C	8	C	8	No tiene	0	C, Visual C++	9	Visual Fox Pro 6	3	
	Promedio Parcial	7.00	7.00	7.00	2.83	8.00	3.33					
	9	FUNCIONALIDADES										
Datos Técnicos		Si	9	Si	9	Si	2	Si	7	Si	4	
Incorporación de Nuevos Clientes		Si	8	Si	8	Si	5	Si	8	Si	6	
Reincorporación de Clientes		No	0	No	0	Si	1	Si	8	Si	1	
Modificación de Contrato		Si	7	Si	7	No	0	Si	8	No	0	
Servicios Generales		No	0	No	0	No	0	Si	8	Si	5	
Modificaciones de Datos Generales		Si	6	Si	6	Si	6	Si	6	Si	6	
Convenios		Si	6	Si	6	Si	5	Si	8	Si	5	
Recaudación		Si	8	Si	8	Si	6	Si	9	Si	6	
Cortes y Reconexiones		Si	6	Si	6	Si	6	Si	6	Si	6	
8	PRESENCIA LOCAL											
	Soporte Local	Buena	8	Buena	8	No tiene	0	Buena	8	Regular	5	
	Experiencia en Sector Eléctrico	Alta	9	Alta	9	No tiene	0	Alta	9	Buena	7	
	Respaldo Empresa Local	Luz del Sur	9	Luz del Sur	9	No tiene	0	Luz del Sur	9	Proveedor	6	
	Promedio Parcial	8.67	8.67	8.67	0.00	8.67	6.00					
	6	MANTENIMIENTO Y OTROS										
		Atención de Problemas / Requerimientos	Buena	8	Buena	8	No	0	Rápida	8	Mala	2
		Posibilidad de venta	No	0	No	0	No	0	Si	7	No	0
	Promedio Parcial	4.00	4.00	4.00	0.00	7.50	1.00					
	TOTAL BENEFICIO		6.52	6.52	6.52	1.71	7.90	3.95				

ANEXO 2

Alternativas	Conexión a LDS por DIGRED	Conexión a LDS por VPN-IP	Desarrollo por Proveedor Externo	Desarrollo por LDS	Renovación de Contrato
Total Inversión	0.00	0.00	9450.00	24000.00	0.00
Meses Desarrollo	0	0	7	8	0
Desarrollo Mensual (Total Inversión / Meses Desarrollo)	0.00	0.00	1350.00	3000.00	0.00
Valor Actual Desarrollo	0.00	0.00	8778.67	22093.48	0.00
Mantenimiento	3200.00	2500.00	0.00	0.00	1500.00
Meses Mantenimiento	60	60	0	0	60
Valor Actual Mantenimiento	114624.84	89550.66	0.00	0.00	53730.39
Servidor Aplicaciones	4000.00	4000.00	0.00	3500.00	0.00
Servidor Base de Datos	0.00	0.00	0.00	4000.00	0.00
Total Equipos	4000.00	4000.00	0.00	7500.00	0.00
Total Costo (Desarrollo + Mantenimiento + Equipos)	118624.84	93550.66	8778.67	29593.48	53730.39
TOTAL COSTO / TOTAL BENEFICIO	18200.61	14353.48	5129.11	3748.37	13587.75
Notas					
Todos los precios están expresados en dolares americanos					
Tasa Efectiva Anual (TEA) 25%					
Tasa Efectiva Mensual calculada (TEM) 1.88%					
Fórmula de Valor Actual Vactual = Mensualidad * ((1+TEM) ^{meses} - 1) / (TEM * (1 + TEM) ^{meses})					